# Table of Contents

Lab Overview - HOL-1857-05-UEM - Workspace ONE - Mobile Application Management and Developer Tools

- Lab Overview
- Lab Guidance

## Module 1 - Introduction to Mobile Application Management (60 Minutes)

- Introduction
- Different types of applications - Internal / Public / Purchased / Web Apps
- Login to the Workspace ONE UEM Console
- iOS Device Enrollment With Directory Account
- Download AppLifecycle Apps
- Internal App Deployment
- Public App Deployment - Workspace ONE Catalog
- Log into Workspace ONE Catalog
- Internal App Versioning
- Web App Deployment
- Remove Apps via AirWatch Console
- Assume Management
- Un-enrolling Your Device
- Conclusion

## Module 2 - VMware AirWatch REST API (30 minutes)

- Introduction
- Login to the Workspace ONE UEM Console
- iOS Device Enrollment
- VMware AirWatch REST API
- Conclusion

## Module 3 - Per-App VPN using VMware Tunnel (30 minutes)

- Introduction
- Login to the Workspace ONE UEM Console
- AirWatch Console Configuration - Publish VMware Tunnel
- Configure VMware Browser for Per-App VPN
- iOS Device Enrollment
- Testing Per App VPN
- Safari Domain Profile Configuration
- Un-enrolling Your Device
- Testing Safari Domains with Per App Tunnel
- Conclusion

## Module 4 - Introduction to AirWatch Android SDK (45 minutes)

- Introduction
- Connect to Windows 10 VM
- Explore AirWatch SDK for Android using Android Studio
- Login to the Workspace ONE UEM Console
- VMware AirWatch Console configuration for the SDK Sample App
- Enroll an Android Device
Explore AirWatch SDK on the enrolled device ............................................................. 280
Conclusion.................................................................................................................... 289
Module 5 - Jenkins Continuous Integration Plugin for AirWatch (45 minutes) ............. 290
Introduction.................................................................................................................. 291
Connect to Windows 10 VM.................................................................................. 292
Add AirWatch App deployment plugin to Jenkins ..................................................... 293
Login to the Workspace ONE UEM Console ......................................................... 301
Download AppLifecycle Apps ............................................................................ 307
iOS Device Enrollment ....................................................................................... 309
Configure Plugin to integrate with AirWatch ....................................................... 324
Run the Plugin......................................................................................................... 334
Un-enrolling Your Device ..................................................................................... 347
Additional Reading ............................................................................................... 356
Conclusion.................................................................................................................. 357
Lab Guidance

Note: It will take more than 90 minutes to complete this lab. You should expect to only finish 2-3 of the modules during your time. The modules are independent of each other so you can start at the beginning of any module and proceed from there. You can use the Table of Contents to access any module of your choosing.

The Table of Contents can be accessed in the upper right-hand corner of the Lab Manual.

Enhance your enterprise application by leveraging AirWatch Developer Tools into an internal app. This lab targets Mobile Application Management (MAM), AirWatch REST APIs, Per-App VPN using AirWatch Tunnel, AirWatch Android SDK and Jenkins integration. At the end of each lab, we will validate the enhancements on an enrolled device to see the app enhancements in action.

Each Module can be taken independently or you can start at the beginning and work your way through each module in sequence. In most cases, a unique "sandbox" instance of AirWatch will be created just for you when you begin a Module. When the Module has ended, this sandbox will be deleted and the device that you are enrolling in the lab will be returned to the state that it was in prior to the lab. The approximate time it will take to go through all the modules is around 4 hours.

Lab Module List:

- **Module 1 - Introduction to Mobile Application Management** (60 Minutes) Explore the basic concepts of Mobile Application Management (MAM) and how to deploy applications in the AirWatch Console.
- **Module 2 - VMware AirWatch REST API** (30 minutes) (Basic) Explore how can automate AirWatch EMM console action without having to log into the console.
- **Module 3 - Per-App VPN using VMware Tunnel** (30 minutes) (Basic) Enable proxy for your application without the need of any wrapping or without the inclusion of any third party resources.
- **Module 4 - Introduction to AirWatch Android SDK** (45 minutes) (Basic) Introduction to AirWatch Android SDK into a sample app using Android studio and then validate the SDK enhancements on an enrolled Android device.
- **Module 5 - Jenkins Continuous Integration Plugin for AirWatch** (45 minutes) (Basic) Integrate Jenkins, an open source continuous integration solution, with AirWatch to help manage the lifecycle of internal applications.

Lab Captains:

- **All modules: Roger Deane, Shardul Navare, Justin Sheets.**

This lab manual can be downloaded from the Hands-on Labs Document site found here:
This lab may be available in other languages. To set your language preference and have a localized manual deployed with your lab, you may utilize this document to help guide you through the process:


**Location of the Main Console**

1. The area in the RED box contains the Main Console. The Lab Manual is on the tab to the Right of the Main Console.
2. A particular lab may have additional consoles found on separate tabs in the upper left. You will be directed to open another specific console if needed.
3. Your lab starts with 90 minutes on the timer. The lab can not be saved. All your work must be done during the lab session. But you can click the **EXTEND** to increase your time. If you are at a VMware event, you can extend your lab time twice, for up to 30 minutes. Each click gives you an additional 15 minutes. Outside of VMware events, you can extend your lab time up to 9 hours and 30 minutes. Each click gives you an additional hour.

**Alternate Methods of Keyboard Data Entry**

During this module, you will input text into the Main Console. Besides directly typing it in, there are two very helpful methods of entering data which make it easier to enter complex data.
Click and Drag Lab Manual Content Into Console Active Window

You can also click and drag text and Command Line Interface (CLI) commands directly from the Lab Manual into the active window in the Main Console.

Accessing the Online International Keyboard

You can also use the Online International Keyboard found in the Main Console.

1. Click on the Keyboard Icon found on the Windows Quick Launch Task Bar.
Click once in active console window

In this example, you will use the Online Keyboard to enter the "@" sign used in email addresses. The "@" sign is Shift-2 on US keyboard layouts.

1. Click once in the active console window.
2. Click on the Shift key.

Click on the @ key

1. Click on the "@" key.

Notice the @ sign entered in the active console window.
Activation Prompt or Watermark

When you first start your lab, you may notice a watermark on the desktop indicating that Windows is not activated.

One of the major benefits of virtualization is that virtual machines can be moved and run on any platform. The Hands-on Labs utilizes this benefit and we are able to run the labs out of multiple datacenters. However, these datacenters may not have identical processors, which triggers a Microsoft activation check through the Internet.

Rest assured, VMware and the Hands-on Labs are in full compliance with Microsoft licensing requirements. The lab that you are using is a self-contained pod and does not have full access to the Internet, which is required for Windows to verify the activation. Without full access to the Internet, this automated process fails and you see this watermark.

This cosmetic issue has no effect on your lab.

Look at the lower right portion of the screen

![Image of Lab Status]

VMware AirWatch - Mobile Application Management and Developer Tools
Please check to see that your lab is finished all the startup routines and is ready for you to start. If you see anything other than "Ready", please wait a few minutes. If after 5 minutes you lab has not changed to "Ready", please ask for assistance.
Module 1 - Introduction to Mobile Application Management (60 Minutes)
Introduction

Let's go through the fundamentals of AirWatch Mobile Application Management with Workspace ONE. We will walk through how to deploy different types of apps via AirWatch Admin console and touch base on some of the basic management capabilities.
Different types of applications - Internal / Public / Purchased / Web Apps

Depending on the type and mode of deployment, AirWatch classifies applications as Internal, Public, Purchased and Web apps.

- **Internal Apps** - These are internally developed apps and uploaded directly to the AirWatch console or can also be imported from an external app repository. These applications are also known as Enterprise apps.
- **Public Apps** - These apps are available on respective app stores of the platforms i.e. App Store, Play Store, Windows Store etc.
- **Purchased Apps** - These apps are categorized as VPP (Volume purchased program) and Custom B2B apps. VPP allows businesses and educational institutions to purchase publicly available iOS applications. However, custom B2B apps are specifically developed third party iOS applications in volume for distribution to corporate devices.
- **Web Apps** - They provide end-users a way to access a URL directly from an icon on menu of their device.

<table>
<thead>
<tr>
<th>Platform/ Type</th>
<th>Internal</th>
<th>Public</th>
<th>Web</th>
<th>Purchased</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows Desktop</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
Login to the Workspace ONE UEM Console

To perform most of the lab you will need to login to the Workspace ONE UEM Management Console.

Launch Chrome Browser

Double-click the Chrome Browser on the lab desktop.
Authenticate to the Workspace ONE UEM Administration Console

The default home page for the browser is https://hol.awmdm.com. Enter your Workspace ONE UEM Admin Account information and click the Login button.

**NOTE - If you see a Captcha, please be aware that it is case sensitive!**

1. Enter your **Username**. This is your **email address** that you have associated with your **VMware Learning Platform (VLP) account**.
2. Enter **VMware1!** for the **Password** field.
3. Click the **Login** button.

**NOTE - Due to lab restrictions, you may need to wait here for a minute or so while the Hands On Lab contacts the Workspace ONE UEM Hands On Labs server.**
Accept the End User License Agreement

Terms of Use

You must accept the following VMware End User License Agreement to use Workspace ONE UEM.

IF YOU HAVE PURCHASED VMWARE IDENTITY MANAGER AS A SERVICE, YOUR USE OF VMWARE IDENTITY MANAGER IS SUBJECT TO THE VMWARE IDENTITY MANAGER TERMS OF SERVICE AVAILABLE AT: HTTP://WWW.VMWARE.COM/DOWNLOAD/EULA.HTML. IF YOU HAVE PURCHASED WORKSPACE ONE AS A SERVICE, YOUR USE OF VMWARE IDENTITY MANAGER IS SUBJECT TO THE WORKSPACE ONE TERMS OF SERVICE AVAILABLE AT: HTTP://WWW.VMWARE.COM/DOWNLOAD/EULA.HTML.

VMWARE END USER LICENSE AGREEMENT

PLEASE NOTE THAT THE TERMS OF THIS END USER LICENSE AGREEMENT SHALL GOVERN YOUR USE OF THE SOFTWARE, REGARDLESS OF ANY TERMS THAT MAY APPEAR DURING THE INSTALLATION OF THE SOFTWARE.

IMPORTANT: READ CAREFULLY. BY DOWNLOADING, INSTALLING, OR USING THE SOFTWARE, YOU (THE INDIVIDUAL OR LEGAL ENTITY) AGREE TO BE BOUND BY THE TERMS OF THIS END USER LICENSE AGREEMENT (“EULA”). IF YOU DO NOT AGREE TO THE TERMS OF THIS EULA, YOU MUST NOT DOWNLOAD, INSTALL, OR USE THE SOFTWARE, AND YOU MUST DELETE OR RETURN THE UNUSED SOFTWARE TO THE VENDOR FROM WHICH YOU ACQUIRED IT WITHIN THIRTY (30) DAYS AND REQUEST A REFUND OF THE LICENSE FEE, IF ANY, THAT YOU PAID FOR THE SOFTWARE.

EVALUATION LICENSE: If you are licensing the Software for evaluation purposes, your use of the Software is only permitted in a non-production environment and for the period limited by the License Key. Notwithstanding any other provision in this EULA, an Evaluation License of the Software is provided “AS-IS” without indemnification, support or warranty of any kind, expressed or implied.

1. DEFINITIONS.

1.1 “Affiliate” means, with respect to a party at a given time, an entity that then is directly or indirectly controlled by, is under common control with, or controls that party, and here “control” means an ownership, voting or similar interest representing fifty percent (50%) or more of the voting or similar interest in the entity.

NOTE - The following steps of logging into the Administration Console will only need to be done during the initial login to the console.

You will be presented with the Workspace ONE UEM Terms of Use. Click the Accept button.
Address the Initial Security Settings

Security Settings

Password Recovery Question 1

Password Recovery Question *

Password Recovery Answer *

Confirm Password Recovery Answer *

Security PIN

A four-digit Security PIN must be entered. It is required in the console for some restricted actions (configured by authorized administrators in System Security settings).

Security PIN *

Confirm Security PIN *

After accepting the Terms of Use, you will be presented with a Security Settings popup. The Password Recovery Question is in case you forget your admin password and the Security PIN is to protect certain administrative functionality in the console.
1. You may need to scroll down to see the Password Recovery Questions and Security PIN sections.
2. Select a question from the Password Recovery Question drop-down (default selected question is ok here).
3. Enter `VMware1!` in the Password Recovery Answer field.
4. Enter `VMware1!` in the Confirm Password Recovery Answer field.
5. Enter `1234` in the Security PIN field.
6. Enter `1234` in the Confirm Security PIN field.
7. Click the Save button when finished.

Close the Welcome Message

Workspace ONE UEM Console Highlights

Powered by VMware AirWatch!

Workspace ONE is powered by VMware AirWatch Unified Endpoint Management (UEM) technology, a unified digital workspace platform delivering a single, secure experience for app management, single sign-on (SSO), and conditional access.

Workspace ONE UEM transforms your business so you can:

- Configure, manage and support devices from any endpoint
- Increase productivity with seamless access to any app
- Safeguard company data at every layer
- Access identity and access management tools with ease
- Enjoy a simplified, consistent look and feel across Workspace ONE

Don't show this message on login
After completing the Security Settings, you will be presented with the Workspace ONE UEM Console Highlights pop-up.

1. Click on the **Don't show this message on login** check box.
2. Close the pop-up by clicking on the **X** in the upper-right corner.
iOS Device Enrollment With Directory Account

You will now enroll your iOS device by using a directory account for use with this module.

Download/Install AirWatch MDM Agent Application from App Store - IF NEEDED

NOTE - Checked out devices will likely have the AirWatch MDM Agent already installed. You may skip this step if your device has the AirWatch MDM agent installed.
At this point, if using your own iOS device or if the device you are using does NOT have the AirWatch MDM Agent Application installed, then install the AirWatch Application.

To Install the AirWatch MDM Agent application from the App Store, open the App Store application and download the free **AirWatch MDM Agent** application.

**Launching the AirWatch MDM Agent**

Launch the **AirWatch Agent** app on the device.

*NOTE - If you have your own iOS device and would like to test you will need to download the agent first.*
Choose the Enrollment Method

Welcome to AirWatch!

AirWatch helps your IT Department to provide your device with secure access to resources.

The multi-step enrollment process begins with authentication.

Choose authentication method:

Email Address

Server Details

QR Code

Click on the Server Details button.
Find your Group ID from AirWatch Console

The first step is to make sure you know what your **Organization Group ID** is.

1. To find the Group ID, hover your mouse over the Organization Group tab at the top of the screen. Look for the email address you used to log in to the lab portal.
2. Your **Group ID** is displayed at the bottom of the Organization Group pop up.

**NOTE - The Group ID is required when enrolling your device in the following steps.**

Attach the AirWatch MDM Agent to the HOL Sandbox

Once the Agent has launched you can enroll the device. To do so, follow the below steps.
1. Enter hol.awmdm.com for the Server field.
2. Enter your Group ID for your Organization Group for the Group ID field. Your Group ID was noted previously in the Finding your Group ID step.
3. Tap the Go button.

**NOTE - If on an iPhone, you may have to close the keyboard by clicking Done in order to click the Continue button.**

**Authenticate the AirWatch MDM Agent**

On this screen, enter the Username and Password for the basic user account.

1. Enter aduser in the Username field.
2. Enter VMware1! in the Password field.
3. Tap the Go button.
Redirect to Safari and Enable MDM Enrollment in Settings

Enable Device Management

To enable your device, you will be redirected to Safari and Settings.

Why?

- Access your company resource
- Remove company data in the event of loss or theft

The AirWatch Agent will now redirect you to Safari and start the process of enabling MDM in the device settings.

Tap on **Redirect & Enable** at the bottom of the screen.
Allow Website to Open Settings (IF NEEDED)

If you prompted to allow the website to open Settings to show you a configuration profile, tap **Allow**.

*NOTE - If you do not see this prompt, ignore this and continue to the next step. This prompt will only occur for iOS Devices on iOS 10.3.3 or later*
Install the MDM Profile

Tap **Install** in the upper right corner of the Install Profile dialog box.
Install and Verify the AirWatch MDM Profile

Tap **Install** when prompted at the Install Profile dialog.

*NOTE - If a PIN is requested, it is the current device PIN. Provided VMware devices should not have a PIN.*
iOS MDM Profile Warning

Installing this profile will allow the administrator at “https://hol.awmdm.com/DeviceServices/AppleMDM/Processor.aspx” to remotely manage your iPad.

The administrator may collect personal data, add/remove accounts and restrictions, and list, install and manage apps on your iPad.

You should now see the iOS Profile Installation warning explaining what this profile installation will allow on the iOS device.

Tap **Install** in the upper-right corner of the screen.
Trust the Remote Management Profile.

You should now see the iOS request to trust the source of the MDM profile.

Tap **Trust** when prompted at the Remote Management dialog.
iOS Profile Installation Complete

You should now see the iOS Profile successfully installed.

Tap **Done** in the upper right corner of the prompt.
Your enrollment is now completed. Tap **Open** to navigate to the AirWatch Agent.
Accept the Authentication Complete Prompt

![Configuration Prompt]

Authentication Complete

- You will receive company resources and settings assigned to your device by your IT department
- You will receive a notification if further action is required

Click on **Done** to continue.

Accept Notification Prompt (IF NEEDED)

![Notification Prompt]

Tap **Allow** if you get a prompt for Notifications.

Accept the App Installation (IF NEEDED)

![App Installation Prompt]
You may be prompted to install a series of applications depending on which Module you are taking. If prompted, tap **Install** to accept the application installation.
Download AppLifecycle Apps

In this section, we are going to download AppLifecycle Apps that we will be using as Internal apps for this lab.

**Download AppLifecycle 101**

1. Open a **new tab** in Chrome Browser.
2. Enter the following URL [https://hol.awmdm.com/MyDevice/s/2239/be759588-38d0-4ad4-949e-88a1f4398f4b](https://hol.awmdm.com/MyDevice/s/2239/be759588-38d0-4ad4-949e-88a1f4398f4b) and hit **Enter**
3. Validate that you have downloaded **App lifecycle_101.ipa**
Download AppLifecycle 102

1. Open a **new tab** in Chrome Browser.
2. Enter the following URL **https://hol.awmdm.com/MyDevice/s/2239/86896741-33e4-43fd-a843-6225742f002c** and hit **Enter**
3. Validate that you have downloaded **Applifecycle_102.ipa**
Internal App Deployment

Use Workspace ONE to distribute, track, and manage your internal applications. These are applications built in-house and not hosted on Public App Stores. You can upload the application files directly to AirWatch console for deployment. However, if you use an external repository to host your internal applications, then you can easily integrate that host with AirWatch, instead of migrating the entire catalog to AirWatch.

Supported File types for different platforms:

<table>
<thead>
<tr>
<th>Platform</th>
<th>File Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td>APK</td>
</tr>
<tr>
<td>iOS</td>
<td>IPA</td>
</tr>
<tr>
<td>macOS</td>
<td>APP Package Bundles</td>
</tr>
<tr>
<td>Windows Desktop</td>
<td>APPX, EXE, MSI, ZIP</td>
</tr>
<tr>
<td>Windows Phone</td>
<td>APPX, XAP</td>
</tr>
</tbody>
</table>

Once the application is installed, you can track the installation status and reason codes in case of failures.

Upload Internal Application with a Local File

In this section, we are going to add an iPA file to AirWatch console as an internal iOS app.

Add Internal Application

In the AirWatch console,

1. Click on Add
2. Click **Internal Application**

**Upload**

![Add Application](image1)

**Choose File**

![Add](image2)

Click **Choose File**
Navigate to the iPA file

1. Click on Downloads
2. Select AppLifecycle_101.ipa
3. Click Open
Continue After Uploading IPA

Click **Continue**.

**Save**
Click Save

Application Details

1. Notice how AirWatch can parse the Application bundle ID from the IPA File. All the application versioning within the AirWatch Console is based on the Application ID.
2. Notice how AirWatch can also parse the Application Version number from the IPA File.
3. Select Category **Productivity (System)**
4. You can also change the minimum OS requirement to be able install this app. For this lab, we are going to keep this value to default **iOS 9.0.0**.
5. Click on **Save & Assign**

Categories are useful to group the apps so that they are easy to find from the catalog.
Internal app version for iOS is determined by `CFBundleVersion` and `CFBundleShortVersionString` from `info.plist`.

**Add an Assignment to the Internal Application**

We will now configure which devices will receive the internal application.

### Add Assignment

![Add Assignment](image)

Click **Add Assignment**

Devices will receive application based on the below configuration.
In the case where devices belong to multiple groups, they will receive policies from the grouping with highest priority (0 being highest priority).
Add Delivery method

1. All the assignments in AirWatch are done via Assignment Groups. These assignment groups can be created by specifying different filter criteria on your devices and/or users. For this lab we are going to select our default group All Devices.

2. Your App Delivery method can be Auto where the application is installed automatically to the enrolled device. However, for this lab, we are going to select On Demand to see how can we download these apps from Workspace One Catalog.

3. You can also select the Deployment Begin Date for a particular app assignment. If you select a date in future, the app will not be available for download until that time. For this lab, we are going to keep the default value, which should reflect the current date.
Enable Remove on Unenroll

1. Scroll down until you see the section **Policies**
2. Select **Enable** for **Remove on Unenroll**
3. Click on **Add** to continue.

By enabling this option, you are flagging this app to be removed when the device is unenrolled after this lab is ended. We will have the same configuration for all the apps that we are going to deploy in this lab.

Enable this flag if you want to un-install the app so that the app data is not leaked when an employee leaves the organization or when the device is lost or stolen.
Save & Publish

### AppLifecycle - Update Assignment

Devices will receive application based on the below configuration. In the case where devices belong to multiple groups, they will receive policies from the grouping with highest priority (0 being highest priority).

<table>
<thead>
<tr>
<th>Name</th>
<th>Priority</th>
<th>App Delivery Method</th>
<th>Effective</th>
<th>Managed Access</th>
<th>Remove On Unenroll</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Devices</td>
<td>0</td>
<td>On Demand</td>
<td>Now</td>
<td>Enabled</td>
<td>Enabled</td>
</tr>
</tbody>
</table>

Click on **Save & Publish** to continue.
Preview your device assignment and click **Publish** to continue.
1. Click **Apps & Books**
2. Expand **Applications**
3. Click **Native**
4. Click the **Internal** tab.
5. Validate that you have the application uploaded with the name as **AppLifecycle**.
6. In the **Version** column, you will see the version as **1.0.1**
7. In the column **Renewal Date**, you will see when the provisioning profile is going to expire for this particular app.

You can renew this provisioning profile from AirWatch console itself (**via Application Details > Files menu**), without having to rebuild and re-upload the app to the AirWatch console. This simplifies the recurring task of profile renewal, without any intervention from App Developer and any interruption on the end user devices.

For this lab, we are not going to renew the provisioning profile of this app.
Public App Deployment - Workspace ONE Catalog

AirWatch offers two app catalogs - Workspace ONE and the AirWatch Catalog. Both catalogs support the features in the Apps Settings of the AirWatch Console.

The Workspace ONE catalog integrates resources from environments that use VMware Identity Manager and AirWatch. If your deployment does not use VMware Identity Manager, you still have access to the features previously released for the AirWatch Catalog.

In this lab, we are going to access the assigned apps via Workspace ONE catalog which is available as a public app from App Store.

Add Workspace One as a public app

1. Click Add
2. Click Public Application
1. Select Platform as **Apple iOS**
2. Select Source as **Search App Store**
3. Select Name as **VMware Workspace One**
4. Click **Next**
Select **VMware Workspace ONE** from the search results.
Save & Assign

Edit Application - VMware Workspace ONE
Public | Managed By: your@email.shown.here | Application ID: com.air-watch...

Details

Name* VMware Workspace

View in App Store

Upload

Categories
Start Typing to Select Category ...

Supported Models
iPad
iPhone
IPod Touch

Save & Assign  Cancel

Click Save & Assign
Add Assignment for Workspace ONE

Click Add Assignment
Select Assignment Group as **All Devices**
2. Select App Delivery Method as **Auto**

**TIP - Automatic App Delivery** ensure that the app is installed on the device automatically, without relying on end users to download it from the catalog. *Use this setting for the apps that you want to make mandatory for your end users.*
Modify Policies

1. Scroll down to the Policies section.
2. Select **Enabled** for Remove on Unenroll.
3. Select **Enabled** for Application Configuration.
Add Application Configuration

In this section, we are going to configure Workspace ONE app, so that it auto-populates the server URL and device UDID at the time of launch.

**TIP - Use AppConfig to pre-configure apps, which reduces end user inputs for a seamless end user experience.**

1. Scroll down to the bottom of the page.
2. Enter key as `AppServiceHost`
3. Enter value as "https://holmam.vidmpreview.com" with the type `String`
4. Click on +Add
5. Enter key as `deviceUDID`
6. Enter value as `{DeviceUid}`
7. Click Add

**NOTE - All the keys and lookup values are case sensitive.**
Save & Publish

VMware Workspace ONE - Update Assignment

Assignment

Devices will receive application based on the below configuration. In the case where devices belong to multiple groups, they will receive policies from the grouping with highest priority (0 being highest priority).

Add Assignment

Name | Priority | App Delivery Method | Managed Access | Remove On Unenroll | Prevent Application
--- | --- | --- | --- | --- | ---
All Devices | 0 | Auto | Enabled | Disabled | Disabled

Click **Save & Publish**
Publish the app

Validate that you are seeing your device enrolled in the assignment list. Click Publish to continue.
Accept the app installation prompt

As soon as the device checks in after the app is assigned, you see a prompt on the device to install the Workspace ONE app. Click **Install** to continue.
Log into Workspace ONE Catalog

Launch Workspace App

Click on the **Workspace** app to launch.

**Create a Passcode (IF NEEDED)**

If you do not already have a device passcode set on the iOS device, you will receive a warning message before being able to access the Workspace ONE app. Please navigate to **Settings > Passcode > Turn Passcode On**, to setup a new passcode, then return to Workspace ONE.
Validate App Service Host URL from AppConfig

This is the value that we entered for the key AppServiceHost while configuring the deployment options for Workspace ONE. This is how easy it is to pre-configure the Workspace ONE app for a seamless end user experience.

1. Validate that the pre-populated URL is https://holmam.vidmpreview.com
2. Tap Next to continue.
Select domain as **corp.local**

1. Select domain as **corp.local**
2. Check the box to **Remember this settings**
3. Click **Next**
1. Enter username as "aduser"
2. Enter password as "VMware1!"
3. Ensure that you are seeing corp.local as the domain.
4. Tap on Sign in
Enter Workspace ONE

Your workspace is ready.

Whenever you see the message **Your Workspace is ready**, tap on **Enter**.

Accept the notifications prompt

![Notification Prompt Image]

Tap **Allow** to enable Notifications for Workspace.
Internal App Versioning

Internal and Enterprise apps get updated on a regular basis to offer latest functionality and security enhancements. Workspace ONE makes it easy to update these apps on end user devices over-the-air automatically, without having to connect the device to a computer. In this section, we are going to add an internal app on-demand and install it from Workspace ONE catalog. We will also see how to update the app in the AirWatch console so that it gets updated on the enrolled device without any app data loss.

Install the Internal app from Workspace ONE Catalog

Since we do not have many apps deployed in this lab, we can see all the apps from the default view. However, we are still going to validate the app category we assigned while deploying our internal app.

Validate the Category of the internal app

1. Tap on Got it! to dismiss the notification.
2. Tap on the Menu icon in the top right corner.
3. Tap on Productivity to filter the apps with that category.
Install 1.0.1 version of the internal app

Install

Validate that you only seeing AppLifecycle v1.0.1 for the category Productivity. Click on Install to continue.

Confirm Installation in Workspace ONE Catalog

Confirm Installation

AppLifecycle
Size: 6.38MB
You will receive a push notification to continue with installation.

Cancel  Install

Click on Install to initiate app installation.
Install the app

Click on Install to accept OS prompt for installing the internal app.

If your device is supervised, then you will not see the OS prompt for installing internal and enterprise apps.

iOS device supervision enables the internal apps to be installed silently.

Confirm app installation on the device

Click on the Home button of the iPad to return to the springboard. Confirm that the app AppLifecycle got installed successfully.

Add an updated version of the Internal app to AirWatch Console

We will upload a new version of our internal app to see how this reflects in the AirWatch Console as well as on our device.
Navigate to the Internal app

Back to the AirWatch console,

1. Click on **Apps & Books**
2. Expand **Applications**
3. Click on **Native**
4. Click the **Internal** tab
5. Click on the hyperlink for **AppLifecycle**
Add Version

Click on + Add Version in the top right corner.
While updating the new version of the app, AirWatch displays the current active version to help determine the app upgrade path.

Click on **Upload**
Choose File

Click **Choose File**
Navigate to the iPA file

1. Click on Downloads
2. Select AppLifecycle_102.ipa
3. Click Open
Click **Save**. The application will take around 45 - 60 seconds to upload.
Complete the app update

1. Validate that you have uploaded **AppLifecycle_102.ipa**
2. Click **Save** to continue.
While retaining all the defaults, click on **Save & Assign**
Save & Publish

AppLifecycle - Update Assignment

Devices will receive applications based on the below configuration. In the case where devices belong to multiple groups, they will receive policies from the grouping with highest priority (0 being highest priority).

Add Assignment

<table>
<thead>
<tr>
<th>Name</th>
<th>Priority</th>
<th>App Delivery Method</th>
<th>Effective</th>
<th>Managed Access</th>
<th>Remove On Unenroll</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Devices</td>
<td>0</td>
<td>On Demand</td>
<td>Now</td>
<td>Enabled</td>
<td>Enabled</td>
</tr>
</tbody>
</table>

Save & Publish
While retaining all the defaults, click on **Save & Publish**

Preview your device assignment and click **Publish** to continue.

**Install the updated version of the app from Workspace ONE Catalog**

Let's view the update process on the device in Workspace ONE when apps are updated through the AirWatch Console.
Launch Workspace ONE

Back to the enrolled iPad, tap on the icon to launch **Workspace ONE app**.

**Enter Passcode For Timeout (IF NEEDED)**

Enter iPad passcode for “Workspace”

In case you encounter a timeout, enter the iPad passcode to log back into Workspace App.
Refresh Workspace ONE Catalog and update the app

You may be seeing the old version of the app (1.0.1) after launching the Workspace ONE app. Swipe down in the screen to refresh.

1. After the refresh, ensure that you are seeing v1.0.2 for AppLifecycle.
2. Click on Update

Confirm Installation in Workspace ONE Catalog

Click on Install to initiate app installation.
Confirm app update on the device

Click on the Home button of the iPad to return to the springboard. Confirm that you are seeing the blue dot next to the app **AppLifecycle** which indicates that app got updated successfully.

Uninstall the app from managed devices

As a part of AirWatch flexible deployment, the app removal from AirWatch has three different phases:

1. **Retire** - Removes an application from all managed devices. For iOS devices, if an older version of the application exists in the AirWatch solution, then this older version is pushed to devices.
2. **Deactivate** - Removes an application and all versions of it from all managed devices.
3. **Delete** - Deletes the app from AirWatch Database. If the application is currently installed on any devices, it puts the app in the Deactivated state first. You can then remove the app by changing the filter to Inactive.

Use the **Retire** option if you want to revert to an earlier version, without uninstalling the app from all the enrolled devices.
Web App Deployment

Web applications are useful for navigating to complex URLs with many characters. You can place Web application icons on the springboard to minimize the frustration with accessing these website. These icons connect end-users to internal content repositories or login screens, so end-users do not open a browser and type out a long or complex URL.

Add Web App from AirWatch console

Continue to walk-through the process of adding a Web app through the AirWatch Console.

Navigate to Web Apps

Back to the AirWatch Console,

1. Click on **Apps & Books**
2. Expand **Applications**
3. Expand **Web**
4. Click **Web Links**
5. Click on **+Add Application**
Select Platform

1. Select platform as **Apple iOS**
2. Click **Continue**
Enter Details

1. Click on **Details** if not selected already.
2. Enter **Name** as "**VMware**"
3. Enter **URL** as "**https://www.vmware.com**"
Add an image

You can associate an icon with your web app so that it is easily distinguishable from the other native apps. Having an icon in alignment with your organization branding guidelines helps your end users to easily identify them and results in better end user experience. Refer to the Branding HOL to learn more about how you can customize the managed content via AirWatch.

1. Click on **Images**
2. Click on the placeholder to add **Icon**
Navigate to Image file

1. Expand **Documents**
2. Expand **HOL**
3. Click on **Intro to MAM**
4. Select **VMware.jpg**
5. Click **Open**
Validate the icon

Confirm that you are now seeing the VMware image file as the icon for the web app.
Configure Assignment

1. Select Assigned Groups as All Devices (your@email.shown.here)
2. Select Push Mode as Auto
3. Click on Save & Publish

By configuring the Push Mode to Auto, the web app will be installed automatically on the device.
Publish the Web App

Click on Publish to continue.

Validate the Web App in AirWatch Console

Validate that you are now seeing VMware web app for iOS added in AirWatch Console.
Access the Web App from enrolled device

Now that the Web app is added to the AirWatch Console and published to devices, let's view and interact with the Web app from our device.

Confirm the Web App Installation

As soon as the device checks back in with AirWatch, a bookmark / web-clip will be installed for the web app VMware we just configured.
Launch the Web App

Tap on the Web app to launch Safari to display the VMware Homepage.

NOTE - The Homepage may differ from the screenshot.
Remove Apps via AirWatch Console

So far, we have seen how to deploy apps using AirWatch. Having the ability to remove the apps from a device is as important as deploying them, especially in the scenarios where a device is lost or stolen or if an employee leaves the organization. This not only clears the sensitive app data from the device but it also revokes access to the corporate resources and functionality that the app has access to.

Uninstall the Web App

In this section, we will see how to remove the apps from the enrolled devices.

Navigate to the Web App

Back in the AirWatch Console,

1. Click on **Apps & Books**
2. Expand **Applications**
3. Expand **Web**
4. Click on **Web Links**
5. Click on **Assigned Users icon with value 1**
Uninstall

1. Click on X to remove the web app from your enrolled device.
2. Click OK from the pop-up window.
3. Click the X on the View Devices screen to close it.

Validate Web App Removal on the device

Validate that the web app was removed from the enrolled device.

In this section, we removed a web app from the device but using similar steps you can remove internal and public apps as well. In the lab, we have configured our apps to remove on un-enroll, so they will be automatically removed upon un-enrolling the device at the end of this lab.
Assume Management

Apple iOS enables AirWatch to assume management of user-installed applications without requiring the deletion of the previously installed application from the device. In this section, we are going to install a public app from App Store and assume the management for it. This will enable us to perform all the mobile application management policies on this user-installed app, including removal upon un-enrollment. We will validate this in the next article.

Consider the scenario where your employee has installed the app from App Store directly (very common in BYOD). In that case this app is unmanaged since it is not pushed down via AirWatch console. As a result, this app can not have MAM enhancements like per-app VPN (to connect to a backend resource), App Config (to auto-configure the app over-the-air), or Data Loss Prevention (removal of the app in case the device is stolen or compromised).

In this section, we will see how to convert such apps as managed apps so that they can leverage the above AirWatch Mobile Application Management (MAM) enhancements and much more.

Install an unmanaged app from App Store

Let's begin by downloading and installing an unmanaged app from the App Store on our device. We will assume management of this app later.

Launch App Store

Tap on App Store to launch.
Search Salesforce

1. Search **Salesforce** in the search box.
2. Tap on **GET** to initiate the install.

Install Salesforce

Tap on **Install**
Open Salesforce

Once the download is completed, tap OPEN to launch the app.
Accept Salesforce EULA

Salesforce.com EULA

Order Form Supplement for Salesforce for iOS

This is a legally binding agreement. Please read it carefully. By clicking "I Accept", or installing or using the Salesforce for iOS software and/or any updates to such software provided by salesforce.com (the "Software"), you:

- agree to the following terms on behalf of the Salesforce customer with which you are employed, affiliated or associated (the "Customer"),
- represent that you have the authority to bind the Customer to these terms, and
- represent that you are an authorized User under the master subscription agreement between salesforce.com and the Customer (the "Master Subscription Agreement").

If you do not have such authority, are not an authorized User, or do not agree to these terms, you may not install or use the Software.

Apple Inc. may, at any time and without notice, restrict, interrupt or prevent use of the Software, or delete the Software from your or the Customer’s Apple devices, or require salesforce.com to do any of the foregoing, without entitling the Customer or you to any refund, credit or other compensation from salesforce.com or any third party (including, but not limited to, Apple Inc. or your network connectivity provider).

This Order Form Supplement was last updated on July 14, 2016. It is effective between the Customer and salesforce.com as of the date you first download, install or use the Software, whichever is earliest.

1. THIS ORDER FORM SUPPLEMENT

This Order Form Supplement is a part of each order form by which the Customer ordered the Salesforce service (including, but not limited to, the Salesforce Files component of the Salesforce Chatter service), or a Salesforce service incorporating the Salesforce service (an "Order Form"). The Order Form and this Order Form Supplement are governed by the Master Subscription Agreement. This Order Form Supplement adjusts certain terms of the Master Subscription Agreement, solely with respect to the Software. Capitalized terms used but not defined in this Order Form Supplement have the meanings given to them in the Master Subscription Agreement.

2. THE SOFTWARE

The Software allows Salesforce customers to use the Salesforce service from Supported Devices. A "Supported Device" is a combination of an Apple device model and relevant iOS software version(s) that is supported by the Software. The Software is provided by salesforce.com as a component of the Salesforce service.

3. WHO YOU ARE CONTRACTING WITH

This Order Form Supplement is being entered into between the salesforce.com entity named on the Order Form (either salesforce.com, inc., salesforce.com Canada Corporation, salesforce.com EMEA Limited, salesforce.com Singapore Pte Ltd, or salesforce.com Koubushii Kaisha ("salesforce.com") and the Customer. In this Order Form Supplement, the term "salesforce.com" means that entity.

4. SOFTWARE LICENSE

The Software, including software embedded in the Software, is licensed, not sold, to the Customer by salesforce.com only under the terms of the Master Subscription Agreement, the Order Form and this Order Form Supplement, and salesforce.com reserves all rights not expressly granted to the Customer. This Software may include some third-party software: please see the Documentation for additional notices required by our licensors. The Customer or you own the media or device on which the Software is recorded or stored but salesforce.com retains ownership of the Software itself.

5. PERMITTED LICENSE USES AND RESTRICTIONS

(a) This Order Form Supplement allows you, as an authorized User under the Master Subscription Agreement, to use the Software on any Supported Devices and on no other devices.
(b) You may not distribute or make the Software available over a network where it could be used by multiple devices at the same time.
(c) With respect to updates to the Software that salesforce.com may make available for download, this Order Form Supplement allows you to download such Software updates to update or restore the Software on any Supported Device.
(d) Except as and only to the extent permitted by applicable law, or by licensing terms governing use of open-sourced components included with the Software, neither you nor any other Customer personnel may copy, decompile, reverse engineer, disassemble, attempt to derive the source code of, decrypt, modify, or create derivative works of the Software or Software updates, or any part thereof. Any attempt to do so is a violation of the rights of salesforce.com and its licensors. If you or any other Customer personnel violate this restriction, you or they, and the Customer, may be subject to prosecution and damages.

Tap I Accept to accept Salesforce EULA.
Accept the notification prompt for Salesforce

Tap **OK** to accept the notification prompt for Salesforce.
Validate Connection options

Tap on the gear icon to validate the available connections. Notice that you are seeing just the default connections, Production and Sandbox.
Add the same application as a public app from AirWatch console

Now that we've downloaded an unmanaged app, we will publish the same app from the AirWatch Console as part of the process of assuming management.

Add Salesforce as a public app

1. Click Add
2. Click Public Application
Search for Salesforce

Add Application

1. Select Platform as **Apple iOS**
2. Select Source as **Search App Store**
3. Select Name as **Salesforce**
4. Click **Next**
Select the Salesforce Result

New Look, same great product! Salesforce1 is now Salesforce. Run your business from your mobile device. Salesforce unifies your Chatter, CRM, custom apps, and business processes in a modern experience. Whether you’re in the office, on an airplane, or checking in from a coffee shop, Salesforce is your key to productivity. Access all your CRM data, existing customizations, and breakthrough productivity tools from anywhere. Manage your day. Stay on top of your day, from viewing your scheduled even... 

Select **Salesforce** from the search results.

### Save & Assign

![Edit Application - Salesforce](image)

**Name**: Salesforce

**Categories**: Start Typing to Select Category...

**Supported Models**: iPad, iPhone, iPod Touch

Click **Save & Assign**
Add Assignment

1. Select Assignment Group as **All Devices**
2. Select App Delivery Method as **Automatic: system push**

**TIP - Automatic App Delivery** ensures that the app is installed on the device automatically, without relying on end users to download it from the catalog. *Use this setting for the apps that you want to make mandatory for your end users.*

Add App Config for Salesforce
Enable Flags

1. Select **Enabled** for Remove on Unenroll
2. Select **Enabled** for Make App MDM Managed if User Installed.

Add App Config

1. Scroll down until you see the Application Configuration section.
2. Select **Enabled** for Application Configuration.
3. Enter Configuration Key as "AppServiceHosts"

NOTE - Now, we will configure Salesforce app to have a connection to a custom domain using App Config. We will validate this new connection on the device at a later step.

1. Enter Key-Value pairs to configure applications for users:
   - **AppServiceHosts**: String, holmam-dev-ed.my.s
4. Enter **Configuration Value** as "holmam-dev-ed.my.salesforce.com"
5. Click **Add**

**Save & Publish**

![Salesforce - Update Assignment](image)

Devices will receive application based on the below configuration.
In the case where devices belong to multiple groups, they will receive policies from the group with the highest priority (0 being highest priority).

**Add Assignment**

<table>
<thead>
<tr>
<th>Name</th>
<th>Priority</th>
<th>App Delivery Method</th>
<th>Managed Access</th>
<th>Remove On Unenroll</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Devices</td>
<td>0</td>
<td>Auto</td>
<td>Enabled</td>
<td>Enabled</td>
</tr>
</tbody>
</table>

Click **Save & Publish**
Publish the app

Click **Publish**

**Salesforce as managed app**

We will now see how the Salesforce app becomes managed by AirWatch on our device.
Close Salesforce app
Double press the Home button to launch app switcher. **Swipe up** the salesforce app to close **Salesforce** app.

**Relaunch Salesforce app**

Tap on the icon to relaunch **Salesforce** app.

**Accept App Management Change prompt**

Review the **App Management Change** prompt confirming the EMM is managing this app now. Tap **Close** to continue.
Review the Connection List

1. Click on the Gear Icon to view the connections that are available.
2. Validate that you are seeing the new connection holmam-dev-ed.my.salesforce.com which we add via App Config.

NOTE - Since we assumed the management of the Salesforce app, we could update the app over-the-air with Application Configuration. This app will also
get removed automatically when we un-enroll the device preventing any data loss from an user-installed app.

Conclusion

This is how easy it is to manage a user installed device via AirWatch. This feature is very powerful in a BYOD scenario to enhance functionality and ensure proper security of the user installed apps.
Un-enrolling Your Device

You are now going to un-enroll the iOS device from Workspace ONE UEM.

**NOTE - The term “Enterprise Wipe” does not mean reset or completely wipe your device. This only removes the MDM Profiles, Policies, and content which the AirWatch MDM Agent controls.**

It will NOT remove the AirWatch Agent application from the device as this was downloaded manually before Workspace ONE UEM had control of the device.

Enterprise Wipe (un-enroll) your iOS device

Enterprise Wipe will remove all the settings and content that were pushed to the device when it was enrolled. It will not affect anything that was on the device prior to enrollment.

To Enterprise Wipe your device you will first bring up the Workspace ONE UEM Console in a web browser. You may need to re-authenticate with your credentials (VLP registered email address and VMware1 as the password).

1. Click **Devices** on the left column.
2. Click **List View**.
3. Check the **checkbox** next to the device you want to Enterprise Wipe.
NOTE - Your Device Friendly Name will very likely be different than what is shown. It will, however, be in the same location as shown on image in this step.

Find the Enterprise Wipe Option

1. Click More Actions. NOTE - If you do not see this option, ensure you have a device selected by clicking the checkbox next to the device.
2. Click Enterprise Wipe under Management.
Enter your security PIN

After selecting **Enterprise Wipe**, you will be prompted to enter your Security PIN which you set after your logged into the console (1234).

1. Enter 1234 for the **Security PIN**. You will not need to press enter or continue, the console will confirm your PIN showing "Successful" below the Security PIN input field to indicate that an Enterprise Wipe has been requested.

   **NOTE** - If 1234 does not work, then you provided a different Security PIN when you first logged into the Workspace ONE UEM Console. Use the value you specified for your Security PIN.

**NOTE - If the Enterprise Wipe does not immediately occur, follow the below steps to force a device sync:**

1. On your device, open the AirWatch **Agent** application.
2. Tap the **Device** section (under **Status**) in the middle of the screen.
3. Tap **Send Data** near the top of the screen. If this does not make the device check in and immediately un-enroll, continue to Step #4.
4. If the above doesn't make it immediately un-enroll, then tap **Connectivity [Status]** under Diagnostics.
5. Tap **Test Connectivity** at the top of the screen.

**NOTE - Depending upon Internet connectivity of the device and responsiveness of the lab infrastructure, this could take a couple of minutes or more if there is excessive traffic occurring within the Hands On Lab environment.**

Feel free to continue to the "**Force the Wipe**" step to manually uninstall the Workspace ONE UEM services from the device if network connectivity is failing.

**Verify the Un-Enrollment**
Press the Home button on the device to go back to the home screen. The applications pushed through Workspace ONE UEM should have been removed from the device.

**NOTE** - The applications and settings pushed through Workspace ONE UEM should have been removed. The Agent will still be on the device because that was downloaded manually from the App Store. Due to lab environment settings, it may take some time for the signal to traverse through the various networks out and back to your device. Continue on to the next step to force the wipe if the needed.
Force the Wipe - IF NECESSARY

If your device did not wipe, follow these instructions to ensure the wipe is forced immediately. Start by opening the iOS Settings app.
1. Tap **General** in the left column.
2. Scroll down to view the **Device Management** option.
3. Tap **Device Management** at the bottom of the list of General settings.

**Force the Wipe - IF NECESSARY**

Tap the **Workspace Services** profile that was pushed to the device.
Force the Wipe - IF NECESSARY

1. Tap **Remove Management** on the Workspace Services profile.  
   *NOTE* - *If prompted for a device PIN, enter it to continue.*  VMWare provisioned devices should not have a device PIN enabled.

2. Tap **Remove** on the Remove Management prompt.
After removing the Workspace Services profile, the device will be un-enrolled. Feel free to return to the **Verify the Un-Enrollment** step to confirm the successful un-enrollment of the device.
Conclusion

In this lab, we went through how to deploy and manage different types of applications using Workspace ONE. We also saw how to remove a managed app from a device and how to assume management of apps installed by the end users.
Module 2 - VMware AirWatch REST API (30 minutes)
Introduction

AirWatch provides a collection of RESTful APIs which allow external programs to use the core product functionality by integrating the APIs with existing IT infrastructures and third-party applications. Leveraging the simplified REST style of software architecture, AirWatch REST APIs currently include Organization Group, Console Administration, Mobile Application Management, Mobile Device Management, Enrollment User Management, Smart Group Management and User Group Management functionalities.

In this lab, you will:

- Configure the AirWatch Management Console to enable AirWatch REST API access.
- Setup a REST API Client in the Chrome Browser on the Control Center server.
- Enroll a device into AirWatch.
- Use the REST API Client to call "GET" API functions to retrieve data about the enrolled device.
- Use the REST API Client to call "POST" API functions to configure the enrolled device.
- Un-Enroll the device from AirWatch using the REST API.
Login to the Workspace ONE UEM Console

To perform most of the lab you will need to login to the Workspace ONE UEM Management Console.

Launch Chrome Browser

Double-click the Chrome Browser on the lab desktop.
Authenticate to the Workspace ONE UEM Administration Console

The default home page for the browser is https://hol.awmdm.com. Enter your Workspace ONE UEM Admin Account information and click the Login button.

NOTE - If you see a Captcha, please be aware that it is case sensitive!

1. Enter your **Username**. This is your email address that you have associated with your VMware Learning Platform (VLP) account.
2. Enter **VMware1!** for the **Password** field.
3. Click the **Login** button.

NOTE - Due to lab restrictions, you may need to wait here for a minute or so while the Hands On Lab contacts the Workspace ONE UEM Hands On Labs server.
Accept the End User License Agreement

Terms of Use

You must accept the following VMware End User License Agreement to use Workspace ONE UEM.

VMWARE END USER LICENSE AGREEMENT

PLEASE NOTE THAT THE TERMS OF THIS END USER LICENSE AGREEMENT SHALL GOVERN YOUR USE OF THE SOFTWARE, REGARDLESS OF ANY TERMS THAT MAY APPEAR DURING THE INSTALLATION OF THE SOFTWARE.

IMPORTANT: READ CAREFULLY. BY DOWNLOADING, INSTALLING, OR USING THE SOFTWARE, YOU (THE INDIVIDUAL OR LEGAL ENTITY) AGREE TO BE BOUND BY THE TERMS OF THIS END USER LICENSE AGREEMENT (“EULA”). IF YOU DO NOT AGREE TO THE TERMS OF THIS EULA, YOU MUST NOT DOWNLOAD, INSTALL, OR USE THE SOFTWARE, AND YOU MUST DELETE OR RETURN THE UNUSED SOFTWARE TO THE VENDOR FROM WHICH YOU ACQUIRED IT WITHIN THIRTY (30) DAYS AND REQUEST A REFUND OF THE LICENSE FEE, IF ANY, THAT YOU PAID FOR THE SOFTWARE.

EVALUATION LICENSE: If you are licensing the Software for evaluation purposes, your use of the Software is only permitted in a non-production environment and for the period limited by the License Key. Notwithstanding any other provision in this EULA, an Evaluation License of the Software is provided “AS-IS” without indemnification, support or warranty of any kind, expressed or implied.

1. DEFINITIONS.

1.1 “Affiliate” means, with respect to a party at a given time, an entity that then is directly or indirectly controlled by, is under common control with, or controls that party, and here “control” means an ownership, voting or similar interest representing fifty percent (50%) or more of the voting interests.

NOTE - The following steps of logging into the Administration Console will only need to be done during the initial login to the console.

You will be presented with the Workspace ONE UEM Terms of Use. Click the Accept button.
Address the Initial Security Settings

Security Settings

Password Recovery Question 1

Password Recovery Question *

Password Recovery Answer *

Confirm Password Recovery Answer *

Security PIN

A four-digit Security PIN must be entered. It is required in the console for some restricted actions (configured by authorized administrators in System Security settings).

Security PIN *

Confirm Security PIN *

After accepting the Terms of Use, you will be presented with a Security Settings pop-up. The Password Recovery Question is in case you forget your admin password and the Security PIN is to protect certain administrative functionality in the console.
1. You may need to scroll down to see the Password Recovery Questions and Security PIN sections.
2. Select a question from the Password Recovery Question drop-down (default selected question is ok here).
3. Enter VMware1! in the Password Recovery Answer field.
4. Enter VMware1! in the Confirm Password Recovery Answer field.
5. Enter 1234 in the Security PIN field.
6. Enter 1234 in the Confirm Security PIN field.
7. Click the Save button when finished.

Close the Welcome Message

Workspace ONE UEM Console Highlights

Powered by VMware AirWatch!

Workspace ONE is powered by VMware AirWatch Unified Endpoint Management (UEM) technology, a unified digital workspace platform delivering a single, secure experience for app management, single sign-on (SSO), and conditional access.

Workspace ONE UEM transforms your business so you can:

- Configure, manage and support devices from any endpoint
- Increase productivity with seamless access to any app
- Safeguard company data at every layer
- Access identity and access management tools with ease
- Enjoy a simplified, consistent look and feel across Workspace ONE

Don't show this message on login
After completing the Security Settings, you will be presented with the Workspace ONE UEM Console Highlights pop-up.

1. Click on the Don't show this message on login check box.
2. Close the pop-up by clicking on the X in the upper-right corner.
iOS Device Enrollment

In this section, we are going to enroll an iOS device to complete the steps on the device side.

Download/Install AirWatch MDM Agent Application from App Store - IF NEEDED

NOTE - Checked out devices will likely have the AirWatch MDM Agent already installed. You may skip this step if your device has the AirWatch MDM agent installed.

At this point, if using your own iOS device or if the device you are using does NOT have the AirWatch MDM Agent Application installed, then install the AirWatch Application.
To Install the AirWatch MDM Agent application from the App Store, open the App Store application and download the free **AirWatch MDM Agent** application.

### Launching the AirWatch MDM Agent

Launch the **AirWatch Agent** app on the device.

*NOTE - If you have your own iOS device and would like to test you will need to download the agent first.*
Choose the Enrollment Method

Welcome to AirWatch!

AirWatch helps your IT Department to provide your device with secure access to resources.

The multi-step enrollment process begins with authentication.

Choose authentication method:

- Email Address
- Server Details
- QR Code

Click on the **Server Details** button.
Find your Group ID from AirWatch Console

1. To find the Group ID, hover your mouse over the Organization Group tab at the top of the screen. Look for the email address you used to log in to the lab portal. 
2. Your **Group ID** is displayed at the bottom of the Organization Group pop up.

**NOTE** - The Group ID is required when enrolling your device in the following steps.

Attach the AirWatch MDM Agent to the HOL Sandbox

Once the Agent has launched you can enroll the device. To do so, follow the below steps.

1. Enter **hol.awmdm.com** for the **Server** field.
2. Enter your **Group ID** for your Organization Group for the **Group ID** field. Your Group ID was noted previously in the **Finding your Group ID** step.
3. Tap the **Go** button.

**NOTE - If on an iPhone, you may have to close the keyboard by clicking Done in order to click the Continue button.**

**Authenticate the AirWatch MDM Agent**

On this screen, enter the **Username** and **Password** for the basic user account.

1. Enter **testuser** in the **Username** field.
2. Enter **VMware1!** in the **Password** field.
3. Tap the **Go** button.
The AirWatch Agent will now redirect you to Safari and start the process of enabling MDM in the device settings.

Tap on **Redirect & Enable** at the bottom of the screen.
Allow Website to Open Settings (IF NEEDED)

If you prompted to allow the website to open Settings to show you a configuration profile, tap Allow.

NOTE - If you do not see this prompt, ignore this and continue to the next step. This prompt will only occur for iOS Devices on iOS 10.3.3 or later.
Install the MDM Profile

Tap **Install** in the upper right corner of the Install Profile dialog box.
Install and Verify the AirWatch MDM Profile

Tap **Install** when prompted at the Install Profile dialog.

**NOTE - If a PIN is requested, it is the current device PIN. Provided VMware devices should not have a PIN.**
iOS MDM Profile Warning

You should now see the iOS Profile Installation warning explaining what this profile installation will allow on the iOS device.

Tap **Install** in the upper-right corner of the screen.
Trust the Remote Management Profile.

You should now see the iOS request to trust the source of the MDM profile.

Tap **Trust** when prompted at the Remote Management dialog.
iOS Profile Installation Complete

You should now see the iOS Profile successfully installed.

Tap **Done** in the upper right corner of the prompt.
AirWatch Enrollment Success

Your enrollment is now completed. Tap **Open** to navigate to the AirWatch Agent.
Accept the Authentication Complete Prompt

Authentication Complete

- You will receive company resources and settings assigned to your device by your IT department
- You will receive a notification if further action is required

Click on Done to continue.

Accept Notification Prompt (IF NEEDED)

Tap Allow if you get a prompt for Notifications.

Accept the App Installation (IF NEEDED)
You may be prompted to install a series of applications depending on which Module you are taking. If prompted, tap **Install** to accept the application installation.
VMware AirWatch REST API

In this section we will go through several REST APIs using both GET and POST commands. We will wrap up the module by un-enrolling the device using a DELETE API request.

Get the REST API Key from console

In this section, we will get the REST API Key.

Click on All Settings

1. Click Groups & Settings.
2. Click All Settings.

Return to the AirWatch console.
Navigate to REST API Settings

1. Click on **System**.
2. Expand the **Advanced** section.
3. Expand the **API** section.
4. Click **REST API**.

---

VMware AirWatch - Mobile Application Management and Developer Tools
1. Select **Override** for **Current Setting**.
2. Select **Enabled** for **Enable API Access**.
3. Using your mouse, highlight the API Key that is in **API Key** text box for the AirWatchAPI service. Right click on the text and select **Copy**.
1. You may need to scroll down to find the **Save** button.
2. Click **Save**.
3. Click the **Close** button in the top right corner.
Open Notepad on the Main Console Server

1. Click the Windows Start Button.
2. Type "Notepad" in the search field.
3. Click on Notepad in the Programs list.
Paste the API Key Into Notepad

1. In Notepad, click on **Edit**.
2. Click on **Paste**. This will paste the API key into notepad.

REST Client Setup

In this section, you will configure a REST Client application on the Main Console server. This application will allow you to easily send REST API calls to AirWatch without having to go through the process of actually creating an application. For this module we will be using an application called **Postman**.

Launch the Postman App

Double-click the **Postman** shortcut from the Desktop to launch Postman.

*NOTE* - The Postman application may take a second to launch after double-clicking, please wait while the application loads.
Configure Authorization Type as Basic Auth

1. Notice the **History** tab, which tracks the history of the REST APIs that you have used. You can reuse the entries from this section if you want to use the same API with any modifications.
2. Click the **Hide Sidebar** button to hide the History and Collections tabs, as we will not be utilizing them during the lab.
3. Select the **Authorization** tab.
4. Click the **Type** dropdown.

In this section, we will prepare the Postman REST client to test AirWatch REST APIs.
5. Select **Basic Auth**.

**Enter The User Credentials**

![Image of Notepad and URL options]

**NOTE** - If there are any existing values in the Username or Password field, remove them and use the details below:

1. Enter the AirWatch Administrator account name. This will be **your email address** that you used to sign into the Lab.
2. Enter the password as "**VMware1!**"
3. Click on **Show Password** to ensure that you have entered the correct password.

**Bring Up Notepad**

![Image of Notepad and options]

Select **Notepad** from the tray.
Select the API Key From Notepad

1. In Notepad, click on **Edit**.
2. Click on **Select All**.

Copy the API Key From Notepad

1. Click on **Edit**.
2. Click on **Copy**.

**Select the Postman App.**

Select **Postman** from the tray.

**Add the API Key to the Header**

1. Click on the **Headers** tab.
2. Enter "**aw-tenant-code**" in the Key field.
3. Click the **Value** field next to the aw-tenant-code Key you just entered to select it.
4. Click **Edit**.
5. Click **Paste**.

Now, we have our REST Client setup with Basic Authentication and another header with the AirWatch REST API key. In the following section, we will use this setup to make REST API calls.

**NOTE - The AirWatch APIs also require the Authorization header, which is built from the username and password provided in the Basic Authorization section. Postman will automation add this Authorization header when we send our first request.**
GET Commands - Enrolled Devices for a User

GET commands are usually used to get some information from the server. The GET commands are primarily targeted towards 'get'ting information from the database without making any change to the data.

The following API command requests information on the enrolled devices for a user.

Search Device API request

1. Confirm that GET is selected.

   NOTE - Please refer the section on how to copy and paste text in VLP in case you want to use that feature to enter URL.

3. Click the Send button.

Validate Search Device API response

NOTE - You may need to scroll down to find the Body section to view the full response.
1. Ensure that you get a status as **200 OK**. That indicates that your query is successful.

2. Click on the response format to **Pretty**.

3. Note the value of key **Id** under **Devices**. We will be using this in one of the POST API queries. You may use notepad to note down this value as you did with the REST API key.

4. Note the value of key **UserId** under **Devices**. We will be using this in the following REST APIs. You may use notepad to note down this value as you did with the REST API key.

**GET Commands - Enrollment User Details**

This API command retrieves details about an enrollment user. For this lab, since we used 'testuser' to enroll the device, we going to use the user id of 'testuser' to retrieve the enrollment user details.

**Enrollment User Details API request**

1. Confirm that **GET** is selected.

2. Enter the URL **https://hol.awmdm.com/api/system/users/{UserId}** in the URL field.
   
   Replace **{UserId}** with the UserId obtained from from the Validate Search Device API Response step.

3. Click on the **Send** button.
Validate Enrollment User API response

NOTE - You may need to scroll down to find the Body section to view the full response.

1. Ensure that you get a status as 200 OK. That indicates that your query is successful.
2. Confirm the information about the enrollment user.

GET Commands - Device Applications

In this step you will use the API to search for all Applications that are in the AirWatch App Catalog for Apple devices.

Application Search API request

1. Confirm that GET is selected.
3. Click on the Send button.

### Validate Application Search API response

In this step, you will use the API to search for all Applications that are in the AirWatch App Catalog for Apple devices.

1. Ensure that you get a status as 200 OK. That indicates that your query is successful.
2. If you have multiple entries of Application entities, scroll down through the response until you find the entry with ApplicationName as AW REST API

### POST Commands - Lock Device

POST REST API commands are usually intended to perform some action. In this section, we will 'post' some data to the database to make changes and we will verify those changes on the enrolled device.

**POST Lock Device**

This API command can remotely lock a device. Prior to performing this step, **please confirm that your enrolled device is on and unlocked.**
1. Change the command type to **POST**.
2. Enter the URL [https://hol.awmdm.com/api/mdm/devices/{deviceID}/lockdevice](https://hol.awmdm.com/api/mdm/devices/{deviceID}/lockdevice). Replace `{DeviceID}` with the DeviceId returned in the Validate Search Device API Response.
3. Click on the button **Send**.

**Confirm Device Lock**

![POST Command Example](image)

Confirm that the API Response status is a **202 Accepted**.

Confirm that the enrolled device is now locked.

**POST Commands - Send a Message**

This API command sends a push message to the enrolled device. In real world scenario, this API can be used to automate notifying managed devices about a certain event/action without requiring to login to the AirWatch UEM console and send push messages manually.

**Send Push Message to the enrolled device - Add Header for JSON**

![JSON Header Example](image)
In this section, we will explore how to make a JSON REST API request. We will add a header to the POST request so that API uses JSON format instead of XML that we have used so far.

1. Confirm that **POST** is selected.
2. Enter the URL `https://hol.awmdm.com/api/mdm/devices/{DeviceId}/sendmessage` in the URL field. Replace `{DeviceId}` with the DeviceId returned in the Validate Search Device API Response step.
3. Add a new header as "Content-Type".
4. Enter the value of the header as "application/json".

### Send Push Message to the enrolled device - Add Body

1. Click the **Body** tab.
2. Click on the **Raw** radio button to change the format.
3. Select **JSON (application/json)** from the dropdown.
4. Enter the following text in the body section. Please refer to the section *How to Copy and Paste in VLP* if you prefer copy and paste the following text:

   ```json
   {
   "MessageBody" : "AirWatch Test Push Message",
   "Application" : "AirWatch Agent",
   "MessageType" : "Push Notification"
   }
   ```

5. Ensure that you are not seeing X marks. If you do then double check quotes and commas in the text body to correct those errors.
6. Click on **Send**.
Confirm Message Delivery

Ensure that you get the API response status as **202 Accepted** is returned. You should now see a push notification with the text **AirWatch Test Push Message** on your device.

**DELETE Request**

In this section, you will see how to use a **HTTP DELETE** command. You will issue a single command to AirWatch to delete the device. Deleting a device will initiate an Enterprise Wipe (or un-enrollment) and will remove the device from the AirWatch database. This will **NOT** perform a factory reset on the device and will not in any way delete any data from the device that was there prior to enrolling the device into AirWatch.

**Creating the DELETE Request**
1. Ensure that you are selecting REST Query type as **DELETE**.
2. Use the following API to Delete the device from AirWatch UEM:
   
   ![API Image](https://hol.awmdm.com/API/mdm/devices/{DeviceId})
   
   Replace `{DeviceId}` with the deviceId returned in the Validate Search Device API Response step.

### Device Delete Confirmation

1. Select **form-data** to clear the API request body.
2. Click on **Send**.
3. Ensure that you get the API response status as **200 OK**.

The device will no longer be enrolled in AirWatch and the AW REST API app has been removed from the device along with any other applications that were present.

### Conclusion and Wrap Up

This concludes the AirWatch REST API Module. There are many more API's available which can be leveraged to automate many of the AirWatch UEM console actions without logging in to the console. REST APIs are powerful tools to perform bulk actions at the trigger of certain events to enhance the existing functionality of the AirWatch UEM solution.
Conclusion

In this lab, we saw how easy it is to use the AirWatch APIs to perform AirWatch Admin console actions externally without compromising on the security. Leveraging REST-based APIs also cloud offer several benefits to enterprises, including eliminated cost and time spent developing applications in-house. AirWatch APIs are fully able and ready to integrate with enterprise servers, programs and processes. Additionally, AirWatch APIs are efficient, run smoothly and are easily branded with enterprises.
Module 3 - Per-App VPN using VMware Tunnel (30 minutes)
Introduction

Leveraging Per-App VPN allows you to control which applications on a device have access to your VPN by automatically enabling disabling VPN access based on which applications are active. This prevents you from needing to provide a device wide VPN on your devices, which allow unintended apps or processes to access your VPN and ensures only authorized apps have access to your VPN.
Login to the Workspace ONE UEM Console

To perform most of the lab you will need to login to the Workspace ONE UEM Management Console.

Launch Chrome Browser

Double-click the Chrome Browser on the lab desktop.
Authenticate to the Workspace ONE UEM Administration Console

The default home page for the browser is https://hol.awmdm.com. Enter your Workspace ONE UEM Admin Account information and click the Login button.

NOTE - If you see a Captcha, please be aware that it is case sensitive!

1. Enter your Username. This is your email address that you have associated with your VMware Learning Platform (VLP) account.
2. Enter VMware1! for the Password field.
3. Click the Login button.

NOTE - Due to lab restrictions, you may need to wait here for a minute or so while the Hands On Lab contacts the Workspace ONE UEM Hands On Labs server.
Accept the End User License Agreement

Terms of Use

You must accept the following VMware End User License Agreement to use Workspace ONE UEM.

NOTE - The following steps of logging into the Administration Console will only need to be done during the initial login to the console.

You will be presented with the Workspace ONE UEM Terms of Use. Click the Accept button.
Address the Initial Security Settings

Security Settings

Password Recovery Question 1

Password Recovery Question *

Password Recovery Answer *

Confirm Password Recovery Answer *

Security PIN

A four-digit Security PIN must be entered. It is required in the console for some restricted actions (configured by authorized administrators in System Security settings).

Security PIN *

Confirm Security PIN *

After accepting the Terms of Use, you will be presented with a Security Settings pop-up. The Password Recovery Question is in case you forget your admin password and the Security PIN is to protect certain administrative functionality in the console.
1. You may need to scroll down to see the Password Recovery Questions and Security PIN sections.
2. Select a question from the Password Recovery Question drop-down (default selected question is ok here).
3. Enter VMware1! in the Password Recovery Answer field.
4. Enter VMware1! in the Confirm Password Recovery Answer field.
5. Enter 1234 in the Security PIN field.
6. Enter 1234 in the Confirm Security PIN field.
7. Click the Save button when finished.

Close the Welcome Message

Workspace ONE UEM Console Highlights

Powered by VMware AirWatch!

Workspace ONE is powered by VMware AirWatch Unified Endpoint Management (UEM) technology, a unified digital workspace platform delivering a single, secure experience for app management, single sign-on (SSO), and conditional access.

Workspace ONE UEM transforms your business so you can:

- Configure, manage and support devices from any endpoint
- Increase productivity with seamless access to any app
- Safeguard company data at every layer
- Access identity and access management tools with ease
- Enjoy a simplified, consistent look and feel across Workspace ONE

Don't show this message on login
After completing the Security Settings, you will be presented with the Workspace ONE UEM Console Highlights pop-up.

1. Click on the **Don't show this message on login** check box.
2. Close the pop-up by clicking on the X in the upper-right corner.
AirWatch Console Configuration - Publish VMware Tunnel

In this chapter you will create a Per-App VPN profile and deploy an Application configured to use the VPN Tunnel on iOS.

Create an iOS VPN Profile

In this step you will configure the iOS profile that will be delivered to the device to configure the VMware Tunnel Client on the device to allow only designated applications to access content on internal servers. If you completed the previous module already, "Introduction to AppConfig", then you have already created the Per-App VPN profile and you may use the iOS Per-App VPN profile created in that lab. You may still walk through these steps if you'd like.

Add a New Profile

1. Click Add.
2. Click Profile.
Select the OS the profile will be used for.

Add Profile

Select a platform to start:

Android
Apple iOS

Click Apple iOS

Configure the General Properties of the Profile

iOS Add a New Apple iOS Profile

General

1. Enter "Per-App VPN" as the Name.
2. Select All Devices (your@email.shown.here) as the Assigned Smart Group.
Add a VPN Payload

1. Click **VPN** from the Payload menu.
2. Click **Configure** to access the VPN payload settings.

Configure the VPN Payload
1. Select **VMware Tunnel** from the **Connection Type** dropdown.
2. Check the **Enable VMware Tunnel** box.
3. Click **Save & Publish**.

**Publish the VPN Profile**

![View Device Assignment Table]

Click **Publish**.

**Add the VMware Tunnel Client as a Public Application**

In order to leverage the VPN profile, the VMware Tunnel Client must be installed on your device. We can leverage AirWatch to deploy the client as a managed application to the device. This step will walk you through the process of adding the client application to the AirWatch Console to automatically install on enrolled devices. Please note, while it is required that the Tunnel client application is installed on any device using Per App Tunnel, it does not have to be a managed application. Users can download the VMware Tunnel client from the App Store.
Add a New Public Application

1. Click **Add**
2. Click **Public Application**

Search for the Application to Add

<table>
<thead>
<tr>
<th>Add Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Managed By</td>
</tr>
<tr>
<td>Platform</td>
</tr>
<tr>
<td>Source</td>
</tr>
<tr>
<td>Name*</td>
</tr>
</tbody>
</table>

Next Cancel
1. Select **Apple iOS** from the Platform dropdown.
2. Enter "**VMware Tunnel**" in the Name field.
3. Select **Next**.

**Select the VMware Tunnel Client from the Search Results**

Click **Select** on the **VMware Tunnel** application.
Click on Save & Assign

Click **Save & Assign**
Add Assignment

Click **Add Assignment**
Assign Groups and Delivery Method

1. Select Assignment Groups as **All Devices**
2. Select App Delivery Method as **Auto**
Configure the Application Policies

1. Scroll down to the bottom of the Policies section.
2. Select Enabled for Remove On Unenroll
3. Click Add
Save & Publish

Click **Save & Publish**
Publish the Application

Click **Publish**
Configure VMware Browser for Per-App VPN

Add the VMware Browser as a Public Application

Now that the Tunnel client is assigned to the appropriate group, this section walks through adding an application that is enabled to use Per App Tunnel. After enabling the setting that allows an application to use VPN, you must select the VPN profile that the app should use. This requires that any application you would like to leverage Per App VPN is pushed to the device from the AirWatch Console as a managed app. There is one exception to this, which is the Safari application on iOS. This is covered in detail in a later section of this lab.

This step will walk you through the process of adding an application from the Public App store that will be associated to the VPN profile you created.

Add a New Public Application

1. Click Add
2. Click Public Application
Search for the Application to Add

1. Select Apple iOS from the Platform dropdown.
2. Enter "VMware Browser" in the Name field.
3. Select Next.

Select the VMware Browser from the Search Results

Click Select on the VMware Browser result.
Save & Assign

Click **Save & Assign**
Add Assignment

Click **Add Assignment**
Assign Group and Delivery Method

1. Select Assignment Groups as **All Devices**
2. Select App Delivery Method as **Auto**

Configure the Application Policies

1. Scroll down to the bottom of the **Policies** section.
2. Select **Enabled** for **Remove On Unenroll**
3. Select **Enabled** for **App Tunneling**
4. Select the profile **Per-App VPN @(your@email.shown.here)**
5. Click **Add**

**Save & Publish**

Click **Save & Publish**
Publish the Application

Click Publish
iOS Device Enrollment

In this section, we are going to enroll an iOS device to complete the steps on the device side.

Download/Install AirWatch MDM Agent Application from App Store - IF NEEDED

NOTE - Checked out devices will likely have the AirWatch MDM Agent already installed. You may skip this step if your device has the AirWatch MDM agent installed.

At this point, if using your own iOS device or if the device you are using does NOT have the AirWatch MDM Agent Application installed, then install the AirWatch Application.
To Install the AirWatch MDM Agent application from the App Store, open the App Store application and download the free **AirWatch MDM Agent** application.

### Launching the AirWatch MDM Agent

Launch the **AirWatch Agent** app on the device.

**NOTE - If you have your own iOS device and would like to test you will need to download the agent first.**
Choose the Enrollment Method

Welcome to AirWatch!

AirWatch helps your IT Department to provide your device with secure access to resources.

The multi-step enrollment process begins with authentication.

Choose authentication method:

- Email Address
- Server Details
- QR Code

Click on the Server Details button.
Find your Group ID from AirWatch Console

1. To find the Group ID, hover your mouse over the Organization Group tab at the top of the screen. Look for the email address you used to log in to the lab portal.
2. Your Group ID is displayed at the bottom of the Organization Group pop up.

**NOTE** - The Group ID is required when enrolling your device in the following steps.

Attach the AirWatch MDM Agent to the HOL Sandbox

Once the Agent has launched you can enroll the device. To do so, follow the below steps.

1. Enter **hol.awmdm.com** for the Server field.
2. Enter your **Group ID** for your Organization Group for the **Group ID** field. Your Group ID was noted previously in the **Finding your Group ID** step.
3. Tap the **Go** button.

**NOTE** - *If on an iPhone, you may have to close the keyboard by clicking Done in order to click the Continue button.*

**Authenticate the AirWatch MDM Agent**

On this screen, enter the **Username** and **Password** for the basic user account.

1. Enter **testuser** in the **Username** field.
2. Enter **VMware1!** in the **Password** field.
3. Tap the **Go** button.
Redirect to Safari and Enable MDM Enrollment in Settings

Enable Device Management

To enable your device, you will be redirected to Safari and Settings

Why?

- Access your company resource
- Remove company data in the event of loss or theft

The AirWatch Agent will now redirect you to Safari and start the process of enabling MDM in the device settings.

Tap on **Redirect & Enable** at the bottom of the screen.
Allow Website to Open Settings (IF NEEDED)

If you prompted to allow the website to open Settings to show you a configuration profile, tap **Allow**.

**NOTE - If you do not see this prompt, ignore this and continue to the next step. This prompt will only occur for iOS Devices on iOS 10.3.3 or later**
Install the MDM Profile

Tap **Install** in the upper right corner of the Install Profile dialog box.
Install and Verify the AirWatch MDM Profile

Tap **Install** when prompted at the Install Profile dialog.

**NOTE - If a PIN is requested, it is the current device PIN. Provided VMware devices should not have a PIN.**
iOS MDM Profile Warning

Installing this profile will allow the administrator at “https://hol.awmdm.com/DeviceServices/AppleMDM/Processor.aspx” to remotely manage your iPad.

The administrator may collect personal data, add/remove accounts and restrictions, and list, install and manage apps on your iPad.

You should now see the iOS Profile Installation warning explaining what this profile installation will allow on the iOS device.

Tap Install in the upper-right corner of the screen.
You should now see the iOS request to trust the source of the MDM profile. Tap Trust when prompted at the Remote Management dialog.
iOS Profile Installation Complete

You should now see the iOS Profile successfully installed.

Tap **Done** in the upper right corner of the prompt.
AirWatch Enrollment Success

Your enrollment is now completed. Tap **Open** to navigate to the AirWatch Agent.
Accept the Authentication Complete Prompt

Authentication Complete

- You will receive company resources and settings assigned to your device by your IT department
- You will receive a notification if further action is required

Click on Done to continue.

Accept Notification Prompt (IF NEEDED)

Tap Allow if you get a prompt for Notifications.

Accept the App Installation (IF NEEDED)
You may be prompted to install a series of applications depending on which Module you are taking. If prompted, tap **Install** to accept the application installation.
Testing Per App VPN

Now that the device is enrolled and has received the settings we configured in the AirWatch Console, we are ready to begin testing the Per-App VPN functionality.

Testing Per App VPN on iOS

The applications assigned in the previous steps should push down during enrollment. The VMware Tunnel and VMware Browser applications should be installed on your device.

Launch the VMware Browser

Press the Home button on the iPad to return to the Launchpad. **Swipe right** to see the downloaded applications if needed.

Tap the **VMware Browser** icon to launch the application. If prompted, select **OK** to allow the Browser to send your device push notifications.
Access the Internal Website with VMware Browser

1. The application will launch and you will see the VPN icon appear indicating the connection is active. The application will now connect to AirWatch and retrieve the settings for your Sandbox Organization Group. These settings include a default homepage that has been pre-configured for this lab. This website is available on an internal web server but not accessible from the public internet.

2. The website will load and you'll see the Welcome message.

Attempt to Access the Website From Safari

We will now show that although the VPN connection is active, other applications on the device will not be able to access the Tunnel or the internal resources.
Select the URL from the VMware Browser

1. **Press & hold** the Navigation Bar in the AirWatch Browser.
2. Choose **Select All** to highlight the URL for the internal site.

Copy the URL from the VMware Browser

Select **Copy**.
Open Safari

Return to the launchpad by pressing the Home button on the iPad. Open **Safari** by selecting the icon form the Launcher.
Paste the URL Into the Safari Browser

1. Open a new tab by selecting the + sign on the navigation bar.
2. Select the entry box on the navigation bar.
3. Press & hold for a count of two then release on the entry box and select **Paste**.
4. Select **Go** on the keyboard.
Notice that the website does not load in the Safari browser due to DNS failure. The website is published to an internal DNS that can only be accessed when the VPN connection is being used. Although the VPN connection may remain active (look for the VPN icon in the status bar), Safari is not designated as an application that is allowed to use the Per-App VPN Tunnel. You may have multiple VPN configurations and multiple apps assigned for each VPN. Most Public applications (apps using Cocoa framework) are compatible with per-app VPN on iOS.
Safari Domain Profile Configuration

In this chapter you create a Per-App VPN profile and deploy an Application configured to use the VPN Tunnel on iOS.

Add a New Version to the iOS VPN Profile

In this step you will update the iOS profile created in the first step to include Safari domains.

Update the Per-App VPN Profile

Return to the AirWatch Console.

1. Click Devices.
2. Click Profiles & Resources.
3. Click Profiles.
4. Select the edit icon next to the Per-App VPN profile.
Add Version to update the existing profile

1. Click **Add Version** to allow editing.
2. Select the **VPN** payload on the left hand side.
Configure Safari Domains

1. Enter "airwlab.com" in the Safari Domains list.
2. Click "Save & Publish".

**NOTE** - They syntax for Safari Domains does not require a wildcard character. Enter only the domain hostname to whitelist the entire domain to initiate VPN in Safari.
Publish the updated VPN Profile

View Device Assignment

<table>
<thead>
<tr>
<th>Assignment Status</th>
<th>Friendly Name</th>
<th>User</th>
<th>Platform / OS / Model</th>
<th>Phone Number</th>
<th>Org</th>
</tr>
</thead>
<tbody>
<tr>
<td>📕 Unchanged</td>
<td>[redacted]</td>
<td>[redacted]</td>
<td>Apple iOS / iOS 10.1.1 / iPad</td>
<td>your</td>
<td></td>
</tr>
</tbody>
</table>

Items 1-1 of 1

Click **Publish**.
Un-enrolling Your Device

You are now going to un-enroll the iOS device from Workspace ONE UEM.

**NOTE - The term "Enterprise Wipe" does not mean reset or completely wipe your device. This only removes the MDM Profiles, Policies, and content which the AirWatch MDM Agent controls.**

It will NOT remove the AirWatch Agent application from the device as this was downloaded manually before Workspace ONE UEM had control of the device.

**Enterprise Wipe (un-enroll) your iOS device**

Enterprise Wipe will remove all the settings and content that were pushed to the device when it was enrolled. It will not affect anything that was on the device prior to enrollment.

To Enterprise Wipe your device you will first bring up the Workspace ONE UEM Console in a web browser. You may need to re-authenticate with your credentials (VLP registered email address and VMware1 as the password).

1. Click **Devices** on the left column.
2. Click **List View**.
3. Click the **checkbox** next to the device you want to Enterprise Wipe.
NOTE - Your Device Friendly Name will very likely be different than what is shown. It will, however, be in the same location as shown on image in this step.

Find the Enterprise Wipe Option

1. Click More Actions. NOTE - If you do not see this option, ensure you have a device selected by clicking the checkbox next to the device.
2. Click Enterprise Wipe under Management.
Enter your security PIN

After selecting **Enterprise Wipe**, you will be prompted to enter your Security PIN which you set after your logged into the console (1234).

1. Enter **1234** for the **Security PIN**. You will not need to press enter or continue, the console will confirm your PIN showing "Successful" below the Security PIN input field to indicate that an Enterprise Wipe has been requested.

**NOTE** - If 1234 does not work, then you provided a different Security PIN when you first logged into the Workspace ONE UEM Console. Use the value you specified for your Security PIN.

**NOTE - If the Enterprise Wipe does not immediately occur, follow the below steps to force a device sync:**

1. On your device, open the AirWatch **Agent** application.
2. Tap the **Device** section (under **Status**) in the middle of the screen.
3. Tap **Send Data** near the top of the screen. If this does not make the device check in and immediately un-enroll, continue to Step #4.
4. If the above doesn't make it immediately un-enroll, then tap **Connectivity [Status]** under Diagnostics.
5. Tap **Test Connectivity** at the top of the screen.

**NOTE - Depending upon Internet connectivity of the device and responsiveness of the lab infrastructure, this could take a couple of minutes or more if there is excessive traffic occurring within the Hands On Lab environment.**

Feel free to continue to the "**Force the Wipe**" step to manually uninstall the Workspace ONE UEM services from the device if network connectivity is failing.

**Verify the Un-Enrollment**
Press the Home button on the device to go back to the home screen. The applications pushed through Workspace ONE UEM should have been removed from the device.

**NOTE - The applications and settings pushed through Workspace ONE UEM should have been removed. The Agent will still be on the device because that was downloaded manually from the App Store. Due to lab environment settings, it may take some time for the signal to traverse through the various networks out and back to your device. Continue on to the next step to force the wipe if the needed.**
Force the Wipe - IF NECESSARY

If your device did not wipe, follow these instructions to ensure the wipe is forced immediately. Start by opening the iOS **Settings** app.
1. Tap **General** in the left column.
2. Scroll down to view the **Device Management** option.
3. Tap **Device Management** at the bottom of the list of General settings.

**Force the Wipe - IF NECESSARY**

Tap the **Workspace Services** profile that was pushed to the device.
Force the Wipe - IF NECESSARY

1. Tap **Remove Management** on the Workspace Services profile.  
   *NOTE - If prompted for a device PIN, enter it to continue. VMware provisioned devices should not have a device PIN enabled.*

2. Tap **Remove** on the Remove Management prompt.
After removing the Workspace Services profile, the device will be un-enrolled. Feel free to return to the Verify the Un-Enrollment step to confirm the successful un-enrollment of the device.
Testing Safari Domains with Per App Tunnel

Now that the VPN profile is updated to include the domain tested in the first example in the Safari Domains list, we can confirm these settings have updated on the device and test in the native Safari application.

Confirm the VPN Configuration Has Updated

This section will walk-through how to confirm that the VPN configuration has successfully updated on your device.

Open Device Settings

Tap **Settings**.
### Open VPN Settings

<table>
<thead>
<tr>
<th>Settings</th>
<th>General</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Airplane Mode" /></td>
<td><img src="image" alt="Airplane Mode" /></td>
</tr>
<tr>
<td><img src="image" alt="Wi-Fi" /> vmwareguest</td>
<td><img src="image" alt="Wi-Fi" /> vmwareguest</td>
</tr>
<tr>
<td><img src="image" alt="Bluetooth" /> On</td>
<td><img src="image" alt="Bluetooth" /> On</td>
</tr>
<tr>
<td><img src="image" alt="Cellular Data" /> Not Connected</td>
<td><img src="image" alt="Cellular Data" /> Not Connected</td>
</tr>
<tr>
<td><img src="image" alt="VPN" /> Not Connected</td>
<td><img src="image" alt="VPN" /> Not Connected</td>
</tr>
<tr>
<td><img src="image" alt="Notifications" /></td>
<td><img src="image" alt="Notifications" /></td>
</tr>
<tr>
<td><img src="image" alt="Control Center" /></td>
<td><img src="image" alt="Control Center" /></td>
</tr>
<tr>
<td><img src="image" alt="Do Not Disturb" /></td>
<td><img src="image" alt="Do Not Disturb" /></td>
</tr>
<tr>
<td><img src="image" alt="General" /></td>
<td><img src="image" alt="General" /></td>
</tr>
<tr>
<td><img src="image" alt="Display &amp; Brightness" /></td>
<td><img src="image" alt="Display &amp; Brightness" /></td>
</tr>
<tr>
<td><img src="image" alt="Wallpaper" /></td>
<td><img src="image" alt="Wallpaper" /></td>
</tr>
<tr>
<td><img src="image" alt="Sounds" /></td>
<td><img src="image" alt="Sounds" /></td>
</tr>
<tr>
<td><img src="image" alt="Siri" /></td>
<td><img src="image" alt="Siri" /></td>
</tr>
<tr>
<td><img src="image" alt="Passcode" /></td>
<td><img src="image" alt="Passcode" /></td>
</tr>
<tr>
<td><img src="image" alt="Battery" /></td>
<td><img src="image" alt="Battery" /></td>
</tr>
</tbody>
</table>

1. Tap **General**.
2. Scroll down to find the VPN section.
3. Tap **VPN**.
Select Your VPN Configuration

1. Tap **VPN Configuration #XXXXXX** from your Per-App VPN profile.
View Included Per-App VPN Apps

Note all managed applications from the AirWatch console enabled to use Per-App VPN and domains listed in Safari Domains in the VPN profile will appear in this list.

Whitelisting a domain in the Safari Domains list will initiate a VPN connection on demand whenever the user browses to a site within this domain.

NOTE - Wildcards are not required when whitelisting a Safari Domain. The entire domain will automatically be whitelisted for VPN On Demand when added to VPN profile.

Attempt to Access the Website From Safari

We will now show that browsing to a site in the domain added to the "Safari Domains" list will initiate a VPN connection.
Open Safari

Return to the launchpad by pressing the Home button on the iPad. Open Safari by selecting the icon form the Launcher. The VPN icon should not be displayed in the toolbar.
1. Open a new tab by selecting the + sign on the navigation bar.
2. Select the entry box on the navigation bar.
3. Press & hold for a count of two then release on the entry box and select Paste or type "internal.airwlab.com"
4. Select Go on the keyboard.
Browse to the Internal Webpage

Notice that the website now loads in the Safari browser after the VPN profile is updated to include airwlab.com in the Safari Domains list, whitelisting the domain for Per App VPN. The website is published to an internal DNS that can only be accessed when the VPN connection is being used.
Conclusion

This lab module reviewed how to leverage native Per-App VPN capabilities by publishing Per-App VPN profiles to your devices to ensure that only authorized apps are accessing your VPN. This prevents users from needing to manually start and end VPN connections based on what apps they are accessing and provides an extra layer of security to your corporate resources by ensuring non-authorized apps are not able to connect to your VPN.

This concludes this lab module.
Module 4 - Introduction to AirWatch Android SDK (45 minutes)
Introduction

The AirWatch Software Development Kit (SDK) for Android allows you to enhance your enterprise applications with MDM capabilities. By incorporating AirWatch SDK code within your Android app project, you can use AirWatch information such as enrollment or compromised status to add a layer of security and business logic however you see fit within your application.

The Android SDK has two primary components or libraries:

1. **Client SDK** - The client SDK is a lightweight library for retrieving basic management and device information such as compromised status, environment info, and user information.
2. **AWFramework** - The AWFramework is a heavier library for more advanced SDK functionality such as application proxy and tunneling, integrated authentication, and encryption functions.

In this lab, we are not going into specifics of each of the components. Rather, this is a walkthrough of how to setup AirWatch Android SDK and deploy the SDK enhanced app to a managed device and validate the SDK integration.

Requirements

Before integrating AirWatch SDK into an app, let us discuss the prerequisites.

1. **Device Operating System** - Android 4.0+ / Ice Cream Sandwich / APILevel14+
2. **IDE** - Android Studio with the Gradle Android Build System *(Gradle)* v1.3.0+
3. **AirWatch Anchor App** - AirWatch Agent v5.3+ for Android. The anchor app facilitates communication between the Enterprise Android App and the AirWatch environment.
Connect to Windows 10 VM

We have provided you a Windows 10 VM to complete the necessary steps for this lab. Let's connect to it to complete the steps in the following section.

Connect to the Windows 10 VM

Double-click the **Win10-01a.rdp** shortcut on the lab desktop.

If prompted, the login credentials for the Windows 10 VM are:

- Username: `corp\holuser`
- Password: `VMware1!`
Explore AirWatch SDK for Android using Android Studio

This section will give an overview of the Android SDK for Android and walkthrough the process of building the sample project provided in Android Studio.

Whitelisting the Signing Key in AirWatch

The AirWatch SDK for Android offers feature enhancements for apps deployed as Internal apps as well as Public Apps deployed via Play Store. However, for the SDK function calls to work, you must ensure your application signing key is whitelisted with the AirWatch environment. Depending upon the mode of app deployment, the process changes slightly.

Internal Apps

For applications which are uploaded locally for internal distribution (either during production or testing), the following steps are taken to establish trust:

1. **Sign the APK file** with Android Studio Keystore. If you use the **Debug** Keystore from Android Studio, you can whitelist the app for debugging from the local IDE. However, if you use **Release** Keystore, the app is not eligible for debugging.
2. **Upload** the signed APK file to **AirWatch Admin Console**.
3. The AirWatch Admin Console **extracts** the application's **public signing key**.
4. The AirWatch Admin Console **whitelists** the application's public signing key with the **AirWatch Agent**.
5. When the application is **launched**, it calls AirWatch SDK for **initial configuration**.
6. The AirWatch Agent **validates** the signing key by comparing it to the one uploaded in the AirWatch Admin Console.

Public Apps

The automatic key whitelisting feature is currently only supported with Internal Apps. However, the trust for the Public Apps using AirWatch SDK can be established by manually whitelisting the public signing key. If the app developer shares the signing key then AirWatch can follow steps similar to the Internal apps flow for whitelisting.

Integrate AirWatch SDK for Android into the Sample App

In this section, we will setup the Android SDK Sample app to use AirWatch Android SDK. We will use Android Studio as Integrated Development Environment (IDE) to include the
SDK libraries into the sample app project. After that, we will build, sign, and export the APK to upload into the AirWatch admin console.

**Launch Android Studio**

From the Task bar, click on the Android Studio icon to launch it.

*NOTE - Android Studio may take several seconds to launch, please be patient while it opens.*

**Open the Sample App Project from Recent Projects**

Click **Open an existing Android Studio Project**.
Open the Sample App Project

1. From the windows, select C:\Users\holuser\Documents\HOL\Android SDK.
2. Select Android SDK 16.02.
3. Select sample code.
4. Select airwath-sdk-test-app.
   NOTE - Android Studio can auto-detect the Android Project files and will put Android Studio icon before for easier identification and access.
5. Click OK.
Open the libs folder of the project and validate the required SDK libs

1. Select the View **Project** from the drop down if it is not already selected.
2. Expand the root **airwatch-sdk-test-app**.
3. **Right click** on the folder **libs**.
4. Select the option **Show in Explorer**.
Open the libs folder in the File Explorer

Double-click the **libs** folder.

**Validate the Necessary Library Files Exist**

1. Validate that you have **AirWatchSDK-16.02.jar** file, **AWFramework 16.02.aar** and **gson-2.4.jar** file included in the libs folder. You will see a lot of other libraries but these three are the primary libraries that are used for this lab.

**NOTE** - **If you do not see these libraries in the libs folder, then you can add those from the SDK package. These files are located in** `\Documents\HOL\Android SDK\Android SDK v16.02\libs\AWFramework and ClientSDK folder`. The following step demonstrates how to add the gson library to the project **libs** folder.
Add the gson library to the project libs folder

For any SDK app, the developer needs to add the related libraries to the libs folder of the project, otherwise the project might not build correctly. To limit the scope of this lab, most of the files are already added in the correct folder for you. In this section, we are going to add the gson library as an example.

For troubleshooting any build errors, we should always check if the required libraries are added or not as the first troubleshooting step.

**NOTE - Perform this step ONLY if you don't see gson library in your project's libs folder. Follow similar steps for AirWatchSDK jar file and/or AWFramework aar file in case they are missing in the libs folder.**

1. From the taskbar at the bottom of the Desktop, click on the icon to launch File Explorer.
2. In the new Window, navigate to Documents\HOL\Android SDK\Android SDK v16.02\libs\ClientSDK.
3. **Right Click** on gson-2.4 executable JAR file.
4. Click on **Copy**.
5. In another Window for libs folder (if you closed it, then reopen from Android Studio as described in the step above), **Right Click** on click on **Paste** to add the gson library. Close or minimize the Explorer window.

**Build the Sample App Project**

![Build Process](image.png)

To limit the scope of this lab, all the coding has already been implemented in the project. We have also verified that the project has the required libraries and added the gson library. Now it is ready to be built.

Back to the Android Studio window,

1. Select the root **airwatch-sdk-test-app**.
2. From the toolbar, select **Build**.
3. Select the option **Build > Rebuild Project**. This will start the build.
4. Select the perspective **Gradle Console** From the bottom right.
5. Scroll to the bottom of the Gradle Console.
6. The build time may vary depending on the VM configuration but after some time (approximately 2 minutes), you should see the status as **BUILD SUCCESSFUL**.

**Generate Signed APK**

1. Select the root **airwatch-sdk-test-app** (if not done already).
2. From the toolbar, select **Build**.
3. Select the option **Generate Signed APK...**
Enter Master Password for keychain

For this lab, keychain and keychain alias is already setup for you and is encrypted with a Master Password. When prompted,

1. Enter the password as "**VMware1!**".
2. Click on **OK**.
3. Keep all the defaults and click on **Next**.

**NOTE - The keychain is located at C:/Users/holuser/Documents/HOL/Android SDK/awks.jks and alias is auto-populated when you make the selection.**
Select APK Destination Folder

1. Click on the Browse icon.
2. Click on the Desktop icon to select the destination folder as Desktop.
3. Click on OK to continue.

For easier access, we are going to save the application on the Desktop.
Select Build Type

1. Select the Build Type as **release**.
   
   **NOTE** - **If we select the build type as debug, it will allow us to run the app in the debug mode by connecting the device to the Computer and troubleshoot via Android Studio debugger. However, to limit the scope of this lab, we are going to sign the app with the release key.**

2. Check the **V1 (Jar Signature)** checkbox.
3. Click **Finish** to start the process. Depending on machine resources, it will take anywhere from 2 to 3 minutes to complete the APK generation process.

While we are waiting to generated the signed APK, let us go ahead and create the SDK profile which we will assign to our sample app while uploading it to the AirWatch Console.
Login to the Workspace ONE UEM Console

To perform most of the lab you will need to login to the Workspace ONE UEM Management Console.

Launch Chrome Browser

Double-click the Chrome Browser on the lab desktop.
Authenticate to the Workspace ONE UEM Administration Console

The default home page for the browser is https://hol.awmdm.com. Enter your Workspace ONE UEM Admin Account information and click the Login button.

NOTE - If you see a Captcha, please be aware that it is case sensitive!

1. Enter your Username. This is your email address that you have associated with your VMware Learning Platform (VLP) account.
2. Enter VMware1! for the Password field.
3. Click the Login button.

NOTE - Due to lab restrictions, you may need to wait here for a minute or so while the Hands On Lab contacts the Workspace ONE UEM Hands On Labs server.
Accept the End User License Agreement

Terms of Use

NOTE - The following steps of logging into the Administration Console will only need to be done during the initial login to the console.

You will be presented with the Workspace ONE UEM Terms of Use. Click the Accept button.
Address the Initial Security Settings

Security Settings

Password Recovery Question 1

Password Recovery Question *

Password Recovery Answer *

Confirm Password Recovery Answer *

Security PIN

A four-digit Security PIN must be entered. It is required in the console for some restricted actions (configured by authorized administrators in System Security settings).

Security PIN *

Confirm Security PIN *

After accepting the Terms of Use, you will be presented with a Security Settings pop-up. The Password Recovery Question is in case you forget your admin password and the Security PIN is to protect certain administrative functionality in the console.
1. You may need to scroll down to see the Password Recovery Questions and Security PIN sections.
2. Select a question from the Password Recovery Question drop-down (default selected question is ok here).
3. Enter VMware1! in the Password Recovery Answer field.
4. Enter VMware1! in the Confirm Password Recovery Answer field.
5. Enter 1234 in the Security PIN field.
6. Enter 1234 in the Confirm Security PIN field.
7. Click the Save button when finished.

Close the Welcome Message

Workspace ONE UEM Console Highlights

Powered by VMware AirWatch!

Workspace ONE is powered by VMware AirWatch Unified Endpoint Management (UEM) technology, a unified digital workspace platform delivering a single, secure experience for app management, single sign-on (SSO), and conditional access.

Workspace ONE UEM transforms your business so you can:

- Configure, manage and support devices from any endpoint
- Increase productivity with seamless access to any app
- Safeguard company data at every layer
- Access identity and access management tools with ease
- Enjoy a simplified, consistent look and feel across Workspace ONE

Don't show this message on login
After completing the Security Settings, you will be presented with the Workspace ONE UEM Console Highlights pop-up.

1. Click on the **Don't show this message on login** check box.
2. Close the pop-up by clicking on the X in the upper-right corner.
VMware AirWatch Console configuration for the SDK Sample App

In this section, we will modify the default SDK profile and assign it to the sample app. If we have more than one set of configurations then we can create custom SDK profiles and assign them individually. However, to limit the scope of this lab, we are going to change only the default profile.

The profile payloads that we are targeting for this lab are, **Authentication, Custom Settings** and **AirWatch App Tunnel**. We will examine how these payloads take effect in the app by sending the configuration over the air. We will discuss each payload and the use case in the individual steps.

Configure the Default SDK Profile in the AirWatch Console

1. Click on **Apps & Books**.
2. Click on **All Apps & Books Settings**.
Navigate to Security Policy

1. Under the Apps section, expand Settings And Policies.
2. Click Security Policies.
The Authentication payload enables the sample app to populate a prompt to authenticate upon launch. The sample app is already equipped with the all the code required to render the authentication box and this payload is used to specify related attributes e.g. type, timeout, allowed attempts etc. The use case here is to require the end user to authenticate in order to use app functionality and data. This restricts the exposure of sensitive resources to unmanaged/ unauthorized users.

While there are multiple combinations possible to fulfill the corporate security requirements, we are going to use the following configuration for Authentication.

1. Change the **Current Settings** to **Override**.
2. Select the **Authentication Type** as **Passcode**.
   
   **NOTE - We are only enabling Authentication Type as Passcode, however, Single Sign on will be disabled.**
3. Select the **Authentication Timeout** to **1 minute(s)**.
4. Select **Minimum Passcode Length** to **6**.

Keep all the other options to default.
AirWatch App Tunnel will allow the application to access backend resources to gather the required data and certain functionality. The advantage of using this payload is that we do not need to enable device level VPN which could potentially expose the internal resources to any unintended third party apps. On top of that, we can also restrict the domains for which the traffic will be tunneled to give more granular network access control.

Now, we will configure the payload to use AirWatch Tunnel already setup at a higher Organization group. We will restrict the traffic to *.airwlab.com to tunnel only.

1. Scroll down until you see the option for **AirWatch App Tunnel**.
2. Click on **Enabled** for **AirWatch App Tunnel** if it is not selected already.
3. Ensure that **Host Name** is **holtunnel.airwlab.com**.
4. In the section for **App Tunnel URLs** enter ".airwlab.com".

**NOTE - Using a Per App VPN Profile is another way of leveraging the AirWatch App Tunnel for proxying.**
Save Security Policy settings

1. Scroll down to the bottom.
2. Click Save. You should see Saved Successfully at the top which confirms that configuration is saved.

Enable Custom Settings

1. Click Settings under Settings And Policies.
2. Click Override for Current Setting.
4. Click Custom Settings to expand the section.
Configure Custom Settings

Custom settings allows the AirWatch admin to push down values which are variable across the organization group structure. e.g. the values such as username are not available before the device is enrolled. Another example would be to send down an authentication URL such as SAML auth. endpoint which may be different for different organization groups depending on the physical location.

By using Custom Settings, an AirWatch admin can send down either hardcoded values (such as a URL) or they can leverage the look up values available within AirWatch console. e.g. when we push down the look up value {EnrollmentUser}, that value will get replaced by the actual enrollment user when the app is installed on a managed device.

In this section, we are going to send one hardcoded value (URL) and one lookup value ({EnrollmentUser})
1. Scroll down to so you can view the **Custom Settings** input field.
2. Type in the following in **Custom Settings** input field: "**URL: http://internal.airwlab.com**" and "**username:{EnrollmentUser}**"
3. Click **Save**. You should see **Saved Successfully** at the top which confirms that configuration is saved.
4. Click on **X** at the top right to close the window.

**NOTE - An app developer can define variables for the values pushed down via Custom Settings and then those variable would be replaced by actual values in runtime.**

**Configure the SDK Sample App in the AirWatch Console**

Now we should have a SDK profile ready to be applied to the app. In this section, we will upload the app, assign the SDK profile we just configured and then setup the deployment option.

**Add Internal Application**

1. Click on **Add** in top right.
2. Click on **Internal Application**.
Upload the SDK Sample App to the AirWatch Admin Console

Add Application

Organization Group ID

Application File

Click **Upload**.

Select Choose File

Click **Choose File**.
Navigate to AirWatch SDK Sample App

1. From the left pane, select the folder **Documents**.
2. Select the app that we signed and exported from Android studio: **AirWatch SDKTest-release.apk**.
3. Click on **Open**.
4. Click on **Save**.
5. Click on **Continue** at the bottom of the Add Application window.
Save the Uploaded File

Click **Save**.
Click **Continue**.
Select enhancement mode as SDK

1. Click the More dropdown tab. Now to select the SDK profile, click on the dropdown More.
2. Select the option enhancement mode as SDK.
Select Android Default Settings as SDK profile

1. Since we did not create a custom profile but rather changed the default profile, from the dropdown we are going to select **Android Default Settings @Global**. We are not concerned with Application Profile for this lab.

2. Click on **Save & Assign**.
Update Assignment

Click on + Add Assignment.
Add Smart Group and Push Mode

1. Select **All Devices (your@email.shown.here)** for the Assignment Group.
2. Select **Auto** for App Delivery Method.
3. Click **Add**.
Save and Publish the App

1. Validate that now you the assignment contains the All Devices group.
2. Click Save & Publish.
Confirm Device Assignment and Publish

Click **Publish**.
Enroll an Android Device

Enrolling an Android Device with the Basic Account

In this chapter you will be enrolling an Android device to install the SDK Sample App and validate the functionality.

NOTE - The device screenshots may differ slightly based on the device model you are using for the lab.

Download/Install AirWatch MDM Agent Application from App Store

At this point, if the device you are using does NOT have the AirWatch MDM Agent Application installed, then install the AirWatch MDM Agent Application from the Google Play store.

To Install the AirWatch MDM Agent application from the Google Play Store, open the Play Store application and download the free AirWatch MDM Agent application.

NOTE - You will need to "Accept" the necessary access policies.
Launching the AirWatch MDM Agent

Launch the AirWatch Agent app on the device. If you have your using your own Android device and would like to test you will need to download the agent first.
Select AirWatch MDM Agent Authentication Method

Welcome to AirWatch!

AirWatch helps your IT Department to provide your device with secure access to enterprise resources.

You will be guided through a three part process to authenticate, secure and configure your device.

Choose Authentication Method:

- EMAIL ADDRESS
- SERVER DETAILS
- QR CODE
Once the Agent has launched you can enroll the device. To do so, you must first select the AirWatch authentication method.

Tap on **Server Details**.

**Finding your Group ID**

1. To find the Group ID, hover your mouse over the GroupID tab at the top of the screen. This may be shown in the form of the original email address used to create the sandbox.
2. The GroupID will be displayed under the Organization Group name. The GroupID is required when enrolling your device.

*NOTE - This screen shot shows an example of what a Group ID is, and not what your Group ID is actually.*

**Attach the AirWatch MDM Agent to the HOL Sandbox**

1. [Server URL]
2. [Group ID]
3. [Continue]
After selecting the Server authentication method, you need to supply the information to authenticate. To do so, follow the below steps.

2. For Group ID, enter the Group ID for your Organization Group. This was noted previously in the "Enroll your Android Device" step.
3. Tap the Continue button.

**NOTE - If on an iPhone, you may have to close the keyboard by clicking "Done" in order to click the "Continue" button.**

### Authenticate the AirWatch MDM Agent

On this screen, enter the credentials for the basic user we created in the earlier step.

1. Type in the Basic User Account Username. This should be 'testuser'.
2. Type in the Basic User Account Password. This should be 'VMware1!'.
3. Tap the Continue button.
Android Authentication Complete

You should now see a screen stating the Authentication is complete and the following steps will be used to ensure the device is compliant with defined policies and profiles.

Tap the **Get Started** button on the device screen.
Grant Permissions to AirWatch MDM Agent

At this point, the AirWatch MDM Agent is ready to install the profile.

When the next screen appears on your device, you will need to tap the **Activate** button. Tap the **Continue** button in the AirWatch MDM Agent wizard on your device.
Activate Device Administrator on Android

You should now see the "Activate device administrator" screen on your device. You will now be taken to the Profile installation Screen.

Tap the **Activate** button.
Install Enterprise Service

Your IT department requires you to install a service to unlock enterprise capabilities on your device.

Note:
Click **Continue** to install this app and **Activate** it as device administrator.

---

You are now taken back to the AirWatch MDM Agent wizard and notified you will be installing the Enterprise Service.

When the next screen appears on your device, you will need to tap the **Activate** button.

Tap the **Continue** button in the AirWatch MDM Agent wizard on your device.
Complete Action Using - IF NEEDED

If you see a "Complete Action Using" pop-up on your Android device, then a specific app package installer has not been selected as default.

1. Check the **Use by default for this action** box.
2. Select the Android **Package Installer**.

**NOTE - If you do not get this notification, ignore this step and go to the next step.**

AirWatch Service Installer

Do you want to install this application?

Allow this application to:

- **System tools**
  change Wi-Fi state, write sync settings

[Install]
You should now see the "AirWatch [device] Service Installer" screen on your device.

**NOTE - In the example here, the device used is an LG Phone. Hence, the service is the "AirWatch LG Service".**

Tap the **Install** button.

**AirWatch Admin Service Installer**

---

You should now see the "AirWatch [device] Admin Service Installer" screen on your device.

**NOTE - In the example here, the device used is an LG Phone. Hence, the service is the "AirWatch LG Admin Service".**

Tap the **Activate** button.
Configure Enterprise Resources

Part 3: Configure

You have successfully completed authentication and securing your device.

You will now configure enterprise resources assigned to your device by your IT department.

Tap the **Continue** button in the AirWatch MDM Agent configuration wizard on your device.

You should now see the device screen on "Part 3: Configure" stating the authentication and securing of the device was successful.

You now would normally need to configure the enterprise resources which have been assigned to your device via AirWatch. These are settings commonly defined by your Information Technology department - which in this case is you!

Tap the **Continue** button in the AirWatch MDM Agent configuration wizard on your device.
Install Applications

At the "Install Applications" configuration screen, you (as the end user) would be given the option to install applications assigned to you by your Employer's Information Technology department.

Since our "Android No Camera" profile did not include additional applications, only the basic support applications have been pushed here for the type of device.

**NOTE - Not all Android devices may need specific services or applications.**

Tap the **Continue** button in the AirWatch MDM Agent configuration wizard on your device.
Exit AirWatch MDM Agent

You have completed the initial configuration for your device. You will receive a notification if further action is required.

You can now exit the wizard

You have now completed the AirWatch MDM Agent configuration wizard.

Tap the **Exit** button in the AirWatch MDM Agent configuration wizard on your device.

**NOTE - You will be taken to the AirWatch Agent app and shown connectivity and device info.**
Explore AirWatch SDK on the enrolled device

In this section we are going to install the SDK Sample app on the device and explore the SDK functionalities related to the payloads we configured.

*NOTE - This article uses device screenshots from a Nexus 5 device. However, they might differ slightly based on the device you are using for this lab.*

Launch the SDK Sample App

Tap the **AirWatch SDK Sample App** to launch it.
Allow the prompt

The SDK sample app is targeting multiple features so you might see different prompts requesting access on the test device. Please go ahead and **ALLOW** all the prompts as they show up.
Enter Basic User Credentials (If Necessary)

If prompted, enter the Username and Password for the AirWatch Basic User account:

1. Enter "testuser" for the Username.
2. Enter "VMware1!" for the Password.
3. Tap Login.
Review the Passcode Restrictions

1. Click the Information button by the New Passcode field.
2. Confirm that the Passcode Restrictions popup confirms the passcode restrictions that were configured earlier.
3. Click **Dismiss**.
Notice how the application presents an authentication screen upon launch. As per the authentication configuration defined in the SDK profile, we have to set up a 6 character long passcode meeting the complexity requirements. This passcode will be required after the authentication timeout is expired to access the app data and app functionality so that it is secured during the idle period.

1. **Setup** a passcode which meet the requirements (e.g. **112233**).
2. **Confirm** the passcode in the next text field.
3. Tap **Submit** to set the passcode.

You will receive a toast notification confirming that Passcode has been set successfully.
Launch Client SDK

AirWatch SDK sample app is primarily composed of two sections:

- AirWatch Client SDK - Lightweight and consists of basic features.
- AirWatch Framework - Deeper integration for advanced features.

Now we are going to validate the custom settings that we pushed as a payload from the custom settings. Custom settings is part of AirWatch Client SDK.

Click to select **AIRWATCH CLIENT SDK**.
Validate Custom Settings

1. Select the option **SDK Manager APIs**.
2. Scroll down until you are at the section **Custom Settings**. Validate that we are seeing the hardcoded value for URL as [http://internal.airwlab.com](http://internal.airwlab.com) and lookup field `{EnrollmentUser}` has been replaced by the actual value of enrollment user which is **testuser**.
   
   **NOTE - In the sample app, we are just displaying the values of the custom settings that we received from the console. In a real world scenario, the developer can assign these values to variables and use them however they like.**

3. Click on **Back** button twice until you return to the Home screen.
Launch AirWatch Framework

In this section, we are going to hit an internal splash page from a basic web view within the app. If you navigate to this URL outside the sample app, the splash page will not load. But since we are assigning the AirWatch Tunnel payload to the sample app, it could use it to proxy the traffic to the whitelisted URLs (in this case, the domain is *.airwlab.com).

AirWatch Tunnel is a part of advanced SDK kit, AirWatch Framework.

Click to select AIRWATCH FRAMEWORK.
1. Scroll down to select **Proxy Tunneling AWWebView/Http/URL/OKHTTP**.
2. Click on **Web View**.
3. Enter the URL in the text field as "**http://internal.airwlab.com**".
   **NOTE - instead of entering the URL manually, we could have used the custom setting value that we pushed down from the last step, as an example.**
4. Click **Go**.
5. Notice the splash page, which is not accessible outside this application, if we hit the same URL from a browser or any other app.
Conclusion

In this lab, we went through how to integrate AirWatch Android SDK using Android Studio. Then we configured the SDK profile to carry certain payloads and validate those on a managed device using the sample app. The basic SDK functionality is included in AirWatch Client SDK and advanced features are bundled in AirWatch Framework.
Module 5 - Jenkins
Continuous Integration
Plugin for AirWatch (45 minutes)
Introduction

Jenkins is an open source continuous integration solution that you can integrate with AirWatch to help manage lifecycle of internal applications. You can now have complete control over different phases of your app deployment, right from application build, testing to application release and retirement of old versions. By automating this end-to-end flow, your application testers and end users will have access to the latest app features and fixes keeping all the AirWatch Mobile Application Management (MAM) enhancements intact without any AirWatch admin activities involved.

In this lab, we will:

1. Add VMware AirWatch App Deployment Plugin to Jenkins build server.
2. Configure the plugin to integrate Jenkins build server with AirWatch for lifecycle management of internal apps.
3. Enroll an iOS device to test the plugin functionality.
4. Test the plugin to update the internal app and validate that on the enrolled device.
5. Test the plugin to delete an old build from AirWatch.
6. Learn more about the current version of the plugin.
Connect to Windows 10 VM

We have provided you a Windows 10 VM to complete the necessary steps for this lab. Let's connect to it to complete the steps in the following section.

Connect to the Windows 10 VM

Double-click the **Win10-01a.rdp** shortcut on the lab desktop.

If prompted, the login credentials for the Windows 10 VM are:

- Username: **corp\holuser**
- Password: **VMware1!**
Add AirWatch App deployment plugin to Jenkins

In this section, we are going to add the AirWatch App Deployment plugin to our Jenkins server. This plugin is hosted on AirWatch Resource portal under the section Developer Tools and anyone with a valid myAirWatch ID can download this plugin. In order to limit the scope for this lab, we have already downloaded this plugin for you.

Open Chrome Browser

Double click on Google Chrome icon on the desktop to launch.

Navigate to Jenkins

1. As you launch Chrome, you will see the default home page as AirWatch admin console URL. Click to open a new tab.
2. Select the bookmark Jenkins.
3. Enter user as "jenkinsadmin".
4. Enter password as "VMware1!".
5. Click on log in.

Navigate to Manage Plugins

1. Click Manage Jenkins.
2. Click Manage Plugins.

Plugin Manager Advanced Settings

Click the Advanced tab.

NOTE - If you see any prompts for updates available for plugin, then ignore them for this lab.
Navigate to the Upload Plugin Section

1. Scroll down until you see the section **Upload Plugin**.
2. Click on **Choose File**.

Select Jenkins Plugin File

1. Click on **Documents**.
2. Click on **HOL**.
3. Select the **Jenkins** folder.
4. Select the file **JenkinsAirWatch.hpi**.
5. Click **Open**.
Upload Jenkins Plugin File

1. Ensure that you have selected `JenkinsAirWatch.hpi`
2. Click on **Upload**.

Install Plugin and Restart Jenkins

1. Click to **ENABLE AUTO REFRESH** if not enabled already.
2. Check the box to **Restart Jenkins**.

**NOTE** - The plugin may take a minute or two to install, please wait until the installation process has finished before continuing.

1. Click to **ENABLE AUTO REFRESH** if not enabled already.
2. Check the box to **Restart Jenkins**.
3. After successful installation of the plugin, you should see a **Success** status for **seamless deployment of airwatchapp**.

Please wait while the Restarting Jenkins task begins. This should take less than a minute.

**Wait While Jenkins Restarts**

Once the Restart task has been started, your browser will refresh and you will see a restarting screen while this processes. Please wait while Jenkins restarts, you will be automatically taken back to the Jenkins login page once this completes.

**Login to Jenkins After the Restart**

Login using same credentials as before:

1. Enter user as "**jenkinsadmin**".
2. Enter password as "**VMware1!**".
3. Click on **log in**.
Return to the Manage Plugins Page

Click on **Manage Plugins**.

**Enable the Plugin**

1. Click the Installed tab.
2. Click the checkbox under the **Enabled** column for the **AirWatch Jenkins Plugin for App Deployment**.
Restart Jenkins After Enabling the Plugin

1. Scroll down to the bottom of the Installed section.
2. Click Restart Once No Jobs Are Running.

Wait While Jenkins Prepares to Shutdown

You will see a status bar update that states Jenkins is going to shut down. Please wait while Jenkins finalizes any jobs and shuts down, you will not need to take any further actions at this time.

NOTE - This task should complete within a minute.

Wait While Jenkins Restarts
Please wait while Jenkins restarts, you will be automatically taken back to the Jenkins login page once this completes.

**Login to Jenkins After the Restart**

1. Enter user as "jenkinsadmin".
2. Enter password as "VMware1!".
3. Click on log in.
Login to the Workspace ONE UEM Console

To perform most of the lab you will need to login to the Workspace ONE UEM Management Console.

Launch Chrome Browser

Double-click the Chrome Browser on the lab desktop.
Authenticate to the Workspace ONE UEM Administration Console

The default home page for the browser is https://hol.awmdm.com. Enter your Workspace ONE UEM Admin Account information and click the Login button.

**NOTE - If you see a Captcha, please be aware that it is case sensitive!**

1. Enter your **Username**. This is your email address that you have associated with your VMware Learning Platform (VLP) account.
2. Enter **VMware1!** for the **Password** field.
3. Click the **Login** button.

**NOTE - Due to lab restrictions, you may need to wait here for a minute or so while the Hands On Lab contacts the Workspace ONE UEM Hands On Labs server.**
Accept the End User License Agreement

Terms of Use

You must accept the following VMware End User License Agreement to use Workspace ONE UEM.

VMWARE END USER LICENSE AGREEMENT

PLEASE NOTE THAT THE TERMS OF THIS END USER LICENSE AGREEMENT SHALL GOVERN YOUR USE OF THE SOFTWARE, REGARDLESS OF ANY TERMS THAT MAY APPEAR DURING THE INSTALLATION OF THE SOFTWARE.

IMPORTANT: READ CAREFULLY. BY DOWNLOADING, INSTALLING, OR USING THE SOFTWARE, YOU (THE INDIVIDUAL OR LEGAL ENTITY) AGREE TO BE BOUND BY THE TERMS OF THIS END USER LICENSE AGREEMENT (“EULA”). IF YOU DO NOT AGREE TO THE TERMS OF THIS EULA, YOU MUST NOT DOWNLOAD, INSTALL, OR USE THE SOFTWARE, AND YOU MUST DELETE OR RETURN THE UNUSED SOFTWARE TO THE VENDOR FROM WHICH YOU ACQUIRED IT WITHIN THIRTY (30) DAYS AND REQUEST A REFUND OF THE LICENSE FEE, IF ANY, THAT YOU PAID FOR THE SOFTWARE.

EVALUATION LICENSE: If you are licensing the Software for evaluation purposes, your use of the Software is only permitted in a non-production environment and for the period limited by the License Key. Notwithstanding any other provision in this EULA, an Evaluation License of the Software is provided ‘AS-IS’ without indemnification, support or warranty of any kind, expressed or implied.

1. DEFINITIONS.

1.1 “Affiliate” means, with respect to a party at a given time, an entity that then is directly or indirectly controlled by, is under common control with, or controls that party, and here “control” means an ownership, voting or similar interest representing fifty percent (50%) or

NOTE - The following steps of logging into the Administration Console will only need to be done during the initial login to the console.

You will be presented with the Workspace ONE UEM Terms of Use. Click the Accept button.
Address the Initial Security Settings

Security Settings

Password Recovery Question 1

Password Recovery Question *

Password Recovery Answer *

Confirm Password Recovery Answer *

Password Recovery Question

VMware!

Password Recovery Answer

VMware!

Confirm Password Recovery Answer

A four-digit Security PIN must be entered. It is required in the console for some restricted actions (configured by authorized administrators in System Security settings).

Security PIN *

1234

Confirm Security PIN *

1234

After accepting the Terms of Use, you will be presented with a Security Settings popup. The Password Recovery Question is in case you forget your admin password and the Security PIN is to protect certain administrative functionality in the console.
1. You may need to scroll down to see the Password Recovery Questions and Security PIN sections.
2. Select a question from the Password Recovery Question drop-down (default selected question is ok here).
3. Enter VMware1! in the Password Recovery Answer field.
4. Enter VMware1! in the Confirm Password Recovery Answer field.
5. Enter 1234 in the Security PIN field.
6. Enter 1234 in the Confirm Security PIN field.
7. Click the Save button when finished.

Close the Welcome Message

Workspace ONE UEM Console Highlights

Powered by VMware AirWatch!

Workspace ONE is powered by VMware AirWatch Unified Endpoint Management (UEM) technology, a unified digital workspace platform delivering a single, secure experience for app management, single sign-on (SSO), and conditional access.

Workspace ONE UEM transforms your business so you can:

- Configure, manage and support devices from any endpoint
- Increase productivity with seamless access to any app
- Safeguard company data at every layer
- Access identity and access management tools with ease
- Enjoy a simplified, consistent look and feel across Workspace ONE

Don't show this message on login
After completing the Security Settings, you will be presented with the Workspace ONE UEM Console Highlights pop-up.

1. Click on the **Don't show this message on login** check box.
2. Close the pop-up by clicking on the **X** in the upper-right corner.
Download AppLifecycle Apps

In this section, we are going to download AppLifecycle Apps that we will be using as Internal apps for this lab.

**Download AppLifecycle 101**

1. Open a **new tab** in Chrome Browser.
2. Enter the following URL `https://hol.awmdm.com/MyDevice/s/2239/be759588-38d0-4ad4-949e-88a1f4398f4b` and hit **Enter**
3. Validate that you have downloaded **Applifecycle_101.ipa**
Download AppLifecycle 102

1. Open a **new tab** in Chrome Browser.
2. Enter the following URL [https://hol.awmdm.com/MyDevice/s/2239/86896741-33e4-43fd-a843-6225742f002c](https://hol.awmdm.com/MyDevice/s/2239/86896741-33e4-43fd-a843-6225742f002c) and hit **Enter**
3. Validate that you have downloaded **Applifecycle_102.ipa**
iOS Device Enrollment

In this section, we are going to enroll an iOS device to complete the steps on the device side.

**Download/Install AirWatch MDM Agent Application from App Store - IF NEEDED**

*NOTE* - Checked out devices will likely have the AirWatch MDM Agent already installed. You may skip this step if your device has the AirWatch MDM agent installed.

At this point, if using your own iOS device or if the device you are using does NOT have the AirWatch MDM Agent Application installed, then install the AirWatch Application.
To Install the AirWatch MDM Agent application from the App Store, open the App Store application and download the free AirWatch MDM Agent application.

**Launching the AirWatch MDM Agent**

Launch the AirWatch Agent app on the device.

*NOTE - If you have your own iOS device and would like to test you will need to download the agent first.*
Choose the Enrollment Method

Welcome to AirWatch!

AirWatch helps your IT Department to provide your device with secure access to resources.

The multi-step enrollment process begins with authentication.

Choose authentication method:

Email Address

Server Details

QR Code

Click on the Server Details button.
Find your Group ID from AirWatch Console

1. To find the Group ID, hover your mouse over the Organization Group tab at the top of the screen. Look for the email address you used to log in to the lab portal.
2. Your **Group ID** is displayed at the bottom of the Organization Group pop up.

**NOTE** - The Group ID is required when enrolling your device in the following steps.

Attach the AirWatch MDM Agent to the HOL Sandbox

Once the Agent has launched you can enroll the device. To do so, follow the below steps.

1. Enter `hol.awmdm.com` for the **Server** field.
2. Enter your Group ID for your Organization Group for the Group ID field. Your Group ID was noted previously in the Finding your Group ID step.
3. Tap the Go button.

NOTE - If on an iPhone, you may have to close the keyboard by clicking Done in order to click the Continue button.

Authenticate the AirWatch MDM Agent

On this screen, enter the Username and Password for the basic user account.

1. Enter testuser in the Username field.
2. Enter VMware1! in the Password field.
3. Tap the Go button.
Redirect to Safari and Enable MDM Enrollment in Settings

Enable Device Management

To enable your device, you will be redirected to Safari and Settings

Why?

- Access your company resource
- Remove company data in the event of loss or theft

The AirWatch Agent will now redirect you to Safari and start the process of enabling MDM in the device settings.

Tap on **Redirect & Enable** at the bottom of the screen.
Allow Website to Open Settings (IF NEEDED)

If you prompted to allow the website to open Settings to show you a configuration profile, tap **Allow**.

*NOTE - If you do not see this prompt, ignore this and continue to the next step. This prompt will only occur for iOS Devices on iOS 10.3.3 or later*
Install the MDM Profile

Tap **Install** in the upper right corner of the Install Profile dialog box.
Install and Verify the AirWatch MDM Profile

Tap **Install** when prompted at the Install Profile dialog.

*NOTE - If a PIN is requested, it is the current device PIN. Provided VMware devices should not have a PIN.*
iOS MDM Profile Warning

You should now see the iOS Profile Installation warning explaining what this profile installation will allow on the iOS device.

Tap **Install** in the upper-right corner of the screen.
Trust the Remote Management Profile.

You should now see the iOS request to trust the source of the MDM profile.

Tap **Trust** when prompted at the Remote Management dialog.
iOS Profile Installation Complete

You should now see the iOS Profile successfully installed.

Tap Done in the upper right corner of the prompt.
Your enrollment is now completed. Tap **Open** to navigate to the AirWatch Agent.
Accept the Authentication Complete Prompt

Authentication Complete

- You will receive company resources and settings assigned to your device by your IT department
- You will receive a notification if further action is required

Click on Done to continue.

Accept Notification Prompt (IF NEEDED)

Tap Allow if you get a prompt for Notifications.

Accept the App Installation (IF NEEDED)
You may be prompted to install a series of applications depending on which Module you are taking. If prompted, tap **Install** to accept the application installation.
Configure Plugin to integrate with AirWatch

VMware AirWatch App Deployment plugin for Jenkins is relevant in the Post-Build action of your Jenkins Job. Since it is independent of your source code management and build configurations, it is very easy to incorporate this plugin into your existing as well as new projects. In conjunction with other Jenkins plugins such as Source Code management and build plugins, it can be a fully automated end-to-end app lifecycle management tool.

In this section, we are going to configure the plugin using values from AirWatch console. We are going to create a new freestyle project and add the plugin as a post deployment action.

Create a new job

1. Click on the tab to switch to Jenkins.
2. Click the Jenkins link to return to the Dashboard.
3. Click on the hyperlink create new jobs.

**NOTE - If you are prompted to login due to a timeout, the username is "jenkinsadmin" and the password is "VMware1!"**
Enter name for the project

1. Enter the project name as **AW Jenkins Test**
2. Select type as **Freestyle project**
3. Click **OK**
Configure Post-build Actions

To limit the scope of this lab, we are going to dive straight into configuring Post-build actions to demonstrate VMware AirWatch App Deployment plugin for Jenkins.

Click on Post-build Actions.

Add Plugin as the Post-build Action

1. Click on Add post-build action.
2. Select AirWatch Jenkins Plugin for App Deployment.
Configure AirWatch Server URL

NOTE - You may need to scroll up to view the AirWatch Server URL field.

This is nothing but the API server URL of the AirWatch instance that we are working with. Since for this lab setup, we have API server hosted on the same server as the console server, it has the same URL. Enter AirWatch Server URL as "hol.awmdm.com".

NOTE - Do not enter http or https before the URL. Ensure that you do not have any spaces in the URL.

Enter the File Path and Application Name

1. Complete the File Path as C:\Users\holuser\Downloads\AppLifecycle_101.ipa
2. Enter Application Name as "App Lifecycle"

Find REST API Key and Group ID from the AirWatch Console

For the next items in Jenkins, we will need to retrieve the REST API Key and your Group ID from the AirWatch Console. Return to the AirWatch Console tab in your browser.
Navigate to All Settings

1. Click Groups & Settings.
2. Click All Settings.
Enable AirWatch REST API Key

1. Click **System**.
2. Expand **Advanced**.
3. Expand **API**.
4. Click **REST API**.
5. Select **Override** for **Current Setting**.
6. Click **Enabled** if not selected already.
Copy the AirWatch REST API Key

1. Click & drag to highlight the **API Key** for the AirWatchAPI service, then **right-click**
2. Click **Copy**

Return to the Jenkins Tab

Click the **Jenkins** tab to return to configuring the Post-build Actions.
Paste the API Key in Jenkins

1. **Right-click** in the **API Key** field.
2. Click **Paste as plain text** to insert the API Key you just copied.

*NOTE - You can also use NOTEPAD from the start menu to copy and paste the REST API Key.*

Enter Additional AirWatch Details

1. Enter **Username** as the **email address** you used to log into the AirWatch Console.
2. Enter **Password** as "**VMware1!**".
3. Enter **Organization Group Name** as the **email address** you used to log into the AirWatch Console.
Switch back to the AirWatch Console

Return to the AirWatch Console tab in your browser.

Close the Settings Screen

Click the Close button on the Settings screen.

Finding your Group ID

The first step is to make sure you know what your **Organization Group ID** is.

1. To find the Group ID, hover your mouse over the Organization Group tab at the top of the screen. Look for the email address you used to log in to the lab portal.
2. Your **Group ID** is displayed at the bottom of the Organization Group pop up.

Remember or copy your **Group ID** and return to the Jenkins tab.
Enter Organization Group ID and Remaining Fields

Back in Jenkins, configure the remaining fields:

1. Enter Organization Group ID as the Group ID you copied from the previous step.
2. Enter Smart Group Name as "All Devices"
3. Set Push Mode to Auto
4. Enter Bundle ID as "com.seinternal.applifecycle"
5. Click Save
Run the Plugin

In the last article, we configured the plugin to integrate with AirWatch admin console. We will now run the job to see the plugin in action. We will first upload version 1.0.1 to the console and install it on the enrolled iOS device. After that, we will upload a new version 1.0.2, while deleting the old version (1.0.1). We will validate this update on our enrolled device.

Build the app

1. Click on **Build Now**
2. The job may take a few minutes to complete. After the job is completed you should see a **Blue** status, indicating **success**.

**NOTE** - *The page will auto refresh while you are waiting for the job to complete, so no need to manually refresh. Please wait while the build finishes.*
Validate the app upload on AirWatch console

1. Click on the AirWatch tab to open the console.
2. Click on Apps & Books.
3. Expand Applications.
4. Click on List View.
5. Click on Internal.
6. Validate that you are now seeing the application **App Lifecycle** with version as **1.0.1**
Install the internal app on your enrolled device

1. Depending on if your device is supervised or not, you will get an App Installation prompt for the app App Lifecycle.
2. Click on Install
3. Confirm that the app is installed on the enrolled device.

Enable the App Catalog

To validate that we are receiving the correct AppLifecycle version on our iOS devices, we are going to enable the App Catalog in the AirWatch Console.
Navigate to All Settings

1. Click **Groups & Settings**.
2. Click **All Settings**.
Override the AirWatch Catalog Publishing Settings

1. Click **Apps**.
2. Expand **Workspace ONE**.
3. Expand **AirWatch Catalog**.
4. Click **General**.
5. Click the **Publishing** tab.
6. Select **Override** for the **Current Setting**.
Enable the iOS Platform Catalog

1. Scroll down to the Platforms section.
2. Select Enabled for iOS.
3. Select Enabled for Full Screen.

Save the Catalog Changes

1. Scroll to the bottom of the page.
2. Click Save.
**Close Settings**

1. Ensure the **Saved Successfully** prompt is displayed.
2. Click **Close**.

**Confirm the app version from catalog**

We will now use the App Catalog we published to our iOS Device to confirm the AppLifecycle app version is correct.

**Open the Catalog App**

Tap the **Catalog** app.
Find the App Lifecycle app and confirm that the version is listed as 1.0.1.
Run the plugin to update the internal app

1. Click on the tab to switch to Jenkins dashboard.
2. Click on Configure.
In this step, we will run the same Jenkins job again but targeting the version 1.0.2 of the same app. This will update the existing app in the console from version 1.0.1 and 1.0.2. At the same time, we will delete the previous version (1.0.1) in the same step. So, in one run, we are not only updating the app on the device but also retiring the old versions which are no longer needed.

**NOTE - Retiring / deleting previous versions is an optional step. You can still update the app to a new version, while keeping the old versions in the console.**

1. Click on **Post-build Actions**.
2. Change the File Path - change only the file name in the end from `AppLifecycle_101.ipa` to `AppLifecycle_102.ipa`.
3. Click on **Get Available Version(s)**.
4. Ensure that you see version 1.0.1 in **Delete Previous Versions**. If it does not display, enter "**1.0.1**" for this field.

5. Click **Save**

**Run the job version 1.0.2**

Click on **Build Now**. Confirm that Build #2 under Build History completes.
Validate in AirWatch Console

1. Click on the AirWatch console tab to switch.
2. Click Apps & Books.
3. Expand Applications.
4. Click List View.
5. Click the Internal tab.
6. Validate that you now see the version **1.0.2** for App Lifecycle.

Validate on the device

The app might update too quickly to notice but as soon as the upgrade is completed, you will see a blue dot before the name, indicating that the app is just updated to a new version.
Confirm the app version from catalog

We will now use the App Catalog we published to our iOS Device to confirm the AppLifecycle app version is correct.

Open the Catalog App

Tap the Catalog app.

Confirm the AppLifecycle Version

Find the App Lifecycle app and confirm that the version is listed as 1.0.2.
Un-enrolling Your Device

You are now going to un-enroll the iOS device from Workspace ONE UEM.

*NOTE - The term "Enterprise Wipe" does not mean reset or completely wipe your device. This only removes the MDM Profiles, Policies, and content which the AirWatch MDM Agent controls.*

It will NOT remove the AirWatch Agent application from the device as this was downloaded manually before Workspace ONE UEM had control of the device.

Enterprise Wipe (un-enroll) your iOS device

Enterprise Wipe will remove all the settings and content that were pushed to the device when it was enrolled. It will not affect anything that was on the device prior to enrollment.

To Enterprise Wipe your device you will first bring up the Workspace ONE UEM Console in a web browser. You may need to re-authenticate with your credentials (VLP registered email address and `VMware1` as the password).

1. Click **Devices** on the left column.
2. Click **List View**.
3. Click the **checkbox** next to the device you want to Enterprise Wipe.
NOTE - Your Device Friendly Name will very likely be different than what is shown. It will, however, be in the same location as shown on image in this step.

Find the Enterprise Wipe Option

1. Click More Actions. NOTE - If you do not see this option, ensure you have a device selected by clicking the checkbox next to the device.
2. Click Enterprise Wipe under Management.
Enter your security PIN

After selecting **Enterprise Wipe**, you will be prompted to enter your Security PIN which you set after your logged into the console (1234).

1. Enter 1234 for the **Security PIN**. You will not need to press enter or continue, the console will confirm your PIN showing "Successful" below the Security PIN input field to indicate that an Enterprise Wipe has been requested.

**NOTE** - If 1234 does not work, then you provided a different Security PIN when you first logged into the Workspace ONE UEM Console. Use the value you specified for your Security PIN.

**NOTE - If the Enterprise Wipe does not immediately occur, follow the below steps to force a device sync:**

1. On your device, open the AirWatch **Agent** application.
2. Tap the **Device** section (under **Status**) in the middle of the screen.
3. Tap **Send Data** near the top of the screen. If this does not make the device check in and immediately un-enroll, continue to Step #4.
4. If the above doesn't make it immediately un-enroll, then tap **Connectivity [Status]** under Diagnostics.
5. Tap **Test Connectivity** at the top of the screen.

**NOTE - Depending upon Internet connectivity of the device and responsiveness of the lab infrastructure, this could take a couple of minutes or more if there is excessive traffic occurring within the Hands On Lab environment.**

Feel free to continue to the "**Force the Wipe**" step to manually uninstall the Workspace ONE UEM services from the device if network connectivity is failing.

**Verify the Un-Enrollment**
Press the Home button on the device to go back to the home screen. The applications pushed through Workspace ONE UEM should have been removed from the device.

**NOTE - The applications and settings pushed through Workspace ONE UEM should have been removed. The Agent will still be on the device because that was downloaded manually from the App Store. Due to lab environment settings, it may take some time for the signal to traverse through the various networks out and back to your device. Continue on to the next step to force the wipe if the needed.**
Force the Wipe - IF NECESSARY

If your device did not wipe, follow these instructions to ensure the wipe is forced immediately. Start by opening the iOS **Settings** app.
1. Tap **General** in the left column.
2. Scroll down to view the **Device Management** option.
3. Tap **Device Management** at the bottom of the list of General settings.

**Force the Wipe - IF NECESSARY**

![Image of General and Device Management settings]

Tap the **Workspace Services** profile that was pushed to the device.
Force the Wipe - IF NECESSARY

1. Tap **Remove Management** on the Workspace Services profile. **NOTE** - If prompted for a device PIN, enter it to continue. **VMware provisioned devices should not have a device PIN enabled.**

2. Tap **Remove** on the Remove Management prompt.
After removing the Workspace Services profile, the device will be un-enrolled. Feel free to return to the **Verify the Un-Enrollment** step to confirm the successful un-enrollment of the device.
Additional Reading

Integrate with Jenkins in Master-Slave Configuration

In this configuration, a master node serves as the central controller, and assigns build jobs to the slave node(s). If using a master-slave configuration for Jenkins, the VMware AirWatch plugin requires the Copy-To-Slave plugin to work. The Copy-To-Slave plugin copies the builds from slave(s) to the master, and then you can use the VMware AirWatch plugin to deliver new builds.

Limitations with the current version of the Plugin

- The plugin configuration does not support variables for the Jenkins environment. Use the full, qualified path to the application build file.
- The system does not extract the icon from the IPA when uploading files using the plugin.
- The plugin defaults to All as the supported iOS version instead of specific version that the app supports.
- The slave node does not support the plugin in a Master-Slave configuration of Jenkins.

Plugin for other Continuous Integration solutions

The plugin discussed in this lab was designed for Jenkins. However, to get support for a different continuous integration solution, you can always build a custom plugin. The VMware AirWatch plugin for Jenkins is composed of certain VMware AirWatch REST APIs called in a specific order:

- ORGANIZATION_GROUP_ID_SEARCH
- SMART_GROUP_ID_SEARCH
- UPLOAD_BLOB
- UPLOAD_APPLICATION
- ASSIGN_SMARTGROUP
- APPLICATION_SEARCH
- DELETE_APPLICATION
Conclusion

VMware AirWatch Plugin for Jenkins can be used with your existing Jenkins setup (Standalone or Master-Slave mode) to deploy and manage Internal applications via AirWatch. You can use AirWatch REST APIs to build customized plugin for other Continuous Integration solutions. Along with other source code management and build plugins, you can build a fully automated, end-to-end App Lifecycle Management solution.
Conclusion

Thank you for participating in the VMware Hands-on Labs. Be sure to visit http://hol.vmware.com/ to continue your lab experience online.

Lab SKU: HOL-1857-05-UEM

Version: 20180822-174717