Deploying Automox via VMware Airwatch

The recommended deployment of the Automox agent consists of 3 configurations:

- Create or Modify an Existing Notifications Profile
- Create an Automox Privacy Preferences Profile
- Deploy the Automox Agent as a Native Application

Create or Modify Existing Notification Profile

macOS Catalina supports notification configuration profiles to ensure delivery of Automox notifications.

If no notification payload currently exists, create a new profile with Custom Settings with the following content to allow all Automox and Microsoft Office notification to run:

```plaintext
NotificationSettings

AlertType
1

BadgesEnabled

BundleIdentifier
com.automox.automox-notifier

CriticalAlertEnabled

GroupingType
0

NotificationsEnabled

ShowInLockScreen

ShowInNotificationCenter

SoundsEnabled
```
AlertType
1
BadgesEnabled

BundleIdentifier
com.microsoft.Word

CriticalAlertEnabled

GroupingType
0

NotificationsEnabled

ShowInLockScreen

ShowInNotificationCenter

SoundsEnabled

AlertType
1
BadgesEnabled

BundleIdentifier
com.microsoft.Excel

CriticalAlertEnabled

GroupingType
0
NotificationsEnabled

ShowInLockScreen

ShowInNotificationCenter

SoundsEnabled

AlertType
1

BadgesEnabled

BundleIdentifier
com.microsoft.Powerpoint

CriticalAlertEnabled

GroupingType
0

NotificationsEnabled

ShowInLockScreen

ShowInNotificationCenter

SoundsEnabled

AlertType
1
BadgesEnabled

BundleIdentifier
com.microsoft.Outlook
CriticalAlertEnabled

GroupingType
0
NotificationsEnabled

ShowInLockScreen

ShowInNotificationCenter

SoundsEnabled

AlertType
1
BadgesEnabled

BundleIdentifier
com.microsoft.onenote.mac
CriticalAlertEnabled

GroupingType
0
NotificationsEnabled

ShowInLockScreen
ShowInNotificationCenter

SoundsEnabled

AlertType
1
BadgesEnabled

BundleIdentifier
com.microsoft.OneDrive

CriticalAlertEnabled

GroupingType
0
NotificationsEnabled

ShowInLockScreen

ShowInNotificationCenter

SoundsEnabled

AlertType
1
BadgesEnabled

BundleIdentifier
com.microsoft.OneDrive-mac

CriticalAlertEnabled
CriticalAlertEnabled

GroupingType
0

NotificationsEnabled

ShowInLockScreen

ShowInNotificationCenter

SoundsEnabled

AlertType
2

BadgesEnabled

BundleIdentifier
com.microsoft.autoupdate.fba

CriticalAlertEnabled

GroupingType
0

NotificationsEnabled

ShowInLockScreen

ShowInNotificationCenter

SoundsEnabled
If you have an existing Notifications profile (macOS only supports a single notifications profiles), add the previous content in your existing NotificationSettings array.

Create an Automox Privacy Preferences Profile

Automox leverages Microsoft’s AutoUpdate Tool to handle Microsoft Office updates. To ensure the Automox agent is also granted access to this, you must create a Privacy Preferences Profile.
1. Devices > Profiles and Resources > Profiles
2. Add Profile
3. Apple macOS
4. Device Profile
5. Select Privacy Preferences
6. Add App

**Identifier:** /usr/local/bin/amagent

**Identifier Type:** Path

**Code Requirement:**
identifier "com.automox.agent" and anchor apple generic and certificate 1[field.1.2.840.113635.100.6.2.6] /* exists */ and certificate leaf[field.1.2.840.113635.100.6.1.13] /* exists */ and certificate leaf[subject.OU] = DAEQ58A4ES

**Static Code Validation:** Off

**Apple Events:** Allow

**Receiver Identifier:** com.microsoft.autoupdate2

**Receiver Identifier Type:** Bundle ID

**Receiver Code Requirement:**
identifier "com.microsoft.autoupdate2" and anchor apple generic and certificate 1[field.1.2.840.113635.100.6.2.6] /* exists */ and certificate leaf[field.1.2.840.113635.100.6.1.13] /* exists */ and certificate leaf[subject.OU] = UBF8T346G9

**Deploy the Automox Agent as a Native Application**

1. Install the VMware AirWatch Admin Assistant
   

2. Download the latest Automox Agent

3. Pull the Automox Agent pkg file into the Admin Assistant

4. Open the generated plist file

5. Find the **display_name** key and remove the trailing version characters.
(Airwatch does not support dashes in display names)

Before:

```xml
<key>developer</key>
<string></string>
<key>display_name</key>
<string>AutomoxAgent-1.0-28</string>
<key>installed_size</key>
<integer>15710</integer>
<key>installer_item_hash</key>
<string>51916b8fbf2cdbea236212aa49fda16846</string>
<key>installer_item_location</key>
<string>AutomoxAgent-1.0-28.pkg</string>
```

After:

```xml
<key>developer</key>
<string></string>
<key>display_name</key>
<string>AutomoxAgent</string>
<key>installed_size</key>
<integer>15710</integer>
<key>installer_item_hash</key>
<string>51916b8fbf2cdbea236212aa49fda16846</string>
<key>installer_item_location</key>
<string>AutomoxAgent-1.0-28.pkg</string>
```

7. Add a Native Application to Airwatch
8. Upload the Automox pkg installer file

9. Upload the plist to as the "Metadata File"
10. For automated installations, add the following code to the "Post Install Script"

```bash
#!/bin/bash
/usr/local/bin/amagent --setkey "ENTER_AUTOMOX_ACCESS_KEY_HERE"
/usr/local/bin/amagent --setgrp "Default Group/ENTER_GROUP_NAME_HERE"
launchctl kickstart -k system/com.automox.agent
```
### Install Scripts

Define a pre-install script to run before attempting to install and/or a post-install script to run after a successful installation.

<table>
<thead>
<tr>
<th>Script Type</th>
<th>Script Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-Install</td>
<td></td>
</tr>
<tr>
<td>Post-Install</td>
<td><code>/bin/bash</code></td>
</tr>
</tbody>
</table>

### Uninstall Scripts

Define a Pre-Uninstall script to run before an attempted uninstall and/or define a Post-Uninstall script to run after a successful uninstall. Uninstall Method should be automatically selected by default from the uploaded metadata. To override the default, select from dropdown and customize the behavior of the uninstallation procedure.

<table>
<thead>
<tr>
<th>Script Type</th>
<th>Script Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-Uninstall</td>
<td></td>
</tr>
<tr>
<td>Post-Uninstall</td>
<td></td>
</tr>
</tbody>
</table>