Okta Identity Engine

Device Context

Updated Sep 20, 2023

Document History

Date	Version	Description
Sep 20, 2023	1.3	Fixed formatting and finalized the document for Okta Support.
Sep 12, 2023	1.2	Update branding
March 1, 2023	1.1	Validate information and update format
November 16, 2022	1.0	Initial publication

How to use this guide

This guide details the various steps and prerequisites required to configure Device Trust across various platforms and third-party unified endpoint management (UEM) solutions. These steps are detailed in two separate desktop and mobile device sections.

This guide discusses a few use cases and how they leverage Device Context. Note that passwordless authentication (FastPass) is discussed in detail in the separate Deploying passwordless authentication guide.

Who this guide is for

This guide is written for technical implementers who design, test, and deploy Okta.

The guide is written for an internal audience and should not be shared externally.

Introducing Device Context

Modern digital businesses face an increasing number of challenges. As enterprises are witnessing an influx of new devices and device types chosen by their end-users, they are required to make access decisions based not only on user risk but also device risk (as well as network risk). Furthermore, administrators seek visibility and easy management of device state, ownership, and user-to-device binding across most common platforms (macOS, iOS/iPad OS, Windows, and Android).

When users authenticate using a device, it presents a *context* — the device type, its state (managed or registered), and additional data, such as the presence of a firewall or an active antivirus solution. This context is critical when determining what authentication methods are required to satisfy the appropriate level of authentication assurance that meets a specific app security requirement.

In OIE, “Device Context” encompasses **Okta Devices** and **Device Trust.** Okta Devices is a set of services and capabilities that embeds Okta on every device to give organizations visibility into devices accessing Okta. Device Trust builds upon Okta Devices by adding the ability to integrate with unified endpoint management (UEM) solutions and provide context for enterprise-managed devices. Both Device Trust and Okta Devices enable contextual access decisions.

Device Context is a cornerstone of Okta Identity Engine (OIE) and is an enabler for App-level Policies and Passwordless experiences. We recommend consulting the Deploying App-level policies and the Passwordless Authentication Deployment Guide for further details.

Device Context Benefits

Device Context provides maximum visibility of the information and state that the device presents to assess the authentication methods required to grant access to Okta-managed Apps. Combined with App-level Policies (powered by authentication assurance levels), Device Context is an enabler of passwordless experiences.

Key benefits are:

- Better visibility: device and user binding in Okta Universal Directory
- Better access control: suspend or deactivate devices and sessions
- Better device analysis which strengthens app-level policies, based upon:
    - Registration with Okta (using Okta Verify)
    - UEM status (managed vs. unmanaged)
    - Endpoint Detection and Response (EDR) Signals

Strategic considerations when deploying Device Context

To create the best and most secure access policies for your apps, consider the following:

- Okta Devices registers users' devices in the Okta Universal Directory once enrolled in Okta Verify, creating a solid binding between users and their devices. This allows some fundamental device administration functionality, such as the ability to suspend/deactivate a user's registered device to prevent it from accessing protected applications. However, Okta Devices is NOT an alternative to the UEM or EDR solution. For the highest assurance and security requirements, consider adopting such solutions.
- Okta Device Trust supports the most common platforms: iOS/iPad OS, macOS, Windows, and Android. For the most consistent experience, consider limiting your supported platforms to these platforms.

- Device Trust supports the following:
    - UEM solutions: Workspace ONE, Jamf, Microsoft Intune
    - EDR solutions: Crowdstrike and Microsoft Windows Security Center
- EUM tools will inform the policy if the device is managed or unmanaged. Consider this when crafting your policies.
- EDR tools capture additional information (such as the presence of a firewall or antivirus application on a PC) that can be very useful in determining assurance level. Consult the OIE documentation to learn the signals that are available and include these in your strategy when crafting IAM policies.

Prerequisites

Okta Devices

Okta Devices requires Okta Verify to be added as an authenticator.
To add Okta Verify as an authenticator:

1. In the Admin Console, navigate to **Security > Authenticators**.
2. Under **Setup**, select **Add Authenticator**. The **Add Authenticator** window appears.
3. Select **Add** on the Okta Verify tile.
4. Configure the Okta Verify options appropriately (see Configure Okta Verify Options for details) and select **Add**.

Device Trust

A device is considered as "managed" when it meets the following requirements:

- Devices must have Okta Verify installed
- Mobile device management requires integration with a third-party unified endpoint management (UEM) solution such as Microsoft Intune or Workspace ONE
- Desktop device management requires a Certificate Authority (CA) that can issue client certificates to devices intended to be managed. OIE allows you to provide your own certificate authority or use Okta as a CA.
- iOS devices must have iOS 13 or 14 installed
- The following operating systems are supported:
    - Android 7.0 or later
    - iOS 13, iOS 14
    - macOS 10.15.x (Catalina) and 11 (Big Sur)
    - Windows 10, 32-bit and 64-bit

- macOS systems must have Apple Extensible SSO configured in your OEM solution if you wish to create policies that provide a password experience (FastPass). Please refer to Configure Extensible SSO for Safari and native apps on managed macOS devices for more details.
- iOS devices must have the Credential SSO Extension configured in your OEM solution if you wish to create policies that provide a password experience (FastPass). Please refer to Configure Credential SSO Extension for managed iOS devices for more details.

Use Cases

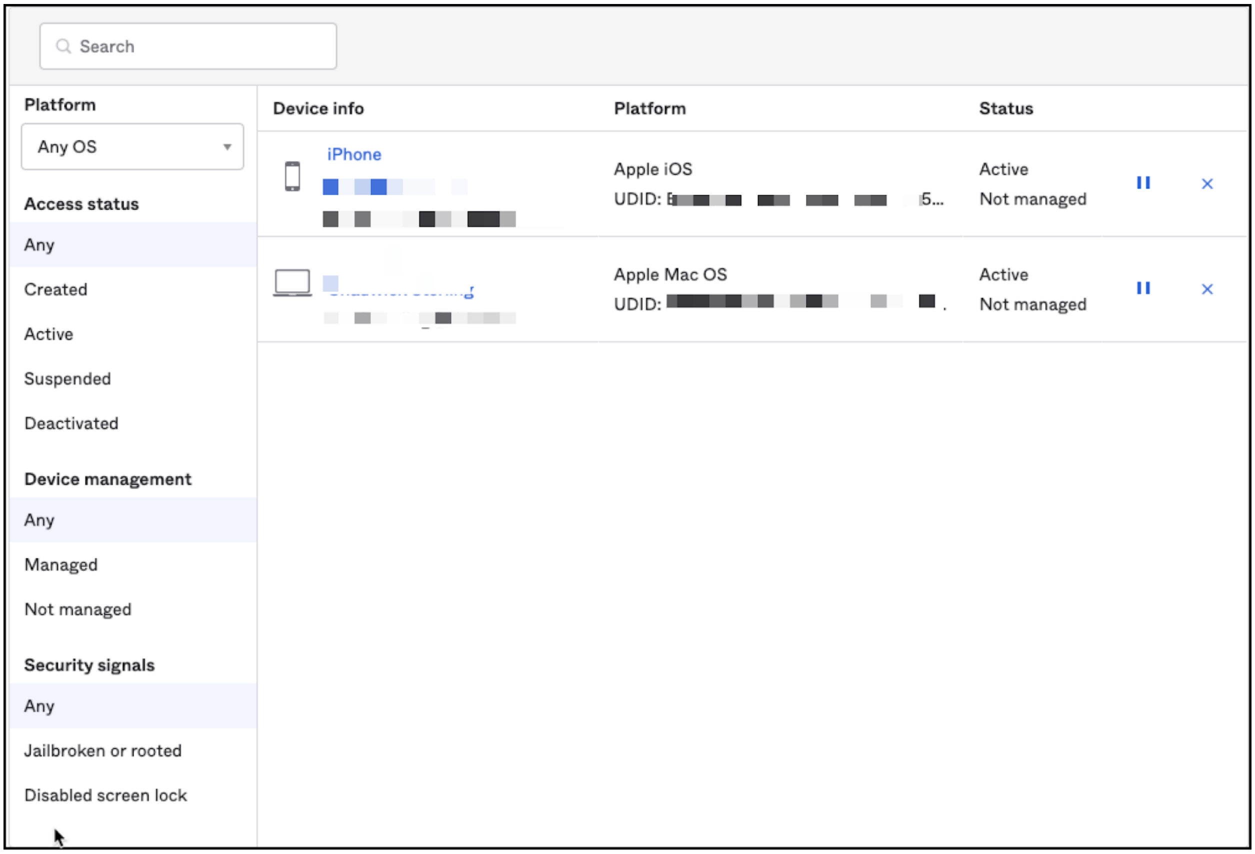
Device Registration (Okta Devices)

When a user installs Okta Verify on a device and enrolls it as an authenticator, the device becomes *registered* as a unique object in the Okta Universal Directory. This registration binds the user to the Okta Verify app instance on the device. It allows admins to see a list of enrolled devices and change their *lifecycle state* in the Okta Admin Console.
All registered devices will be in one of three lifecycle states, as detailed below:

Active	● All Okta Verify factors associated with the device are supported. ● Users can access protected resources from the device if permitted by the app sign-on policies applied to the resources.
Suspended	This is intended to be a temporary state. It is useful if you want to pause (and later resume) device access for users such as contractors or employees who take a leave of absence. Suspended devices can be unsuspended from the **Devices** page in the Okta Admin Console. When a device is suspended: ● All active sessions established on that device using Okta Verify are terminated. ● Active sessions established without Okta Verify are unaffected until the session ends. ● New sessions using Okta Verify can't be established. ● Okta Verify authentication factors can't be used from the device, but users can continue using a password, an email, or a WebAuthN authentication factor. ● Users can't add or remove accounts from Okta Verify on the device. ● Device certificates are unaffected (applies to desktop devices).

	● The device can’t be unsuspended by the user trying to enroll in Okta Verify from the device
Deactivated	To be used if a user reports a device as lost or compromised. A deactivated device can be reactivated from the **Devices** page in the Okta Admin Console. ● All active sessions that were established on that device using Okta Verify are terminated. ● Active sessions established without Okta Verify are unaffected until the session ends. ● New sessions using Okta Verify can’t be established. ● The device can no longer use Okta Verify factors, but users can continue to use password, email, or WebAuthN authentication factors from the device. ● Users can't add or remove accounts from Okta Verify on the device. ● Enrolled factors on the device are deactivated, and users must re-enroll them when the device is reactivated. ● Device certificates are revoked (applies to desktop devices). ● If all rules in the app sign-on policy protecting a resource require devices to be registered, a user on a Deactivated device is denied access to that resource. If the policy includes rules that allow access from unregistered devices, an end-user on a Deactivated device might be able to access the resource but cannot use Okta Verify as an authenticator.

To see all enrolled devices and manage their lifecycle state, launch the Okta Admin Console and navigate to **Directory > Devices**:



Search

Platform

Any OS

Access status

- Any
- Created
- Active
- Suspended
- Deactivated

Device management

- Any
- Managed
- Not managed

Security signals

- Any
- Jailbroken or rooted
- Disabled screen lock

Device info	Platform	Status
iPhone [redacted] [redacted]	Apple iOS UDID: [redacted]	Active Not managed
[redacted] [redacted]	Apple Mac OS UDID: [redacted]	Active Not managed

- The left pane of the Devices page allows the admin to filter the device list based upon the following criteria:
  - **Platform:** the device's operating system (for example, Android, Windows)
  - **Access Status:** Any, Created, Active, Suspended, Deactivated
  - **Device Management:** Any, Managed, Not managed
  - **Security Signals:** Jailbroken or rooted, Disabled screen lock
- The **Device info** column displays:
  - The device's name. Select to provide device details such as the model, OS Version, and enrollment date
  - The device owner's full name (entered in their Okta user profile).
  - The device owner's Okta username
- The **Platform** column displays the device's OS.
- The **Status** column displays whether the device is currently active, suspended, or deactivated, as well as whether it is managed or not managed by UEM.
  - To **suspend** a device, select the ⏸ icon in the Status column. When a device is suspended, this icon is replaced by the ✓ icon. Select this to Unsuspend the device.

    - To **deactivate** a device, select the ✕ icon in the Status column. A deactivated device can be reactivated by selecting the ✓ icon OR permanently removed from the devices list by selecting the 🗑 icon.

In addition to the company-wide list of devices shown above, it is also possible to navigate directly to a particular user's list of registered devices. To do so:

1. In the Okta Admin console, navigate to **Directory > People**
2. Search for or select the name of the user whose devices you wish to see
3. In the user's profile, select **Devices**
4. selecting a device will display further details and will allow you to suspend/deactivate it

Deploying Device Trust on Desktop devices

Okta and your UEM solution must be configured before creating App-level Policies that leverage Device Trust. This guide will refer to computers running Windows or macOS as "desktop devices" and mobile devices running Android or iOS as "mobile devices." Each device type requires a separate configuration process to be performed.

Note: If you wish to use Device Trust to enable a password experience (FastPass) for macOS users, Apple Extensible SSO must be configured in your UEM solution. Please refer to Configure Extensible SSO for Safari and native apps on managed macOS devices for more details.

Choosing a Certificate Authority

Windows and macOS device management requires a **Certificate Authority (CA)** that can issue client certificates to targeted devices. Device Trust uses these client certificates to determine whether devices are managed. This allows application sign-on policies to grant or deny access to an application or prompt for more authentication factors, based on the device's managed status.

OIE allows you to use Okta as a CA, or you can use your own existing CA if you already have one in place that you'd prefer to use.

Providing your own Certificate Authority for Device Context

To provide your own Certificate Authority (CA), your environment requires a PKI infrastructure integrated with your UEM solution to distribute Okta-provided client certificates to targeted devices. In addition to distributing certificates, your UEM takes care of renewing certificates before they expire and revoking certificates from your UEM server and managed devices when devices are no longer managed.

In addition to devices managed by your existing UEM solution, Okta can manage devices that have a certificate deployed by an existing Active Directory Certificate Services (ADCS) infrastructure. To do so, the device must have a certificate deployed from the same CA that is set up in Okta.

To use your own CA, perform the following steps:

Configure the SCEP Payload

Ensure SCEP profiles are targeted at the **USER** level, not the **DEVICE** level. This ensures that the certificate is deployed to the login keychain and accessible to Okta Verify. Your SCEP policy requires a user context. Multiple users using the same device is supported, but each user must be in a separate org. The enrolled user must be managed by your UEM and possess a certificate.

Configure the SCEP payload using the following settings:

Key	Type	Value
KeyUsage	Integer	Set to **signing** so Okta Verify can sign the nonce sent from the Okta server.
AllowAllAppAccess	Boolean	Set to **true** so Okta Verify can sign requests without prompting users to sign in. Otherwise, users are prompted to allow Okta Verify to access the key.
KeysExtractable	Boolean	Set to **false** so that it cannot be copied to another device easily.

Configure management attestation and upload your certificate

1. In the Okta Admin Console, go to **Security > Device Integrations**.
2. Select the **Endpoint Management** tab.
3. Select **Add Platform**.
4. Select **Desktop (Windows and macOS only)**.
5. Select **Next**.
6. Select **Use my own certificate authority** for the Certificate authority.
7. Select **Save**.

8. Select the **Certificate Authority** tab.
9. Select **Add Certificate Authority**.
10. In the **Add Certificate Authority** dialog box, browse to the Intermediate CA used to issue the Client Certificate. If you have multiple such issuers, upload all of them one at a time.
    - **Note:** Okta doesn't support PKCS#7, PKCS#12, or PFX certificate formats.
    - Certificates are uploaded automatically. A message appears if uploads are successful. To view details, select **View root certificate chain details**.
11. Select **Close**.

Using Okta as a CA

UEMs use the Simple Certificate Enrollment Protocol (SCEP) to issue certificates to managed devices. When configuring Okta as a CA, the SCEP "challenge type" can be set to Static, Dynamic, or Delegated. Some UEMs (such as Jamf Pro) support multiple challenge types while others support only one. Before configuring Okta as a CA, determine what challenge type your UEM supports or recommends.

macOS using Jamf Pro - static SCEP challenge

Configure management attestation and generate a SCEP URL and Secret Key

Note: This procedure applies to any UEM solution that supports pushing the Apple SCEP UEM payload.

1. In the Okta Admin Console, navigate to **Security > Device Integrations**.

2. Select the **Endpoint Management** tab.

3. Select **Add Platform**.
   Note: If you add more than one configuration for the same platform type, see this Known Issue.

4. Select **Desktop (Windows and macOS only)**.

5. Select **Next**.

6. On the **Add Device Management Platform** page, enter the following:

    a. **Certificate authority**: Select **Use Okta as certificate authority**.
    b. **SCEP URL challenge type**: Select **Dynamic SCEP URL**.

    c. Select **Generate**.
    d. **SCEP URL:** Copy and save the value. You will need this value later.
    e. **Secret Key:** Copy and save the value. You will need this value later.
    Note: Save the **SCEP URL** and **Secret Key** in a safe place. This is the only time they will appear in the Okta Admin Console.

7. Select **Save**.

Create a static SCEP profile

1. In Jamf Pro, go to **Computers > Configuration Profile**s.
2. Select **+ New**.
3. Navigate to **Options** > **General**.
4. On the General profile page, enter the following:
    a. **Name**: Enter a name for the profile.
    b. **Description**: Optional. Enter a description of the profile.
    c. **Level**: Select User Level.
5. Navigate to **Options > SCEP**.
6. Select **Configure**.
7. On the SCEP profile page, enter the following:
    a. **URL**: Enter the **SCEP URL** you saved in step 6b above.
    b. **Name**: Enter a name for the SCEP profile.
    c. **Subject**: Enter a subject.
    Choose a name that indicates that the certificate is used as the device management signal to Okta. As a best practice, you can also include profile variables provided by Jamf Pro to include the device ID (UDID). For a list of supported variables, see the Jamf Pro document Payload Variables for Computer Configuration Profiles.
    d. **Challenge type**: Static.
    e. **Challenge**: Copy and paste the Secret Key you generated in step 6e above.
    f. **Verify Challenge**: Copy and paste the Secret key again.
    g. **Key Size**: select 2048.
    h. **Use as digital signature**: Check this option.

    i. **Allow export from keychain**: Uncheck this option.
    j. **Allow all apps access**: Check this option
    k. Select **Save**.

macOS using Jamf Pro - dynamic SCEP challenge

Configure management attestation and generate a SCEP URL

Note: This procedure applies to any UEM solution that supports pushing the Apple SCEP UEM payload.

1. In the Okta Admin Console, go to **Security > Device Integrations**.
2. Select the **Endpoint Management** tab.
3. Select **Add Platform**.
   Note: If you add more than one configuration for the same platform type, see this Known Issue.
4. Select **Desktop (Windows and macOS only)**.
5. Select **Next**.
6. On the Add Device Management Platform page, enter the following:
    a. **Certificate authority**: Select **Use Okta as certificate authority**.
    b. **SCEP URL challenge type**: Select **Static SCEP URL** and select **Generic**.
    c. select **Generate**.
    d. **SCEP URL**: Copy and save the value. You will need this value later.
    e. **Challenge URL**: Copy and save the value. You will need this value later.
    f. **Username**: Copy and save the value. You will need this value later.
    g. **Password**: To reveal the password, select **Show password** 👁. Copy and save the value. You will need this value later.
       Note: Save the Password in a safe place. This is the only time it will appear in the Okta Admin Console.
7. Select **Save**.

Create a dynamic SCEP profile in Jamf Pro

1. In Jamf Pro, go to **Computers > Configuration Profiles**.

2. Select **+ New**.

3. Navigate to **Options** > **General**.

4. On the General profile page, enter the following:
    a. **Name**: Enter a name for the profile.

    b. **Description**: Optional. Enter a description of the profile.
    c. **Level**: Select **User Level**.

5. Navigate to **Options > SCEP**.

6. Select **Configure**.
7. On the SCEP profile page, enter the following:
    a. **URL**: Enter the **SCEP URL** you saved in step 6b above.
    b. **Name**: Enter a name for the SCEP profile.
    c. **Subject**: Enter a subject.
    Okta recommends choosing a name that indicates that the certificate is used as the device management signal to Okta. As a best practice, you can also include profile variables provided by Jamf Pro to include the device ID (UDID). For a list of supported variables, see Jamf Pro document Payload Variables for Computer Configuration Profiles.
    d. **Challenge type:** Select **Dynamic-Microsoft CA.**
        - **URL To SCEP Admin**: Enter the **Challenge URL** you saved in step 6b above.
        - **Username**: Enter the **UserName** you saved in step 6f above.
        - **Password**: Enter the **Password** you saved in Step 6g above.
        - **Verify Password**: Re-enter the **Password** you saved
    e. **Key Size**: Select **2048**, and then select **Use as digital signature**.
    f. **Allow export from keychain**: Leave this unselected. It is good security practice to mark the certificate as non-exportable.
    g. **Allow all apps access**: Select this option.
8. Select **Save**.
9. Configure the targets that the profile will be deployed to:
    a. select **Configuration Profiles**.
    b. select the applicable configuration profile name.
    c. select the **Scope** tab.
    d. select **Edit**.
    e. select **+ Add**.
    f. Locate the required deployment targets, and then select **Add**.
10. Select **Save**.

Verify that the Okta CA was installed on your devices

On a macOS device managed by Jamf Pro, make sure the SCEP profile is installed.

1. Go to **System Preference > Profiles**.

2. Verify that your dynamic SCEP profile is installed.
3. Open **Keychain > Login**.
4. Verify that a client certificate and associated private key exists.

Windows using Workspace ONE

Download the x509 certificate in Okta

The x509 certificate you download from Okta is the Organization Intermediate certificate.

1. In the Okta Admin Console, navigate to **Security > Device Integrations**.
2. Select the **Certificate Authority** tab.
3. For the **Okta CA** Certificate Authority, select the **Download x509 certificate** icon in the Actions column.
   You will upload the certificate to Workspace ONE later.

Configure management attestation, generate a SCEP URL and a Secret Key

Configure management attestation and generate a SCEP URL and a Secret Key in Okta by performing the following steps:

1. In the Okta Admin Console, navigate to **Security > Device Integrations**.
2. Select the **Endpoint Management** tab.
3. Select **Add Platform**.
   Note: If you add more than one configuration for the same platform type, see this Known Issue.
4. Select **Desktop (Windows and macOS only)**.
5. Select **Next**.
6. On the **Add Device Management Platform** page, enter the following:
   a. Select **Use Okta as certificate authority** as the Certificate authority.
   b. Select **Static SCEP URL** as the SCEP challenge type.
   c. Select **Generate**.
   d. Copy and save the Okta SCEP URL and the Secret key. You will paste these in Workspace ONE in the Create a static SCEP profile phase.
      Note: Save the **SCEP URL** and **Secret Key** in a safe place. This is the only time they will appear in the Okta Admin Console.
7. Select **Save**.

Create a static SCEP profile in Workspace ONE

1. If not already, log in to Workspace ONE as an administrator.
2. In Workspace ONE, select **DEVICES** (left ribbon bar).
3. Select **Certificates > Certificate Authorities**.

4. Select **+ ADD**.
5. On the Certificate Authority - Add/Edit page, enter the following:
    a. **Name**: Enter a name for the CA.
    b. **Description**: Optional. Enter a description for the CA.
    c. **Authority type**: Select Generic SCEP.
    d. **SCEP Provider**: Basic is entered automatically and can't be changed.
    e. **SCEP URL**: Copy and paste the SCEP URL you generated in step 6b above.
    f. **Challenge Type**: select **STATIC**.
    g. **Static Challenge**: Copy and paste the Secret Key you generated in step 6d above.
    h. **Confirm Challenge Phrase**: Copy and paste the Secret Key you generated in step 6d above.
    i. **Retry Timeout**: Accept the default value of 30.
    j. **Max Retries When Pending**: Accept the default value of **5**, or specify a different number of retries the system allows while the authority is pending.
    k. **Enable Proxy**: Accept the default value of **DISABLED** or select **ENABLED** if appropriate for your environment. If you select Enabled, Workspace ONE UEM acts as a proxy between the device and the SCEP endpoint defined in the CA configuration.
6. Select **TEST CONNECTION**. If you select **SAVE** before **TEST CONNECTION**, the error **Test is unsuccessful** appears.
7. After the **Test is successful** message appears, select **SAVE AND ADD TEMPLATE**. If the test doesn't succeed, ensure you can access the Okta SCEP URL generated in step 6b above from Workspace ONE UEM.

Add/Edit a Certificate Template in Workspace ONE

1. In Workspace ONE, select the **Request Templates** tab.
2. Select **+ ADD**.
3. On the Certificate Template - Add/Edit page, enter the following:
    a. **Name**: Enter a name for the template.
    b. **Description**: Optional. Enter a description for the template.
    c. **Certificate Authority**: Select the CA you created in Step 3.
    d. **Issuing Template**: Leave blank or configure as appropriate for your implementation.
    e. **Subject Name**: Enter **CN = {EmailUserName} managementAttestation {DeviceUid}.**
    f. **Private Key Length**: Select **2048**.
    g. **Private Key Type**: Select **Signing**.

    h. **SAN Type**: N/A.
    i. **Automatic Certificate Renewal**: select **DISABLED**.
    j. **Publish Private Key**: select **DISABLED**.

4. Select **SAVE**.

Define a device profile

Define a device profile to deploy the Okta Intermediate CA to the Intermediate Store on devices in by performing following steps:

1. In Workspace ONE, select **RESOURCES** (left ribbon bar).
2. Select **Profiles & Baselines > Profiles**.
3. Select **ADD**, and then select **Add Profile**.
4. Select **Windows > Windows Desktop > Device Profile**.
5. On the General page, enter the following:
    a. **Name**: Enter a name for the device profile.
    b. **Description**: Optional. Enter a description for the device profile.
    c. **Deployment**: Select **Managed**.
    d. **Assignment Type**: Accept the default or configure as appropriate for your implementation.
    e. **Allow Removal**: Accept the default or configure as appropriate for your implementation.
    f. **Managed By**: Enter the person or group with administrative access to the profile.
    g. **Smart Groups**: Begin typing the group's name and select it from the list.
    h. **Exclusions**: Allows you to exclude groups from the profile. Accept the default or configure as appropriate for your implementation.
    i. **Additional Assignment Criteria**: Allows you to specify a deployment schedule.
    j. **Removal Date**: Allows you to specify a date when the profile is removed from the device.
6. Select **Credentials** in the left pane.
7. Select **CONFIGURE**.
8. In the Credentials page, enter the following:
    a. **Credential Source**: Select **Upload**.
    b. **Certificate**: select **Upload** and browse to the certificate you downloaded in Step 1.
    c. **Key Location**: Accept the default or configure as appropriate for your implementation.

   d. **Certificate Store**: Select **Intermediate**.
9. Select **SAVE AND PUBLISH**.

Define a user profile in Workspace ONE

Define a user profile to deploy the Okta CA-issued client certificate to the Personal Store on devices for management attestation. This step creates the management payload that pushes the client certificate information and credentials to the client, allowing the client to connect to Okta and request a new client certificate. The client certificate is used for management attestation in Okta Verify-enabled flows.

1. In Workspace ONE, select **RESOURCES** (left ribbon bar).
2. Select **Profiles & Baselines > Profiles**.
3. Select **ADD**, and then select **Add Profile**.
4. Select **Windows > Windows Desktop > User Profile**.
5. In the General page, enter the following:
   a. **Name**: Enter a name for the user profile.
   b. **Description**: Optional. Enter a description for the user profile.
   c. **Deployment**: Select **Managed**.
   d. **Assignment Type**: Select **Auto**.
   e. **Allow Removal**: Select **Always**.
   f. **Managed By**: Optional. Enter additional admin names.
   g. **Smart Groups**: Enter the same group(s) you specified in step 5g above.
   h. **Exclusions**: Allows you to exclude groups from the profile. Accept the default or configure as appropriate for your implementation.
   i. **Additional Assignment Criteria**: Allows you to schedule a deployment schedule.
   j. **Removal Date**: Allows you to specify a date when the profile is removed from the device.
6. Select **Credentials** in the left pane.
7. Select **CONFIGURE**.
8. In the Credentials page, enter the following:
   a. **Credential Source**: Select **Defined Certificate Authority**.
   b. **Certificate Authority**: Select the same Certificate Authority that you configured previously.
   c. **Key Location**: Select **TPM If Present** to support devices with or without TPM.
   d. **Certificate Store**: Select **Personal**.
9. Select **SAVE AND PUBLISH**.

On a Windows computer, verify the certificate installation

1. Verify that the client certificate was installed:
    a. On the Windows computer, select **Start**, and then type **cert**.
    b. select **Manage user certificates**.
    c. In **Certificates — Current User**, select **Personal > Certificates**.
    d. Make sure the client certificate exists.
2. Verify the certificate authority (CA):
    a. In **Certificates — Local Computer**, select **Intermediate Certificate Authority > Certificates**.
    b. In the **Issued To** column, find **Organization Intermediate Authority**.
    c. Ensure the **Issued By** column specifies **Organization Root Authority** for **Organization Intermediate Authority**.

Windows using Microsoft Intune

Download the x509 certificate from Okta

1. In the Okta Admin Console, go to **Security > Device Integrations**.
2. Select the **Certificate Authority** tab.
3. In the **Actions** column, select the **Download x509 certificate** icon.
   You will upload the certificate to Microsoft Endpoint Configuration Manager (MECM) later.

Create a Trusted Certificate profile in MECM

1. In Microsoft Endpoint Configuration Manager (MECM), go to **Devices**.
2. Select **Configuration profiles**.
3. Select **+ Create profile**.
4. In Create a profile, do the following:
    a. **Platform**: Select **Windows 10 and later**.
    b. **Profile**: Select **Trusted certificate**.
    c. Select **Create**.
5. In the Trusted Certificate Wizard, do the following:
    a. Enter a name and (optionally) a description.
    b. Select **Next**.
    c. Select the x509 certificate that you downloaded from Okta in step 3 above.
    d. In **Destination store**, select **Computer Certificate store — Intermediate**.
    e. Select **Next**.

f. Assign the trusted certificate profile to one or more user groups.
Note: the user group(s) must be the same as the group(s) you will assign the SCEP profile to in Create a SCEP profile in MECM.

g. Select **Next**.

h. Set Applicability Rules.

i. Select **Next**.

j. Review the configuration, and then select **Create**.

Register the AAD app credentials for Okta in Microsoft Azure

1. In Microsoft Azure, select **App registrations**.
2. Select **+ New** registration.
3. On the **Register an application** page, enter the following:
    a. **Name:** Enter a meaningful name for the application. Make a note of this for later use.
    b. **Supported account types**: Select the appropriate supported account type. Okta tested with **Accounts in this organizational directory only ([Your_Tenant_Name] only — Single tenant)** selected.
    c. **Redirect URI (optional)**: Leave blank or select **Web**, and then enter a redirect URI.
4. Select **Register**.
5. On the app page under **Essentials**, copy and note the **Application (client) ID**.
   You will paste this value in the Okta Admin Console later.
6. Add a client secret:
    a. In the left pane, select **Certificates & secrets**.
    b. Under **Client secrets**, select **+ New client secret**.
    c. In the **Add a client secret** section, enter the following:
        - **Description**: Optional. Enter a description of the client secret.
        - **Expires**: Select an expiration time period.
    d. Select **Add**.
    The secret appears under **Client secrets**.
    e. In the **Client secrets** section, copy and note the **Value**.
7. Set Intune scep_challenge_provider permissions:
    a. In the left pane, select **API permissions**.
    b. Select **+ Add a permission**.
    c. In the **Request API permissions** section, scroll down and select **Intune**.
    d. Under **What type of permissions does your application require?** select **Application permissions**.

    e. In the **Select permissions** search field, enter `scep`, and select the **scep_challenge_provider** check box.
    f. Select **Add permissions**.
    g. In the **Configured permissions** section, select G**rant admin consent for [Your_Tenant_Name] in the Configured permissions section**.
    h. Select **Yes** in the message that appears.

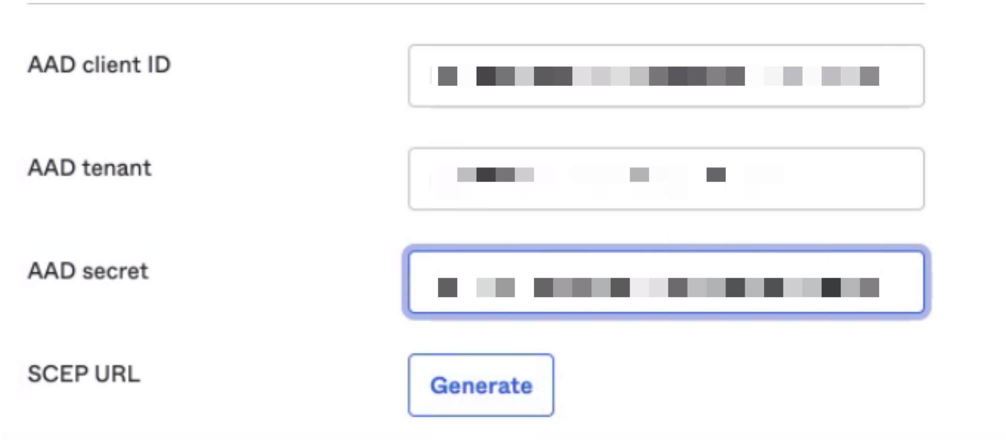
8. Set Microsoft Graph Application.Read.All permissions:
    a. Select **+ Add a permission**.
    b. In the **Request API permissions** section, select **Microsoft Graph**.
    c. Under **What type of permissions does your application require?** select **Application permissions**.
    d. In the **Select permissions** search field, enter `application`, expand **Application**, and select the **Application.Read.All** check box.
    e. Select **Add permissions**.
    f. In the **Configured permissions** section, select **Grant admin consent for [Your_Tenant_Name]**.
    g. Select **Yes** in the message that appears.
9. Set Azure Active Directory Graph Application.Read.All permissions:
   Note: Microsoft Azure portal no longer supports Azure Active Directory Graph. As a workaround, use a PowerShell script to set the Azure Active Directory Graph Application.Read.All permissions.
    a. Create and run a PowerShell script that sets the **Azure Active Directory Graph** Application.Read.All permissions.
       See the following PowerShell resources:
        - https://docs.microsoft.com/en-us/powershell/module/azuread/set-azureadapplication?view=azureadps-2.0 for information about using Set-AzureADApplication with the -RequiredResourceAccess option.
        - https://docs.microsoft.com/en-us/powershell/module/azuread/new-azureadserviceapproleassignment?view=azureadps-2.0 for information about using New-AzureADServiceAppRoleAssignment.
    b. Go to the Microsoft Azure portal and verify the **Azure Active Directory Graph** Application.Read.All permission is assigned to the application in the API permission blade.

Configure management attestation and generate a SCEP URL in Okta

1. In the Admin Console, go to **Security > Device Integrations**.
2. Select the **Endpoint Management** tab.

3. Select **Add Platform**.
   Note: If you add more than one configuration for the same platform type, see this Known Issue.
4. Select **Desktop (Windows and macOS only)**.
5. Select **Next**.
6. Configure the following:
   a. **Certificate authority:** Select **Use Okta as certificate authority**.
   b. **SCEP URL challenge type**: Select **Delegated SCEP URL (Microsoft Intune only)**.
   c. Enter the values that you copied from Microsoft Azure into the following fields:
      - **AAD client ID:** Enter the value you copied from step 5 above.
      - **AAD tenant:** Enter your AAD tenant name followed by .onMicrosoft.com.
      - **AAD secret:** Enter the value you copied from step 6e of above.

   Example:



AAD client ID [redacted]

AAD tenant [redacted]

AAD secret [redacted]

SCEP URL

7. Select **Generate**.
8. Copy and save the Okta SCEP URL. You will paste the URL in Microsoft Endpoint Configuration Manager later.

Create a SCEP profile in MECM

1. In Microsoft Endpoint Configuration Manager (MECM), go to **Devices**.
2. Select **Configuration profiles**.
3. Select **+ Create profile**.
4. In Create a profile, enter the following:
   a. **Platform**: Windows 10 or later
   b. **Profile**: SCEP certificate
   c. select **Create**.

5. In the SCEP Certificate Wizard enter a name and (optionally) a description.
6. Select **Next**.
7. Enter the following:
    a. **Certificate type**: User
    b. **Subject name format** (recommended; other formats will also work): CN={{UserPrincipalName}} ManagementAttestation {{AAD_Device_ID}}
    c. **Key storage provider**: Enroll to Trusted Platform Module (TPM) KSP if present, otherwise Software KSP
    d. **Key usage**: Digital signature.
    e. **Key length**: 2048.
    f. **Hash algorithm**: Select **SHA-2**.
8. Select **+ Root Certificate**.
9. In the **Root Certificate** pane, select the trusted certificate that you created earlier in Step 2, and then select OK.
10. Under **Extended key** usage, set **Predefined values** to **Client Authentication**.
11. Copy the SCEP URL you generated in Step 8 above and paste it into the SCEP Server URLs field.
12. Select **Next**.
13. Assign the SCEP certificate to the same user group(s) to which you assigned the **Trusted certificate profile** in step 5f above.
14. Select **Next**.
15. Set Applicability Rules.
16. Select **Next**.
17. Review the configuration, and then select **Create**.

Verify the certificate installation on a Windows computer

1. Verify the client certificate installation:
    a. On the Windows computer, select **Start** and type `cert` and then select **Manage user certificates**.
    b. Look in **Personal > Certificates**.
2. Verify the Certificate Authority:
    a. On the Windows computer, select **Start** and type `cert` and then select **Manage user certificates**.
    b. Look in Intermediate **Certificate Authority > Certificates**.
    c. In **Issued To**, find and double-select **Organization Intermediate Authority**.
    d. See Issuer: Organization Root Authority.
3. Verify successful SCEP certificate installation and flow:

a. On the Windows computer, select **Start**, type **Event**, and then select **Event Viewer**.
b. Look in **Applications and Service Logs > Microsoft > Windows > DeviceManagement-Enterprise > Admin**.
c. In the **General** tab, find:
   - **SCEP: Certificate installed successfully.**
   - **SCEP: Certificate request generated successfully**

Deploying Device Trust on Mobile devices

In addition to being enrolled in your UEM, the device must also have Okta Verify installed. Integrate with an UEM solution that can silently install Okta Verify to all UEM-enrolled devices for best results.
Okta has tested the following UEM solutions:

- Android: VMware Workspace ONE Unified Endpoint Management, Microsoft Intune
- iOS: VMware Workspace ONE Unified Endpoint Management

Note: To provide a passwordless experience (FastPass) to iOS users, configure the Credential SSO Extension in your UEM solution. Refer to [Configure Credential SSO Extension for managed iOS devices]() for more details.

Configure Okta Endpoint Management for mobile devices

When evaluating an app sign-on policy that requires devices to be managed, Okta determines the management status of your targeted Android and iOS devices by verifying whether there's a key installed on the device that matches a key generated in the Okta Admin Console and entered in your UEM provider's managed app configuration.

To generate this key:

1. In the Admin Console, go to **Security > Device Integrations**.

2. Select the **Endpoint management** tab.

3. Select **Add Platform**.
   Note: If you add multiple configuration for the same platform type, see Device Trust on Identity Engine known issues.

4. Select **Android** or **iOS** as applicable and select **Next**.

5. In **Configure management attestation**:

    a. Copy the provided **Secret key** to your clipboard by selecting the copy icon 📋 adjacent to the field. You'll enter the Secret key later in your UEM provider's app configuration.

        - Note the provided Secret key value as this is the only time it will appear in Okta. If you generate a new Secret key by selecting **Reset secret key**, make sure to update your UEM configuration with the new key.
        - The **Device management provider** field is pre-populated with the name of your UEM but you can change it. The contents of this field will be displayed to end-users when they enroll their device.

    b. In the **Enrollment link** field, enter a web address to redirect end-users with unenrolled devices. For example, you may want to redirect these users to a page with enrollment instructions or the enrollment page of your selected UEM (assuming the UEM provider supports web-based enrollment).

    c. Select **Save**.

Integrate Okta with your third-party UEM provider

Regardless of which UEM provider you choose to integrate with Okta, you must complete the following two steps:

1. Configure your UEM provider to manage Okta Verify and install Okta Verify on end-user devices that do not have it installed.
   Note: If you are configuring your UEM to deploy Okta Verify to Android devices, make sure that Okta Verify is installed in the **work profile** of the device.
2. Configure the key-value pair by using your UEM provider's managed app configuration as described in their documentation:
    a. **Domain:** Enter the URL of your Okta org
    b. **Key:** enter `managementHint`
    c. **Value:** Enter the Secret Key value you saved during the Configure Device Management for mobile devices procedure.
    Note: The key-value pair is case-sensitive.

We suggest using the settings and steps detailed below if you are using Workspace ONE or Microsoft Intune. UEM configurations can change without notice, so Okta recommends that you always consult your UEM's documentation for the most up-to-date information.

Workspace ONE for Android

To add, assign, and manage Okta Verify with Workspace ONE UEM, perform the procedures as described in the following Workspace ONE's Add Assignments and Exclusions to your Android Applications
Configure the following settings:

- **App Delivery Method:** Automatic
- **Managed Access:** Enable

Workspace ONE for iOS

- In **Add Application**:
    - **Platform:** Apple iOS
    - **Source**: Search App Store
    - **Name**: Enter the name of the app. A search finds the app after you select Next.
    - **Details**: Keep the defaults, and then select **Save & Assign**
- In **Assignment**:
    - Distribution:
        - **Name**: Enter a name.
        - **Assignment Groups**: Specify a group(s).
        - **App Delivery Method**: Auto
    - **Restrictions**:
        - **Make App UEM Managed if User Installed**: Enable
    - Application Configuration:
        - **Managed Access**: Enable
        - **Send Configuration**: Enable
        - select **+Add** and configure settings:
            - **Configuration Key**: *managementHint*
            - **Value Type**: String
            - **Configuration Value**: Enter the Secret Key that you generated in step 5a above

Microsoft Intune for Android

To manage Okta Verify with Microsoft Intune for Android devices, perform the procedures described in the Microsoft Intune document Add app configuration policies for managed Android Enterprise devices.

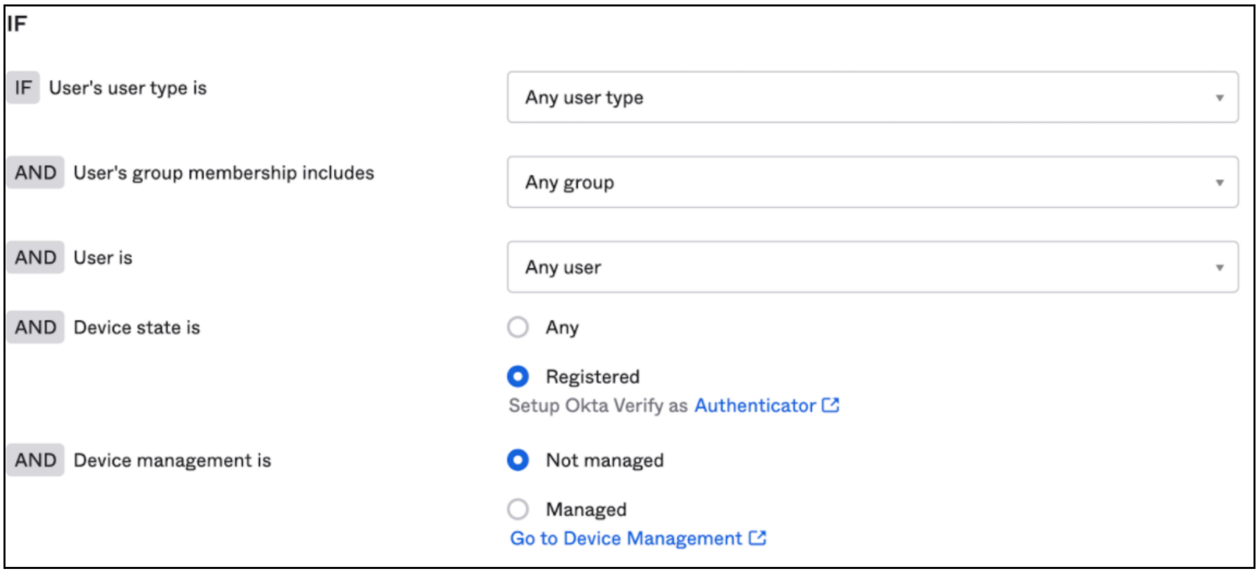
- **Device enrollment type**: Managed devices
- **Associated App**: Okta Verify
- **Configuration settings format:** Use configuration designer
- **Username (string)**: Enter your username for your Okta org

Using Device Context in app-level policies

Device state and device management

In OIE, application sign-on policies and rules can be configured to apply to devices based on the following device state and device management selections:

- **Any:** the rule will be applied to all devices
- **Registered/Not managed:** the rule will only be applied to devices enrolled in Okta Verify, but UEM management is not required.
- **Registered/Managed:** the rule will only be applied to devices enrolled in Okta Verify AND managed by a third-party UEM solution.



IF

IF User's user type is: Any user type

AND User's group membership includes: Any group

AND User is: Any user

AND Device state is:
○ Any
● Registered
Setup Okta Verify as Authenticator

AND Device management is:
● Not managed
○ Managed
Go to Device Management

When the conditions selected above are met or not met, the rule can be configured to deny/allow access or prompt for additional authentication accordingly. For example:

- Low sensitivity applications might consist of only one rule that allows access from all devices regardless of registrations and management
- Medium sensitivity applications might consist of:

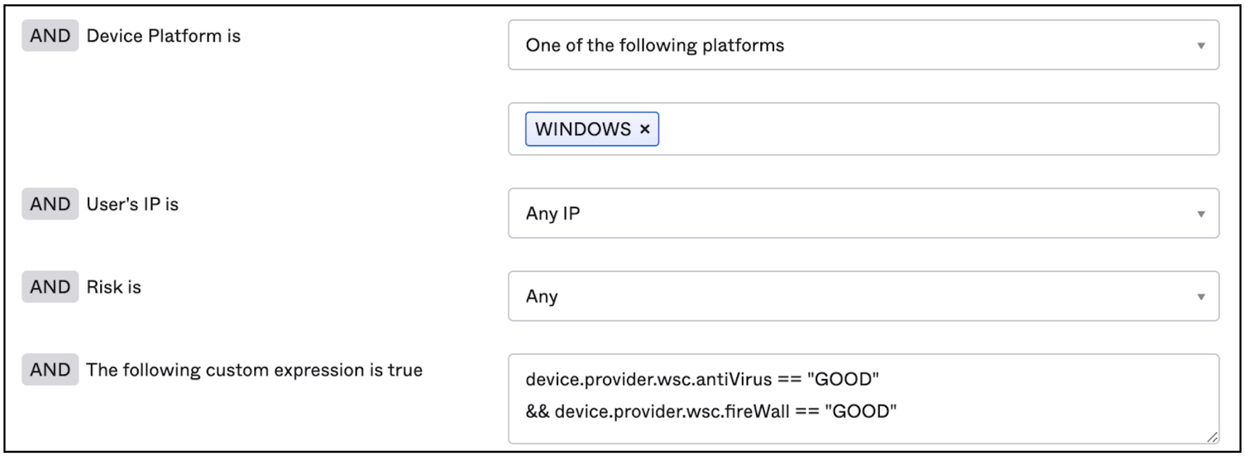
    - A rule that requires unregistered devices to use two factors to authenticate
    - A rule that allows registered and managed devices to authenticate without a password
- High sensitivity applications can be configured to:
    - Require that iOS and Android devices are managed by third-party UEM solutions
    - Require biometric authentication every time the application is launched, regardless of the device state

To see a detailed list of the available sign-on policy IF/THEN conditions and how they behave, please consult Add an app sign-on policy rule.

Endpoint Protection and Response (EDR) Signals

In addition to evaluating the device's state and management, sign on policies can be configured to evaluate signals from a third-party EDR solution such as Crowdstrike or Windows Security Center. These signals can then be used by the policy to determine an access or authentication decision when a user tries to access a protected resource.

For example, you can create a policy that only allows access to a sensitive application from Windows devices that have active firewall and endpoint protection software, as shown below:



AND Device Platform is: One of the following platforms

WINDOWS ×

AND User's IP is: Any IP

AND Risk is: Any

AND The following custom expression is true:

```
device.provider.wsc.antiVirus == "GOOD"
&& device.provider.wsc.fireWall == "GOOD"
```

To learn more about leveraging an EDR integration in your app sign-on policies, please consult our EDR Integrations documentation.