VMware AirWatch API Programming Quick Start Guide

AirWatch v9.1

Have documentation feedback? Submit a Documentation Feedback support ticket using the Support Wizard on support.air-watch.com.

Copyright © 2017 VMware, Inc. All rights reserved. This product is protected by copyright and intellectual property laws in the United States and other countries as well as by international treaties. VMware products are covered by one or more patents listed at http://www.vmware.com/go/patents.

VMware is a registered trademark or trademark of VMware, Inc. in the United States and other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

Copyright © 2017 VMware, Inc. All rights reserved. Proprietary & Confidential.

Revision Table

The following table displays revisions to this guide since the release of AirWatch v9.1.

Date	Reason
April 2017	Initial upload. Document posted for AirWatch v9.1 GA.

Copyright © 2017 VMware, Inc. All rights reserved. Proprietary & Confidential.

Table of Contents

Copyright © 2017 VMware, Inc. All rights reserved. Proprietary & Confidential.

Chapter 1:
Overview

Copyright © 2017 VMware, Inc. All rights reserved. Proprietary & Confidential.

Introduction

This guide walks you through the steps required to access AirWatch APIs. The quick starts in this guide are short, descriptive, and simple intended to help you get started using APIs right from generating API Client certificates to calling an API function in code.

Perform the following steps to quick start the process:

1. Generate the API Client Certificate in AirWatch
2. Export the Client Certificate
3. Add Client Certificate to Certificate Store
4. Add Service Reference to your Project
5. Modify the app.config file in your C# Project
6. Calling an API Function in Code

Each step is explained in detail in the following sections.

Copyright © 2017 VMware, Inc. All rights reserved. Proprietary & Confidential.

Process Flow

Step 1: Generate the API Client Certificate in AirWatch

Navigate to **Groups & Settings > All Settings > System > Advanced > API > SOAP API** in the AirWatch Console and click **Generate Client Certificate**.

Step 2: Export the Client Certificate

After the Certificate has been generated, the AirWatch Console displays the new certificate information:

1. Copy the thumbprint displayed to Notepad or similar text application for later use.
2. Click **Export Client Certificate**.
3. Enter and confirm a password and then click **Export**.
4. Note where the certificate is saved on the local machine.

Step 3: Add Client Certificate to Certificate Store

1. From the Windows desktop on the machine that will be accessing the API, click **Start > Run**, enter MMC, and then click OK.
2. Select **File > Add/Remove Snap-ins**.
3. Select **Certificates** from the available snap-ins at left and then click **Add**.
4. Select **Computer Account** and then click **Next**.
5. Select **Local computer** and then click **Finish**.
6. Click **OK**.
7. Expand the **Certificates** option from the left-hand pane of the display.
8. Right-click **Personal** and then select **All Tasks > Import** to launch the Certificate Import Wizard.
9. Click **Next**.
10. Click **Browse** and then use the file type drop-down menu to select **Personal Information Exchange**.
11. Select the certificate file you exported from the AirWatch previously and then click **Open** to return to the wizard.
12. Click **Next**.
13. Ensure the **Include all extended properties** check box is selected.
14. Enter the password for the certificate you created previously and then click **Next**.
15. Ensure **Personal** is displayed as the Certificate store and select **Place all certificates in the following store**.

Copyright © 2017 VMware, Inc. All rights reserved. Proprietary & Confidential.

16. Click **Next**.
17. Click **Finish**.

Repeat this process on the same machine for the "Trusted People" store. Then, add the certificate to both certificate stores on the AirWatch Device Services server as well.

Step 4: Add Service Reference to your Project

In your project, right-click the project name in the Solution Explorer and select **Add Service Reference**.

In the address section of the Add Service Reference, enter the URL of the AirWatch Server services folder. For example, you would enter "https://www.myairwatchserverurl.com/airwatchservices/DeviceServiceEndpoint.svc" for the Device Services API. Click Go and once the service has been found it will be displayed in the Services pane. Enter a unique name in the **Namespace** text box at the bottom of the dialog box. For example, "DeviceServices" for the Device Service API. Click **OK**. Repeat this process for each of the seven API's that you want to include in the project. The seven Web Service API's are:

1. **Device Services** – /airwatchservices/DeviceServiceEndpoint.svc
2. **Device Data** – /airwatchservices/DeviceDataServiceEndpoint.svc
3. **Enrollment Services** – /airwatchservices/EnrollmentServiceEndpoint.svc
4. **Location Services** – /airwatchservices/LocationServiceEndpoint.svc
5. **Location Group Services** – /airwatchservices/LocationGroupServiceEndpoint.svc
6. **Pick List Services** – /airwatchservices/PickListServiceEndpoint.svc
7. **User Services** – /airwatchservices/UserServiceEndpoint.svc

Step 5: Modify the app.config file in your C# Project

You need to add the behavior section to the app.config file and modify the client section. The behavior section should look like this (replace "your thumbprint goes here" with the actual thumbprint that you saved in Step 2.

```
<behaviors>
    <endpointBehaviors>
        <behavior name="ClientPeerToPeerCertificateSecuredBehavior">
            <clientCredentials>
                <clientCertificate findValue="your thumbprint goes here"
                x509FindType="FindByThumbprint" storeLocation="LocalMachine"
                storeName="My"/>
                <serviceCertificate>
                <authentication certificateValidationMode="PeerTrust"/>
                </serviceCertificate>
            </clientCredentials>
        </behavior>
    </endpointBehaviors>
</behaviors>
```


Copyright © 2017 VMware, Inc. All rights reserved. Proprietary & Confidential.

In the <client> section, you need to modify each endpoint that corresponds to the AirWatch API services by adding a property that refers to the <behavior> that was added above. The property should look like this:

```
behaviorConfiguration="ClientPeerToPeerCertificateSecuredBehavior"
```

Once complete, each <endpoint> section in the <client> area should look like this (replace “your.url.here” with your actual url):

```
<endpoint
address="https://your.url.here/AirWatchServices/DeviceServiceEndpoint.svc"
    behaviorConfiguration="ClientPeerToPeerCertificateSecuredBehavior"
    binding="wsHttpBinding" bindingConfiguration="DeviceServiceEndPoint"
    contract="DeviceService.IDeviceService" name="DeviceServiceEndPoint" />
```

Step 6: Calling an API Function in Code

To call a function in code, you will need to set the appropriate parameters of the function.Below is an example using the “LockDevice” function of the DeviceServices API Service (written in C#).

Create a deviceEntity variable and set the DeviceUid Property, the DeviceType property and the ActivationCode (Location Group ID) property.

```
var deviceEntity = new DeviceService.DeviceEntity { Id = new
DeviceService.Entity { Value = {device ID goes here}, DeviceUid = {device UID
goes here}, DeviceType = new DeviceService.EntityKey { Value = {device type goes
here, 2=Apple}}, ActivationCode = {GroupID goes here} };
```

Create an operationContext instance.

```
DeviceService.AWOperationContext operationContext = new
DeviceService.AWOperationContext();
```

Set the Identity property of operationContext.

```
operationContext.Identity = new DeviceService.ServiceIdentity { Name =
"Administrator" };
```


Copyright © 2017 VMware, Inc. All rights reserved. Proprietary & Confidential.

Create a DeviceServiceClient instance.

```
DeviceService.DeviceServiceClient client = new DeviceService.DeviceServiceClient
();
```

Perform the call to the service with deviceEntity and operationContext as parameters.

```
bool result = client.LockDevice(deviceEntity, operationContext);
```

Results for the LockDevice function returns a Boolean value.

Copyright © 2017 VMware, Inc. All rights reserved. Proprietary & Confidential.

Sample Code

Sample VB Code in the “app.config” file

This code goes after the <bindings> section. The <client> section will be added automatically when the Service Reference is added to the project. The line behaviorConfiguration needs to be added manually to each Service Reference endpoint that is added. The <behaviors> section needs to be added manually.

```
<behaviors>
    <endpointBehaviors>
        <behavior name="ClientPeerToPeerCertificateSecuredBehavior">
            <clientCredentials>
                <clientCertificate findValue="{thumb print goes here}"
                x509FindType="FindByThumbprint" storeLocation="LocalMachine"
                storeName="My"/>
                <serviceCertificate>
                    <authentication certificateValidationMode="PeerTrust"/>
                </serviceCertificate>
            </clientCredentials>
        </behavior>
    </endpointBehaviors>
</behaviors>
<client>
    <endpoint
    address="https://shairwatch.rccl.com/AirWatchServices/DeviceServiceEndpo
    int.svc"
    behaviorConfiguration="ClientPeerToPeerCertificateSecuredBehavior"
    binding="wsHttpBinding" bindingConfiguration="DeviceServiceEndPoint"
    contract="DeviceService.IDeviceService" name="DeviceServiceEndPoint" />
</client>
```

Sample VB Code In A Form

This sample code locks a device when a button is clicked. This code requires that step 4 and step 5 be completed.

```
Public Class Form1
    Private Sub Button1_Click(sender As System.Object, e As
    System.EventArgs) Handles Button1.Click
        Dim results As Boolean
        lblResults.Text = ""

        ' create the instances of the AirWatch objects
        Dim awopcon = New DeviceService.AWOperationContext()
        Dim deventity = New DeviceService.DeviceEntity()
        Dim client = New DeviceService.DeviceServiceClient()
```


Copyright © 2017 VMware, Inc. All rights reserved. Proprietary & Confidential.

```
        awopcon.Identity = New DeviceService.ServiceIdentity()
        ' specify the name in the AWOperationContext Identity
        awopcon.Identity.Name = "Administrator"

        ' create instances of objects inside the deviceentity object
        deventity.Id = New DeviceService.EntityKey()
        deventity.DeviceType = New DeviceService.EntityKey()

        ' Load the parameters with the required values
        ' For lock device need Device UID, Device ID, Location GroupID
        (ActivationCode) and
        ' Device Type
        ' (Apple = 2).
        deventity.DeviceUid = txtDeviceUID.Text
        deventity.ActivationCode = txtGroupID.Text
        deventity.DeviceType.Value = 2
        deventity.Id.Value = txtDeviceID.Text

        ' Call LockDevice with paramaters.
        results = client.LockDevice(deventity, awopcon)

        If results = True Then
            lblResults.Text = "Success"
        Else
            lblResults.Text = "Failed"
        End If
    End Sub
End Class
```


Copyright © 2017 VMware, Inc. All rights reserved. Proprietary & Confidential.