Introduction to Product Provisioning and Staging

Overview

The following document describes how a customer can push products containing files/actions, profiles, and/or applications over the air from AirWatch to rugged devices with a set of ordered instructions. The product provisioning system can be governed by conditions and rules set by you.

In This Guide

- Before You Begin – Details device hardware and software supported, requirements, recommended reading, and things you should know and do before proceeding.
- Relay Server Configuration– Details how to create content delivery servers used to push MSP staging to devices. These FTP(S) servers help ease strain on congested servers. They can also be used for Product Provisioning.
- Motorola Device Staging – Describes the process for creating auto-enrollment configurations including Wi-Fi profiles and download agents for Motorola rugged devices using Windows Mobile or Android.
- Product Provisioning – Explains the creation of ordered installations of products consisting of files/actions, Profiles, and/or Applications that are pushed to devices and installed based on conditions set.
    - Creating Profiles – Details how to create rugged device specific profiles used in provisioning. The profiles are installed as part of product provisioning or staging to make updating and installing easier.
    - Creating Files/Actions – Covers how to create files and actions for devices as well as file management. These files/actions can later be sent to devices as part of a product.
    - Uploading Applications – Details how to upload internal apps for your rugged devices. Apps uploaded here can be pushed to all your rugged devices as part of a product.
    - Defining Conditions – Explains how to create specific conditions to test before downloading or installing products. These tests ensure end users are updating and installing at a specified time and location.
    - Pushing Products – Details the creation of products and the ordered installations contained within.
- Product Provisioning Management – Details how to manage devices and products used in product provisioning.
- Appendix A –Android OS Upgrade – Covers the creation of an OS Upgrade File/Action and how the process works on the device side.
- Appendix B – XML Provisioning – Details how to use Product Provisioning to download and install XML updates to a Windows Mobile device.
- Appendix C – Pull Server Configuration – Explains how to configure your FTP(S) server as a pull relay server. Covers downloading and installing the pull service.
- Appendix D – Windows Mobile OS Upgrade – Details how to download, extract, provision, and install the Windows Mobile OS package.
- Appendix – Custom Attributes – Details how to configure, gather, and apply custom variables from Windows Mobile devices for use with Product Provisioning.

Before You Begin

Overview

This product provisioning guide is for AirWatch administrators and explains the complete process from creating relay servers to managing those devices in AirWatch. This guide simplifies the entire process by explaining each process step-by-step in a logical sequence. By following procedures in this guide, you can ensure a successful deployment of product provisioning for rugged devices.

In This Section

This section provides all the information you need to know prior to advancing to the procedures in this guide:

- Requirements – Details useful and/or required information you need before continuing with this guide.
- Supported Devices, OS, Agents, Versions, and Browsers – Lists the devices and software versions supported by AirWatch.
- Recommended Reading – Provides a list of helpful guides to better your understanding of mobile device management and aspects of Product Provisioning.

Requirements

Before you can push products to rugged devices, you must have the following:

- A relay server created for your devices to connect to if you are using the staging configuration
- A staging configuration created and devices enrolled for Motorola devices.
- A product created containing files/actions,profiles, and/or applications.

Supported Devices, OS, Agents, Versions, and Browsers

Product Provisioning is for devices with the following operating systems:

- Windows CE 5, 6, and 7
- Windows Mobile 5.x/6.1/6.5 (Professional and Standard)
- Windows Embedded 6.5
- Android on the following devices:
  - ET1
  - MC40
  - TC55 (with and without GMS)

- TC70

Recommended Reading

AirWatch provides on the Resource Portal many documents, videos, and webinars on a multitude of related subjects that give you additional background and knowledge to help you in the processes explained within this guide. If this is your first time using this guide, you might find the following information helpful:

- **AirWatch Windows Mobile Platform Guide** – Provides additional information regarding the use of Windows Mobile devices.
- **AirWatch Android Platform Guide** – Provides additional information regarding the use of Android devices.
- **AirWatch Mobile Device Management Guide** – Provides additional information regarding the general aspects of MDM.

Relay Server Configuration

Overview

The main purpose of a relay server is to act as a content distribution node that provides help in bandwidth and data utilization control. Relay servers are used to push products to the devices. The relay servers act as a proxy between the MDM server and the managed devices. This proxy basically serves as an FTP(S) server that will distribute products to the device for download and installation. Using relay servers allows the product to distribute to all devices without consuming all of the bandwidth to the main/central MDM server.

Why do you need this? Relay servers are absolutely required for Motorola Rapid Deployment Barcode Enrollment. Relay servers are optional for pushing products to downloaded apps and content from a relay server as opposed to downloading directly from AirWatch Server. Relay servers also add redundancy through the fallback feature. If a devices relay server is down, the device will fallback to the next relay server in the hierarchy system until it finds a working server or connects to the AirWatch Server. If you are not using a relay server, the device downloads apps and contents directly from the AirWatch Server.

Relay servers can be configured for either push or pull configuration. A pull relay server pulls content from AirWatch based on certain variables established in the server configuration. A push server will push content from AirWatch to devices when ever it is published. For more information on installing a pull server, see Appendix C – Pull Relay Server Configuration.

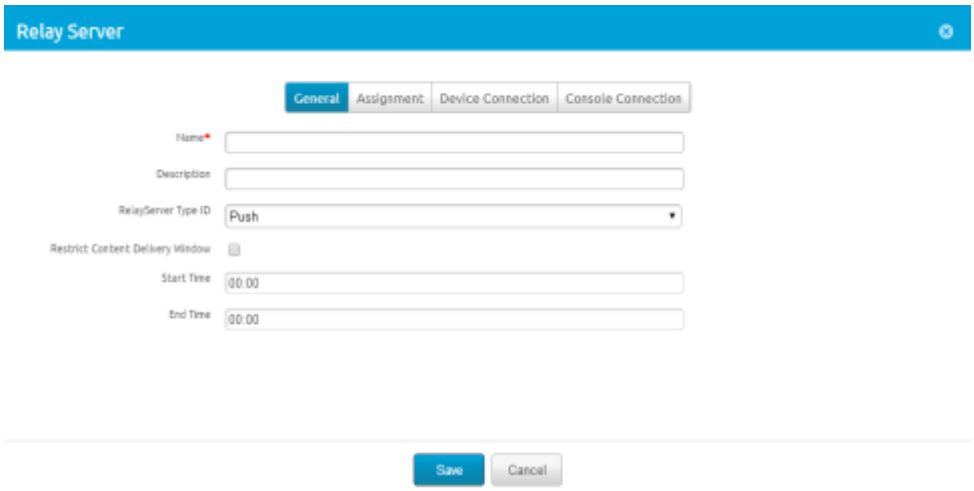
Note: Relay servers, both push and pull configurations, fallback to the next available relay server in its hierarchy and continues to fallback until the device finds a suitable server or reaches AirWatch. This ensures devices with products provisioned to them will have access to their content.

In This Section

Configuring Relay Servers

To configure a relay server, follow the steps outlined below:

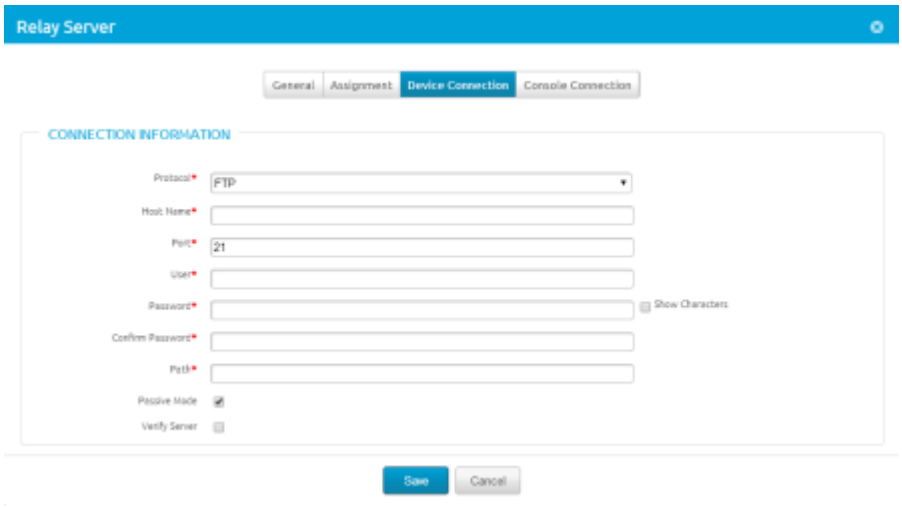
1. Navigate to **Devices ▶Product (New) ▶Relay Servers ▶List View ▶Add Relay Server**.



2. Complete the **General** fields:
   - Choose a Server **Name**.
   - Provide a **Description**.
   - Choose the **Relay Server Type** from the drop-down menu:
     - **Push** –This method is typically used in on-premise models. The AirWatch Admin Console pushes content and applications contained in the product or staging to the relay server from.
     - **Pull** – This method is typically used in SaaS models. A web-based application stored in the relay server pulls content and applications contained in the product or staging from the AirWatch Admin Console through an outbound connection. For more information on installing a pull server, see Appendix C – Pull Relay Server Configuration.
   - Enable **Restrict Content Delivery Window** to limit content delivery to a specified time window.

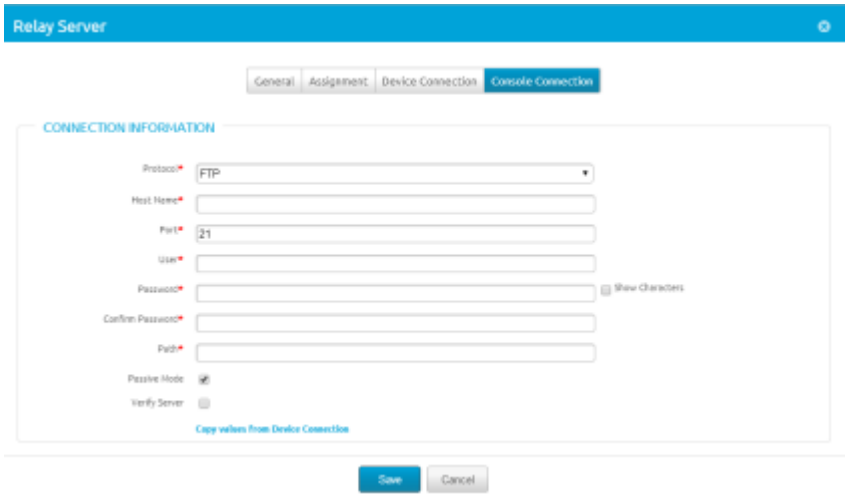
     If enabled, provide a **Start Time** and **End Time** based on local time.
3. Select the **Device Connection** tab and complete the connection information fields. This is the information the device will use to authenticate with the FTP(s) server when downloading apps and content.
   - Choose an **FTP** or **FTPS Protocol**. If using FTPS, your FTPS server must have a valid SSL certificate. This is configured on the FTPS server.
   - Provide a **Host Name**.
   - Select the **Port** established for your FTP(S) server.
   - Provide a **User** and a **Password**.
   - Supply the **Path** for the server.
   - Enable **Passive Mode** to have the client establish both the data and command channels.
   - Check **Verify Server** to ensure the connection is trusted and there are no SSL errors. Use only if you use FTPS servers.

     If left unchecked, then the certification used to encrypt the data can be untrusted and data can still be sent.



Note: The ports you configured when you create your FTP(S) server must be the same ports you enter when creating a relay server in the AirWatch Admin Console.

4. Select the **Console Connect** tab and complete the fields. This is the information that the AirWatch Admin Console uses to authenticate with the FTP(S) server when pushing apps and content. The fields are typically identical to the Device Connection tab.



5. Choose the **Assignment** tab and **Assign Organization Groups** to the **Staging Server** and the **Production Server**. The default settings are recommended.

   A staging server only works for the Staging process involving the Rapid Deployment Client. A production server works with any device with the proper agent installed on it.

   If you want to use the FTP(S) server for Motorola Rapid Deployment Barcode Enrollment only and not for Product Provisioning, remove all assigned Organization Groups under the production server section.

6. Select **Save**.

Note: In order to edit a product, the product must be deactivated in the List View first.

Viewing Relay Server Status

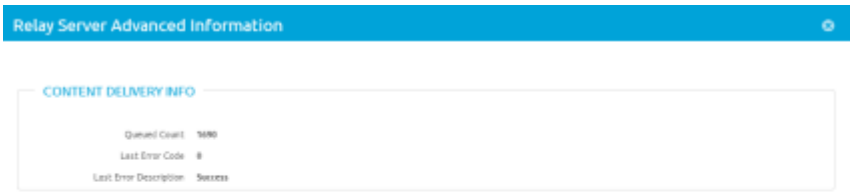
After creating a relay server, you can refresh the relay server detail page to get the real-time status of the connection.

The status colors are as follows:

- **Green** – There are no errors and the connection is ready.
- **Yellow** – The connection setup is in progress.
- **Red** – There is an error. Hovering over the status color provides the error.

Advance Info Action

Along with the Relay Server Status, you can access the **Advance Info** action for more detailed information pertaining to the server. The Advance Info action displays the **Queued Count** of files, the **Last Error Code** displayed, and the **Last Error Description**. This action can be found in the **More** options drop-down menu located to the right of the Status color.



Importing Relay Servers in Bulk

The Relay Server Import feature loads relay servers into the system in bulk. The relay servers users can associate with an Organization Group.

To bulk import relay servers, follow the steps detailed below:

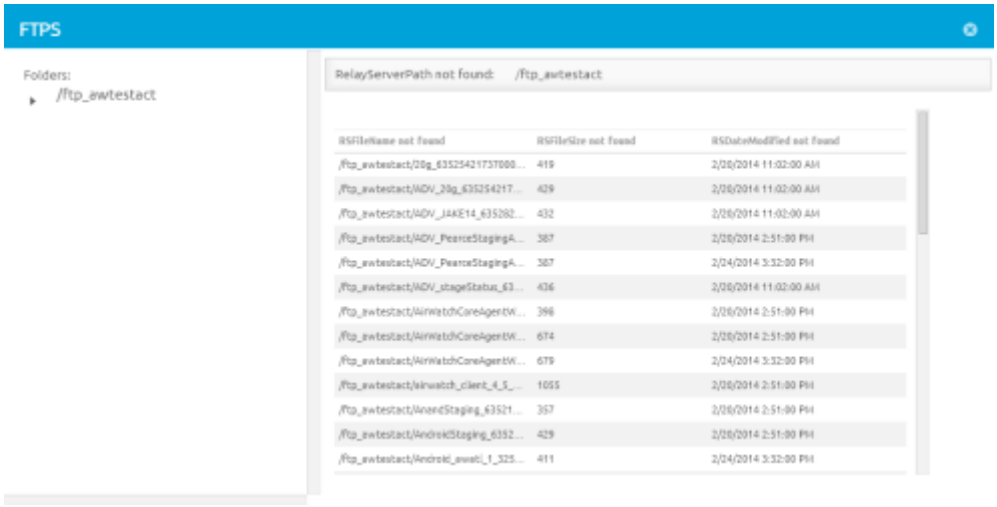
1. Navigate to **Devices ▶Product (New) ▶Relay Servers ▶Batch Status ▶Batch Import**.
2. Enter a **Batch Name**.
3. Enter a **Batch Description**.
4. Select **Choose File**to upload the **Batch File**. Batch files must be in .CSV format. Select the **Information** icon to download a template.
5. Select **Save** to upload the batch import.

Remote Viewing Files on Relay Server

Files that have been sent to a relay server for distribution to devices can be viewed through the Remote File Viewer.

To access the Remote File Viewer, follow the steps detailed below:

1. Navigate to **Devices ▶Product (New) ▶Relay Servers ▶List View**.
2. On the far right of a server listing, select the **More** option.
3. Select **Remote File List**. This opens the Remote File List for your viewing. The list allows you to see what files are on a relay server.



Device Staging

Overview

Staging prepares a Motorola rugged device running Windows Mobile or Android for enrollment. For non-Motorola devices, sideloading packages offer many of the same features. The staging package automatically configures a device with a Wi-Fi profile, specific download Agent, and then enrolls it into AirWatch. This section also covers the advanced option of creating a ordered list of actions to be taken during staging.

Note: The Staging section is only for the enrollment of Motorola rugged devices. For information on staging and enrolling other devices, please consult the related platform guides.

In This Section

1. Creating a Wi-Fi Profile – Details the steps required for creating a Wi-Fi profile to be used in staging.
2. Staging – Covers the steps needed to create a Staging configuration.
3. Advanced Staging – Explains how to create a manifest of installation/uninstallation instructions.
4. Enrolling a Device – Describes how to use a barcode to enroll devices.
5. Generating a Side Staging – Details how to create a sideload staging package for use on devices that do not support Motorola Rapid Deployment Barcode Enrollment.

Creating a Wi-Fi Profile

It is mandatory that your staging configuration include a Wi-Fi profile. This is the network that the device uses to connect to the FTP(s) server after the barcode is scanned to download the AirWatch MDM Agent.

A Wi-Fi profile is either a staging or production profile. The staging Wi-Fi profile is created under the Products section and connects the device to the relay server so the device can receive the staging configuration. The production Wi-Fi profile is created under Device Profiles and pushes the production Wi-Fi a device uses if it is staged at a location that is different than where the device will be used on a daily basis.

To create a Wi-Fi profile, follow the steps detailed below for each OS.

Android Wi-Fi Profile:

1. Navigate to **Devices ▶Product (New) ▶Profiles ▶Add** and choose **Android** from the list of platform options.
2. Fill in **General** settings
   - **Staging Wi-Fi Profile** – Connects a device to the Wi-Fi used for staging.
   - **Production Wi-Fi Profile** – Connects a device to the Wi-Fi used for everyday use. Production Wi-Fi profiles are under **Device ▶Profiles ▶List View ▶Add**. You must use auto deployment and publish the profile before staging a device with it.

3. Select **Wi-Fi**.
4. Provide the **Service Set Identifier** to name the network to which the device will connect.
5. Indicate if the Wi-Fi network is a **Hidden Network**.
6. Indicate if the device connects to the network with no end-user interaction as an **Active Network**.
7. Specify the **Security Type** of access protocol used and whether certificates are required.
8. Provide the **Password** required for the device to connect to the network.
9. Select **Save**.

Windows Mobile Wi-Fi Profile:

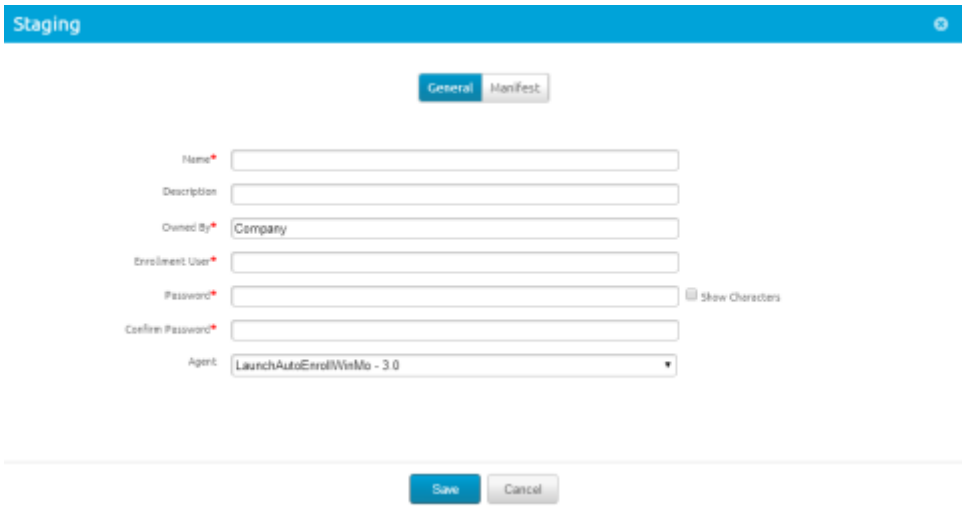
1. Navigate to **Devices ▶Product (New) ▶Profiles** and select **Add**. Select **Windows Mobile**.
2. Fill in **General** settings
    - **Staging Wi-Fi Profile** – Connects a device to the Wi-Fi used for staging.
    - **Production Wi-Fi Profile** – Connects a device to the Wi-Fi used for everyday use. Production Wi-Fi profiles are under **Device ▶Profiles ▶List View ▶Add**. You must use auto deployment and publish the profile before staging a device with it.
3. Select **Wi-Fi**.
4. Configure the Wi-Fi settings, including:
    - Peer-to-peer networks
    - Basic and enterprise Wi-Fi authentication protocols, including:
        - Open Authentication
        - Shared Authentication
        - WPA, WPA-PSK, WPA NONE
        - WPA2, WPA2-PSK
        - WPA Enterprise (Fusion only)
        - WPA2 Enterprise (Fusion only)
    - Encryption standards such as 802.1x
    - Certificate-based and smart card authentication methods
5. Select **Save & Publish** to push the profile to devices.

Staging

To create a Staging configuration, follow the steps outlined below:

1. Navigate to **Devices ▶Product (New) ▶Staging ▶List View ▶Add**.
2. Select the Platform type you want to create a staging configuration for.

3. Complete the required fields on the **General** tab.
    - Choose a Wi-Fi Profile **Name**.
    - Select who the profile is **Owned By**.
    - Provide an **Enrollment User** and **Password**.

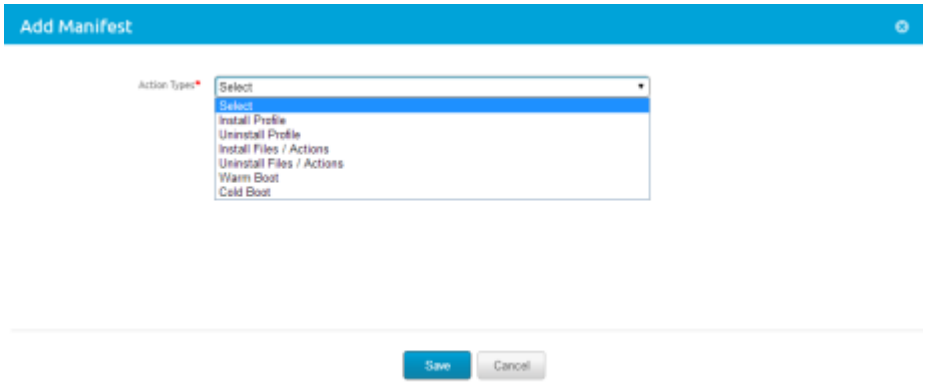


4. Select an **Agent** from the drop-down menu.
5. Select **Save**.

Advanced Staging

To establish a list of ordered steps during staging, follow the steps detailed below:

1. After completing the **General** tab of the Staging window, select the **Manifest** tab.
2. Select **Add**.



3. Select the action you want to take place during staging.
    - **Install Profile** – Select the profile you want to install. For more information on creating profiles, see the Profiles section.
    - **Uninstall Profile** – Select the profile you want to uninstall.

- **Install Files/Actions** – Select the files/actions you want to install. For more information on creating files/actions, see the files/actions section.
- **Uninstall Files/Actions** – Select the files/actions you want to uninstall.
- **WarmBoot** – Perform a soft reset of the device.
- **Cold Boot** – Perform a hard reset of the device (Windows Mobile devices only).

Note: For more information on creating files, profiles, actions, see the Product Provisioning section.

4. Select **Add** again to add additional actions to the manifest if desired.
5. When you are finished adding actions, select **Save**.
6. View the newly created staging profile in the List View. Take additional actions on the profile from the menus on the right:
   - **Edit** your configuration.
   - **Copy** your profile.
   - Select **Barcode** and complete the fields on the **Generate Barcode** subpage.

Enrolling a Device With Barcode Staging

End users can scan a barcode you create to begin the auto-enrollment process for their rugged devices. These barcodes are used for any Motorola rugged devices.

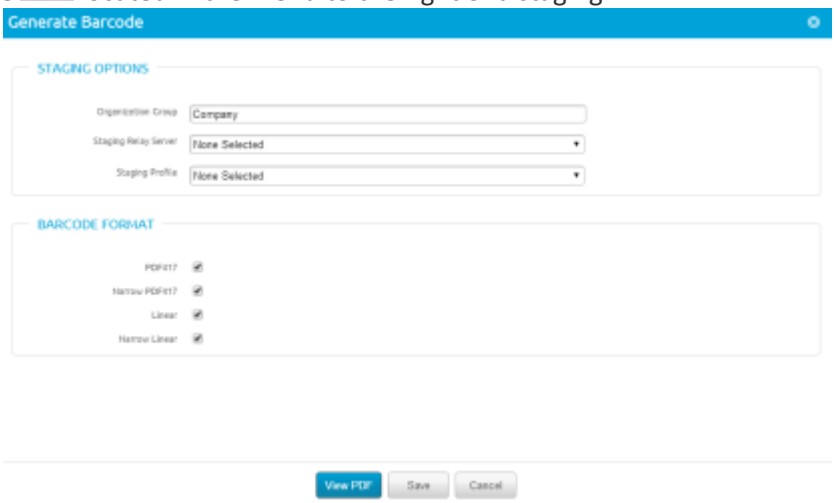
Barcode enrollment is only supported on the following devices:

- Windows Mobile
  - MC45
  - MC55
  - MC65/67
  - MC75
  - MC3090
  - MC3190
  - MC9090
  - MC92N0
- Android
  - ET1
  - MC40
  - TC55 (with and without GMS)
  - TC70

To generate a barcode, follow the steps outlined below:

1. Navigate to **Devices ▶Product (New) ▶Staging ▶List View**

2. Select **Barcode**  located in the menu to the right of a staging



configuration.

3. Select the **Staging Options**:
   - **Organization Group**
   - **Staging Relay Server**
   - **Staging Profile**
4. Select the **Barcode Format** for the device you want to enroll.
5. Select **View PDF**. This generates the barcode for end users to scan.

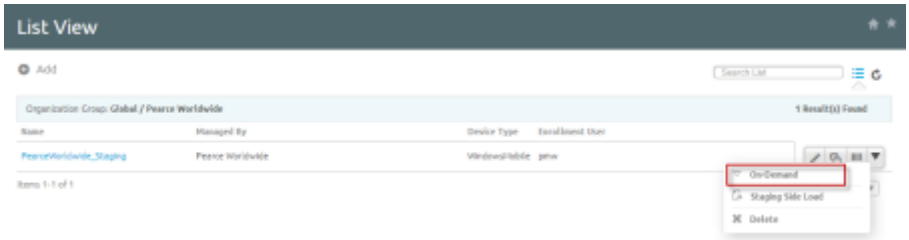
On-Demand Enrollment

On-Demand Enrollment allows you to use a staging profile to stage a device without the use of a barcode.

To use On-Demand Enrollment, follow the steps detailed below:

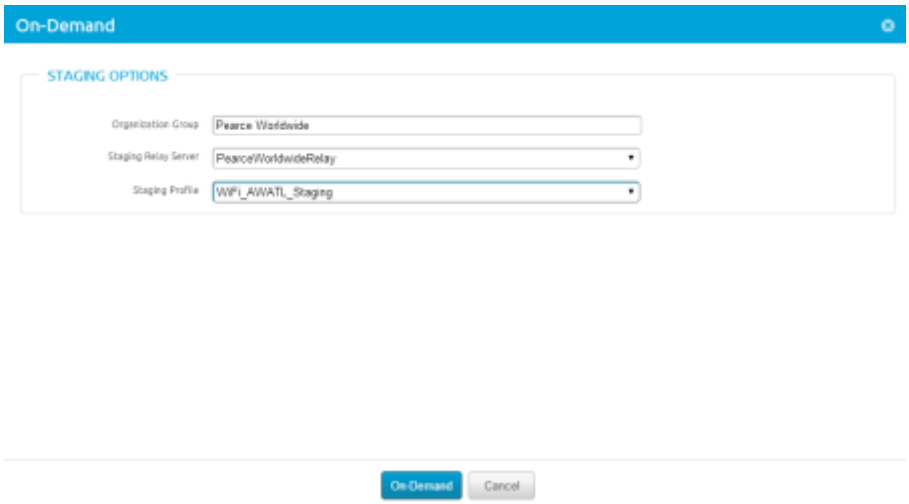
1. Navigate to **Devices ▶Product (New) ▶Staging ▶List View**.

2. Find the Staging configuration you want to use and select the **More** option. 

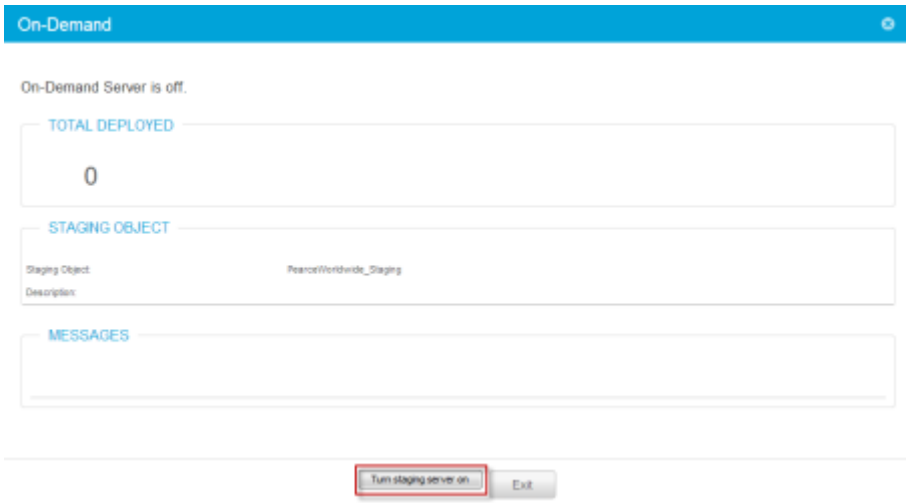


3. Select **On-Demand** from the drop-down menu.
4. Specify the staging options including:
   - **Organization Group** – Defines the AirWatch Organization Group the device enrolls in.

- **Staging Relay Server** – Defines which relay server the device retrieves the agent and other staging content from.
- **Staging Profile** – Defines which Wi-Fi profile to use during staging to connect to the relay server.

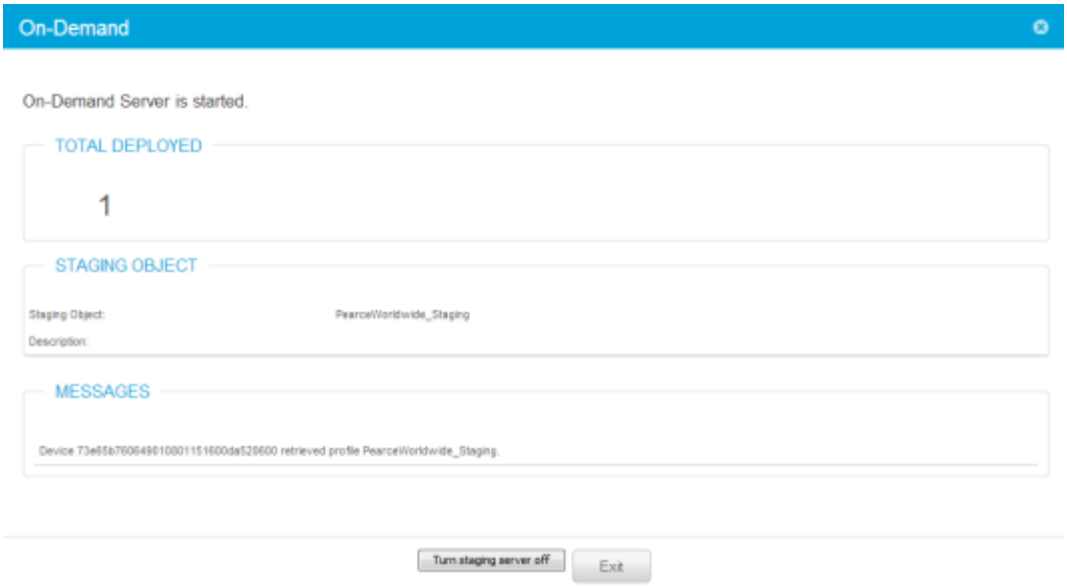


5. Select **On-Demand** to launch the On-Demand Enrollment application.



6. Select **Turn staging server on**.
7. On the device you want to enroll, start the **Rapid Deployment** client and use the following settings:
   - **Search Connected Networks** – Rapid deployment client will search for an On-Demand staging server over any Wi-Fi profiles that exist on the device, or via LAN if cradled. Motorola devices come with a generic Wi-Fi profile out of the box, which you use when setting up a Wi-Fi access point.
   - **Search Unconnected Networks** – Rapid deployment client will search for an On-Demand staging server using ActiveSync. The device must be cradled and connected to the admin's machine hosting the On-Demand server via USB.

Once a device is connected to an On-Demand server, the staging profile configuration passes to the device. The device then retrieves all staging content from the relay server. Once all staging content has been retrieved and installed, the device enrolls in AirWatch.

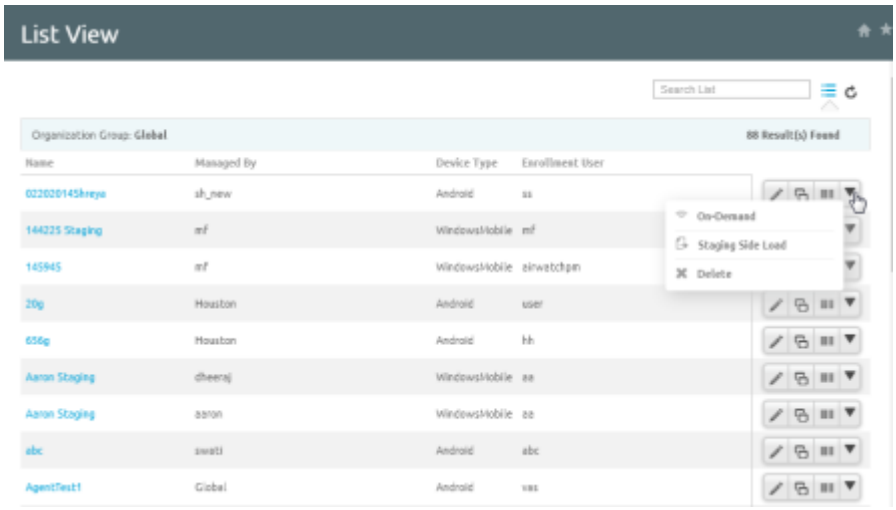


Generating Side Staging

Not all devices that use Product Provisioning support Motorola Rapid Deployment Barcode Enrollment. In such cases, AirWatch can create a side staging package to use with non-Motorola devices that mirrors the staging process for Motorola devices.

To create a Side Staging package, follow the steps detailed below:

1. Navigate to **Devices ▶Product (New) ▶Staging ▶List View**.
2. Choose a staging you want to create a side-loaded staging package for then select the **More** option and select **Staging Side Load**.



3. Choose the **Organization Group** this staging applies to.
4. Select **Download** to start downloading the zip file of the staging side load.

The next steps depend on if the device is a Windows Mobile device or an Android device.

For Android device staging follow the steps below:

1. Unzip the file and connect your device to the staging device once the download is complete.
2. Ensure that the Android Debug Bridge is enabled and running (Android Only).
3. Double-click the batch file located in the unzipped folder.

This script installs the MX Service and agent then applies the Wi-Fi profile you defined in the staging manifest as well as any other manifest items. Once the Wi-Fi connects, the device auto-enrolls into AirWatch.

For Windows Mobile device, follow the steps below:

1. Unzip the file and connect your device to the staging device once the download is complete.
2. Move the unzipped file to the device.
3. Double-click the batch file located in the unzipped folder.

The script installs the agent then applies the Wi-Fi profile you defined in the staging manifest as well as any other manifest items. Once the Wi-Fi connects, the device auto-enrolls into AirWatch.

Product Provisioning

Overview

The main feature of the Product Provisioning system is the ability to create an ordered installation of Files/Action, **Profiles**, and/or **Applications** into one product to be pushed to devices based on the **Conditions** you create.

Once products are created and activated, they are pushed to the device based on the conditions set. Conditions are an optional tool that determine when a product is downloaded as well as when it is installed. Applications and content provisioning by products can be pushed to devices through the use of optional relay servers.

Products are pushed to devices that are chosen by Smart Group Assignments. These groups control which devices get which product based on how the group is created. For more information on Smart Groups, consult the **AirWatch Mobile Device Management Guide**.

In addition, the AirWatch Admin Console periodically syncs with a device and checks the content of the device against those assigned via a product. Should content be missing, the console pushes the content to the device again to ensure compliance between the product and the device.

Note: You must create and/or upload the content of the product before a product can be created.

In This Section

- Creating Profiles – Details how to create rugged device specific profiles used in provisioning. The profiles are installed as part of product provisioning or staging to make updating and installing easier.
- Creating Files/Actions – Covers creating files and actions for devices as well as file management. These files/actions can later be sent to devices as part of a product.
- Uploading Applications – Details how to upload internal apps for your Android rugged devices. Apps uploaded here can be pushed to all your rugged devices as part of a product.
- Defining Conditions – Explains how to create specific conditions to test before downloading or installing Products. These tests ensure end users are updating and installing at a specified time and location.
- Pushing Products – Details the creation of products and the ordered installations contained within.

Creating Profiles

The product provisioning system allows you to create profiles for your rugged devices.The profiles created for rugged devices are installed or uninstalled as part of a product.

Note: Assignment happens at the product level and not at the profile level as it is in smartphone profiles.

To configure a profile, follow the steps detailed below:

1. Navigate to **Devices ▶Product (New) ▶Profiles ▶Add**.
2. Select the Platform type you want to create a staging configuration for.

3. Complete the **General** fields.
    - Enter a **Name**.
    - Enter a **Description**.
    - Select the **Profile Scope**.
        - Choose **Production** for profiles to be used as part of product provisioning.
        - Choose **Staging** for profiles to be used in stating configurations.
        - Choose **Both** for profiles to be used in both staging and provisioning.
    - Select if you will **Allow Removal** of the profile.
    - Select who the device will be **Managed By**.
4. Select the type of **Profile** you want to configure depending on the Platform chosen:

    Note: AirWatch recommends setting up one payload per profile to minimize multiple settings being pushed down at once. Pushing multiple payloads at once makes troubleshooting difficult, requires pushing down all settings at once rather than the one that needs update and is less granular of a configuration.

    - **Windows Mobile**
        - **Passcode**
        - **Restrictions**
        - **Wi-Fi**
        - **Exchange ActiveSync**
        - **Credentials**
        - **Launcher**
        - **VPN**
        - **Time Sync**
        - **Shortcut**
        - **Time Zone**
    - **Android**
        - **Passcode**
        - **Restrictions**
        - **Wi-Fi**
        - **VPN**
        - **Email Settings**
        - **Exchange ActiveSync**
        - **Application Control**
        - **Bookmarks**

- **Credentials**
- **Secure Launcher**
- **Global Proxy**
- **Date/Time**
- **Sound**
- **Display**
- **Advanced**
- **Custom Settings**

5. Select **Save** when you are finished configuring the profile.

For more information on creating profiles, see the **AirWatch Windows Mobile Platform Guide** or the **AirWatch Android Platform Guide**.

Updating Profiles

When you edit an existing profile, the version number automatically increases. After saving the edits, AirWatch runs a check against all active products to find any that contain the newly edited profile.

If any active products contain the profile, a warning prompt displays listing all active products affected by the edited profile. You can then choose to **Activate** or **Deactivate** a product using the profile.

Deleting Profiles

AirWatch checks any attempt to delete a profile against the list of active products. Should a profile be part of an active product, a warning prompt appears listing any product that uses the profile.

In order to delete a profile, it must be detached from all products as detailed below:

1. Select the **Profile** listed in the Warning prompt.
2. Select **Edit**.
3. Remove the profile from the product.
4. Select **Save**.
5. Repeat for all products containing the profile.
6. Once the profile detaches from all products, you may delete the profile.

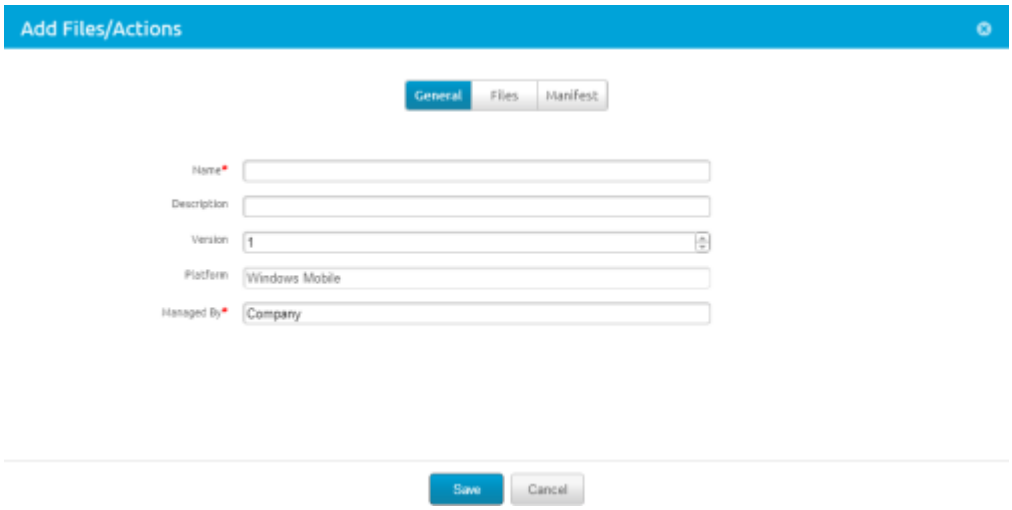
Creating Files/Actions

You can upload files and actions for use in **Product Provisioning**. The files/actions section also contains ways to manage the file system of a device. You can use this wizard to upload multiple files/actions at once. It is important to take the device's storage space into account when adding files/actions. When creating a file/action, at least one file or one action is required.

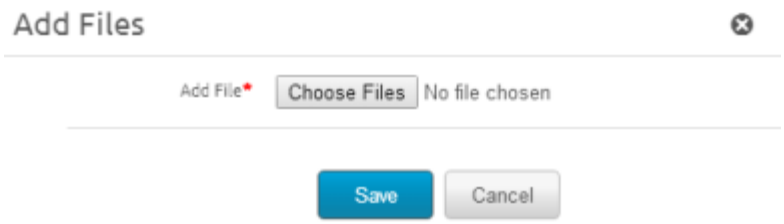
Note: Smart Group assignment happens at the Product level and not at the file/action level.

To add files/actions, follow the steps outlined below:

1. Navigate to **Devices ▶Product (New) ▶File/Actions ▶Add**.
2. Select the Platform you want to create a staging configuration for.
3. Complete the **General** fields.
   - Enter a **Name**.
   - Enter a **Description**.
   - View the **Version**, which the AirWatch Admin Console automates.
   - Enter who the files/actions are **Managed By**.



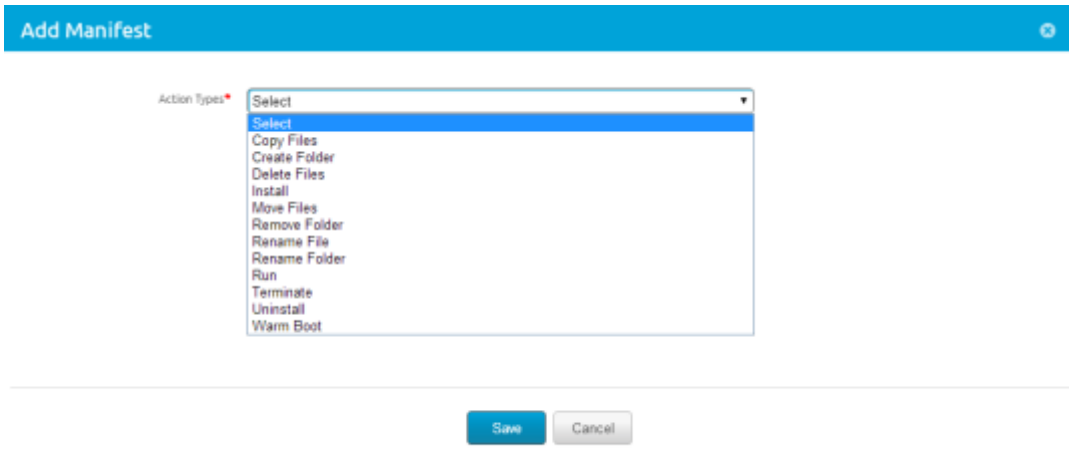
4. Select the **Files** tab
5. Select **Add Files**.



6. Select **Choose Files** to browse for a file or multiple files to upload.

Note: Windows Mobile devices can use the files/actions option to install XML onto a device. For more information, see Appendix B – XML Provisioning.

7. Define the **Download Path** the device will use to store the File(s) in a specific device folder.
8. Select **Save**. You may repeat the previous steps for as many files as you want.
9. Select the **Manifest** tab. Actions are no required as long as you have at least one file uploaded.



10. Add actions to the **Install Manifest** or the **Uninstall Manifest** if needed:

    - **Copy Files** – Copy files from one location to another on the device.
    - **Create Folder** – Create a new folder on the device.
    - **Delete Files** – Delete folders from the device.
    - **Install** – Install files on the device (Windows Mobile devices only). Supports the following: .reg, .cab, and .xml.
    - **Move Files** – Move files from one location to another on the device.
    - **Remove Folder** – Remove a folder from the device.
    - **Rename File** – Rename a file located on the device.
    - **Rename Folder** – Rename a folder located on the device.
    - **Run Intent** – Run command lines and arguments on the device (**Run** for Windows Mobile devices).
    - **Terminate** – Ends a process or application running on the device (Windows Mobile devices only).
    - **Uninstall** – Uninstalls a program or application on the device (Windows Mobile devices only).
    - **Warm Boot** – Restarts the device for Windows Mobile devices.
    - **Reboot** – Restarts the device for Android devices.
    - **OS Upgrade** – Installs a new OS upgrade as well as the relevent AirWatch MDM Agent. For more information on this option, see Appendix – OS Upgrade (Android devices only).
    - **AirWatchMDM Agent Upgrade** – Installs the new MDM Agent for Android devices that cannot access the Google Play store (Android devices only).

Note: The Uninstall Manifest is used for deleting files when a product is removed. If you remove a product from a device, any files installed remains on the device until uninstalled with using an Uninstall Manifest. Android can only remove products through unenrollment.

11. When finished adding actions to the **Manifest**, select **Save**.
12. View the newly created File/Action in the List View. Take additional actions on the File/Action from the menus on the right:

    - **Edit** your files/actions.

- **Copy** your files/actions.
- **Delete** your files/actions.

Updating Files/Actions

When you edit any existing files/actions, the version number automatically increases. After saving the edits, AirWatch runs a check against all active products to find any that contain the newly edited files/actions.

If any active products contain the files/actions, a warning prompt displays listing all active products affected by the edited files/actions. You can then choose to **Activate** or **Deactivate** a product using the files/actions.

Deleting Files/Actions

AirWatch checks any attempt to delete files/actions against the list of active products. Should the files/actions be part of an active product, a warning prompt appears listing any product that uses the files/actions.

In order to delete files/actions, it must be detached from all products as detailed below:

1. Select the **Files/Actions** listed in the Warning prompt.
2. Select **Edit**.
3. Remove the files/actions from the product.
4. Select **Save**.
5. Repeat for all products containing the files/actions.
6. Once the files/actions destaches from all products, you may delete the files/actions.

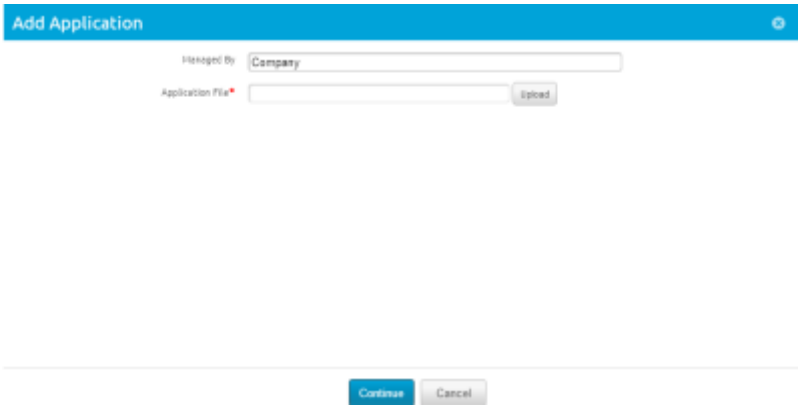
Uploading Applications (Android Only)

The Application section of Product Provisioning allows you to upload applications to the console for distribution as part of a product.

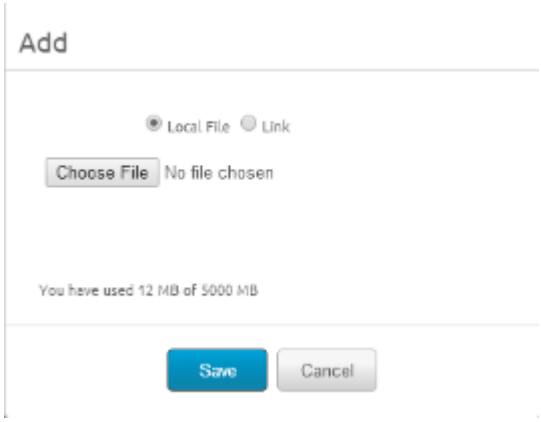
Note: Smart Group Assignment happens on the Product level and not on the Application level.

To add an Application, follow the steps detailed below:

1. Navigate to **Devices ▶Product (New) ▶Applications ▶Add Application**.



2. Enter who the application will be **Managed By**.
3. Select **Upload** to browse for the **Application File**.
4. Select **Choose File** to add a local file or select **Link** to enter a link.



5. Select **Save** to finish uploading the application.
6. Select **Continue** to add the application to the Product Provisioning application list.

For more information on adding applications, please consult the **Mobile Application Management Guide**.

Adding New Application Versions

You can add a new version of an already uploaded application. This allows you to push the newest version of an application to end users using the existing products you have already created.

To add a new version, follow the steps detailed below:

1. Navigate to **Devices ▶Product (New) ▶Applications ▶More**.
2. Select the **Add Version** option from the drop-down menu.
3. Upload the new version of the application as described above.
4. When you select **Save** to save the new version of the application, AirWatch runs a check to see if the application is currently attached to an active product.

   If the application is a part of an active product, a warning prompt displays showing a list of all the products that is affected by the update. You can then choose to **Update** or **Deactivate** any of the listed products.

Deleting Applications

AirWatch checks any attempt to delete an application against the list of active products. Should an application be part of an active product, a warning prompt appears listing any product that uses the application.

In order to delete an application, it must be detached from all products as detailed below:

1. Select the **Product** listed in the Warning prompt.
2. Select **Edit**.
3. Remove the application from the product.
4. Select **Save**.
5. Repeat for all products containing the application

6. Once the application detaches from all products, you may delete the application.

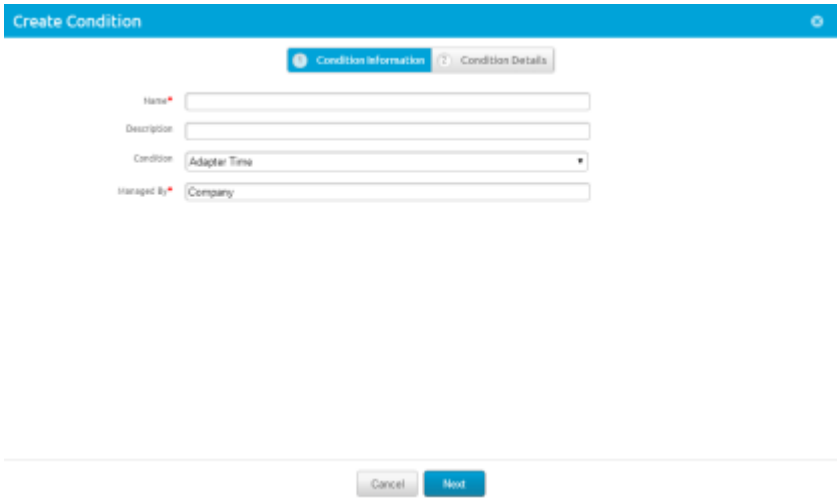
Defining Conditions

A condition dictates when the product or OS upgrade package should be downloaded and/or installed. Conditions are checked when a product is pushed to a device.

Note: This section is optional and contains options for advanced users.

To create a condition, follow the steps listed below:

1. Navigate to **Devices ▶Product (New) ▶Conditions ▶Add Condition**.



2. Complete the **Condition Information** fields:
   - Enter a **Name**.
   - Enter a **Description** for the Condition.
   - Select a **Condition Type** - the type chosen affects the parameters on the **Condition Details** screen.
     - **Adapter Time**
     - **Adapter** (**Windows Mobile** device only) – This condition is a legacy from Motorola System Provisioning and AirWatch recommends using the Adapter Time condition instead.
     - **Confirm**
     - **Power**
     - **Time** (**Windows Mobile** device only) – This condition is a legacy from Motorola System Provisioning and AirWatch recommends using the Adapter Time condition instead.
     - **SD Card Encryption** (**Android** device only)
   - Select who the condition will be **Managed By**.
3. Select **Next**.
4. Complete the **Condition Details** field based on the condition type chosen above:
   - **Adapter Time**:

This condition type tests for various combinations of constraints related to **Network Adapters** and/or local date and time on the device.

Note: Each Scenario is an OR statement and each option inside a Scenario is an AND statement. For example, a device will check to see if Scenario #1 OR Scenario #2 is true. If Scenario #1 is true, it will check if all the constraints listed are true because they are AND statements.

a. **Specify Scenario #1**:
    a. Choose to **Constrain Network Adapters** or not:
        - Specify any **Included or Excluded Network Adapters**.
        - Choose to either select adapters from a drop-down list or to type in an adapter name.
        - Up to five network adapters may be included or excluded.
    b. Choose to **Constrain Days of Week**:
        - For each day of the week, choose whether it will be included or excluded.
    c. Choose to **Constrain Months**:
        - For each month, choose whether it will be included or excluded.
    d. Choose to Constrain Days of Month:
        - Enter a **Start Day of Month** and a **End Day of Month**.
    e. Choose to **Constrain Years**:
        - Enter a **Start Year** and an **End Year**.
    f. Choose to **Constrain Time of Day**:
        a. Enter a **Start Hour** and **Start Minute**.
        b. Enter an **End Hour** and **End Minute**.
    g. Choose to **Set a Frequency Limit**:
        - **Every 15 Minutes**
        - **Every 30 Minutes**
        - **Every 1 Hour**
        - **Every 2 Hours**
        - **Every 4 Hours**
        - **Every 8 hours**
        - **Every 12 hours**
        - **Every 1 day**
        - **Every 1 Week**

Up to 5 scenarios may be entered each with their own constrain choices.

- **Adapter**

This condition type tests to see which, if any **Network Adapters** are connected. This can be highly relevant if network connectivity is a scarce or expensive resource and certain operations should be limited to use over certain **Network Adapters** or prohibited from use over certain **Network Adapters**.

a. Choose whether to **Exclude Adapters**.
   - If False is entered, the adapters selected in the following steps are included in the condition.
   - If True is entered, the adapters selected in the following steps are excluded in the condition.

b. Choose to **Select Adapter from List**:
   - If False is entered, the **Select Adapter** list changes to a field to enter **Adapter Name**.

c. Choose whether to specify second and/or third adapters.

Note: It is strongly recommended to use the Adapter Time Condition Type instead of the Adapter Condition Type as support for the Adapter Condition Type has been deprecated and may be withdrawn completely.

- **Confirm**:

  This condition type prompts the end user to determine whether or not the condition is met. The prompts displayed can be controlled, effectively allowing the "question" being asked to be customized.

  a. Enter a header of the prompt into the **First Line Prompt** field.

  b. Enter the body of the prompt into the **Second Line Prompt** field.

  c. If you enable a countdown, you can enter a countdown phrase into the **Third Line Prompt** field.

     For example, "You have %count% seconds to comply" where %count% will be the countdown clock.

  d. Enter a **Delay** (in seconds).

     Use this to delay for a specified time or until the end user makes a selection. If you enter a non-zero value, the prompt will wait for that value worth of seconds. Then if the end user does not make a selection in the time allowed, the condition is automatically considered not met. If a value of zero is entered, then the prompt will display indefinitely until the end user makes a selection

  e. Choose to **Enable Countdown**.

     This allows delay time to be “counted” down on the device so the end user knows how much time is remaining for the user to make a selection.

  f. Choose to **Enable Cancel**.

  g. Enter a **Defer Time** (in seconds).

     This controls the minimum time after the condition is considered to be not met before the end user will be prompted again to determine the state of this condition. If a non-zero value is entered, the end user will not be prompted again for at least that number of seconds. If a value of zero is entered, then the end user could be prompted again as soon as the next execution of the Check-In command.

  h. Enter a **Maximum Number of Defers**.

     This controls the maximum number of times the condition is not met. Once the condition has not been met this number of time, it will either be met or failed, depending on the setting of the next feature. If a value of zero is entered, then the condition will be met or failed on the first time.

  i. Choose an **Action after Maximum Defers** is met.
     - **Fail Condition**

- **Display Cancel Button**
- **Pass Condition**

- **Power**

  This condition type tests how a device is being powered including whether the device is on A/C and/or has a suitably high battery level. A **Power** condition type can also be used to prompt the end user such as to ask him to place the device into the cradle or insert a more fully charged replacement battery.

  a. Enter a header of the prompt into the **First Line Prompt** field.

  b. Enter the body of the prompt into the **Second Line Prompt** field.

  c. If you enable a countdown, you can enter a countdown phrase into the **Third Line Prompt** field.

     For example, "You have %count% seconds to comply" where %count% will be the countdown clock.

  d. Enter a **Delay** (in seconds).

     Use this to delay for a specified time or until the end user makes a selection. If you enter a non-zero value, the prompt will wait for that value worth of seconds. Then if the end user does not make a selection in the time allowed, the condition is automatically considered not met. If a value of zero is entered, then the prompt will display indefinitely until the end user makes a selection.

  e. Choose to **Enable Countdown**.

     This allows delay time to be “counted” down on the device so the Device User knows how much time is remaining for the user to make a selection.

  f. Enter the **Required Power Level**.

     - **A/C**
     - **A/C or Full Battery**

- **Time**

  This condition type tests the local date and time on a device.

  a. Enter a specified start time of the time window in the following fields:

     - **Month**
     - **Day**
     - **Year**
     - **Hour**
     - **Minute**

  b. Enter an end time of the time window in the following fields:

     - **Month**
     - **Day**
     - **Year**
     - **Hour**
     - **Minute**

  c. Choose to **Enable Time Check 2** and/or **3**.

Note: It is strongly recommended to use the Adapter Time Condition Type instead of the Time Condition Type as support for the Time Condition Type has been deprecated and may be withdrawn completely.

- **SD Card Encryption**

  This condition type tests whether an SD card is encrypted or not encrypted. This can be highly relevant if you need to wait for the SD card to be encrypted before downloading a file to it.

  a. Choose an **Encryption State**:

  - **Encrypted**.
  - **Unencrypted**.

5. Select **Finish**.
6. View the newly created condition in the List View. Take additional actions on the profile from the menus on the right:

   - **Edit** your condition.
   - **Copy** your condition.
   - **Delete** your condition.

Adding New Condition Versions

You can add a new version of an already uploaded condition. This allows you to push the newest version of an condition to end users using the existing products you have already created.

To add a new version, follow the steps detailed below:

1. Navigate to **Devices ►Product (New) ►Applications ►More**.
2. Select the **Add Version** option from the drop-down menu.
3. Upload the new version of the condition as described above.
4. Select **Save** and AirWatch runs a check to see if the condition is currently attached to an active product.

   If the condition is a part of an active product, a warning prompt displays showing a list of all the products that are affected by the update. You can then choose to **Activate** or **Deactivate** any of the listed products.

Deleting Conditions

AirWatch checks any attempt to delete an condition against the list of active products. Should a condition be part of an active product, a warning prompt appears listing any product that uses the condition.

In order to delete an condition, it must be detached from all products as detailed below:

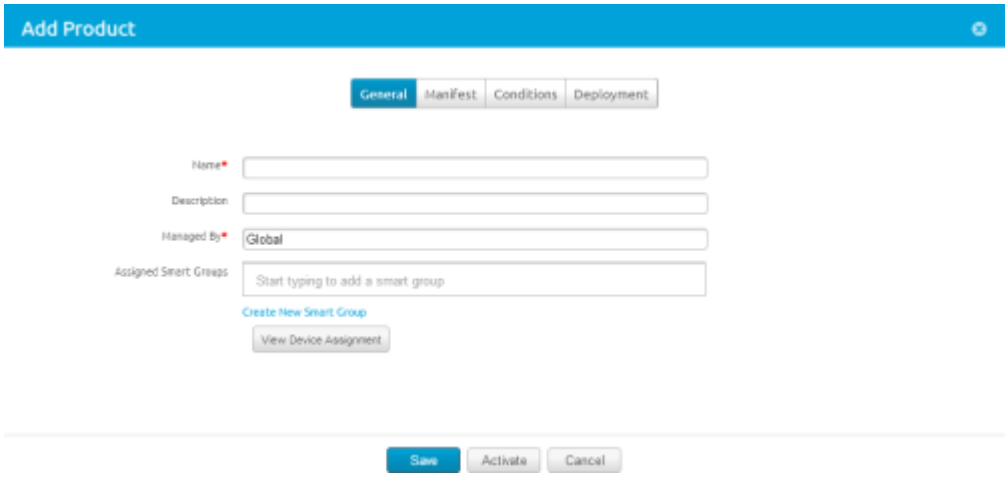
1. Select the **Product** listed in the Warning prompt.
2. Select **Edit**.
3. Remove the condition from the product.
4. Select **Save**.
5. Repeat for all products containing the condition.
6. Once the condition detaches from all products, you may delete the condition.

Pushing Products to Devices

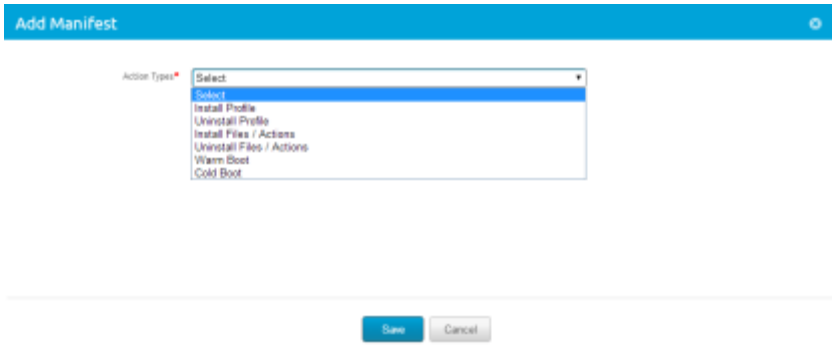
After creating the content you want to push to devices, such as files/actions, apps, or profiles, you must create a product that controls when the content is pushed as well as the ordered installation of the product.

To create and configure a product, follow the steps detailed below:

1. Navigate to **Devices ▶Product (New) ▶List View ▶Add Product**.
2. Select the Platform you want to create a staging configuration for.
3. Complete the General fields:
   - Enter a **Name**.
   - Enter a **Description**.
   - Enter who the Product will be **Managed By**.
   - Enter the **Assigned Smart Groups**. If a Smart Group is not available, one can be created by clicking the link.



4. Select **Add Rules** to use **Assignment Rules** to control which devices receive the product.
5. Select **Add Rule** to create a rule for product provisioning based on the following:
   - **Attribute** – This is the custom attribute created separately. For more information see Appendix – Custom Attributes.
   - **Operator** – This operator compares the Attribute to the Value to determine if the device qualifies for the product.
   - **Value** – This is the value of the custom attribute. All values from all applicable devices are listed here for the **Attribute** selected for the rule.
6. Select **Add Logical Operator** to create more complex rules.
7. Select **Save** to add the **Assignment Rule** to the product.
8. Select the **Manifest** tab.
9. Select **Add** to add a actions to the **Manifest**. At least one manifest action is required.



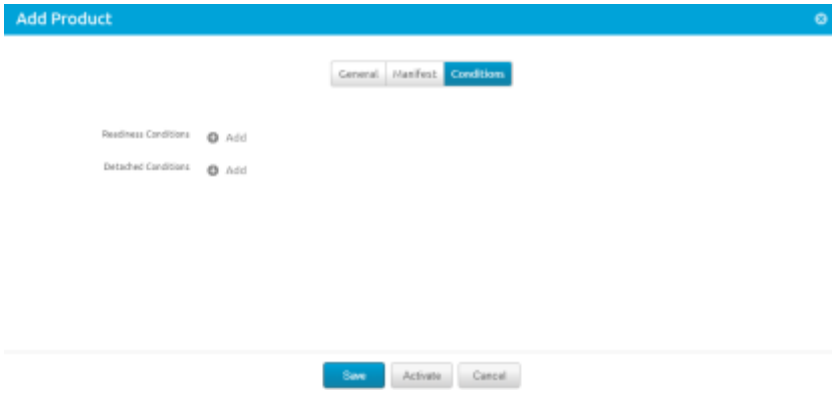
10. Select an **Action Type**:

    - **Install Profile**.
        - Select whether you want the Profile to be **Persistent through enterprise reset** or not. For more information, see the Product Persistence section.
    - **Uninstall Profile**.
    - **Install Applications** (Android devices only).
        - Select whether you want the Application to be **Persistent through enterprise reset** or not. For more information, see the Product Persistence section.
    - **Uninstall Applications** (Android devices only).
    - **Install Files/Actions**. This option runs the Install Manifest.
        - Select whether you want the File/Action to be **Persistent through enterprise reset** or not. For more information, see the Product Persistence section.
    - **Uninstall Files/Actions**. This option runs the Uninstall Manifest.
    - **Warm Boot** (Windows Mobile devices only).
    - **Reboot** (Android devices only).
    - **Cold Boot** (Windows Mobile devices only).

    After choosing an action type, choose the corresponding file/action, profile, or application from the drop-down menu and select **Save** to add it to the manifest.

    Note: Profiles and files/actions that were selected to persist through an Enterprise Reset are stored in the flash memory of the device upon install. Once a device initiates the restore process from an Enterprise Reset and installs the AirWatch MDM Agent, any persisted files/actions will be restored after Profiles are installed. For more information, see the Product Persistence section.

11. Add additional **Manifest** items if desired.
12. You can adjust the order of manifest steps using the up and down arrows in the Manifest list view. You may also edit or delete a manifest step.
13. Select the **Conditions** tab if you want to use conditions with your product. These conditions are optional and are not required to create and use a product.



14. Select **Add** to add either **Readiness Conditions**, **Detached Conditions**, or both.
    - A **Readiness Condition** determines when a products should be downloaded.
    - A **Detached Condition** determines when a products should be installed on a device.
15. Select the **Deployment** tab if you want to control the time and date that products are activated and deactivated. This tab is optional and is not required to create and use a product.
    - An **Activation Date** determines the time when a product automatically activates for device job processing. If the activation date is defined and the product is saved, the product will is stay inactive until the activation date is met according to the AirWatch server time. The policy engine will wake up and automatically activate the product. Products with activation dates can be manually activated beforehand. Manually activating a product will override the activation date.
    - A **Deactivation Date** determines the time when a product automatically deactivates from current and new device job processing. If the deactivation date is defined and the product is saved and currently active, it will stay active until the deactivation date is met according to the AirWatch server time. The policy engine will wake up and automatically deactivate the product.

    Note: A deactivation date cannot be set earlier than the activation date.
16. Choose to deploy the product immediately by selecting **Activate** or wait to deploy later and select **Save**.

Note: In order to edit a product, the product must be deactivated in the list view first.

Product Persistence

Product Provisioning allows you to enable profiles, files/actions, and applications to remain on a device following an enterprise reset. Content marked to persist following an enterprise reset reinstalls following the device restart after the agent installs.

Note: Product Persistence only applies to Motorola devices running Windows Mobile or Android.

Persistence works as follows:

1. A device must contain a staging configuration so that the agent and enrollment reinstall following the enterprise reset.

   Note: Staging configurations automatically persist on a device.

2. Set to persist any profiles, files/actions, and/or apps that you want to remain on the device after the enterprise reset.
3. The device will be reset when the enterprise reset command is sent (see Product Management). After resetting, the restore process starts.
4. The AirWatch MDM Agent for the device reinstalls during the restore process.
5. After the agent is installed, any persisted profiles, such as Wi-Fi, will be reinstalled.
6. Finally any persisted files/actions and/or apps will be reinstalled.

Product Persistance is ideal for help-desk type support as it allows the device to be wiped to clear away any problems without needing the device to be renrolled and products provisioned again.

Note: Enterprise Reset is only availble to Motorola devices.

Product Management

Overview

Product Provisioning offers some alternative methods to manage products and devices. The majority of the management tools are explained in the **AirWatch Windows Mobile Platform Guide** and the **AirWatch Android Platform Guide**. The tools listed in this section are meant to be used in addition to those covered in the guides.

In This Section

- Using the Product List View – Details how to use the Product List View to see what products are active and view the devices the product is provisioning.
- Using the Device Details View – Shows how the Device Detail View lists the products, files/actions, apps, and profiles a device has provisioned to it.
- Using Enterprise Reset – Explains how to perform an enterprise reset to wipe a device and allow profiles, files/actions, and applications that are set to persist to remain on the device.
- Using Remote Management for Android – Details how to use remote management to gain access to an end-user's Android device for troubleshooting.
- Using Remote Management for Windows Mobile – Details how to use remote management to gain access to an end-user's Windows Mobile device for troubleshooting.

Using the Product List View

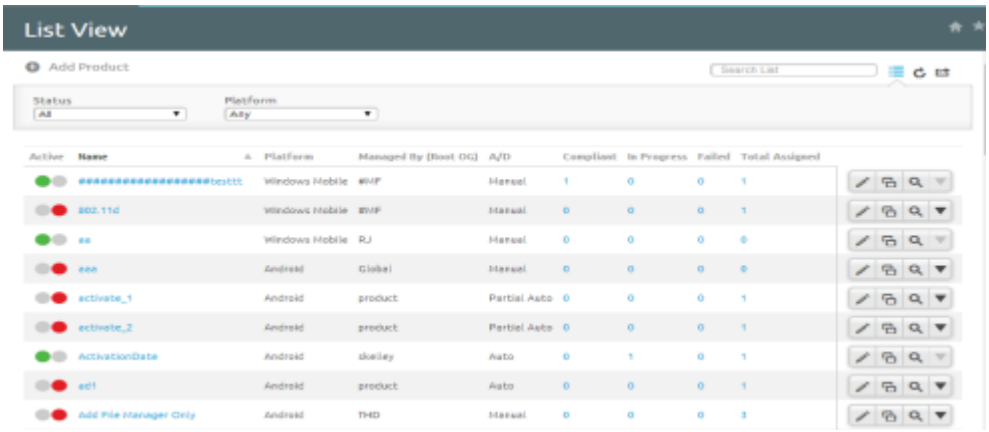
The Product List view allows you to view, edit, copy, and delete products as well as view the devices a product is provisioning.

Editing Products

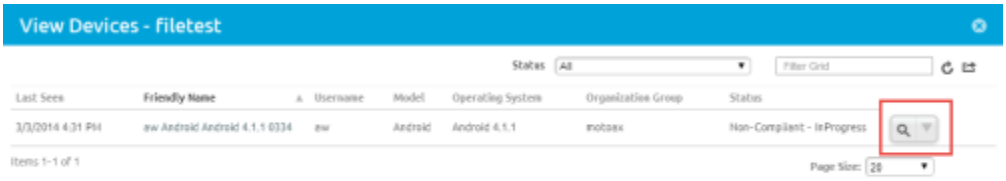
By selecting **Edit** , you can edit a product. You can only edit products after they are deactivated first. **Edit** brings up the Product Wizard allowing you to change any part of a product.

Product List View

Navigate to **Devices ▶Product (New) ▶List View ▶View Devices**. This opens the device listing of a specific product. The products can be sorted using the columns. Platform sorts by the device platform. Managed By sorts by the organization group the product is assigned to. A/D sorts by if the product uses activation/deactivation dates or manual. Compliant, In Progress, Failed, and Total Assigned sort by the status of the product on devices.



You can also use **View History** from the Device Details page to see the past and future products pushed to the device based on Product sync.



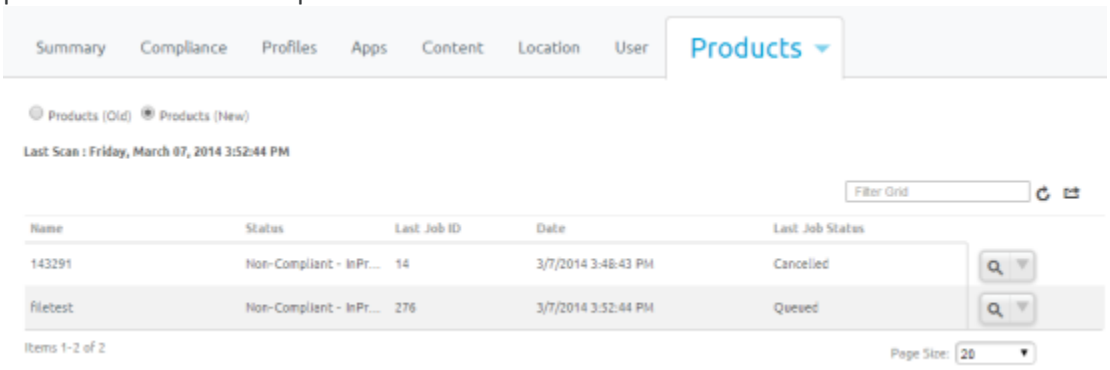
The **Log** listing shows the actions taken by the AirWatch Admin Console to keep the product and device in sync.

Using the Device Details View

You can use the Device Details View to see the products, files/actions, apps, and profiles pushed to a device.

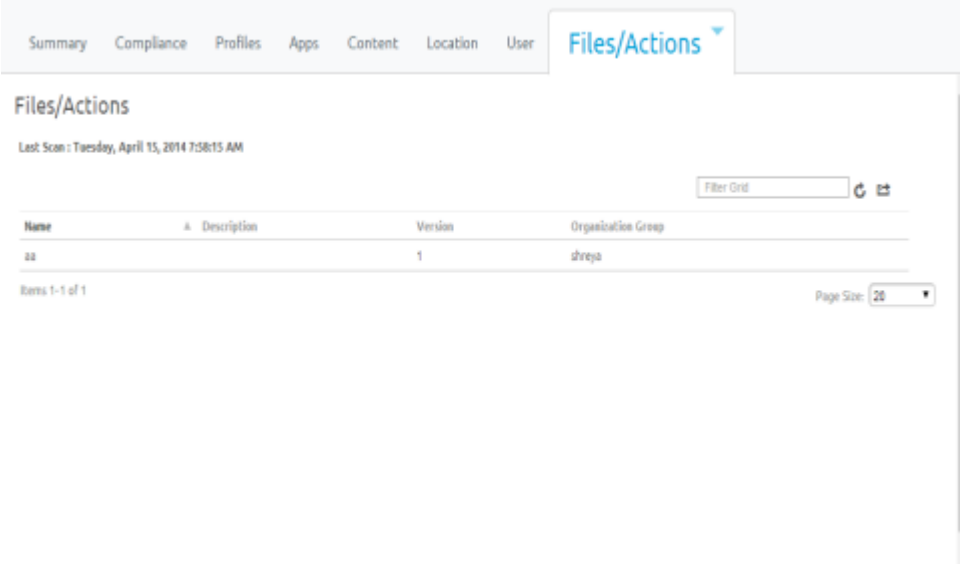
Products

To view the products on a device, navigate to **Devices ▶Details View ▶Additional Options ▶Products**. This displays the products available on a specific device.



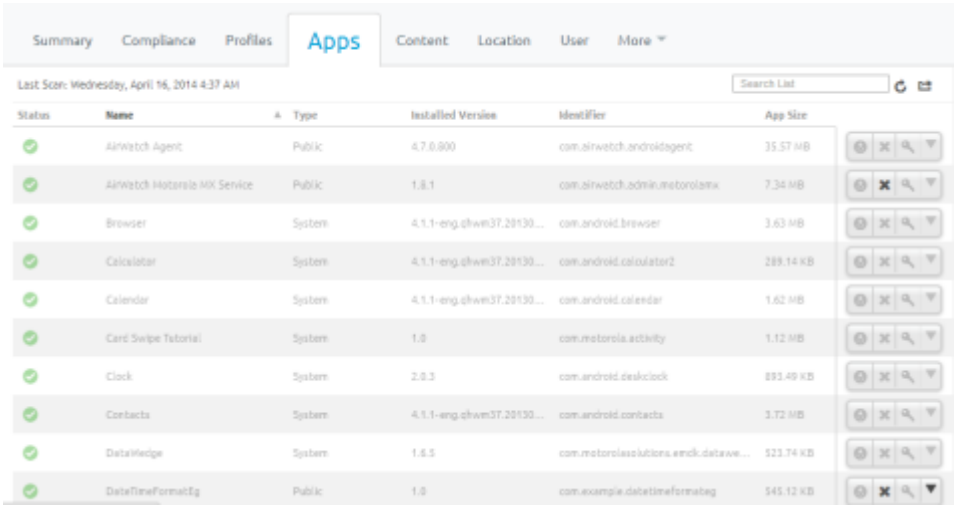
Files/Actions

Navigate to **Devices ▶Details View ▶Additional Options ▶Files/Actions** to access the files/actions on the device.



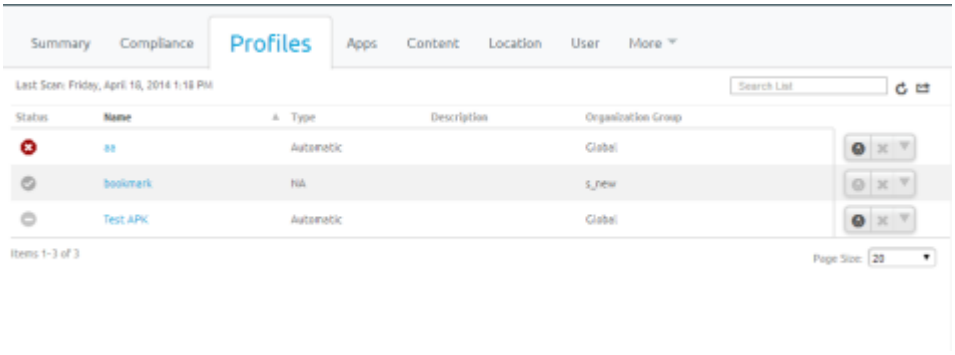
Applications

Navigate to **Devices ▶Details View ▶Apps** to access the Applications on the device.



Profiles

Navigate to **Devices ▶Details View ▶Additional Options ▶Profiles** to access the Profiles on the device.

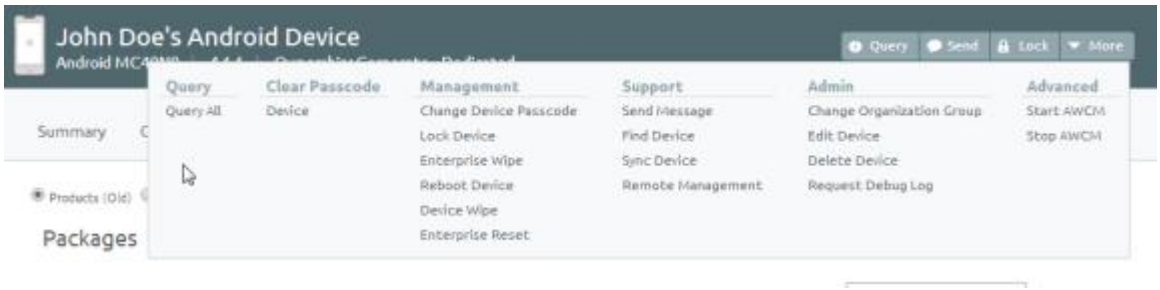


Using Enterprise Reset

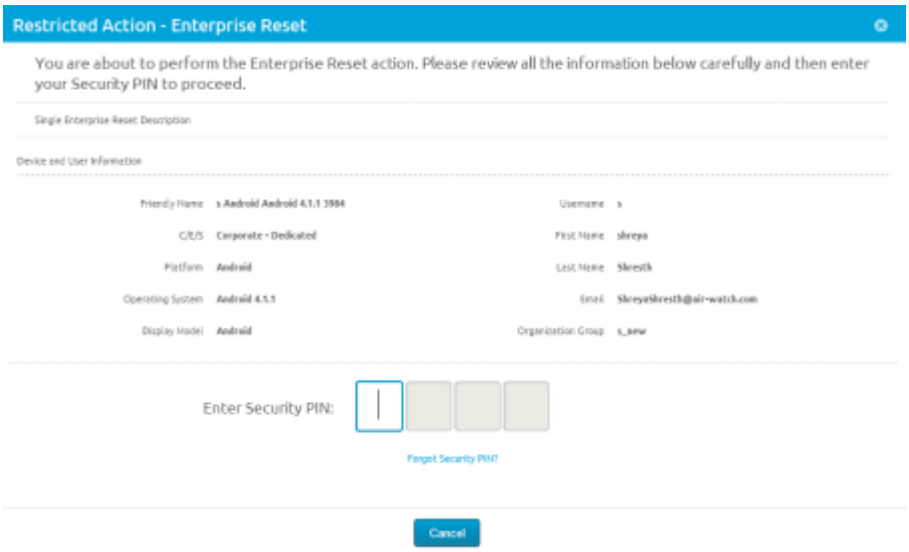
Enterprise Reset allows you to reset a device similar to an Enterprise Wipe, but with one important difference. Profiles and files/actions set to persist on a device are not removed and reinstall on a device following the first reboot after an Enterprise Reset.

To use an Enterprise Reset, follow the steps detailed below:

1. Navigate to **Devices ►List View** and select a Windows Mobile or Android device you want to Enterprise Reset.
2. On the Device Details View, select the **More** option ( ) to bring up an expanded list of management options .



3. Select **Enterprise Reset**.
4. Enter your **Security Pin** in the **Restrict Action** prompt to perform the Enterprise Reset.



Using Remote Management for Android

Remote Management allows you to directly control a device for troubleshooting or to ensure a device is properly provisioned.

Prerequisites

- AirWatch OEM service – The AirWatch OEM Service (MX Service) must be installed and bound to the AirWatch MDM Agent with device admin privileges.

- Remote Management Service – The remote management service must be installed and bound. On-premise customers can refer to the Appendix – Remote Control Tunnel Server Installation.
- Java installed on admin computer in order to run the remote management applet.

Configuring Remote Management for Android devices.

To configure Remote Management, follow the steps detailed below:

1. Navigate to **Devices ▶Settings ▶Windows ▶Android ▶Agent Settings**.
2. Under the Remote Management section, complete the following settings related to the use of Remote Management:
    - **Enable Encryption** to encrypt the data using AES 128 bit encryption.
    - Enter a **Passphrase** for the encryption.
    - Set the **Device Log Level** to control the verbosity of the remote control application on the device.
    - Define the **Log Folder Path** where the application will save the remote control log file on the device.
    - Enable **Display Tray Icon** to show the remote management applet on the device.
    - Enable **Seek Permission** if you want to prompt the end user to accept/decline the remote management request from the admin.
        - Enter a **Seek Permission Message** that the end user will see when a remote request is sent.
        - Enter the **Yes Caption** message for the accept button the end user will see on the Seek Permission request.
        - Enter the **No Caption** message for the decline button the end user will see on the Seek Permission request.
    - Enter the **Max Sessions** allowed through Remote Management.
    - Enter the **Remote Management Port** used to communicate between the applet and the device.
    - Select the **Mode** to define how the remote management applet and the device communicate over the network:
        - **Off** – Communication happens directly between the applet and the device. This mode is used when the computer that has the applet and the device you want to remotely manage are on the same network or virtual network.
        - **Inbound** – Communication flows from the applet to the device. There is no direct connection available between the applet and the device. The applet initiates a connection with the tunnel server and the tunnel server reaches out to the tunnel agent on the device to establish a connection. It is assumed that the device and the tunnel server are on the same network.
            - **Tunnel Agent Port** – The port used for communication from the applet to the device.
        - **Outbound** – Communication flows from the device to the applet. There is no direct connection available between the applet and the device. The applet and the device both establish connections with the tunnel server proactively. This is most likely used when the device and the tunnel server are on different networks and the device can connect to the tunnel server on a public IP. An example of outbound connections would be a device out in the field and the applet and tunnel are located in a central location.
            - **Number of Retries** – The number of retries allowed before communication attempts stop.
            - **Heart Beat Interval (Seconds)** – The amount of time (in seconds) that passes between status updates are sent from the device.

- **Connection Loss Retry Frequency (Seconds)** – The amount of time (in seconds) that passes between attempts to reestablish connection.
- **Retry Frequency (Seconds)** – The amount of time between attempts to communicate.

3. If you are using Inbound or Outbound mode, a tunnel server configuration must be defined. Navigate to **Settings ► Systems ►Advanced ►Site URLs**.

Note: The tunnel server must be installed on a server that both the remote control applet and the device can communicate with. The following step is only required if you are using Outbound or Inbound modes.

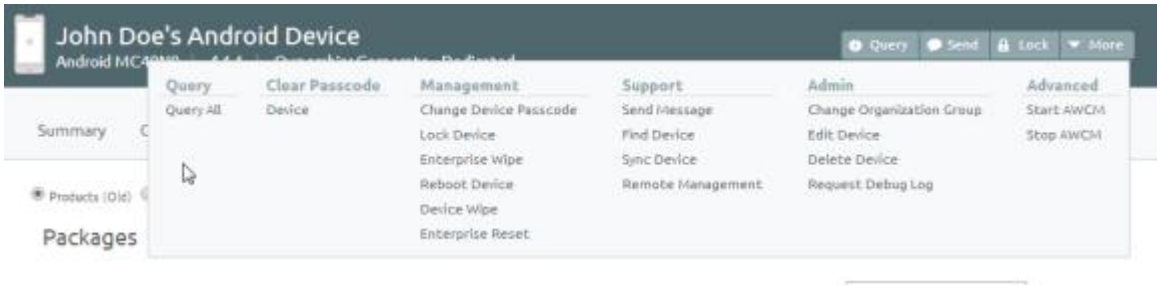
4. Under the Remote Management section, complete the following settings related to the use of Remote Management:
   - **Enable Tunnel Server** to allow the applet and the device to communicate with each other.
   - Enter the **Tunnel Server External URL** that communicates with device. The url should be a public facing url. This is used only for outbound communication where the device needs to reach out to the tunnel server on a public IP.
   - Enter the **Tunnel Server External Port** that communicates with device. The default port value is 7779. This is used only for outbound communication where the device needs to reach out to the tunnel server on a public IP.
   - Enter the **Tunnel Server Internal URL** that communicates with the computer the applet is running on. The url can be internal or external depending on the network(s) the applet and the tunnel servers are on. This is used for both inbound and outbound modes since the applet establishes the connection with the tunnel server in both cases.
   - Enter the **Tunnel Server Internal Port**that communicates with the computer the applet is running on.The default value of the port is 7778. This is used for both inbound and outbound modes since the applet establishes the connection with the tunnel server in both cases.
5. After configuring the agent settings and the optional tunnel server, download the Remote Control .apk from the Resources portal.
6. Deploy the .apk as an internal application sent to devices through AirWatch. For more information on deploying internal applications, please see the **AirWatch Mobile Application Management Guide**.

Using Remote Management

To use Remote Management, follow the steps detailed below:

1. Navigate to **Devices ►List View** and select a Windows Mobile or Android device you want to manage.
2. On the **Device Details View**, select the **More** option ( ) .

   to bring up an expanded list of management options.



3. Select **Remote Management**. A new window opens listing the device details as well as showing the Remote Management window.

4. Use the Remote Management window to accomplish the tasks you want:

   - **Connection** – Shows the connection settings used to communicate with the device.
   - **Retry** – Attempts to make a connection with the device again.
   - **Screen Share** – Displays the device's screen so you can remotely view and control the device screen as well as receive any actions performed by the device user. The **Screen Share** toolbar also allows you to adjust the view depth and size. The toolbar also allows you to take and save screenshots and videos to your computer. You can also record sequence of actions macros that can be used again later.

   - **File Manager** – Displays the device file system so you can remotely manage by viewing, creating, copying, moving, and deleting files and directories. Files and directories can also be dragged and dropped between the file system on the device and the remote control admin's machine.
   - **Task Manager** – Displays a list of the processes currently running on the device. Select a process from the list to stop or kill the process. You can also start an executable on the device by entering the full path and any parameters to be passed.
   - **Applications** – Displays a list of any applications currently managed by the OS on the device. Uninstall a managed application by right-clicking on an entry. You may also load a managed application from your local machine and install it to a specified directory on a device.
   - **Command Prompt** – Displays the command prompt. For a full list of supported commands and details, type 'help' into the command prompt and press enter.
   - **Applet Display and Log Settings** – Allows you to configure and save the settings for the remote management applet as well as the log.
   - **About** – Displays the version information about the remote management applet in use.

5. When finished with your remote management session, select the cancel button located below the applet window to close the connection.

Using Remote Management for Windows Mobile

Remote Management allows you to directly control a device for troubleshooting or to ensure a device is properly provisioned.

Prerequisites

- The remote management service must be installed and bound. On-premise customers can refer to the Appendix – Remote Control Tunnel Server Installation.
- Java installed on admin computer in order to run the remote management applet.

Configuring Remote Management for Windows Mobile devices

To configure Remote Management, follow the steps detailed below:

1. Navigate to **Devices ▶Settings ▶Windows ▶Windows Mobile ▶Agent Settings**.
2. Under the Remote Management section, complete the following settings related to the use of Remote Management:
    - **Enable Encryption** to encrypt the data using AES 128 bit encryption.
    - Enter a **Passphrase** for the encryption.
    - Set the **Device Log Level** to control the verbosity of the remote control application on the device.
    - Define the **Log Folder Path** where the application will save the remote control log file on the device.
    - Enable **Seek Permission** if you want to prompt the end user to accept/decline the remote management request from the admin.
        - Enter a **Seek Permission Message** that the end user will see when a remote request is sent.
        - Enter the **Yes Caption** message for the accept button the end user will see on the Seek Permission request.
        - Enter the **No Caption** message for the decline button the end user will see on the Seek Permission request.
    - Enter the **Max Sessions** allowed through Remote Management.
    - Enter the **Remote Management Port** used to communicate between the applet and the device.
    - Select the **Mode** to define how the remote management applet and the device communicate over the network:
        - **Off** – Communication happens directly between the applet and the device. This mode is used when the computer that has the applet and the device you want to remotely manage are on the same network or virtual network.
        - **Inbound** – Communication flows from the applet to the device. There is no direct connection available between the applet and the device. The applet initiates a connection with the tunnel server and the tunnel server reaches out to the tunnel agent on the device to establish a connection. It is assumed that the device and the tunnel server are on the same network.
            - **Tunnel Agent Port** – The port used for communication from the applet to the device.
        - **Outbound** – Communication flows from the device to the applet. There is no direct connection available between the applet and the device. The applet and the device both establish connections with the tunnel server proactively. This is most likely used when the device and the tunnel server are on different networks and the device can connect to the tunnel server on a public IP. An example of outbound connections would be a device out in the field and the applet and tunnel are located in a central location.
            - **Number of Retries** – The number of retries allowed before communication attempts stop.
            - **Heart Beat Interval (Seconds)** – The amount of time (in seconds) that passes between status updates are sent from the device.
            - **Connection Loss Retry Frequency (Seconds)** – The amount of time (in seconds) that passes between attempts to reestablish connection.
            - **Retry Frequency (Seconds)** – The amount of time between attempts to communicate.
3. If you are using Inbound or Outbound mode, a tunnel server configuration must be defined. Navigate to **Settings ▶ Systems ▶Advanced ▶Site URLs**.

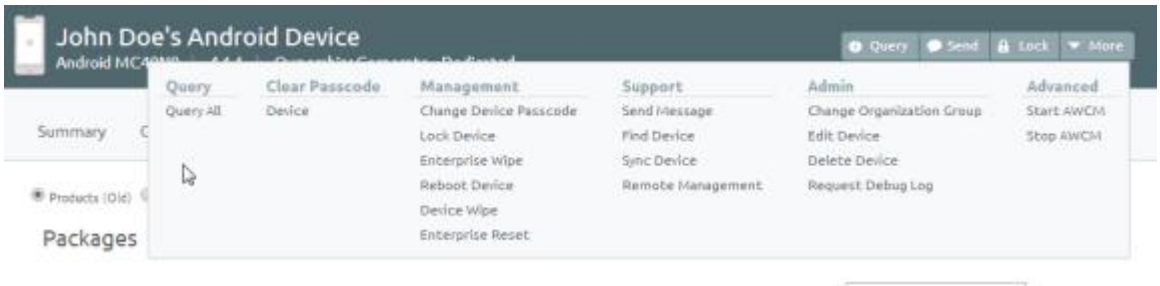
Note: The tunnel server must be installed on a server that both the remote control applet and the device can communicate with. The following step is only required if you are using Outbound or Inbound modes.

4. Under the Remote Management section, complete the following settings:
   - **Enable Tunnel Server** to allow the applet and the device to communicate with each other.
   - Enter the **Tunnel Server External URL** that communicates with device. The url should be a public facing url. This is used only for outbound communication where the device needs to reach out to the tunnel server on a public IP.
   - Enter the **Tunnel Server External Port** that communicates with device. The default port value is 7779. This is used only for outbound communication where the device needs to reach out to the tunnel server on a public IP.
   - Enter the **Tunnel Server Internal URL** that communicates with the computer the applet is running on. The url can be internal or external depending on the network(s) the applet and the tunnel servers are on. This is used for both inbound and outbound modes since the applet establishes the connection with the tunnel server in both cases.
   - Enter the **Tunnel Server Internal Port**that communicates with the computer the applet is running on.The default value of the port is 7778. This is used for both inbound and outbound modes since the applet establishes the connection with the tunnel server in both cases.
5. After configuring the agent settings and the optional tunnel server, download the Remote Control Cab from the Agent Settings page.
6. Using product provisioning, upload the Remote Control Cab file as a file/action and either:
   - Add it to a staging configuration as an Install file/action manifest item.
   - Add it to a product as an Install file/action manifest item.
7. Stage the device or push the product to download and install the application onto the device.

Using Remote Management

To use Remote Management, follow the steps detailed below:

1. Navigate to **Devices ►List View** and select a Windows Mobile or Android device you want to manage.
2. On the **Device Details View**, select the **More** option ( ) to bring up an expanded list of management options.



3. Select **Remote Management**. A new window opens listing the device details as well as showing the Remote Management window.

4. Use the Remote Management window to accomplish the tasks you want:

   - **Connection** – Shows the connection settings used to communicate with the device.
   - **Retry** – Attempts to make a connection with the device again.
   - **Screen Share** – Displays the device's screen so you can remotely view and control the device screen as well as receive any actions performed by the device user. The **Screen Share** toolbar also allows you to adjust the view depth and size. The toolbar also allows you to take and save screenshots and videos to your computer. You can also record sequence of actions macros that can be used later.

   - **Registry Manager** – Displays the device registry so you can remotely manage by viewing, creating, editing, and deleting registry keys and values.
   - **File Manager** – Displays the device file system so you can remotely manage by viewing, creating, copying, moving, and deleting files and directories. Files and directories can also be dragged and dropped between the file system on the device and the remote control admin's machine.
   - **Task Manager** – Displays a list of the processes currently running on the device. Select a process from the list to stop or kill the process. You can also start an executable on the device by entering the full path and any parameters to be passed.
   - **Registered DLL List** – Displays a list of the registered .DLLs on the device. Select a .DLL to unregistered by right-clicking. Load and register a .DLL from your local machine to a directory on the device.
   - **Device Info** – Displays the information about the device from the remote management applet instead of returning to the Dashboard.
   - **Applications** – Displays a list of any applications currently managed by the OS on the device. Uninstall a managed application by right-clicking on an entry. You may also load a managed application from your local machine and install it to a specified directory on a device.
   - **Display/Volume Settings** – Displays the configurable display and volume settings on the device.
   - **Send Message** – Displays a prompt for a message to be entered that displays on the device-user's screen.
   - **Command Prompt** – Displays the DOS-like command prompt. For a full list of supported commands and details, type 'help' into the command prompt and press enter.
   - **Applet Display and Log Settings** – Allows you to configure and save the settings for the remote management applet as well as the log.
   - **About** – Displays the version information about the remote management applet in use.

5. When finished with your remote management session, select the cancel button located below the applet window to close the connection.

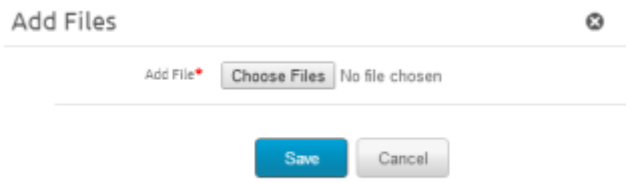
Appendix A – Android OS Upgrade

This appendix explains the additional steps required for using the OS Upgrade action of the files/actions wizard of Product Provisioning. OS Upgrade action is only available for Android devices. Windows Mobile devices use the Install action. For more information on upgrading Windows Mobile devices, see Appendix D – Windows Mobile OS Upgrade.

Creating OS Upgrade Product

The following steps explain how to create an OS upgrade File/Action for a product.

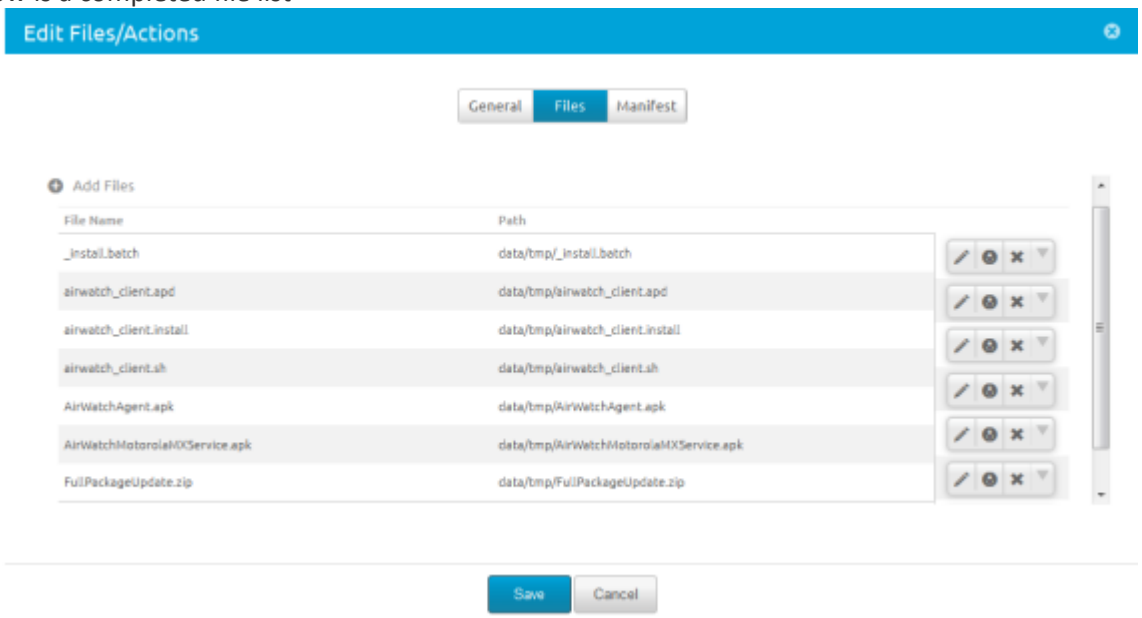
1. Navigate to **Devices ▶Product (New) ▶Files/Actions ▶Add**.
2. Select **Android** as your device platform.
3. Complete the General fields.
   - Enter a **Name**.
   - Enter a **Description**.
   - View the **Version** automated by AirWatch.
   - Enter who the files/actions are **Managed By**.
4. Select the **Files** tab.
5. Select the **Add Files** button.



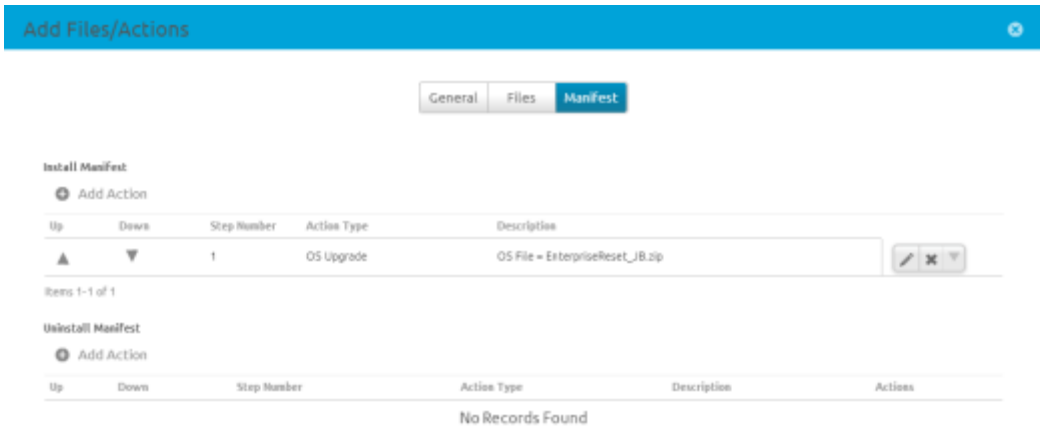
6. Upload the following files and specify the path as: data/tmp/
   - install.batch
   - airwatch_client.apd
   - airwatch_client.install
   - airwatch_client.sh
   - AirWatchAgent.apk
     - Rename the .apk to AirWatchAgent.apk as the above script files are referencing this name specifically.
     - Use the same Agent .apk that is currently installed on your device.
     - AirWatch can provide this apk.
   - AirWatchMotorolaMXService.apk
     - It must be renamed as the above script files are referencing this name specifically

- This should ideally be the same Agent apk that is currently installed on your device.
- AirWatch can provide this apk.

- OS Update zip file
  - This can be a major or minor OS upgrade file.
  - This can also be an enterprise reset package.
- Below is a completed file list



7. Select the **Manifest** tab and select **Add Action** under the **Install Manifest**.
8. Add OS Upgrade command to the manifest and select the corresponding OS upgrade file that was uploaded earlier. Your Manifest should look similar to below:



9. Select **Save**.

Deploy OS Upgrade Through a Product

The following steps cover how to deploy an OS Upgrade File/Action as part of a Product.

1. Navigate to **Devices ▶Product (New) ▶List View ▶Add Product**.
2. Select **Android** as your device platform.
3. Complete the **General** fields.
    - Enter a **Name**.
    - Enter a **Description**.
    - View the **Version** automated by AirWatch.
    - Enter who the files/actions are **Managed By**.
4. In the **Assigned Smart Groups** for assignment, add the smart group of devices that should be receiving the upgrade. You can view the list of assigned devices by selecting **View Device Assignment**.
5. Select the **Manifest** tab and select **Add**.
6. Choose **Install Files/Actions** as your **Action Type**.
7. Enter the name of the OS Upgrade file/action you created.
8. Select **Save**.
9. Select **Activate** to push to the devices assigned in the smart group.

From the List View for products you will be able to see the status of the devices receiving the Product. The In-Progress field will update showing when the products have been pushed

Deploying OS Update on the Device

This section covers the process taken by the device when the OS Update is pushed to it.

1. Device receives the product which you can verify in **Agent ▶Products** tab.
2. Download all the files including the OS update zip which you can verify in the Product logs found in **Agent ▶Product ▶Product Name**.
3. Once the downloads complete, the AirWatch MDM Agent backs up its data and any installed managed applications to the device Enterprise folder which is persistent.
4. The agent then fires the intent to start the OS update by passing the OS Upgrade zip file.
5. Device then applies the OS upgrade.
6. Once complete, the device reboots.
7. Upon reboot, the Rapid Deployment client re-installs the agent applications and launch them.
8. Upon launch, the agent restores its data and re-installs managed apps.

Appendix B – XML Provisioning

Overview

The 5.X version of the AirWatch MDM Agent for Windows Mobile now supports XML provisioning. XML provisioning allows a client admin to take a custom designed XML file and download it to a device in a provisioning product. After the file is downloaded, it will execute an install command to extract the settings from the XML file and install them on the device. This feature provides added flexibility in terms of implementing custom configuration options for Windows Mobile devices which are currently enrolled in AirWatch.

Note: XML Provisioning is for Windows Mobile devices only and not Windows CE.

Creating an XML Product

1. Navigate to **Devices ▶Product (New) ▶File/Actions ▶Add**.
2. Select **Windows Mobile**.
3. Enter in the required settings on the **General** tab then select the **Files** tab and upload the desired XML file and enter in the destination path on the device .
4. Select the **Manifest** tab and **Add** an **Install Action** for the XML file.
5. Select **Save**.
6. Navigate to **Devices ▶Product (New) ▶List View ▶Add**.
7. Select **Windows Mobile.**
8. Enter the **General** information.
9. Select the **Manifest** tab.
10. Select **Install Files/Actions** and choose the files and actions just created.
11. **Save** and **Activate** the product.

The product downloads to all assigned devices and the XML file should be successfully installed.

Note: The old product provisioning should also work with XML provisioning as long as the 5.X agent is installed on the device.

Below is a sample of an XML file which updates a registry setting that can be installed on a device through product provisioning.

```
<?xml version="1.0"?>
-<wap-provisioningdoc>
  -<characteristic type="Registry">
    -<characteristic type="HKLM\Software\AirWatch\Test">
      <parm datatype="integer" value="5" name="TestValue"/>
      <parm datatype="boolean" value="1"name="TestValueBoolean"/>
```

```
      </characteristic>
   </characteristic>
</wap-provisioningdoc>
```

Appendix C – Pull Relay Server Configuration

Overview

When using relay servers for staging or for product provisioning, you can configure the server to be either a push or pull server. These severs are designed to be located at each store/distribution center/corporate location where devices will be used. When content is published, it is pushed to the push relay servers at that time which then provides the profiles, file/actions, and/or applications for the devices to download. Push servers are ideal for an On-Premise locations as the server and the devices will be on the same network as the AirWatch Admin Console. Pull servers work in a different manner.

With a pull server, when content is published, it is not distributed to the servers but is pulled when the Pull service connects and checks for new or changed content.

A pull relay server will periodically contact the AirWatch Admin Console to check for new products, profiles, files/actions, and/or applications have been assigned to devices under the pull relay servers purview. If changes or additions have been made, the server will create an outbound connection to the AirWatch Admin Console to download the new content to the server before pushing it to its devices. Pull service is best used when traversing any NAT firewall or SaaS to On-Premise hybrid environments as SaaS customers typically do not want the service to tie-up bandwidth as content is delivered from AirWatch to the store server.

To create a pull relay server, you must first have an FTP or FTPS server to function as the relay server. The instructions below detail how to create a pull relay server from an FTP or FTPS server.

Note: The ports you configured when you create your FTP(S) server must be the same ports you enter when creating a relay server in the AirWatch Admin Console.

In This Section

Prerequisites

- An FTP or FTPS server
- .NET must be installed on Windows-based servers.
- Java must be installed on Linux-based servers.
- The relay server requires network access from where it is installed (in-store, distribution center, etc.) to the AirWatch SaaS environment.
- Each server requires disk storage of 2 MB for the pull server installer as well as storage space for all the content that will be pulled to the server.

Windows-Based Pull Relay Server

To create a windows-based pull relay server, follow the steps detailed below:

1. Using your preferred server management system, download the .exe and an xml config file onto the server.
2. Open the xml config file and edit the following information:

```
<PullConfiguration>
   <libraryPath>C:\AirWatch\PullService\</libraryPath>
   <endPointAddress>https://[endpoint URL]/contentpull /</endPointAddress>
</PullConfiguration>
```

3. Run the WindowsPullServiceInstaller.exe.

   Note: .NET will be installed before the MSI is extracted.

4. Follow the instructions prompted by the installer.

If you are using the silent install from the command prompt, use the following commands:

1. WindowsPullServiceInstaller.exe" /s /v"/qn
2. To include log: "WindowsPullServiceInstaller.exe" /s /v"/qn /l WindowsPullServiceInstaller.txt"

The installer looks for the PullserviceInstaller.config file in the installer execution directory. If the file is missing, the installer prompts you to let you know the file is missing.

Note: This guide covers the installation of one server at a time. For bulk installation, you must use a third-party application. AirWatch supports importing servers in bulk through the Bulk Import option.

Linux-Based Pull Relay Server

To create a Linux-based pull relay server, follow the steps detailed below:

1. Download the .bin and an xml config file onto the server using your preferred server management system.
2. Open the xml config file and edit the following information:

```
<PullConfiguration>
   <libraryPath>C:\AirWatch\PullService\</libraryPath>
   <endPointAddress>https://[endpoint URL]/contentpull /</endPointAddress>
</PullConfiguration>
```

1. In the command prompt, enter:

```
sudo ./LinuxCentOSPullServerInstaller.bin
```

2. Enter the following command to silently install:

```
sudo ./LinuxCentOSPullServerInstaller.bin -I silent
```

The installer looks for the PullserviceInstaller.config file in the installer execution directory. If the file is missing, the installer prompts you to let you know the file is missing.

Note: This guide covers the installation of one server at a time. For bulk installation, you must use a third-party application. AirWatch supports importing servers in bulk through the Bulk Import option.

Appendix D – Windows Mobile OS Upgrade

This appendix explains how to push an OS Upgrade to Windows Mobile-based Motorola devices using product provisioning.

All OS upgrade files are restricted access files and require a valid user account with Motorola to log in with as well as a valid device serial number. The OS upgrade files are specific to both device model and OS version. The AirWatch OS upgrade process uses the .APF files and requires the administrator to first install the 'MSP Package Builder' utility.This utility is required to extract the contents of the .APF file into individual components that will then be pushed to the device and used to upgrade the OS.

In This Section

- Prerequisites – Lists the requirements for using the OS Upgrade process for Windows Mobile-based Motorola devices.
- Updating Device Registry Settings – Details the changes to the device registry needed to allow the OS Upgrade process to work.
- Extracting Required OS Update Files – Explains how to use the MSP Package Builder utility to extract the OS Upgrade files to be used in a product provisioned to devices.
- Creating a Product for OS Upgrade – Details the steps required in creating a product that will push required files to the device as well as initiate the OS Upgrade.

Prerequisites

In order to use the Windows Mobile OS Upgrade, you must have the following:

- The MSP Package Builider utility installed on your computer.
- A Motorola user account to download the OS Upgrade.
- The serial number for the device you want to upgrade.
- The OS update utility which should be included with the extracted .APF files.

  This can be downloaded from Motorola (https://portal.motorolasolutions.com/Support/US-EN).
- The registry file to install on the target device to update registry settings for OS update to run correctly.

Step 1 – Updating Device Registry Settings

Before your device can receive the OS Upgrade product, you must edit the device registry to allow the process to run. The 5.x agent has been designed to update these settings based on the AirWatch Admin Console configuration settings. However, you should check these settings before hand and confirm they are set correctly. If not, you can update these settings manually or can export a registry file containing these settings and download and install them on a device using files/actions via Product Provisioning as mentioned in Appendix B – XML Provisioning.

The following registry entries are required:

HKEY_LOCAL_MACHINE\SOFTWARE\AIRBEAM\	
IGNORESERVER	string “1”
SERVERRIP	string [Enter the Server URL or IP Address]
FTPUSER	string [Enter the FTP(s) username]
FTPPASSWORD	string [Enter the FTP(s) password] - THIS FIELD MUST BE ENCRYPTED
TFTP	string "0"
PASSIVEMODE	string "1"
FTPPORT	string "21"
FTPS	string "0" or "1"
VERIFYSERVER	string "0"
SOFTKEY1	double "123"
SOFTKEY2	double 124"
ENTERKEY	double "13"

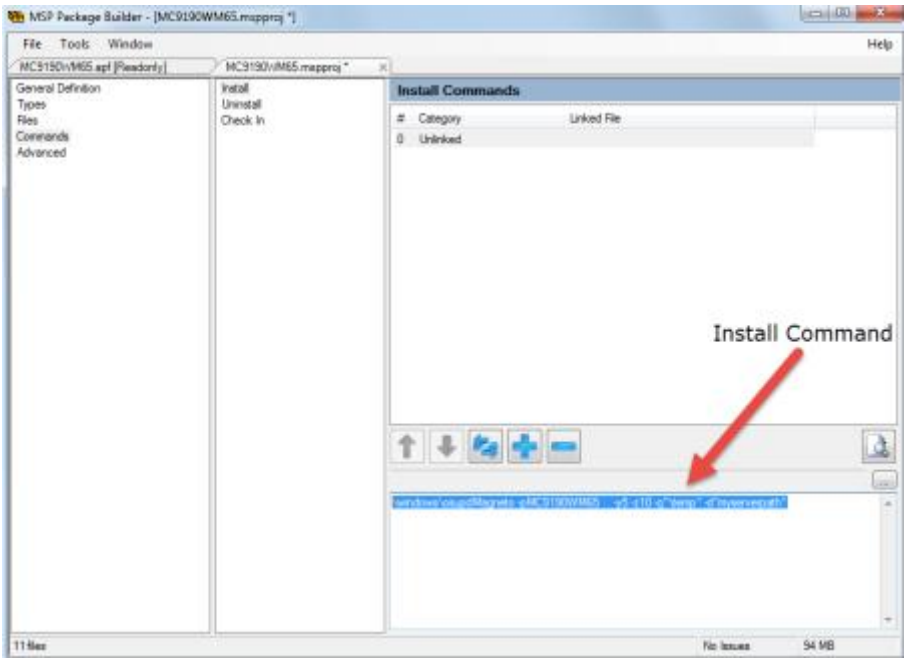
Step 2 – Extracting Required OS Update Files

Once you edit the device registry, you must extract the OS update files from the Motorola Upgrade Files.

To extract the OS update files, follow the steps detailed below:

1. Launch the MSP Package Builter Utility.
2. Navigate to **File ▶Open Project** and select the appropriate .APF file and open it in MSP Package Builder.
3. Navigate to **Tools ▶Convert Project** to open the **Convert to Project** dialog.
4. Complete the following fields:
   - **Name** – Enter the name for the OS Upgrade project.
   - **Extract FilesTo** – Enter the location the files should be saved to or select the Browse button to select the location.
5. Select **OK** to close the Convert to Project dialog.
6. After closing the dialog, select **Command** from the left window pane.
7. Select **Install** and copy the install command from the bottom right window pane.

Note: You may want to paste and save this install command to a file (notepad, word, etc.) so you can access this command while creating the File/Action for the OS Upgrade.



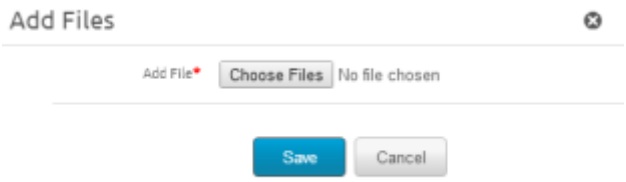
Step 3 – Creating a Product for OS Upgrade

Once you extract the OS upgrade files from the package, you are ready to great a product to push the files to devices.

Note: If you need to update the device registry using a product, create a separate files/actions for the Registry Edit file and list that Files/Action first in the product manifest.

To create a product for OS Upgrade, follow the steps detailed below:

1. Navigate to **Devices ▶Product (New) ▶File/Actions ▶Add**.
2. Select the **Windows Mobile** platform.
3. Complete the General fields:
   - Enter a **Name**.
   - Enter a **Description**.
   - The **Version** is automated.
   - Enter who the files/actions are **Managed By**.
4. Select the **Files** tab.
5. Select **Add Files**.



6. Select **Choose Files** and upload the extracted OS update files (.bin files, .sig files, .bmp files, etc.).

7. Specify the **Download Path** as:
   - \[directory]\[full file name]
8. Enable **Relay Server Only** if the device to be updated is a Windows Mobile device. For Windows CE devices, **do not** enable this option.
9. After uploading all the extracted OS update files, upload the OS Update Utility .exe as well as the "package.TXT" manifest file.

   Windows Mobile devices always use a file named "osupdMagneto.exe." For Windows CE devices, the file name varies with the device. CE devices also require a package.APD. The contents of this file are not important and can be a simple text format file containing something benign.
10. Specify the separate **Download Paths** for both the OS Update Utility and the package.TXT manifest file.

    The **Download Path** must be as formated: "\Windows\[directory]\[full file name]". If you are updating a Windows CE device, the .APD file will be placed in: "\Application\AirBeam\PKG."

    Note: Do not select **Relay Server Only** as these files need to be delivered directly to the device.
11. Once all the files have been uploded, select the **Manifest** tab and select **Add Action** to add the **Install Actions**.
12. Select the **Run** action in the **Action Types** field.
13. Copy the Install Command from the MSP Package Builder utility and paste it into the **Command Line and Arguments to Run**.

    Note: The syntax of the Install Command must be edited to match AirWatch syntax. The command syntax should look similar to this: ""\windows\osupdMagneto.exe" -p"MC9190WM65" -y" -z10 -q"\temp\" -d"\PFILES\MC9190OSUpdate_7."" Note that double quotes are used around the entire string. See Run Commnd Argument Information for more on the install command.
14. Select **Save**.
15. Navigate to **Devices ▶Product (New) ▶List View ▶Add**.
16. Select **Windows Mobile** as your platform.
17. Complete the General fields:
    - Enter a **Name**.
    - Enter a **Description**.
    - Enter who the Product will be **Managed By**.
    - Enter the **Assigned Smart Groups**. If a Smart Group is not available, one can be created by selecting the link.
18. Select the **Manifest** tab.
19. Select the **Add** button and select **Install Files/Actions**.
20. Add the files/actions created above. Repeat steps 20 and 21 for each files/actions created for the OS Upgrade.
21. Once all files/actions are assigned, select **Activate** to assign it to the devices.

Once the device receives the product, the OS Update process initiates. A notification screen displays when the upgrade is complete.

Run Command Argument Information

-d option describes the AirWatch relay server folder path containing files required to update the device.

-y option describes the maximum number of retrys while the –z option describes the retry delay.

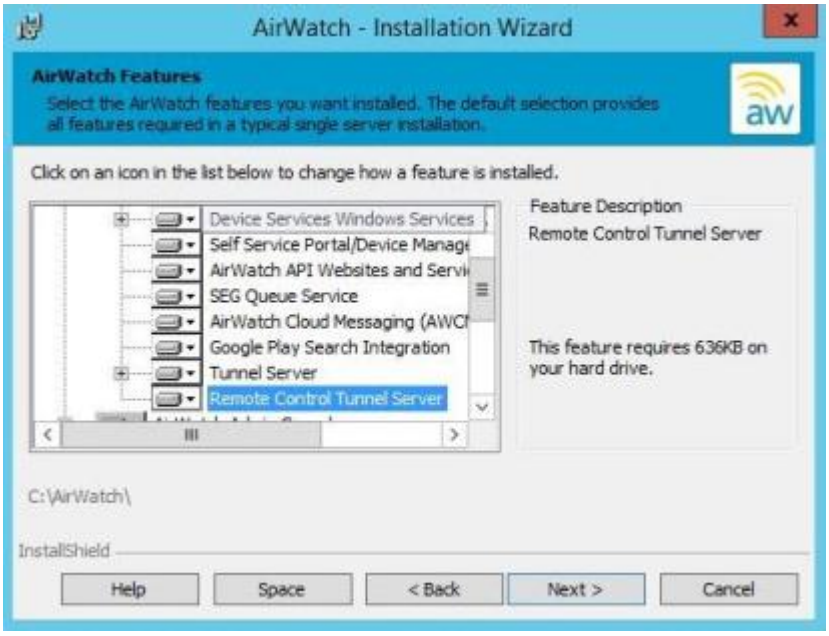
-q is the folder on the device that will be used to bring down the files from the FTP server when updating WM. This will typically be “\Temp” or “\Storage Card”. In CE updates, the files are brought down one at a time into device memory so the parameter is not used in that case.

-p is the OSUpdate project name. It is whatever the user wants it to be. This is usually named something to indicate what the update package contains. Remember this would also be the name of the Motorola .APF file. The project (APF) should be named in such a way as to be able to know the target device, OS type, etc.

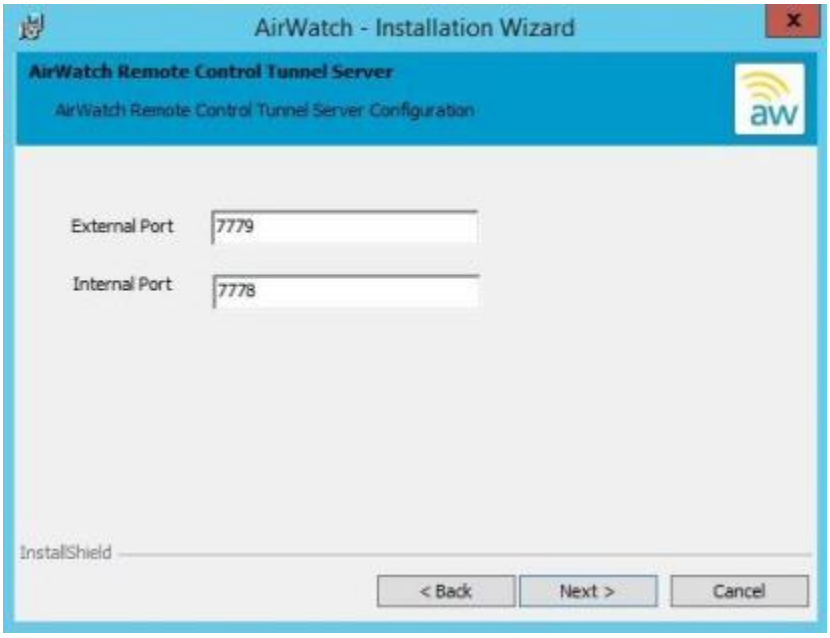
For the AW process, this name will be the same as whatever you call the dummy .APD file (which can be an empty text file). So if your command line contained –p”WT41N0”, you need to have a WT41N0.apd file put down to the device prior to doing the update. THIS IS ONLY NEEDED FOR CE devices. For WM devices –p”MC9190WM65” (as in the doc) is required by the command line parsing, but in effect will not be used.

Appendix E – Remote Control Tunnel Server Installation

The Remote Tunnel Server is an optional component that on-premise customers can install as part of their AirWatch installation to provide remote management capabilities to your Windows Mobile and rugged Android devices. If you did not enable this feature when you installed AirWatch then you will need to re-run the installer to enable it. You can do so by selecting the check box for Remote Control Tunnel Server under **AirWatch Device Services,** as shown below:



Continue through the installation process until you reach the following screen, where you can enter the internal and external ports that should be used to establish a secure connection between your Windows Mobile/rugged Android devices and the AirWatch Admin Console.



For additional information about the installation process, please refer to the **AirWatch Installation Guide**.

Appendix – Custom Attributes

Overview

Custom attributes enable administrators to extract particular values from a managed device and return it to the AirWatch Admin Console. Apply these attributes to other uses such as associating them with rules to further assign products to devices.

For the Windows Mobile platform the most common application will be through the syncing of registry settings, which allows administrators to specifically assign products to devices based on common registry settings.

Implementation

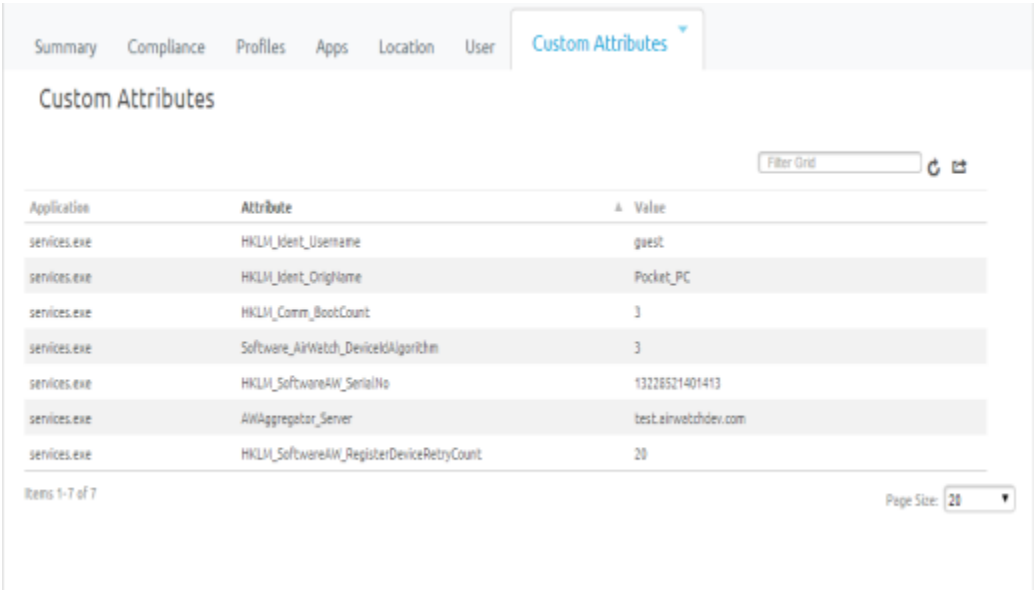
To begin collecting custom attributes, follow the steps detailed below:

1. Navigate to **Devices ►Product (New) ►File/Actions ►Add** and select **Windows Mobile** as your platform.
2. Complete the steps to create an XML Product as mentioned in Appendix B – XML Provisioning. The Manifest should include an action to download the XML file to **\Program Files\Airwatch\Cache\Profiles**.

Upon receiving the XML file, the AirWatch MDM Agent for Windows Mobile creates a custom attributes output file.

During the next check-in with AirWatch, the agent will send the output file to the AirWatch Admin Console.

Once the XML file installs, the custom attributes requested in the file exported to the console. These values display in the console in the Device Details page under Custom Attributes.This page allows you to view the name of the attribute as well as the values returned from each device. These values can be used to create product assignment rules using the Custom Rules system.



You may also view existing custom attributes for all devices at a particular Organization Group as well as manually creating custom attributes directly in the console. Navigate to **Groups & Settings ►All Settings ►Devices & Users ► Advanced ►Custom Attributes** to see these custom attributes listed. Any custom attribute created in this manner automatically associates with a device and its respective custom attribute value that is successfully transmitted to the console.

Syncing Registry Settings

In order to synchronize the registry settings on a Windows Mobile device with the console, which most likely is the most common use of custom attributes for Windows Mobile devices, the you need to create a custom XML file. Below is an example of the format of an XML file that can pull information from the registry on a device:

```
<?xml version="1.0"?>
-<wap-provisioningdoc allowRemoval="True" name="GetTypicalRegValues/V_1"
id="5a63204f-848c-42d5-9c14-4ca070743920">
   -<characteristic uuid="f49a9cb5-48e9-47cd-84cc-ef122dcb5d50"
   type="com.airwatch.getregistryinfo.winmo">
      <reg_value value_name="Username" key_name="HKEY_LOCAL_MACHINE\Ident"
      custom_attribute_name="HKLM_Ident_Username"/>
      <reg_value value_name="OrigName" key_name="HKEY_LOCAL_MACHINE\Ident"
      custom_attribute_name="HKLM_Ident_OrigName"/>
      <reg_value value_name="BootCount" key_name="HKEY_LOCAL_MACHINE\Comm"
      custom_attribute_name="HKLM_Comm_BootCount"/>
      <reg_value value_name="DeviceIdAlgorithm"
      key_name="HKEY_LOCAL_MACHINE\Software\AirWatch"
      custom_attribute_name="Software_AirWatch_DeviceIdAlgorithm"/>
   </characteristic>
</wap-provisioningdoc>
```

It must be in the previous format for the XML file to get correctly parsed and the registry settings to be outputted to a key value pair that can be exported back to the AirWatch Console. In this example, the registry key path is “HKEY_LOCAL_MACHINE\Ident” for two of the values and within that key path it is reading the values of “Username” and “OrigName”. The ‘custom_attribute_name’ parameter is simply the name of the custom attribute that displays in the console and corresponds to the value read from the device.

Using Third-Party Applications to Create Custom Attributes

If you want to create custom attributes using a third party application, you will need that application to export an XML file with a key value pair to the **Program Files\AirWatch\Cache\CustomAttributes** directory on the device. Once an XML file with a key value pair is present in this directory it will be parsed by the agent and included in the next interrogator sample. The XML key/value pair should be in the following format:

```
<?xml version="1.0"?>
-<attributes>
   <attribute name="HKLM_Ident_Username" value="guest"/>
   <attribute name="HKLM_Ident_OrigName" value="Pocket_PC"/>
   <attribute name="HKLM_Comm_BootCount" value="1"/>
   <attribute name="Software_AirWatch_DeviceIdAlgorithm" value="3"/>
   <attribute name="HKLM_SoftwareAW_SerialNo" value="13233521403231"/>
</attributes>
```

‘Attribute name’ will simply be the name of the attribute in the console while ‘value’ will be the corresponding value that will be associated with that attribute.