Cogito Group

DIGITAL IDENTITY AND SECURITY

SCEP Client Configuration

Cogito PKI Services

21 August 2025

Version 4.2

Owner:	Cogito Group
Contact details:	Email: Security.Services@cogitogroup.net
Program name:	PKIaaS Services
Division/Unit:	Cogito Group
Document status:	Released and uploaded to Cogito Group web site

1 Overview

This document describes the procedure for adding the Cogito Jellyfish Intune SCEP Connector as a third party SCEP enrollment server.

The procedure can be summarized as a set of goals to accomplish within the Microsoft Azure Portal. These are:

- Create a Jellyfish SCEP *Application* within the Microsoft Azure *Azure Active Directory* service. Provision *API Permissions* and *Secrets* for the Jellyfish Intune SCEP Connector.
- Create a *Trusted certificate Configuration profile* within the Microsoft Azure *Microsoft Endpoint Manager admin center* deploying trust of the Jellyfish Certificate Authority trust chain.
- Create a *SCEP certificate Configuration Profile* within the Microsoft Azure *Microsoft Endpoint Manager admin center* deploying a 'policy' by which Intune may generate and validate Certificate Signing Requests.

Disclaimer

Instructions in this document are correct as of 22 August 2025.

Some screenshots and interfaces may be updated by external platforms. Microsoft Azure Portal, Microsoft Entra admin centre, and Microsoft Intune admin centre configurations are subject to change and have a history of modifying workflows. Therefore, the document's content may not directly correspond to Microsoft's web tooling appearance or functionality.

2 Cogito Product Requirements

Using the Cogito Jellyfish Intune SCEP Connector requires a subscription to one of Cogito's Jellyfish PKIaaS services (a government service or https://securesme.com/) or a Jellyfish software license.

To arrange a subscription or license, please contact: sales@cogitogroup.net

3 Self Service SCEP CA and Template Configuration

This section describes the SCEP CA and Template configuration that is required before using Intune SCEP or Simple SCEP.

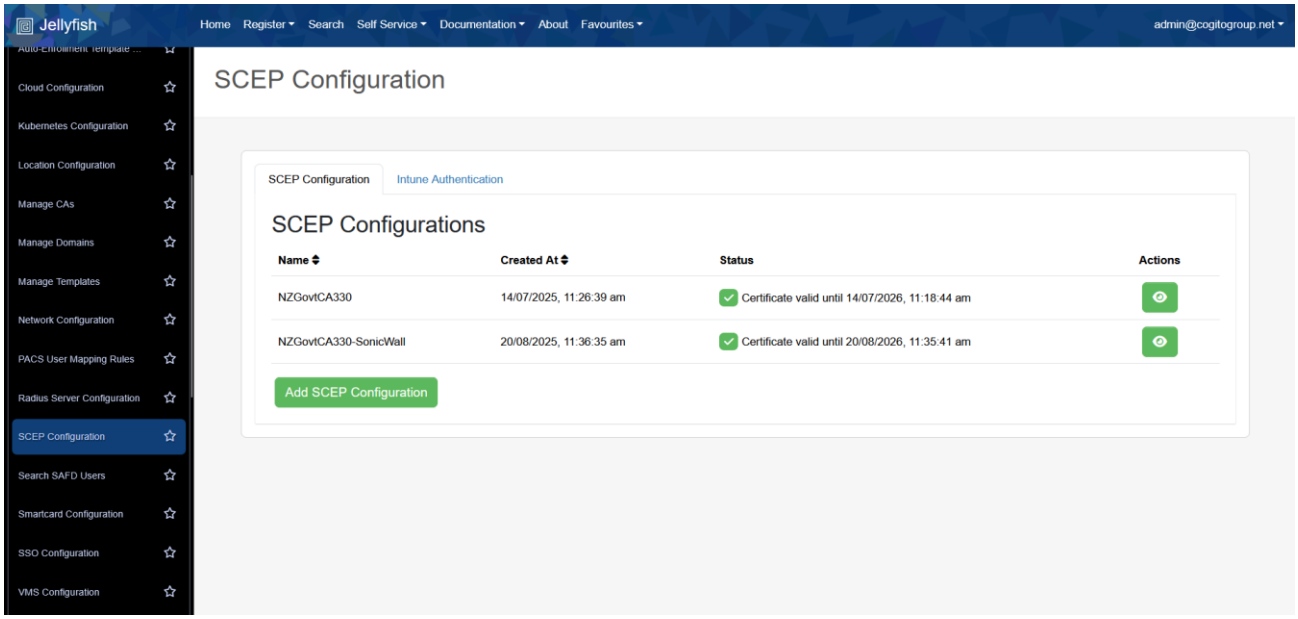
3.1 Jellyfish Product Prerequisites

The following Jellyfish Products are required before Jellyfish SCEP can be provisioned:

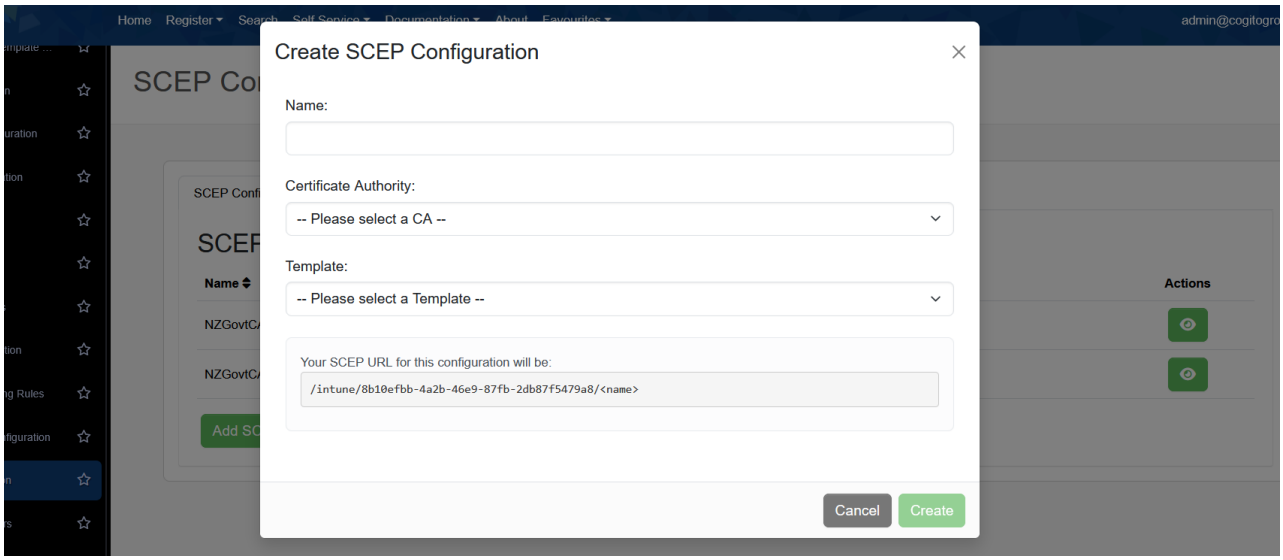
- Optional: Issuing Certificate Authority - base service (a shared Certificate Authority may be used in the case where one is already available to the service).
- Required: Certificate Template under management - uplift. For the purposes of issuing SCEP certificates.
- Required: Autoenrollment services server - uplift.

3.2 Creating a Jellyfish SCEP Configuration

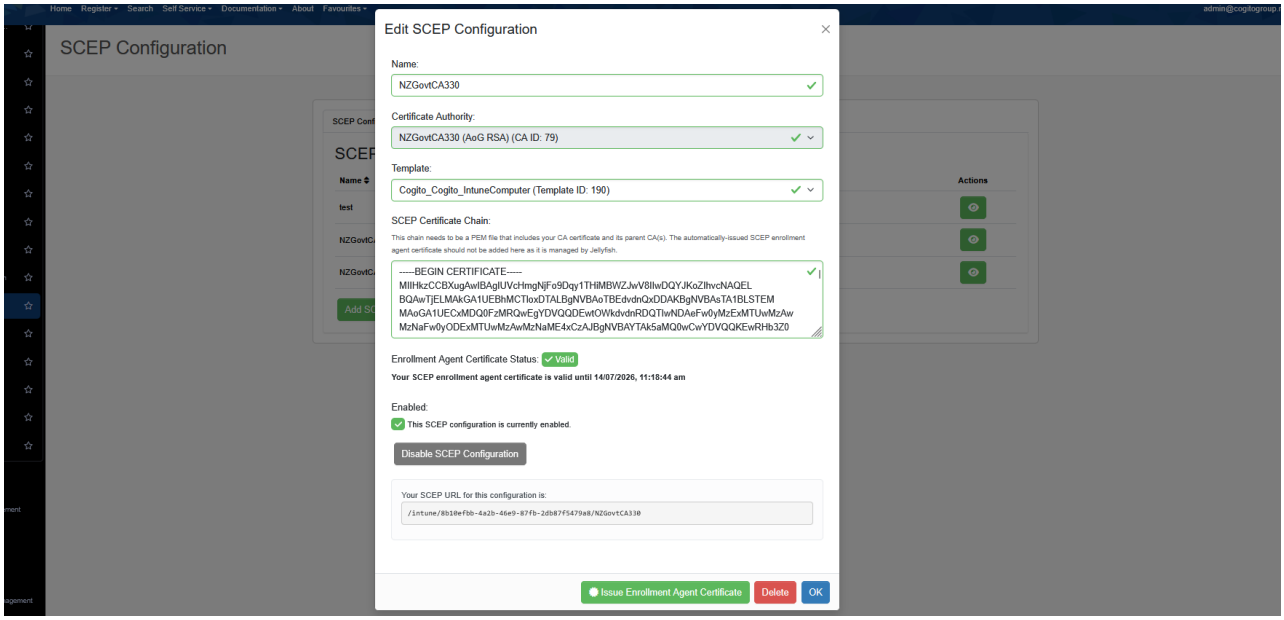
1. In the Jellyfish Portal, navigate to **Configuration > SCEP Configuration**.



2. Click the green “Add SCEP Configuration” button
3. Enter a configuration name and select a CA and template.



4. Click “Create”
5. View your SCEP configuration and add your CA’s certificate chain. This should include your CA certificate and its parent CA(s) if applicable, but not include any SCEP enrollment agent certificate.
6. Click the “Issue Enrollment Agent Certificate” button.
7. Click “Enable SCEP configuration” and save changes.



You can now proceed with either the Intune SCEP or Simple SCEP instructions, depending on how you intend to use SCEP.

4 Intune SCEP

4.1 Software Security

4.1.1 Preface

Cogito Jellyfish Intune SCEP Connector (usSCEP) provides the highest level of security possible within the Microsoft Intune implementation of the Simple Certificate Enrollment Protocol (SCEP).

Although usSCEP provides an implementation of the Simple SCEP Pre-shared Secret method of Certificate Signing Request (CSR) authentication method, the Microsoft Intune SCEP enrollment process does not use this method of authentication or validation.

usSCEP has a more secure method of authentication and validation than that provided by Microsoft's NDES service. usSCEP does not require the installation of any 'on-site' software such as Microsoft Intune Connector for the NDES service. usSCEP does not require Kerberos authentication and as such does not depend on an 'on-site' deployment of Active Directory.

4.1.2 Intune SCEP Request Flow

Cogito Jellyfish Intune SCEP Connector (usSCEP) conforms to the requirements of the Microsoft Intune SCEP request flow.

The authentication and validation procedure of this flow is implemented through an API request from usSCEP to the Microsoft Graph API. This request is a sequence of two subsequent requests, the first request reads the Service Principal Endpoints for the Intune 'Azure Application', this provides an address for which to submit a SCEP challenge request.

In the Microsoft Intune SCEP request flow, the validation procedure occurs after the CSR is generated, but before the CSR is signed.

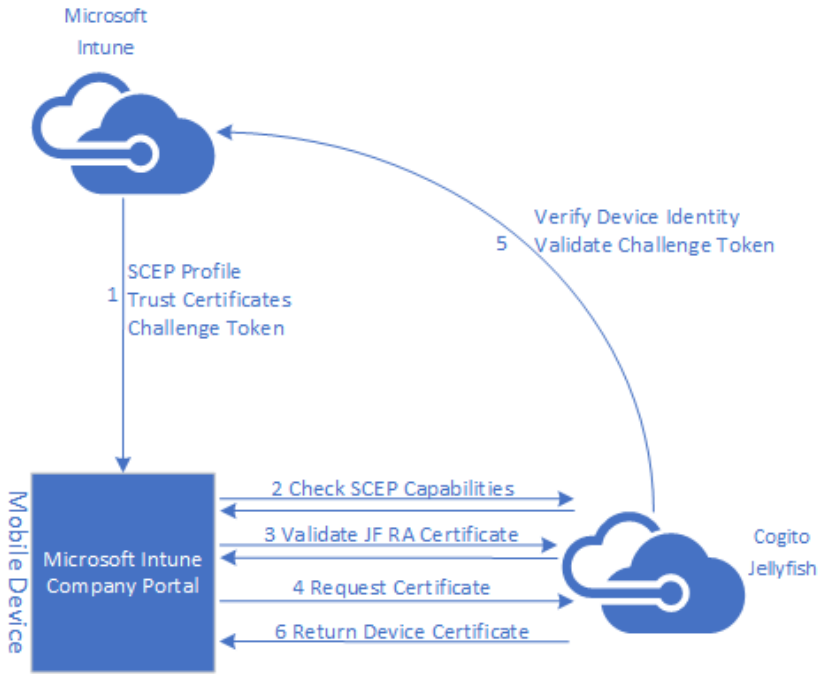


Figure 1: In step 5 of the Microsoft Intune SCEP request flow, the CSR requested by the device is sent back to Azure Graph API to ensure it conforms to the profile by which it has been generated.

Azure Intune uses the whole Certificate Signing Request (CSR) as the 'challenge'. This is a distinction from the Simple SCEP implementation in which a 'pre-shared secret' is used as validation.

The Azure Intune method of sending the whole CSR is a significant improvement to a 'pre-shared secret'. However, additional information is required to perform validation using a CSR. In the case of Azure Intune this additional information is already available and used to generate the CSR in the first place. The CSR generation and validation information is saved within the *SCEP certificate Device Configuration Profile* defined in the *Microsoft Endpoint Manager admin center*.

The *SCEP certificate Device Configuration Profile* (SCEP profile) takes the form of an exact match whitelist. The SCEP profile defines exactly which Subjects, Subject Alternatives Names, Key Usages, Extended Key Usages, Key Sizes, Hash Algorithms, and more **MUST** be included in a CSR intended for enrollment using this SCEP Profile. The SCEP Profile is permissioned against a set of Included and Excluded *Azure Active Directory Users or Groups*.

For the purposes of usSCEP validation: every CSR received by the services is sent to the Azure Graph API for validation. Azure Intune compares the contents of the CSR against the SCEP profile. Azure Intune verifies the Device ID contained within the CSR belongs to, or is assigned to a user that belongs to, one of the included Azure Users or Groups, and is **NOT** included in an excluded group. Azure Intune then verifies that all details within the CSR exactly match those of the SCEP profile, including any variable object identifiers exactly match those of the user or device the CSR has been determined to belong to.

Cogito and the Jellyfish Software including usSCEP are not involved in the validation procedure beyond requesting validation of a CSR.

4.1.3 Microsoft Graph API

Cogito Jellyfish Intune SCEP Connector (usSCEP) uses the OAuth authentication procedure for establishing a connection to Microsoft Graph API. The establishment of the OAuth connection requires the usSCEP to store two Object Identifiers associated with a customer's Azure Active Directory environment, and one 'pre-shared secret' in the form of an *App Registration Secret*.

The three stored values can be described in more detail as:

- *Directory (tenant) ID*: Used by usSCEP as part of the OAuth authentication procedure prior to transmitting validation requests.
- *Application (client) ID:* Used by usSCEP in the first stage of the validation process to identify which *Service Principal* to enquire for a service provider for the second stage of the validation process.
- *App Registration Secret:* Used by usSCEP as part of the OAuth authentication procedure prior to transmitting validation requests. Directly associated with the permissions we are granted for the purpose of request validation. Revocation of this token effectively disables all SCEP enrollments, and our access to your Azure tenancy and services.

An OAuth session is created each time a usSCEP certificate enrollment is triggered. The OAuth session is reused for both the first and second stage of the validation process. Subsequent enrolments will use distinct sessions.

SCEP enrolments for other Azure tenancies may occur on the same service. Each validation procedure is handled in a silo, and there is no 'cross-pollination' of customer Object Identifiers or secrets.

4.2 Prerequisites

Before configuration of Azure Intune may be performed the following requirements must be provisioned and available.

4.2.1 Technical Prerequisites

The following accounts, credentials and keys must be available to complete this configuration:

- Administrative access to the following Microsoft Azure platform services
    - Microsoft Azure Portal
    - Microsoft Azure Active Directory
    - Microsoft Endpoint manager admin center (in some instances this may simply be referred to as *Intune* with the *Microsoft Azure Portal*)
- Provision of the Cogito Jellyfish Intune SCEP Connector service by Cogito Group Operators. This must be confirmed through the following email inbox:
    - Security.Services@cogitogroup.net
- Your unique customer SCEP Server URL. This is sent to you by Cogito Group Operators as part of the Provisioning of the Cogito Jellyfish Intune SCEP Connector.
    - Example Server URL: https://jellyfish.securesme.com/intune/a5e8ee4c-eed9-4a7b-be8e-a97a875cdcf5/OperationsCA302/scep
- The Certificate Authority certificates required for establishing trust for your device configuration. This is at least the Root Certificate Authority, and at most the Root, Intermediate, and Issuing Certificate Authority certificates.
    - For NZTaaS customers, certificates can be downloaded from: http://pki.govt.nz
    - Cogito Group Operators are available for assistance with which Certificates you require and may be contacted through the following email inbox:
        - Security.Services@cogitogroup.net

4.3 Configuring Azure Intune

This document describes the most common configuration for the Cogito Jellyfish SCEP Intune Connector. If you organisation requires deviations from the below configuration, and the setup does not result in successfully issued SCEP certificates, please contact:

Security.Services@cogitogroup.net

For further assistance. Standard Professional Services charges may be incurred in some cases.

4.3.1 Microsoft Azure Portal

This document describes procedures for configuring within the *Azure Active Directory* service as well as from the *Microsoft Endpoint Manager admin centre* server. Both services are accessible through the *Microsoft Azure Portal* website. Access to the *Microsoft Azure Portal* website is a prerequisite requirement to the Configuring Azure Intune section of this document.

It is recommended to search for the following services through the *Search resources, services, and docs* toolbar at the top of the *Microsoft Azure Portal*.

4.3.1.1 Azure Active Directory

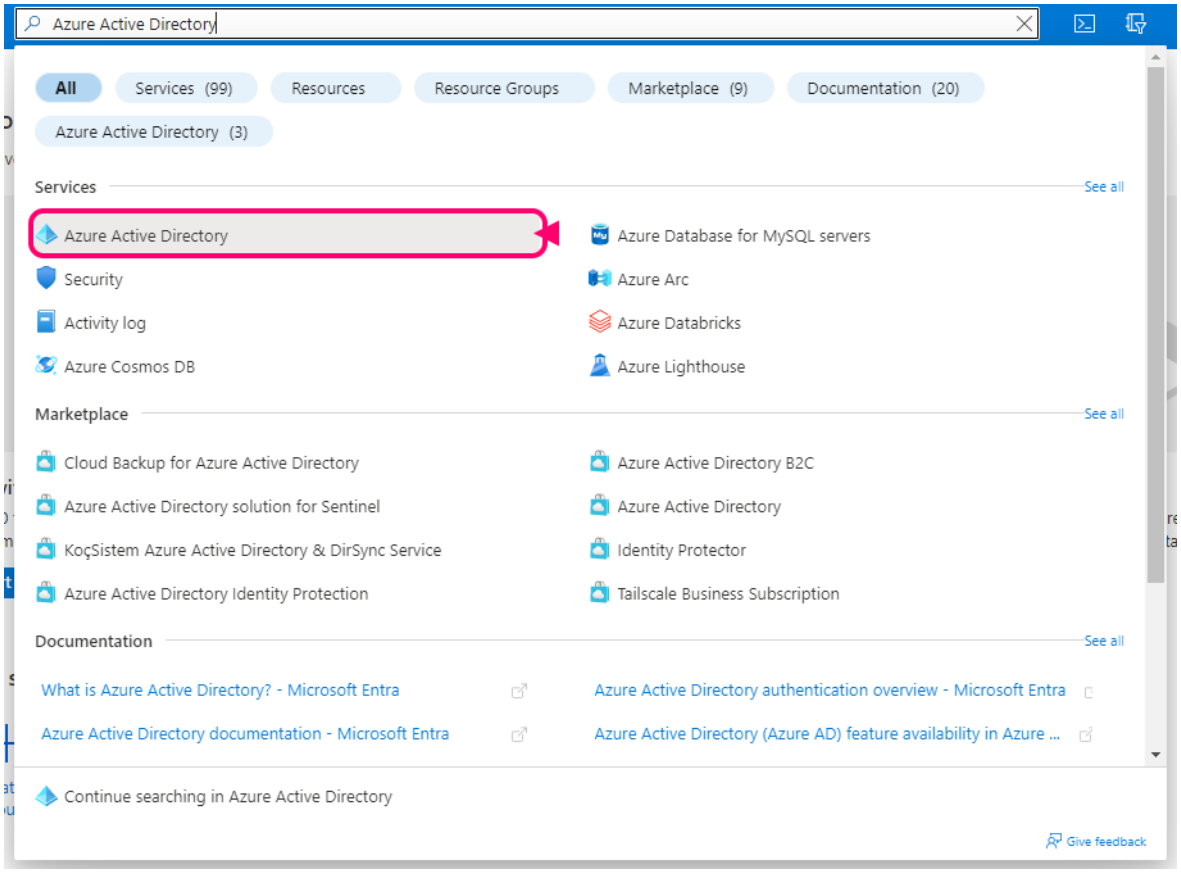


Figure 2: When searching for "Azure Active Directory" the service appears as the first result.

4.3.1.2 Microsoft Endpoint Manager admin center (Intune)

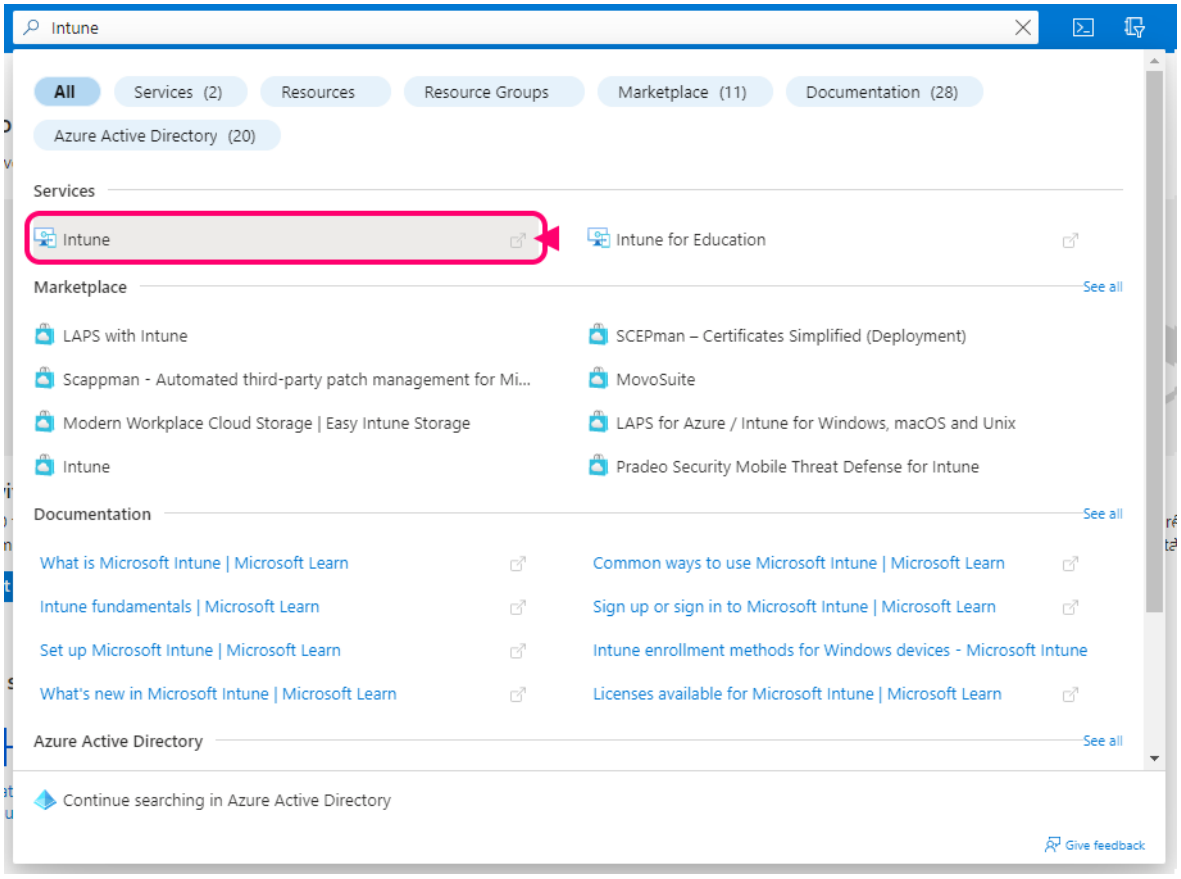


Figure 3: When searching for "Intune" the service appears as the first result. Note that despite the service name, the Intune service directs to the Microsoft Endpoint Manager admin center.

4.3.2 App Registrations

Application registration occurs in the *Azure Active Directory* service.

1. In the left menu bar, under the *Manage* heading: select *App Registrations*.
2. In the top of the *App Registrations* blade click the *+ New registrations* button.

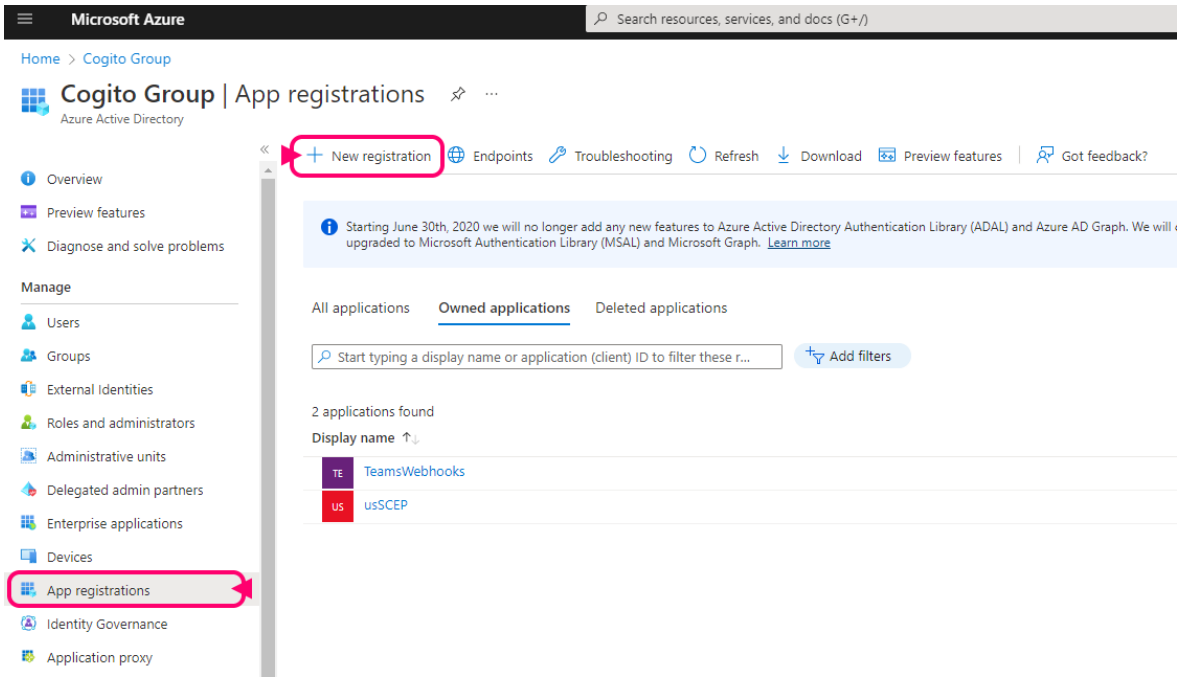


Figure 4: Within Microsoft Azure Active Directory Service, App registrations appears in the left menu. Within the App registrations blade, + New registrations appears at the top.

3. Fill out the details on the first page of the *Register an application* page:
    a. Enter the Name of the application. We recommend this includes the words: *Cogito Jellyfish SCEP Intune Connector.* We additionally recommend you include any identifying information your organization will require to identify this application in the future.
    b. Select *Accounts in this organizational directory only (Single tenant)*.
    c. Do not specify a *Redirect URI*
    d. Click *Register.* You will be redirected to the application summary for the application just created.
4. Record the Two Object Identifiers required to later complete your Intune configuration in Jellyfish:
    a. *Application (client) ID:* A Globally Unique Object Identifier unique to this application.
    b. *Directory (tenant) ID*: A Globally Unique Object Identifier used to identify your tenancy within Azure.

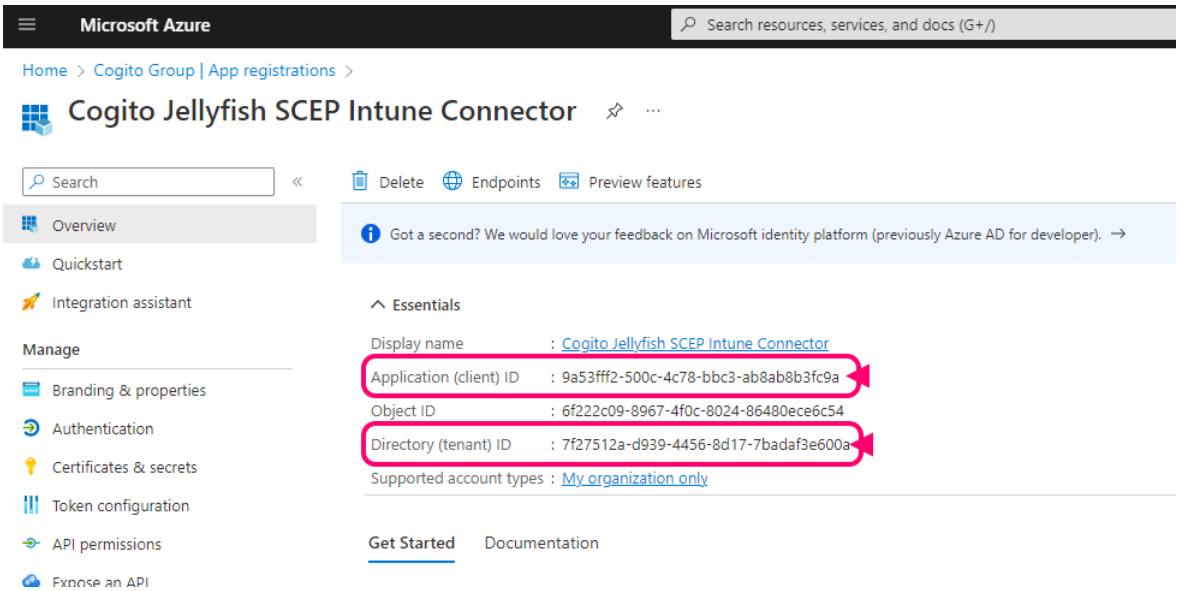


Figure 5: Application ID appears second in the list of *Essentials*. Directory ID appears fourth in the list of *Essentials*. Both GUIDs will have a copy button appears on the right-hand side when hovered.

5. In the left menu bar select *API Permissions.*
6. Remove all existing permissions, the default permissions are not required by the Cogito Jellyfish SCEP Intune Connector. This is done using the ellipsis to the right on the Microsoft Graph heading. After clicking the ellipsis select *Remove all permissions.*

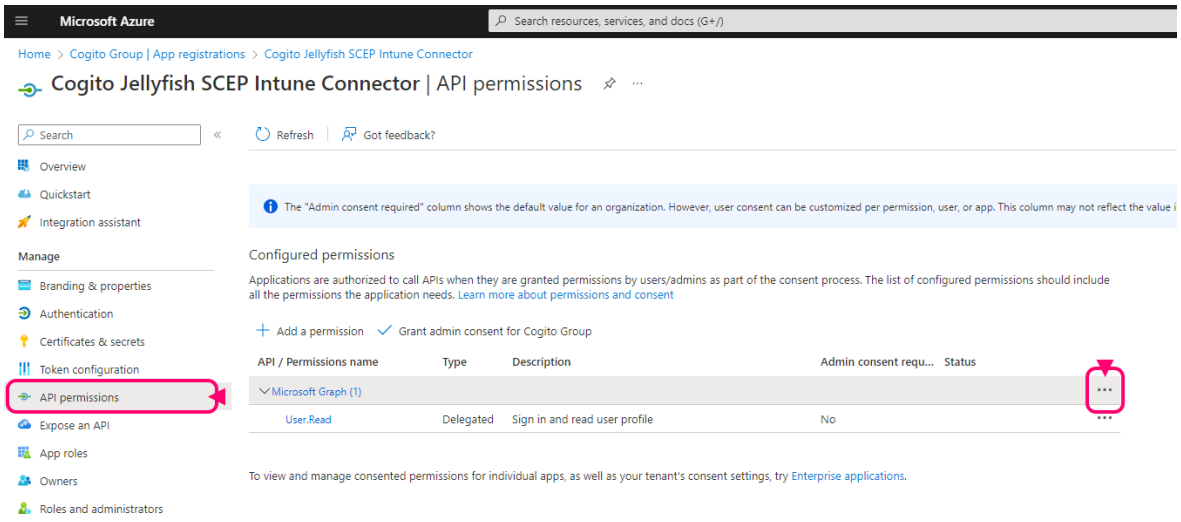


Figure 6: The *API permissions* menu item is on the left. All default permissions are removed using the ellipsis on the right under the Microsoft Graph heading.

7. Click *+ Add a permission* near the top of the *API permissions* blade.
    a. In the *Request API permissions* tab select *Microsoft Graph*
    b. Click *Application permissions*.
    c. Search for: *ServicePrincipalEndpoint.Read.All*

d. Expand *ServicePrincipalEndpoint* and tick *ServicePrincipalEndpoint.Read.All*

e. Click *Add permissions* at the bottom.

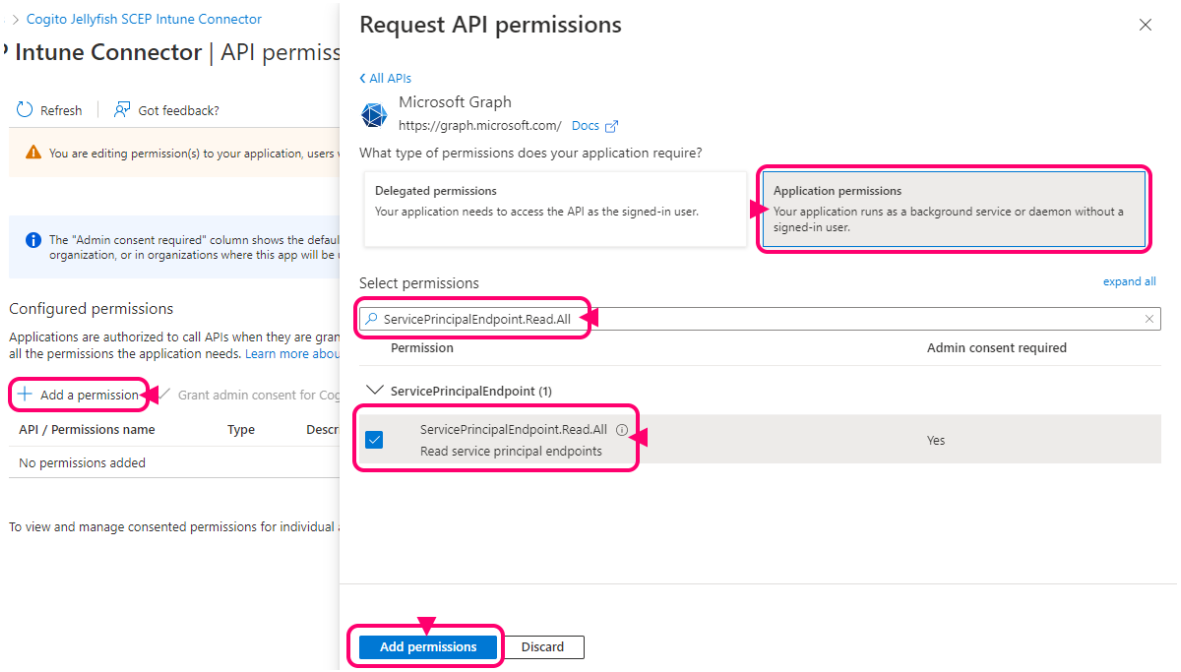


Figure 7: Click + *Add a permission* on the left. Choose *Microsoft Graph*. Then in order from top to bottom: select *Application permissions*, search for *ServicePrincipalEndpoint* tick *ServicePrincipalEndpoint.Read.All*, click *Add permissions*.

8. Click + *Add a permission* near the top of the *API permissions blade.*

   a. In the *Request API permissions tab* select *Intune*, it is mid-way down the list of service permissions.

   b. Click *Application permissions*.

   c. Search for *scep_challenge_provider*.

   d. Expand *Permissions* and tick *scep_challenge_provider*

   e. Click *Add permissions* at the bottom.

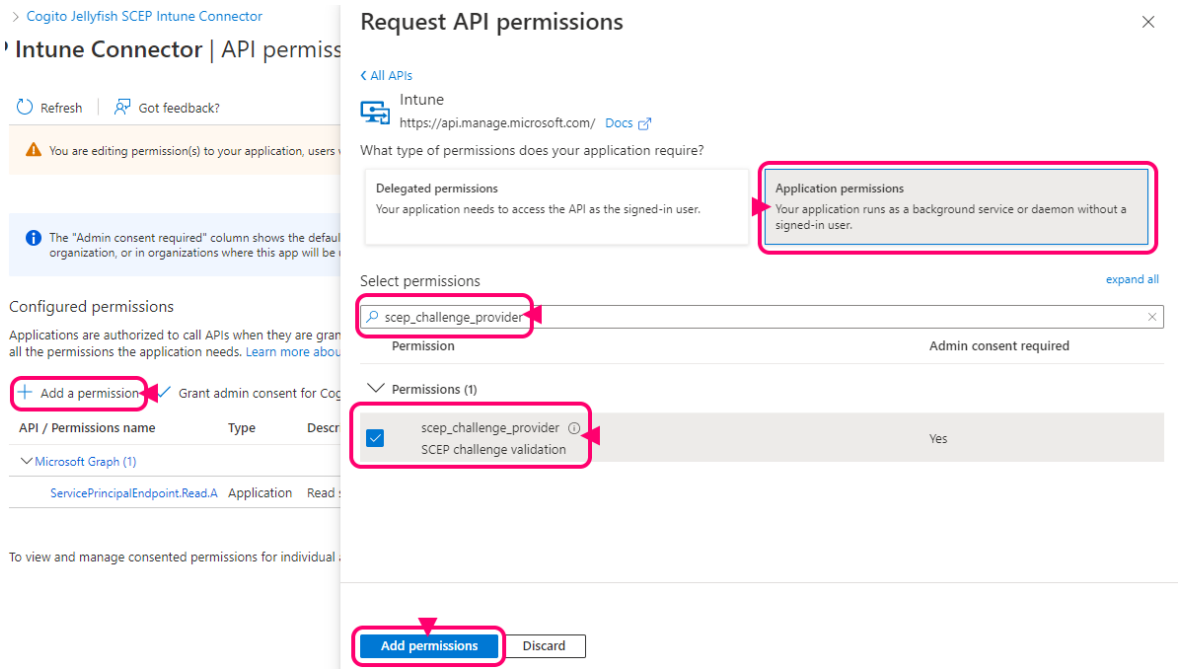


Figure 8: Click + *Add a permission* on the left. Choose *Intune*. Then in the order from top to bottom: select *Application permissions*, search for *scep_challenge_provider*, tick *scep_challenge_provider*, click *Add permissions.*

9. Click ✓ *Grant admin consent for*. This confirms the permissions changes. This requires administrative privileges for the Azure Active Directory tenancy.

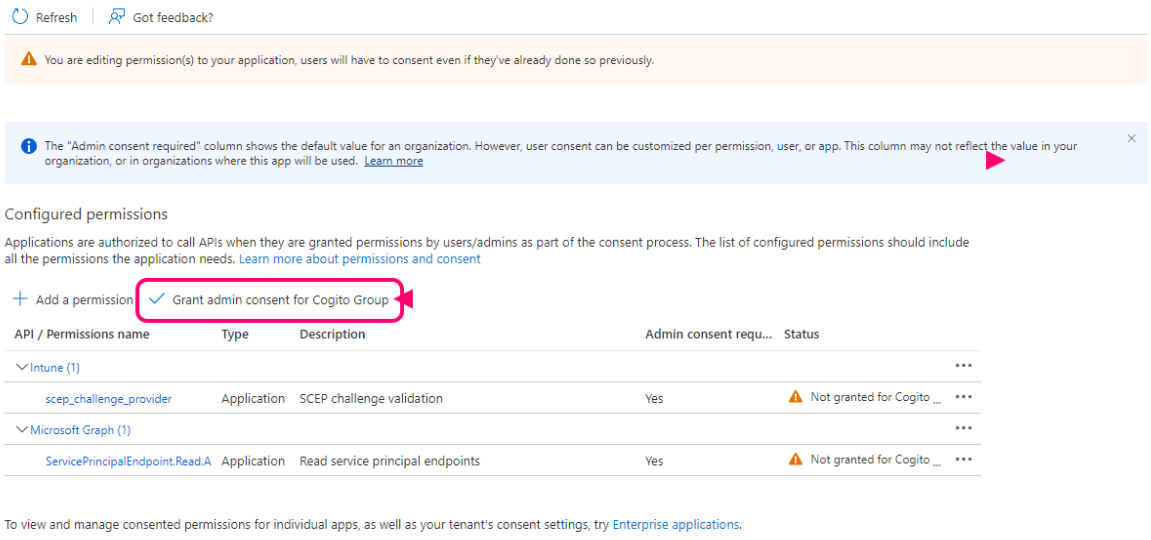


Figure 9: Grant admin consent applies the API permission changes.

10. Click *Certificates & secrets* in the left menu.
    a. Click + *New client secret* near the top of the page.

b. In the *Add a client secret* tab, enter a description that identifies this client secret. We recommend a description that contains the word: "Cogito Jellyfish Intune SCEP Connector Access".

c. Set an expiry compliant with your security policies. The secret value can be updated within jellyfish as detailed below.

d. Click *Add* at the bottom

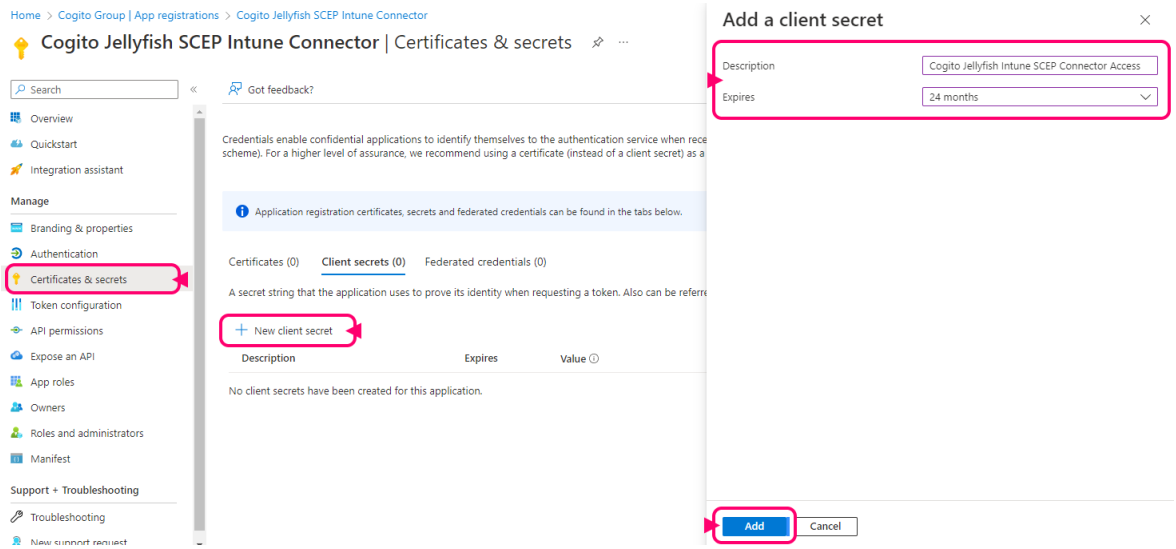


Figure 10: Certificates & secrets appears in the left menu bar. + *New client secret* near the top of the *Certificates & secrets blade*. Fill out the details in the *Add a client secret* tab, and click *Add.*

11. Immediately copy and record the *Value* of the secret that been created. The secret value will only appear this one time, after leaving this page the secret is lost forever and must be re-created. Note: the *Secret ID* is not required.

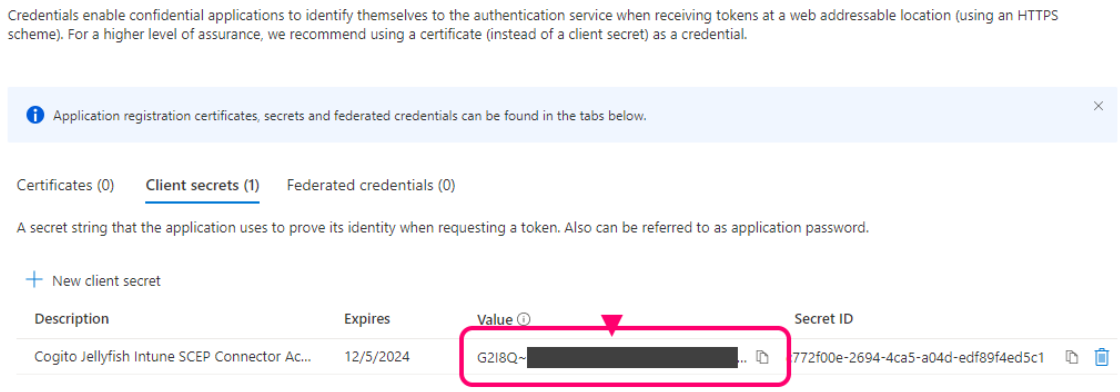


Figure 11: A copy button appears next to the *Value*. Note: the *Value* in this figure has been redacted, and will appear without censor during configuration.

4.3.3 Jellyfish Configuration

1. In Jellyfish, navigate to **Configuration -> SCEP Configuration** from the vertical menu.
2. Switch to the second tab “Intune Authentication”

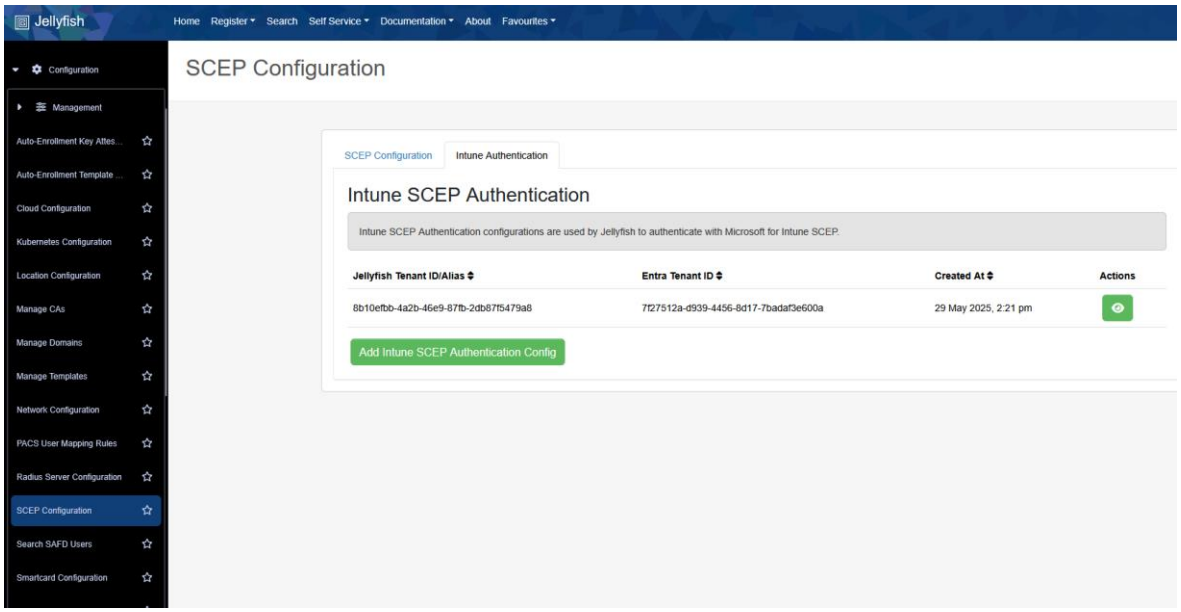


Figure 12 - Intune SCEP Configuration

3. Select “Add Intune SCEP Authentication Config”
4. Enter the three values recorded earlier:
    a. *Entra Application ID*
    b. *Entra Tenant ID*
    c. *Entra App Key (Secret Value)*
5. Confirm the configuration.
6. View your new config and click “Validate Configuration” to test that the details are correct and that the configuration works.

4.3.4 Trusted Certificate

Trusted Certificate configuration occurs in the *Microsoft Endpoint Manager admin center* service.

Prior to completing the following steps, ensure you have the Root Certificate Authority Certificate accessible. For more information on how to retrieve this certificate consult the *Prerequisites* section of this documentation.

1. In the far-left menu select *Devices*.
2. In the left sub menu select *Configuration profiles*. This may be searched for or found under the *Policy* section.
3. Near the top of the page click + *Create profile*.
4. In the *Create a profile* tab, select the platform you are configuring (in the example case *Windows 10 and later*)
5. Select the Profile type of *Templates*.
6. Search for *Trusted Certificate* and select it.
7. Click the *Create* button at the bottom.

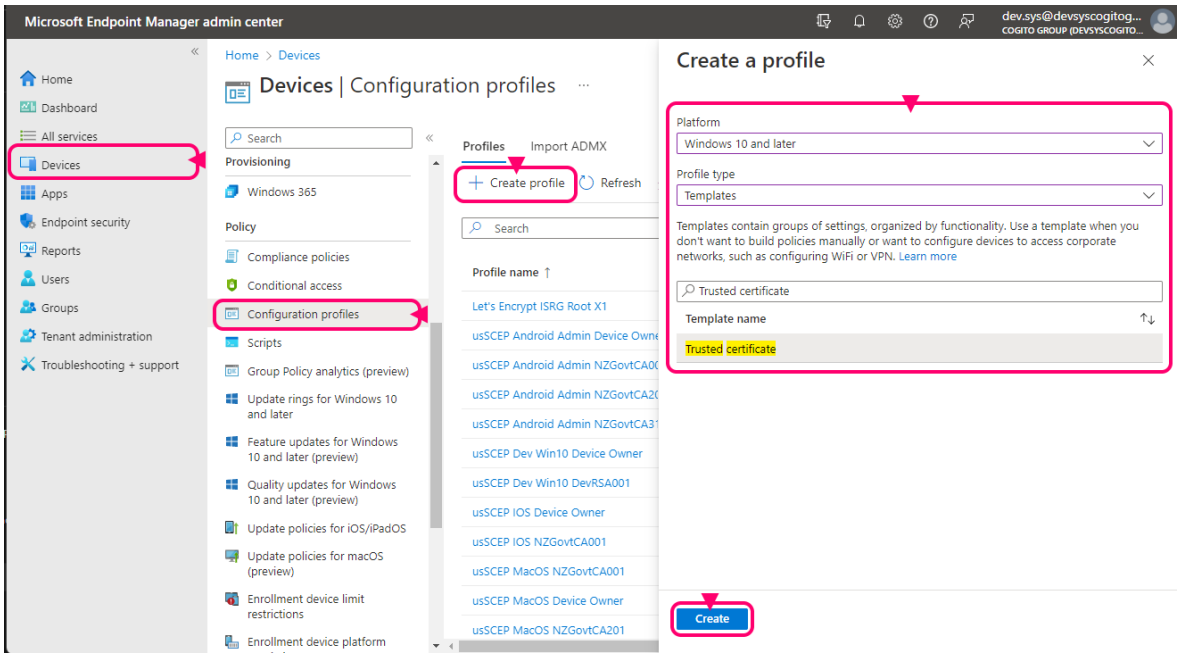


Figure 13: From left to right: Select *Devices*, select *Configuration profiles*, click + *Create profile*, enter the details as per the configuration documentation, click *Create*.

8. On the *Basics* tab of the *Trusted certificate* page, enter a Name to describe the profile both applied to Jellyfish SCEP and the Certificate Authority it represents. We recommend including the words: "Jellyfish SCEP - Trusted Certificate <CA-Name>" where CA Name is the name of the root certificate authority being deployed.
9. Optionally fill in a description. This is useful only for your organizations operational team.
10. Click *Next*.

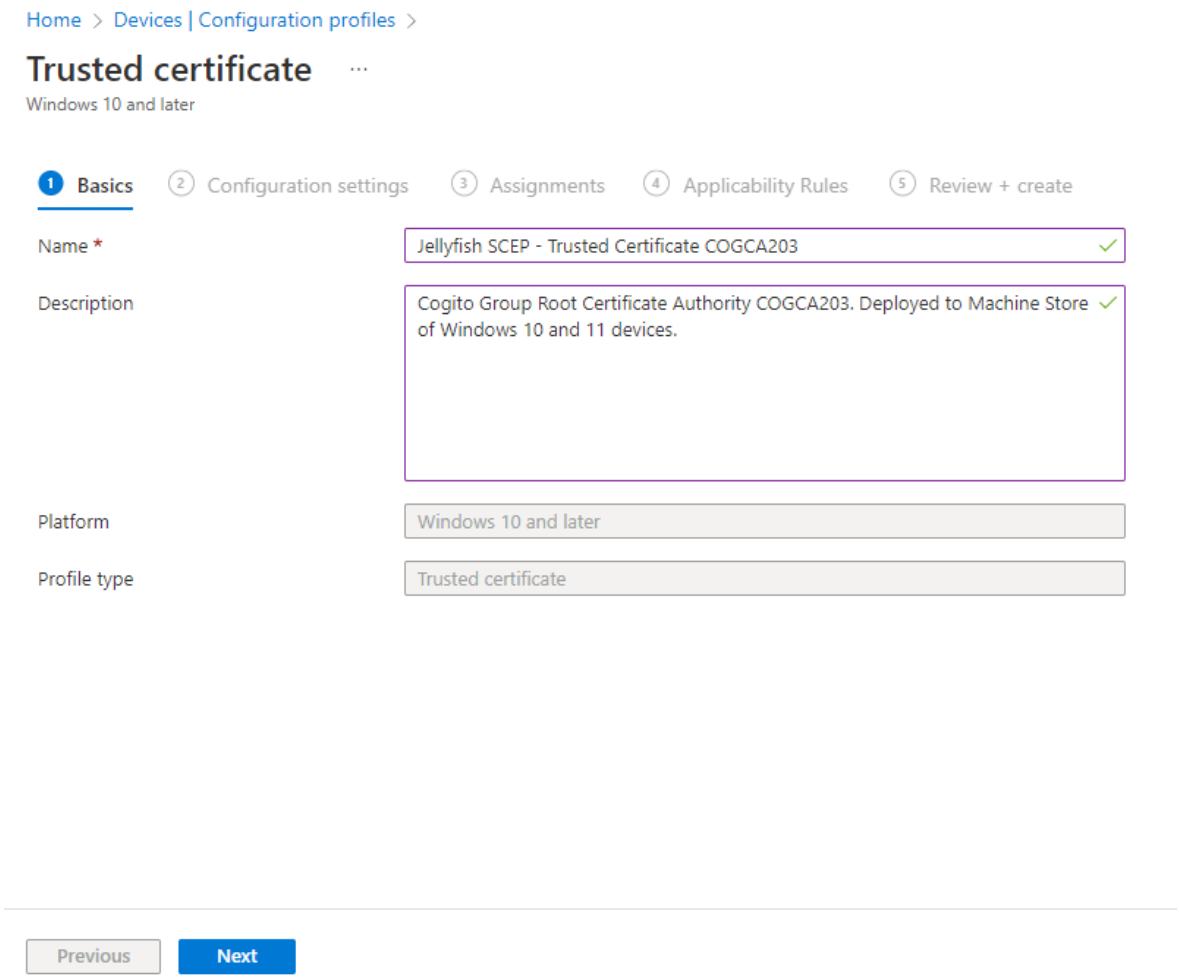


Figure 14: An example of a Name and Description that describes the COGCA203 Certificate Authority *Trusted Certificate Configuration profile.*

11. On the *Configuration settings* tab, upload the Root Certificate Authority Certificate collected earlier.
12. Choose the appropriate Destination Store, this is usually *Computer certificate Store - Root*.
13. Click *Next* near the bottom.
14. On the *Assignments* tab, under *Included groups* add the groups appropriate for your organization.
    a. When deploying Cogito Jellyfish SCEP Intune Connector for the first time, we recommend creating a Staged Rollout group, which may initially include only one user, additional users may be added to this group as confidence in the deployment grows.
    b. We do **NOT** recommend using the *Add all users* or *Add all devices* options for tenancies with a large number of users or devices.
    c. The larger the number of users or devices are configured simultaneously, the greater the load on our services to facilitate enrollment requests. If requesting a large number of enrolments be aware of some time delay between deploying the *Configuration profile* and all members of the associated groups completing their enrollment.

15. Click *Next* at the bottom.
16. Leave the *Applicability Rules* tab empty and click *Next* at the bottom.
17. Review the details of the *Trusted certificate Configuration profile.*
    a. We recommend ensuring the *Configuration settings* and *Assignment* sections are correct before continuing, a fault in this information may result in erroneous deployment of trusted certificates.
18. Click *Create* at the bottom.

4.3.5 Trusted Certificate - Android Specific

Android requires not only the Root Certificate Authority Certificate to be deployed as a *Trusted certificate Configuration profile*, but all element of the Certificate Authority Chain.

The number of entities may differ depending on the Certificate Authority that your Cogito Jellyfish SCEP Intune Connector's Registration Authority certificate was issued from.

- For NZTaaS customers using a Cogito hosted Certificate Authority, you will be using a Three Tier PKI including a Root, Intermediate, and Issuing Certificate Authority. E.g.:
  - NZGovtCA003 => NZGovtCA204 => NZGovtCA330

If you are unsure about which Certificate Authority Certificate must be deployed in your environment specifically for Android, contact your Cogito Operator at: Security.Services@cogitogroup.net

Ensure you mention you are configuring Android Trusted Certificates.

1. Follow the instructions in section: *Configuring Azure Intune - Trusted Certificate* for any subsequent Certificate Authority Certificates.

4.3.6 SCEP Certificate

SCEP Certificate configuration occurs in the *Microsoft Endpoint Manager admin center* service

1. In the far-left menu select *Devices*.
2. In the left sub menu select *Configuration profiles*. This may be searched for or found under the *Policy* section.
3. Near the top of the page click + *Create profile*.
4. In the *Create a profile* tab, select the platform you are configuring (in the example case *Windows 10 and later*)
5. Select the Profile type of *Templates*.
6. Search for *SCEP certificate* and select it.
7. Click the *Create* button at the bottom.

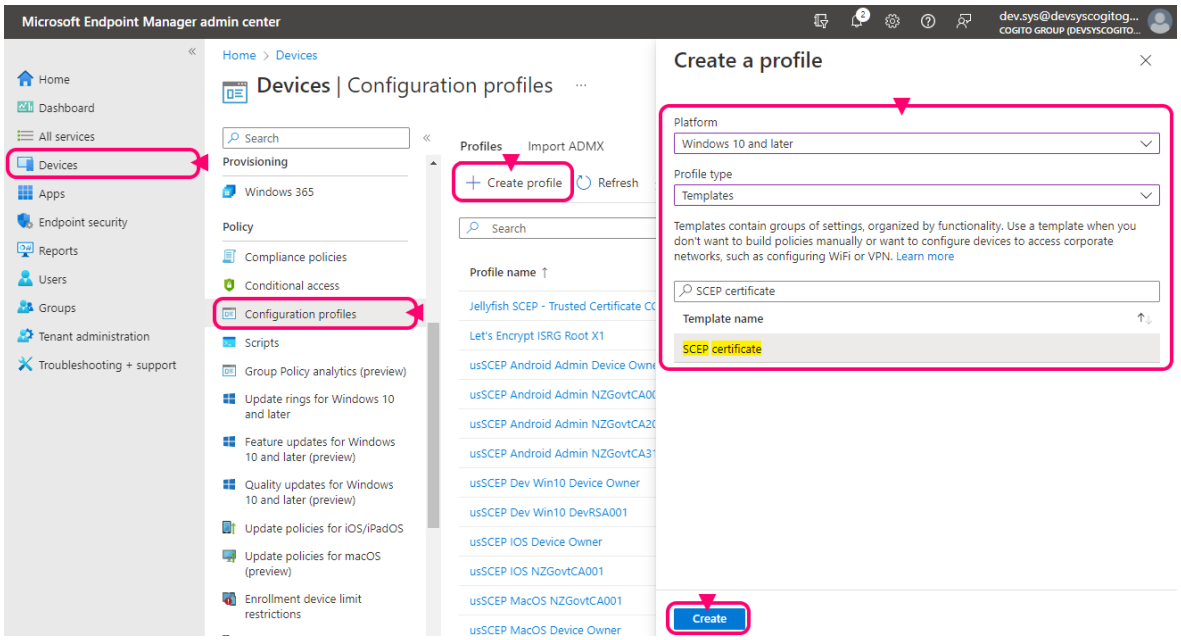


Figure 15: From left to right: Select *Devices*, select *Configuration profiles*, click + *Create profile*, enter the details as per the configuration documentation, click Create.

8. On the *Basics* tab of the *SCEP certificate* page, enter a Name to describe the profile both applied to Jellyfish SCEP and that this is a SCEP certificate profile, when deploying more than one certificate profile, ensure the names can be distinguished. We recommend including the words: "Jellyfish SCEP - SCEP Certificate".
9. Optionally fill in a description. This is useful only for your organizations operational team.
10. Click *Next*.

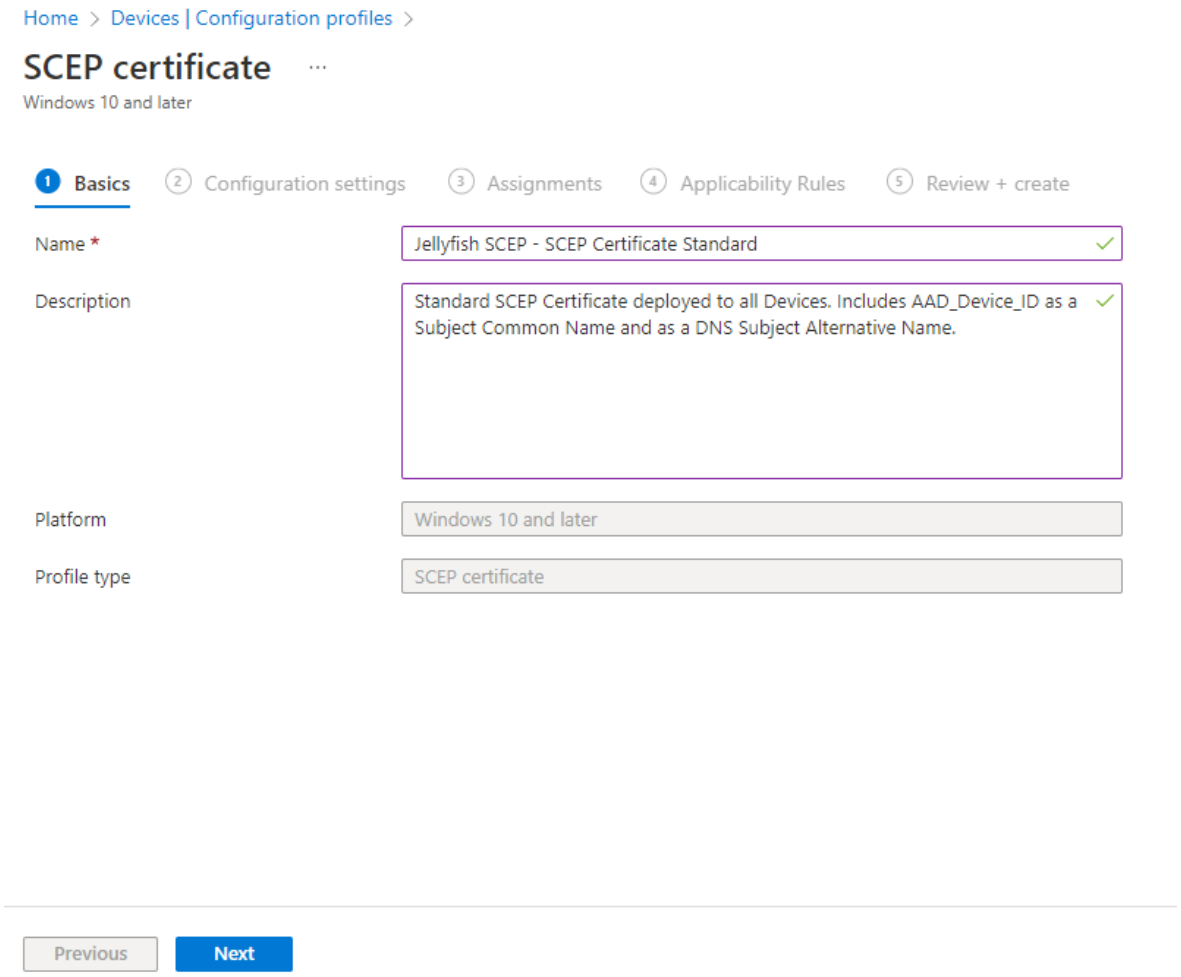


Figure 16: An example of a name and description of a Standard SCEP Certificate, configured as per the steps in this documentation.

11. On the *Configuration settings* tab, the following details are recommended. More specific details can be configured if required:
    a. Certificate type: *User,* or *Device* depending on your organization requirements, and which certificate profile is current being configured.
    b. Subject name format: *CN={{AAD_DEVICE_ID}}*, additional Subject Names may be configured as required.
    c. Subject alternative name:
        i. Attribute: DNS
        ii. Value: {{AAD_DEVICE_ID}}
    d. Certificate validity period: Years 1
    e. Key storage provider (KSP): Enroll to Trusted Platform Module (TPM) KSP if present, otherwise Software KSP.
    f. Key usage: Digital signature and Key encipherment
    g. Key Size (bits): 2048
    h. Hash algorithm: SHA-2

i. Root Certificate: The *Trusted certificate Configuration profile* as configured in the above section *Trusted Certificate*. In the case of an Android configuration, ensure all Root Certificates created are selected here.

j. Extended Key Usage:

   i. Name: Client Authentication

   ii. Object Identifier: 1.3.6.1.5.5.7.3.2

   iii. Predefined Values: Client Authentication

k. Renewal threshold (%): 20

l. SCEP Server URLs: The URL provided by your Cogito Operator as part of your onboarding procedure. For more details refer to the Prerequisites section of this document.

12. For more details regarding configuring SCEP certificate details, consult the Microsoft SCEP Certificate configuration documentation available here: https://learn.microsoft.com/en-us/mem/intune/protect/certificates-profile-scep

13. Click *Next* at the bottom.

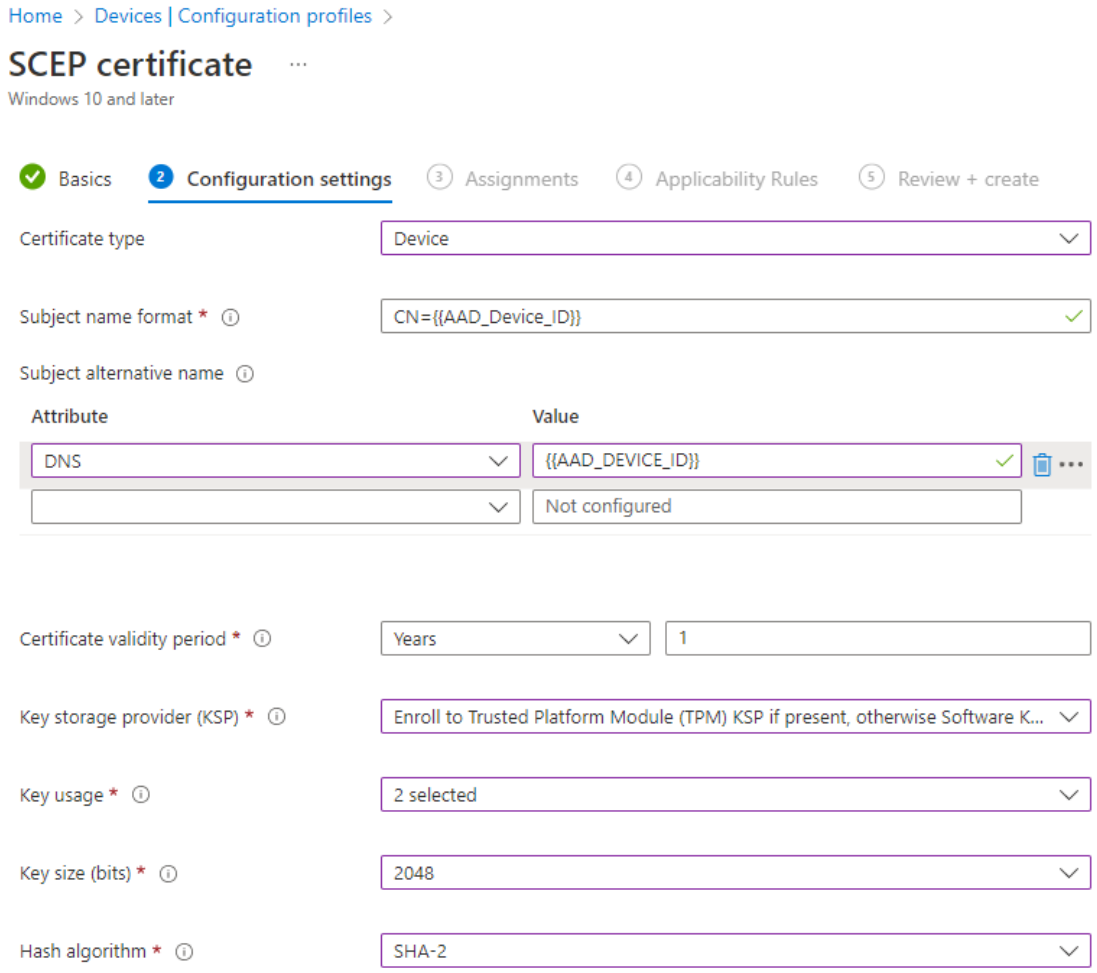


Figure 17: Example details of recommended SCEP certificate settings Part 1.

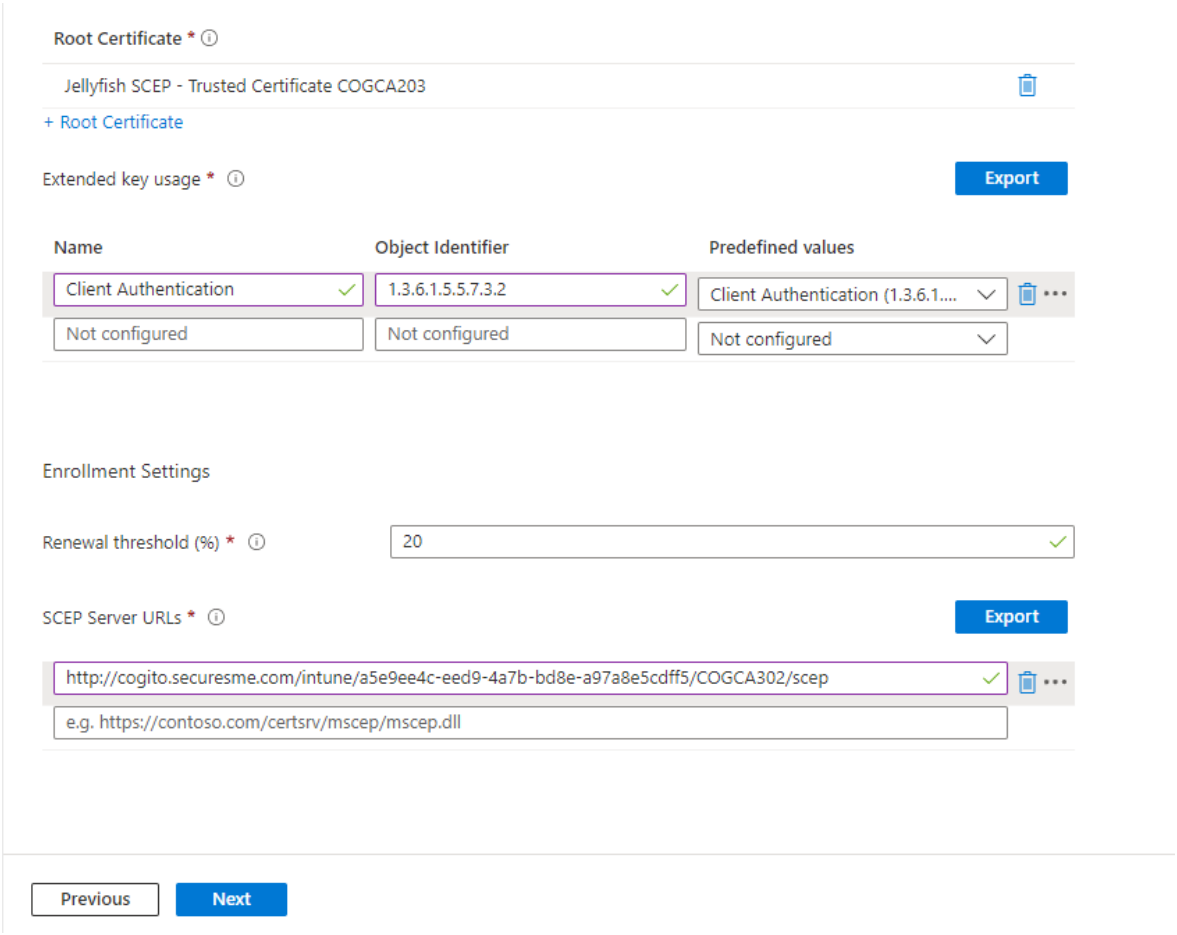


Figure 18: Example details of recommended SCEP certificate settings Part 2.

14. On the *Assignments* tab, under *Included groups* add the groups appropriate for your organization.
    a. When deploying Cogito Jellyfish SCEP Intune Connector for the first time, we recommend creating a Staged Rollout group, which may initially include only one user, additional users may be added to this group as confidence in the deployment grows.
    b. We do **NOT** recommend using the *Add all users* or *Add all devices* options for tenancies with a large number of users or devices.
    c. The larger the number of users or devices are configured simultaneously, the greater the load on our services to facilitate enrollment requests. If requesting a large number of enrolments: be aware of some time delay between deploying the *Configuration profile* and all members of the associated groups completing their enrollment.
15. Click *Next* at the bottom.
16. Leave the *Applicability Rules* tab empty and click *Next* at the bottom.
17. Review the details of the *SCEP certificate Configuration profile.*
    a. We recommend ensuring the *Configuration settings* and *Assignment* sections are correct before continuing, a fault in this information may result in erroneous enrollment of SCEP certificate, or more likely a failure to issue any certificates.

b. We recommend triple checking the SCEP Server URLs at this point, comparing against the URL provided in the SCEP onboarding process. Ensure devices can hit this URL and that there is no restriction within your network on this URL.

c. Network connectivity to this URL can be tested by suffixing the URL with the following query parameters:
?operation=getcacaps

E.g.: https://jellyfish.securesme.com/intune/0/0/scep?operation=getcacaps

The expected results in a plain text list of Cogito Jellyfish Intune SCEP Connector capabilities. If this webpage fails to load, the device will not be able to complete a SCEP enrollment.

18. Click *Create* at the bottom.

4.3.7 Application specific setup

4.3.7.1 Jellyfish TeSA / Radius Certificate

Radius certificates require the Client Authentication key usage to authenticate. security.services@cogitogroup.net can setup a template with this EKU enabled.

If Jellyfish controlled VLANs are required, certificates additional require a domain to be configured for the network in Jellyfish Portal. And all certificates for that network require the domain to be part of the DNS, UPN or Common Name.

5 Simple SCEP

5.1 CertMonger / General

5.1.1 Overview

Simple Certificate Enrollment Protocol (SCEP) is a form of Automated Certificate Enrollment most used in combination with Microsoft Azure Intune, or Microsoft Autopilot. SCEP is similar to the Microsoft NDES technology, the difference being NDES is designed for use solely within the Microsoft Active Directory environment and is not intended for portable devices like phones or tablets. SCEP is capable of all NDES is capable of, with the added benefits of integration into various Microsoft Azure platforms, and additional compatibility with devices like printers, office phones, and Linux machines.

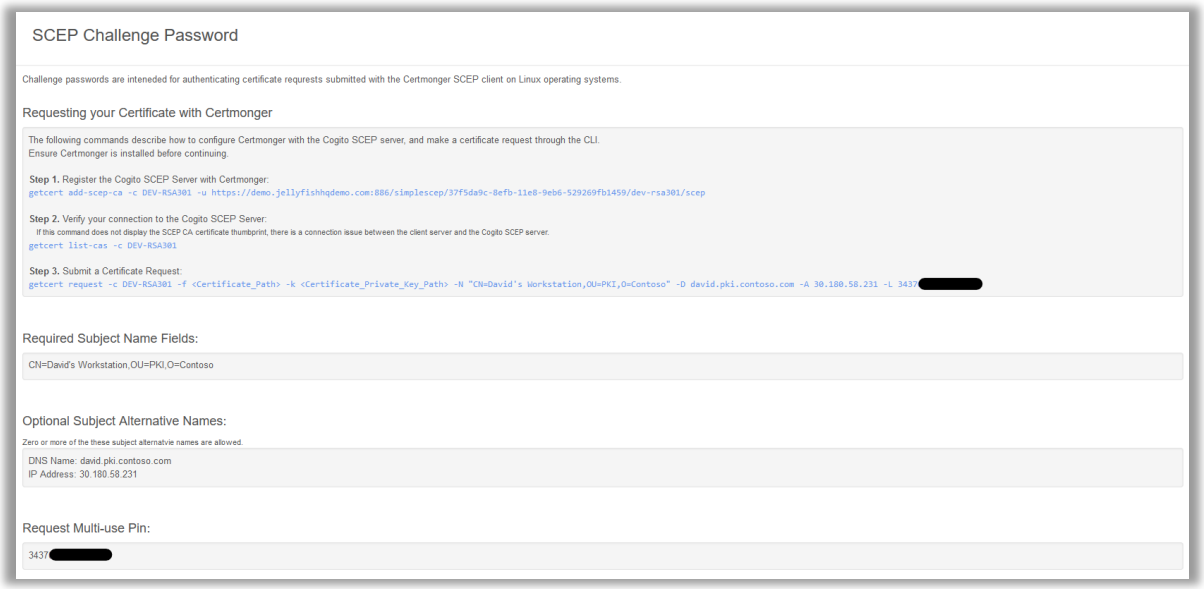
Cogito Jellyfish SCEP is also available to various tools that support a pre-Shared key form of the SCEP protocol. This type of SCEP Enrollment is most used for devices such as office phones, printers, or Linux servers.

Cogito Jellyfish SCEP may be used without the requirement of installing client-side software, certificate exchange is orchestrated through a negotiation process like a TLS handshake.

5.1.2 Request SCEP Certificate

1. Simple Certificate Enrolment Protocol (SCEP) certificate requests can be made from Jellyfish.
2. To make a SCEP request use:
    a) **Credential Management -> Certificate -> Request SCEP Certificate** from vertical menu.
2. Jellyfish uses Multi-Use SCEP tokens to validate CSRs sent to the SCEP endpoint. These tokens must be accompanied by a CSR that exactly match the values defined on the SCEP Certificate Request Page, with a couple of exceptions:
    a) Any part of the Subject Name must be defined, in the case where the Subject Name fields filled out are less specific than a Common Name, the more specific components become wildcard where any value of the more specific fields is acceptable.
    E.g.
        i) OU=PKI,O=Contoso: will allow subjects such as: CN=David's Workstation,OU=PKI,O=Contoso.
        ii) C=NZ: will allow subjects such as: CN=David's Second Certificate,OU=Authentication,O=Cogtoso,C=NZ
        iii) Be careful when specifying the Subject Name and ensure it is as specific as possible for your purpose.
    b) If one or more Subject Alternative Names are defined within the SCEP Certificate Request, zero or more of those specified may exist on the SCEP CSR. However, a SCEP CSR containing a Subject Alternative Name not defined within the SCEP Certificate Request will result in the CSR being declined.
3. To generate a SCEP token, fill out the fields you require in the same manner a normal certificate would be requested (with the notable exception that any of the Subject Name fields 'may' be filled out).

4. Be sure to set the contact email to inbox you have control over, as a Hyper Link to your Multi-Use SCEP token will be included in an email send to this email address.
5. Once you submit the SCEP Certificate Request, it will be sent for certificate approval which needs to be approved by someone in the tenancy with the appropriate permissions. In the case where the requesting use has permission for self-approvals this step is not required.
6. Once the request is approved, an email will be sent to the contact email specified in the SCEP Certificate Request containing a hyper link to a location in the Jellyfish Anonymous Portal in which your SCEP Multi-Use token can be retrieved.
7. Follow the link and arrive at a one-time use page describing how to consume your SCEP Certificate Request:



8. Follow the instructions under the *Requesting your Certificate with Certmonger* section of the page. The command here are dynamic and have the fields described within you SCEP Certificate Request pre-filled. It may be the case that *Step 3. Submit a Certificate Request* needs to be modified slightly to include different Subject and Subject Alternative Names. **NOTE:** Certmonger is an open source available on Linux machines, other options for consuming the SCEP Challenge Password are available.

5.2 FortiGate Firewall

Fortinet offers FortiGate Firewalls which support SCEP Autoenrollment.

The firewall's SCEP client however *doesn't support HTTPS* and SCEP enrolment must therefore be performed over HTTP to integrate with Jellyfish.

5.2.1 Request SCEP Certificate

Once logged into the FortiGate Web interface, navigate to **System -> Certificates** and select the **Create/Import** dropdown -> **Generate CSR**.

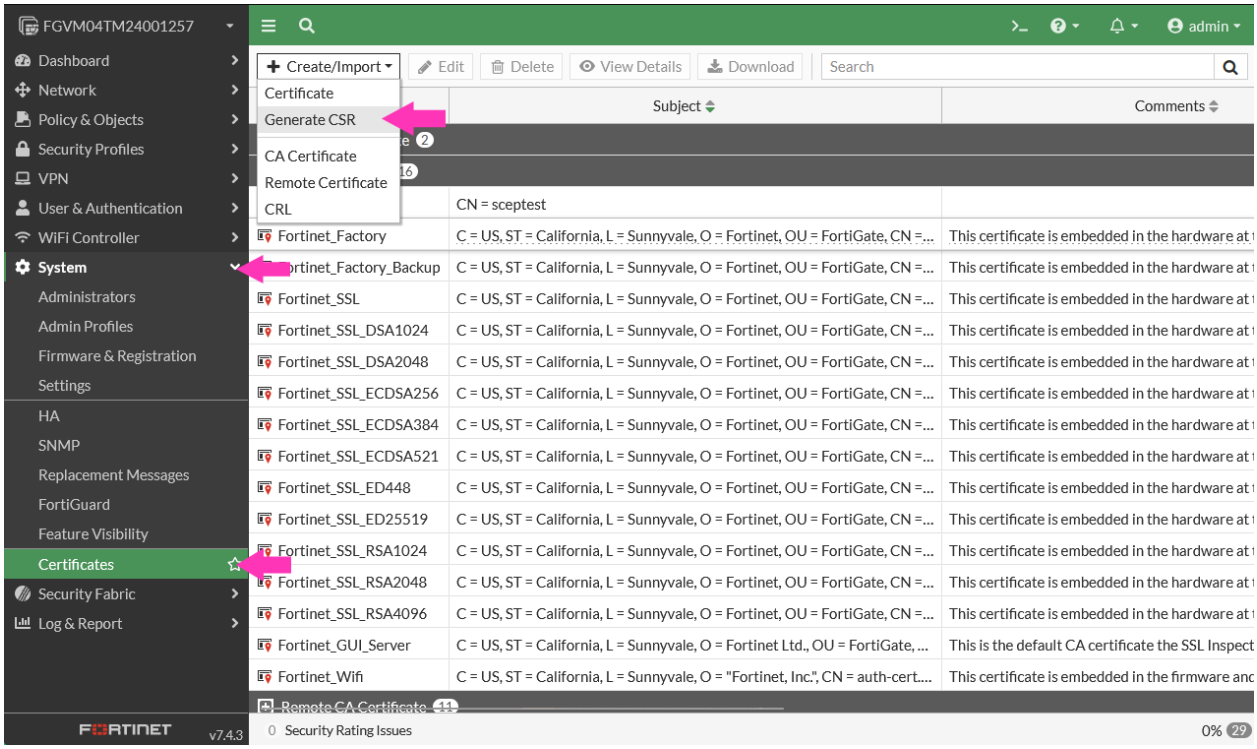


Figure 19 - FortiGate's certificate page

In the Generate Certificate Signing Request page, supply a Certificate Name (only used as a display name in the certificate list in the previous page)

- Supply Subject information, such as **IP, DNS, or Email address**
- Supply any optional fields in the **Optional Information** section.
- Select a **Key Type** and **Key Size** or **Curve**
- **For Enrollment Method** select **Online SCEP**
- Provide the **Jellyfish SCEP Server URL** & **Challenge Password**
- Click **OK**

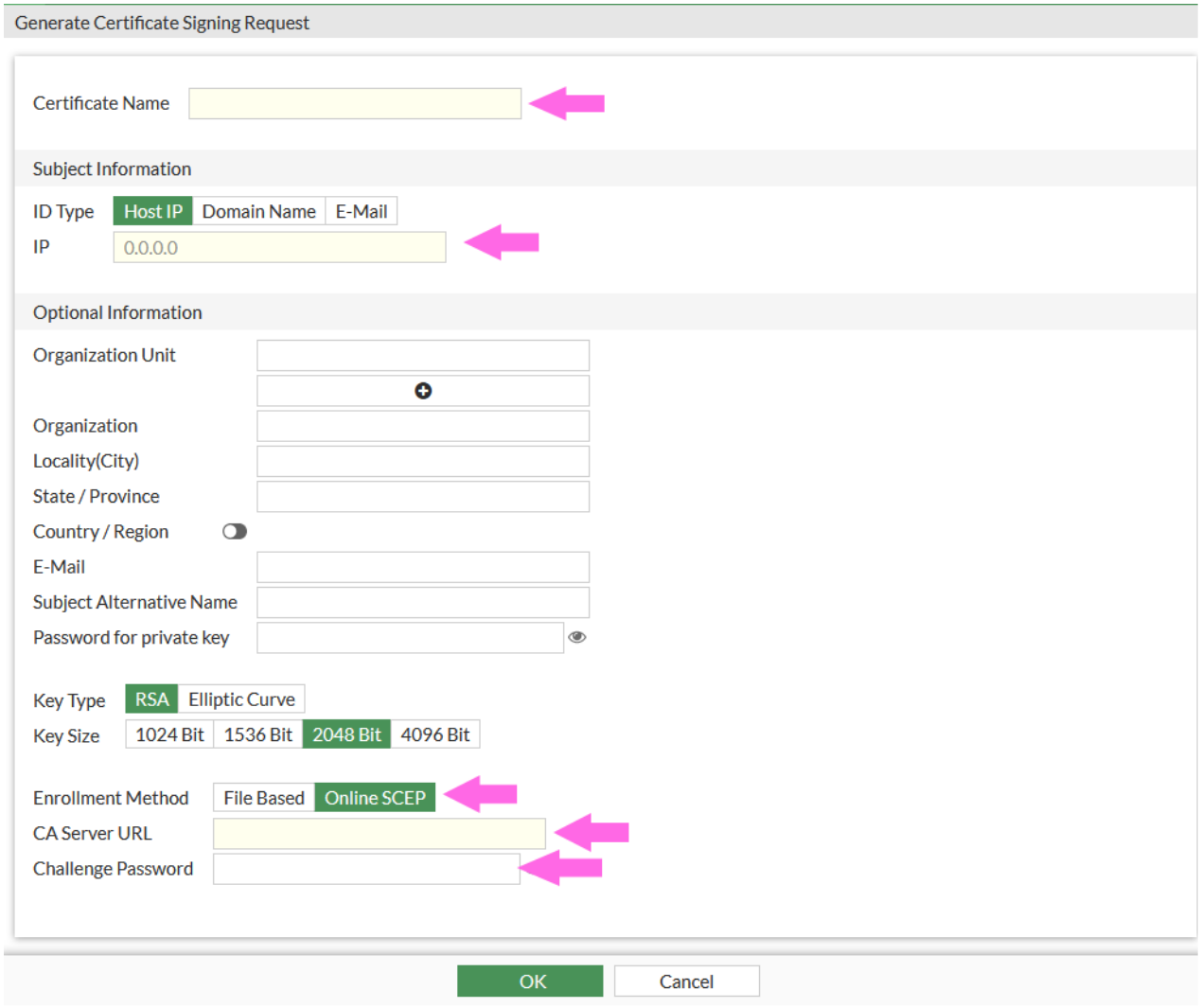


Figure 20 - FortiGate generate certificate signing request page

Once submitted, you will be taken back to the Certificate List page, your new certificate will have a Pending status until it is successfully issued and returned to the FortiGate device.

5.2.2 Auto Enrollment

After successfully requesting a SCEP Certificate, the process can be automated so that the certificate is re-issued and rotated before expiry.

Once the certificate is successfully imported, the auto-regenerate option can be configured in the CLI if it is required. It will ensure that the certificate will automatically renew before expiry:

```
config vpn certificate local
    edit <name>
        set auto-regenerate-days {integer}
        set auto-regenerate-days-warning {integer}
    next
end
```

auto-regenerate-days = Number of days to wait before the expiry of an updated local certificate is requested (0 = disabled).

auto-regenerate-days-warning = Number of days to wait before an expiry warning message is generated (0 = disabled).

Note: As of FortiOS 7.0.4: if a certificate signing is made by an intermediate CA, the root certificate needs to be in the SCEP client certificate repository so that the intermediate CA's issuer can be checked.

5.2.3 Debug Logs

To view Debug logs to troubleshoot possible issues during SCEP enrolment. Log into the Appliance's CLI, and execute the following commands:

Set the log level

```
diagnose debug application scep -1
```

Enable debug logging

```
diagnose debug enable
```

Successful SCEP Enrollment can be observed with the following debug logs:

```
scep_parse_header: server returned status code 200
scep_parse_header: MIME header: application/x-x509-ca-cert
__process_ca_cert_reply: Reply type 1
__process_ca_cert_reply: loaded cacert
__read_ca_cert_cb: loaded signing cert
__read_ca_cert_cb: loaded CA
new_scep_transaction: transaction id: 2B9E443E5E0A9BA816785A1915AA2E99
pkcs7_wrap:1120 creating inner PKCS#7
pkcs7_wrap: data payload size: 775 bytes
pkcs7_wrap: successfully encrypted payload
pkcs7_wrap: envelope size: 1280 bytes
pkcs7_wrap: creating outer PKCS#7
pkcs7_wrap: signature added successfully
pkcs7_wrap: adding signed attributes
__add_attribute_string: adding string attribute transId
__add_attribute_string: adding string attribute messageType
__add_attribute_octet: adding octet attribute senderNonce
pkcs7_wrap: PKCS#7 data written successfully
pkcs7_wrap: applying base64 encoding
pkcs7_wrap: base64 encoded payload size: 3953 bytes
scep_parse_header: server returned status code 200
scep_parse_header: MIME header: x-pki-message
pkcs7_unwrap: reading outer PKCS#7
pkcs7_unwrap: PKCS#7 payload size: 756 bytes
```

```
pkcs7_unwrap: PKCS#7 contains 0 bytes of enveloped data
pkcs7_unwrap: verifying signature
pkcs7_unwrap: signature ok
pkcs7_unwrap: finding signed attributes
__get_attribute: finding attribute transId
__get_signed_attribute: allocating 32 bytes for attribute
pkcs7_unwrap: reply transaction id: 2B9E443E5E0A9BA816785A1915AA2E99
__get_attribute: finding attribute messageType
__get_signed_attribute: allocating 1 bytes for attribute
pkcs7_unwrap: reply message type is good
__get_attribute: finding attribute senderNonce
__get_signed_attribute: allocating 16 bytes for attribute
pkcs7_unwrap: senderNonce in reply: 65DCB45E9FCB8CBB4395C99D8B17BD92
__get_attribute: finding attribute recipientNonce
__get_signed_attribute: allocating 16 bytes for attribute
pkcs7_unwrap: recipientNonce in reply: 57E9E8B7D23E2912BCADE9BCACA90F08
__get_attribute: finding attribute pkiStatus
__get_signed_attribute: allocating 1 bytes for attribute
pkcs7_unwrap: pkistatus: PENDING
pkcs7_wrap:1138 creating issuer_and_subject PKCS#7
pkcs7_wrap: data payload size: 267 bytes
pkcs7_wrap: successfully encrypted payload
pkcs7_wrap: envelope size: 776 bytes
pkcs7_wrap: creating outer PKCS#7
pkcs7_wrap: signature added successfully
pkcs7_wrap: adding signed attributes
__add_attribute_string: adding string attribute transId
__add_attribute_string: adding string attribute messageType
__add_attribute_octet: adding octet attribute senderNonce
pkcs7_wrap: PKCS#7 data written successfully
pkcs7_wrap: applying base64 encoding
pkcs7_wrap: base64 encoded payload size: 3271 bytes
scep_parse_header: server returned status code 200
scep_parse_header: MIME header: x-pki-message
pkcs7_unwrap: reading outer PKCS#7
pkcs7_unwrap: PKCS#7 payload size: 5996 bytes
```

```
pkcs7_unwrap: PKCS#7 contains 2899 bytes of enveloped data
pkcs7_unwrap: verifying signature
pkcs7_unwrap: signature ok
pkcs7_unwrap: finding signed attributes
__get_attribute: finding attribute transId
__get_signed_attribute: allocating 32 bytes for attribute
pkcs7_unwrap: reply transaction id: 2B9E443E5E0A9BA816785A1915AA2E99
__get_attribute: finding attribute messageType
__get_signed_attribute: allocating 1 bytes for attribute
pkcs7_unwrap: reply message type is good
__get_attribute: finding attribute senderNonce
__get_signed_attribute: allocating 16 bytes for attribute
pkcs7_unwrap: senderNonce in reply: 976C51687F9263B79BF8CDF964136EF6
__get_attribute: finding attribute recipientNonce
__get_signed_attribute: allocating 16 bytes for attribute
pkcs7_unwrap: recipientNonce in reply: 40460901E83309A3022FF353C7EA1183
__get_attribute: finding attribute pkiStatus
__get_signed_attribute: allocating 1 bytes for attribute
pkcs7_unwrap: pkistatus: SUCCESS
pkcs7_unwrap: reading inner PKCS#7
pkcs7_unwrap: decrypting inner PKCS#7
pkcs7_unwrap: PKCS#7 payload size: 2384 bytes
scep_write_local_cert: found certificate with
subject:
   /ST=NH/L=Amsterdam/O=Fortinet/OU=AS/CN=myvpn/emailAddress=ipsecvpntest@forti
   net.com
issuer:
   /C=NL/ST=NH/L=Amsterdam/O=Fortinet/OU=AS/CN=sceptest/emailAddress=sceptest@f
   ortinet.com
scep_write_local_cert: writing cert
scep_write_local_cert: certificate written as /tmp/IPSECVPNTest
```

5.3 Sonicwall Firewall

5.3.1 Configure Jellyfish

The Sonicwall Firewall requires unique and unusual certificate chain configuration that is not compatible with standard Simple SCEP or Intune SCEP. If you already consume SCEP certificates on another device, create a new configuration with a distinct configuration 'Name', e.g.: "Intune Computer - Sonicwall".

When setting a configuration specifically for a Sonicwall appliance, the certificate chain MUST INCLUDE the enrolment agent certificate, in addition to the enrolment agent certificate being provided to the service. This will result in the 'CA Certs' response including a chain with a duplicate enrolment agent certificate, e.g.: enrolment agent, enrolment agent, issuer, intermediate, root.

This deviates from a typical configuration where only a single enrolment agent certificate is include in the 'CA Certs' response.

In the Jellyfish portal, configure a certificate template profile to ignore signature validation. The Sonicwall signs some pkcs10 objects with MD5, regardless of the key configuration. Without this configuration Jellyfish will fails to verify the MD5 signature and fail certificate enrolment.

5.3.2 Create SCEP request

Create a simple SCEP request using the Jellyfish interface. We recommend setting the appliances Fully qualified distinguished name as the Subject and DNS SAN.

Create the same SCEP certificate request through the certificate portal page on the Sonicwall appliance. We recommend an RSA256-2048 or ECDSA256 algorithm and or key size.

5.3.3 Submit SCEP request

Using the certificate portal page on the Sonicwall submit the certificate request created in above step. Use the details provided by the SCEP request retrieval. Ensure a HTTP URL is used, enrolment will fail if an HTTPS URL is attempted.

6 Troubleshooting

6.1 Intune Certificates Issued but Intune Errors

If the retrieved certificate is missing Key Usages, Microsoft Intune will report errors on issuance despite the certificate being in the client device certificate store. Ensure the Intune portal has the Key Usages configured as per 4.3.1 Microsoft Azure Portal.