MyID Enterprise

Version 12.13

Mobile Identity Management

Lutterworth Hall, St Mary's Road, Lutterworth, Leicestershire, LE17 4PS, UK
www.intercede.com | info@intercede.com | @intercedemyid | +44 (0)1455 558111

Document reference: INT1958-12.13.0-Enterprise

December 2024

Copyright

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

© You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License. ---

Bouncy Castle

Copyright © 2000 – 2011 The Legion Of The Bouncy Castle (http://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

KSoap2

Copyright © 2003,2004 Stefan Haustein, Oberhausen, Rhld., Germany

Copyright © 2006, James Seigel, Calgary, AB., Canada

Conventions used in this document

- Lists:
  - Numbered lists are used to show the steps involved in completing a task when the order is important.
  - Bulleted lists are used when the order is unimportant or to show alternatives.
- **Bold** is used for menu items and for labels.

  For example:
  - Record a valid email address in **'From' email address**.
  - Select **Save** from the **File** menu.
- *Italic* is used for emphasis:

  For example:
  - Copy the file *before* starting the installation.
  - Do *not* remove the files before you have backed them up.
- ***Bold and italic*** hyperlinks are used to identify the titles of other documents.

  For example: "See the ***Release Notes*** for further information."

  Unless otherwise explicitly stated, all referenced documentation is available on the product installation media.
- A `fixed width` font is used where the identification of spaces is important, including filenames, example SQL queries and any entries made directly into configuration files or the database.
- **Notes** are used to provide further information, including any prerequisites or configuration additional to the standard specifications.

  For example:

  Note: This issue only occurs if updating from a previous version.
- Warnings are used to indicate where failure to follow a particular instruction may result in either loss of data or the need to manually configure elements of the system.

  For example:

  Warning: You must take a backup of your database before making any changes to it.

Contents

1 Introduction

This document provides information on the support for MyID® Mobile Identity Management (MIM), including details on the following:

- Configuring the system to support the installation of mobile identities on your mobile devices.
- Requesting mobile identities through MyID.
- Managing mobile identities through MyID.

This release provides support for a range of Android and iOS mobile devices.

In this document, the words *mobile device* may refer either to a smartphone or a tablet. Some devices are unable to receive SMS messages but can receive email messages capable of starting the identity management process.

For information on using the mobile identities on your mobile device, see the information accompanying the individual client applications.

Note: There is a significant overlap in the configuration and use of mobile identities and mobile identity documents. For information on mobile identity documents, see the ***Mobile Identity Documents*** guide.

2 Overview

To receive credentials from MyID CMS, the end user must use an app that has been integrated with the solution. Intercede works with a range of services to achieve this; for example, Mobile Device Management systems and third-party app developers. Typically, this requires you to install the service's own mobile app. For more information, see section *6*, *MDMs and derived credentials*.

This feature allows you to request a mobile ID from your MyID system and store it on your mobile device; this allows you to use secure certificates with your email application for reading and writing encrypted and signed emails, display an identity badge, and so on.

The process is as follows:

1. You install the mobile app on your mobile device.
2. Using MyID, a MyID operator requests (and optionally approves) an ID for your mobile device.

   You can use the MyID Operator Client or MyID Desktop. Alternatively, you can use the MyID Core API or the Credential Web Service API to make the request.
3. MyID uses email or an SMS gateway to send a message to the user's email address or phone number that is stored as the **Cell** or **Mobile** (depending on the language setting) number in the MyID record.
4. When the message is received on your mobile device, you click the link or notification.
5. The type of notification depends on your mobile device type and whether the message is sent through SMS or email. Follow the instructions displayed on your mobile device.

   Alternatively, you can request an ID for your own device, and rather than use an email or SMS notification, you can choose to display a QR code on screen you can scan with the mobile app.

   Alternatively, you can use the MyID Operator Client or MyID Desktop to request an ID for your own device. If you are using MyID Desktop to request your own ID, rather than use an email or SMS notification, you can choose to display a QR code on screen that you can scan with the mobile app.
6. You use the mobile app to download the certificates and badge layouts to your mobile device from the MyID web service.
7. You can now use the credentials delivered to your mobile device.

2.1 Supported devices

Devices running the following operating systems are supported:

- iOS.
- Android.

See the information provided with your wallet app for information on the specific versions supported. For more information about mobile operating system support, contact Intercede customer support quoting reference SUP-49.

2.2 Supported key stores

When credentials are delivered to mobile devices, they must be securely stored to protect the integrity of the credential. Depending on the mobile app you are using, the location may vary; for example, an app that is used with a Mobile Device Management system typically uses its own key store on the device.

For further details on how the mobile app stores credentials, contact the app vendor.

Note: Issuing and recovering certificates with elliptic curve cryptography (ECC) keys to mobile devices is not currently supported.

2.3 Supported Mobile Device Management integration

MyID currently supports integration with the following Mobile Device Management (MDM) and associated systems:

- Microsoft Intune.
- VMWare Workspace ONE.

Contact the relevant vendor for full details of how to configure these MDM systems for integration with MyID.

Note: You can set up more than one MDM connector for each instance of MyID; you must specify which MDM connector to use in the credential profile. You can set up multiple MDM connectors of the same type, or multiple MDM connectors of mixed types; for example, you could set up two Intune connectors and one VMWare connector on the same MyID system.

For information on configuring a credential profile with MDM restrictions, see section *3.11.4*, *Configuring credential profiles for MDM restrictions*.

For extra control and security, you can set up an external system for certain MDMs.

- For information on setting up an external system for Microsoft Intune, see section *3.11.1*, *Setting up an external system for Intune*.
- For information on setting up an external system for VMWare Workspace ONE, see section *3.11.2*, *Setting up an external system for Workspace ONE*.

If you want to integrate with any of the following MDM systems, contact your Intercede account manager to discuss your requirements:

- MobileIron.
- Citrix XenMobile.
- Centrify Identity Service.

2.4 Prerequisites

2.4.1 SMS gateway

You can configure the system to use any SMS gateway. To set up the system to communicate with your SMS gateway and allow MyID to send text messages to the users' mobile devices, you must have some knowledge of ASP and JavaScript.

See section *3.4.3*, *Configuring the SMS gateway for MyID Desktop* for details.

Alternatively, you can use email for notifications.

2.4.2 Communication with the MyID mobile web services

To allow your mobile device to obtain and work with mobile IDs, your device must be able to communicate with the URLs of the MyID mobile web services; for example:

```
https://myserver/MyIDProcessDriver/
```

```
https://myserver/MyIDDataSource/
```

Where `myserver` is the name of the server on which the MyID web services are installed.

Note: If you attempt to browse to these URLs from the mobile device, you will see an error due to the security set up on the web service folders; this does not mean that the connection has failed.

Your PC-based MyID clients must also be able to communicate with these web services. For example, QR codes are generated on the web services server by the MyIDDataSource web service, and embedded in the workflow.

2.4.3 SSL certificate

Before you start provisioning mobile devices, you must issue an SSL certificate from a trusted root CA.

Issuance will fail if the SSL certificate used on the MyID web server is untrusted by the mobile device. Intercede recommends that either an SSL certificate is issued by a trusted public root CA, or that devices have a trusted root CA for the issuing CA added to their Trusted Root stores.

2.4.4 IIS and web service users

The MyID IIS and MyID web service users must be members of the IIS_IUSRS Windows group; this is necessary for .NET 4 to operate correctly.

2.4.5 Logon mechanisms for mobile identities

The server update installation program turns on the **Password Logon** logon mechanism, which is essential for the correct operation of mobile identities. You must review your settings for logon mechanisms for the end user roles – you can switch off password logon for individual roles by using the **Assign Logon Mechanisms** feature in the **Edit Roles** workflow.

2.4.6 REST API for provisioning mobile credentials

MyID provides a REST API for provisioning mobile credentials (rest.provision). This API is currently used only for Mobile Identity Management integration projects where a mobile application written by third party vendors is using the Identity Agent Framework version 3.2 and above for iOS or Android. Where other mobile apps are used, for example apps built with earlier versions of the Identity Agent Framework, the existing APIs remain in place and continue to operate as before.

If you are using the Identity Agent Framework, make sure you select the **Provisioning API (rest.provision)** option on the Server Roles and Features screen in the MyID Installation Assistant.

3 Configuring the system

You must configure your system to allow you to request mobile IDs and collect them on the mobile device. This chapter contains instructions for configuring your MyID system for mobile identities, including:

- Setting the content signing certificate.
- Setting the MyID configuration options.
- Configuring access to workflows.
- Configuring notifications.
- Registering mobile devices.
- Setting up iOS OTA provisioning.
- Setting up the credential profile.
- Creating a custom mobile card format file.
- Setting up your MDM system.

3.1 Setting the content signing certificate

MyID must be able to sign the content for the mobile IDs before issuing them to mobile devices. Before MyID can use a certificate to sign the mobile IDs, the certificate must be available to the MyID COM user account.

1. On the MyID application server, log on using the account that you use to run the MyID components.
2. Request a certificate. You can issue a certificate from any certificate authority as long as it is available to CAPI or CNG.

   Notes:

   - Do not enable strong private key protection on the certificate, as this will prevent processing of the request by the MyID account.
   - By default, MyID uses SHA256 as the hashing algorithm when signing using this certificate. The certificate that you use for signing must therefore have been produced using a KSP or CSP that supports SHA256.
3. Once the certificate has been generated, install and save it as a `.cer` file (in binary format). You must save it in a location accessible to the MyID application, so save it to the `Components` folder within the MyID installation folder.

   Note: You may need administrative privileges to save files to this area.
4. Enter the filename of the certificate in the system registry.

   Note: You must log in as a user with sufficient privileges to edit the registry.

   If the keys and values do not already exist, you must create them.

   a. From the **Start** menu, click **Run** and type `regedit` in the dialog displayed. Click **OK**.

   b. Navigate to:

   `HKEY_LOCAL_MACHINE\SOFTWARE\Intercede\Edefice\ContentSigning`

c. Check that the value of the following string is set:

- `Active` – set to `WebService`

d. Set the value of the following string to the full path of the certificate on the application server:

- `WebService`

For example:

```
C:\Program Files\Intercede\MyID\Components\mycert.cer
```

An example `.reg` file for setting the content signing certificate might be:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Intercede\Edefice\ContentSigning]
"WebService"="C:\\Program Files\\Intercede\\MyID\\Components\\mycert.cer"
"Active"="WebService"
```

3.2 Setting the configuration options

3.2.1 Web service location

Within MyID, you must set the location of the MyID web service that allows a mobile device to collect a mobile ID.

To set the location of the web service:

1. From the **Configuration** category, select the **Operation Settings** workflow.
2. Click the **Certificates** tab.
3. Set the **Mobile Certificate Recovery Service URL** option to the location of the MyID Process Driver web service host.

   Note: This option is used for more operations than certificate recovery, despite the name.

   For example:

   `https://myserver`

   Replace `myserver` with the name of the server on which the web service is installed.

   You are recommended to use SSL on this connection. Make sure you specify the correct protocol: `http` or `https`.

   Note: The users' mobile devices must be able to access this URL. To be able to access the other MyID web services, all three MyID web services must be installed on the same server.
4. If you have installed MyID in a distributed network where the web server is in a separate domain, you may have to supply a separate URL for your MyID client workstations to retrieve a QR code for mobile issuance. In this case, set the **Web Server External Address** option to the URL of the MyID web services server that hosts the ProcessDriver web service. Make sure this URL is accessible to your MyID clients.

   In the majority of network configurations, you can leave this option blank.
5. Click **Save changes**.

3.2.2 Setting the authentication code complexity

To set up the single-use authentication code that is used to secure mobile IDs sent to the mobile device, you must use the **Certificate Recovery Password Complexity** configuration option to require numeric characters only.

To set the password complexity:

1. From the **Configuration** category, select the **Operation Settings** workflow.
2. Click the **Certificates** tab.
3. Set the **Certificate Recovery Password Complexity** option.

   The format is `xx-yyN`, which is made up of:

   - `xx` = minimum length.
   - `yy` = maximum length.

   The default is `04-08N` which means a code of 4 to 8 numbers.
4. Click **Save changes**.

3.2.3 Biometric authentication

MyID PIV systems support biometric authentication when updating and unlocking credentials. These features are not supported for mobile devices, therefore, on PIV systems, you must disable them before you can issue mobile identities successfully.

To set the biometric authentication options:

1. From the **Configuration** category, select the **Operation Settings** workflow.
2. Click the **Biometrics** tab.
3. Set the following options:
   - Set the **Verify fingerprints during card update** option in the **Operation Settings** workflow set to `No`.

     If this option is set to `Yes`, provisioning a mobile identity will fail with an error similar to:

     ```
     Your mobile device is not compatible with biometric authentication
     ```
   - Set the **Verify fingerprints during card unlock** option in the **Operation Settings** workflow set to `No`.

     If this option is set to `Yes`, unlocking a mobile identity will fail with an error similar to:

     ```
     Your mobile device is not compatible with biometric authentication
     ```
4. Click **Save changes**.

Note: When you set these options to `No`, you are removing the requirement to use biometrics when unlocking or updating smart cards as well as mobile identities.

3.2.4 Configuring the image location

To allow MyID to send badge images to the mobile device, you must make sure that the **Image Upload Server** configuration option (on the **Video** page of the **Operation Settings** workflow) is set to a value that can be resolved (to the name or IP address of the MyID web server) from the MyID Web Services server. For more information, see the *Configuring the image location* section in the ***Administration Guide***.

3.2.5 Maximum session count

If too many clients (whether mobile clients, or other clients such as MyID Desktop, the Self-Service App, or the Self-Service Kiosk) access the server at the same time for issuance or update processes, you may experience performance issues, and end users may experience errors.

If too many clients overload the server infrastructure, the errors may be generated from various points in the system (for example, from the database or the web server) and there may be a wide variety in the messages displayed; some error messages may be generic errors, with the details visible only in the MyID server logs.

If a user sees an "unexpected" error on the mobile device:

1. Review the MyID server logs for the time period involved. Check for timeout issues.
2. Review your infrastructure for high resource usage; for example, CPU, RAM, and so on.
3. Consider restricting the number of mobile sessions using the **Maximum session count** configuration option.

To set the maximum number of mobile sessions allowed.

1. From the **Configuration** category, select the **Operation Settings** workflow.
2. Click the **Identity Agent Policy** tab.
3. Set the following option:
   - **Maximum session count**

     This determines the number of concurrent sessions (whether from mobile clients or other clients such as MyID Desktop, the Self-Service App, or the Self-Service Kiosk) that are allowed by the server while still allowing mobile issuance and update operations.

     Values:

     `0` – Do not allow mobile issuances or updates.

     `-1` – No limits.

     Any other number determines the number of client sessions allowed. If this number is exceeded, the server returns HTTP 503 – service unavailable – to all mobile clients. This will also be recorded in the local event log.

     Only mobile clients are prevented from connecting.

     You are recommended to tailor this value to your hardware: too high a value, and your server may experience performance issues; too low and your server will be under-used.

     As server deployments differ in computing capability, functionality usage, and data load, it is impossible to recommend precise values. You are recommended to try various values on a test system that mirrors the resources and data load of your production system.
4. Click **Save changes**.

3.2.6 Setting up support for historic certificates

You can set up MyID to provide historic encryption certificates for mobile identities. This feature allows users to decrypt their old email messages on their mobile device. The historic encryption certificates are delivered to the mobile device when the mobile identity is issued.

To configure MyID to provide historic certificates, you must use the certificate options in the credential profile. See the *Selecting certificates* section in the ***Administration Guide*** for details of the **Issue new**, **Use existing**, and **Historic Only** options.

3.2.7 Configuring MyID for 3DES encryption

New installations of MyID are configured for AES encryption for low level processes; for example, secure communication between MyID clients and servers.

If you are issuing mobile identities using the following versions of the mobile apps (or earlier):

- iOS
  - MyID Identity Agent v4.2
  - MyID Authenticator v1.5
- Android
  - MyID Identity Agent v4.1.2813
  - MyID Authenticator v1.4.171

you must configure MyID to use 3DES instead; a later update for these mobile apps will provide support for AES.

To configure MyID to use 3DES:

1. From the **Configuration** category, select the **Security Settings** workflow.
2. Click the **Server** tab.
3. Set the following option:
   - **Envelope Transport Key Algorithm** – make sure this option is set to `3DES`.
4. Click **Save changes**.

Note: Apps developed using Identity Agent Framework version 3.9 or later, which use the rest.provision mobile provisioning API, can support AES; for apps developed using earlier versions, set the option to `3DES`. The MyID Wallet app for mobile identity documents supports AES.

3.3 Granting access to the workflows

The system makes use of the following workflows:

- **Cancel Credential** – used within MyID to cancel a mobile ID and revoke its certificates.
- **Enable / Disable ID** – used within MyID to enable or disable a mobile ID, and suspend or enable its certificates.
- **Request ID** – used within MyID for operator-guided requests for mobile IDs to be installed on a mobile device.
- **Request My ID** – used within MyID for self-service requests for mobile IDs to be installed on a mobile device.

- **Request Replacement ID** – used within MyID to request a replacement for a missing or damaged mobile ID.
- **Unlock Credential** – used within MyID to retrieve an unlock code for an issued mobile ID.
- **Collect My Updates** – used by the Identity Agent app to obtain a mobile ID.
- **Issue Device** – used by the Identity Agent app to obtain a mobile ID.

  Note: The **Collect My Updates** and **Issue Device** workflows are not used within MyID; they are used to control access from a mobile device to the features of the web service. You must make sure that these workflows are available to the recipient through a role that is set to allow **Password** as a **Logon Mechanism**; for example, you can use the **PasswordUser** role.

Use the **Edit Roles** workflow to grant access for these workflows to the roles you want to be able to access them.

3.3.1 Roles

You must add the **Collect My Updates** workflow to the Server Credentials role if the user does not already have access to this workflow through one of their other roles.

Note: You can use the Server Credentials role to control access to the collection service; allocate this role to the users who you want to be able to collect mobile IDs.

Alternatively, you can add the **Collect My Updates** workflows to an existing role to allow users in that role to collect mobile IDs.

3.3.2 Scope

When a mobile device user, for example a guard, requests the details for another mobile device user, the guard must have the correct scope within MyID to view the details of the other user; for example, the user must be in the same group as the guard if the guard has Department scope.

3.4 Configuring SMS and email notifications

You can choose whether to allow SMS, email, or both types of notification when sending provisioning messages to mobile devices.

MyID sends two notifications:

- A link to the collection URL.

  MyID sends this notification as an email.
- An authentication code.

  MyID sends this one time password either as a separate email, or as an SMS.

  Note: The complexity of the authentication codes is determined by the **Certificate Recovery Password Complexity** configuration option (on the **Certificates** page of the **Operation Settings** workflow). See section *3.2.2, Setting the authentication code complexity* for details.

The two components of the notification (the collection URL and the authentication code) are sent separately for security, and you are recommended to configure MyID to send the collection URL as an email and the authentication code as a SMS for additional security.

3.4.1 Configuring SMS and email notifications for the MyID Operator Client

You control the way MyID sends notifications for the issuance of mobile identities through the MyID Operator Client by setting the notification scheme in the credential profile; see section *3.8.1*, *Creating the Identity Agent credential profile*.

You must enable the notification methods using configuration options.

To enable SMS and email notifications:

1. From the **Configuration** category, select the **Operation Settings** workflow.
2. On the **General** tab, set the following options:
   - **SMS email notifications** – set to `Yes` to allow authentication codes to be sent through SMS.

     If you do *not* set this option to `Yes`, you must configure the credential profile to send the authentication code as an email, or display the authentication code on screen when you request the mobile device.
   - **SMS gateway URL for notifications** – set to the URL of your SMS gateway.

     By default, SMS messages are sent to through an email to SMS gateway, in the format `<cellnumber>@<gateway>`, where:
     - `<cellnumber>` – the cell phone number from the person's record.
     - `<gateway>` – the URL from the **SMS gateway URL for notifications** option.

     For example: `00447700900123@smsgateway.com`

     If this is not suitable, you can customize the `sp_CustomPrepareSMS` stored procedure in the MyID database.
3. On the **Notifications** tab, set the following option:
   - **Send Email Notifications** – set to `Yes` to allow notifications to be sent through email.

     You must configure an SMTP server in the **External Systems** workflow; see the *Setting up email* section in the ***Advanced Configuration Guide***.
4. On the **Issuance Processes** tab, set the following options:
   - **App Download URL – ANDROID** – the URL where the Android version of the Identity Agent app is available for download.
   - **App Download URL – iOS** – the URL where the iOS version of the Identity Agent app is available for download.

   If you click on a provisioning URL on a mobile device, but do not have the Identity Agent app installed, these links are displayed to allow you to download the app and try again. See the *Issuance Processes page (Operation Settings)* section of the ***Administration Guide*** for further details of these options.
5. Click **Save changes**.

3.4.2 Configuring SMS and email notifications for MyID Desktop

You control the way MyID sends notifications for the issuance of mobile identities through MyID Desktop by setting configuration options.

To allow provisioning messages:

1. From the **Configuration** category, select the **Operation Settings** workflow.
2. On the **Devices** tab, set the following options:
   - **Mobile Provision Via Email** – set this option to allow the notifications of mobile IDs to be sent to the user's email address.
   - **Mobile Provision Via SMS** – set this option to allow the notifications of mobile IDs to be sent to the user's mobile phone number.

   Note: You can select one or both of these options. If you select both options, you can select which method to use when you request the mobile identity.
3. On the **Notifications** tab, set the following options:
   - **Send Mobile OTP via SMS** – set this option to allow the operator to send the OTP authentication code directly to the mobile device.

     Note: If you set **Send Mobile OTP via SMS** to Yes, as a security feature, the OTP is sent as an SMS while the notification message must be sent using email and *not* SMS; make sure you select the **Mobile Provision Via Email** option on the **Devices** tab.
   - **Mail Format** – make sure this option is set to **HTML**.
4. Click **Save changes**.

3.4.3 Configuring the SMS gateway for MyID Desktop

You can configure the system to use any SMS gateway. You must customize the following file:

`customSMS.asp`

Versions of this file are installed to the MyID web server in the following locations:

- `Web\<edition>\untranslated\res\custom\js\`
- `Web\<edition>\en\res\custom\js\`
- `Web\<edition>\us\res\custom\js\`

Where `<edition>` is `WebPIV` for PIV, and `WebENT` for non-PIV editions of MyID.

You must make the same changes in each version of the file. If you have created any custom translations of the MyID website, you must also make the same change in the custom versions.

The sample file installed with the system is set up to use the SMS gateway provided by `www.2sms.com` – if you are using this service, edit the `username` line to include your 2sms account, and the `password` line to include your 2sms password.

If you are using any other system, you must customize the ASP file to conform to the calling requirements of your own SMS gateway.

This ASP file implements the following function:

```
customSendSMS(message, mobileNumber, userRS)
```

where:

- `message` – the body of the SMS text message to be sent to the mobile device.
- `mobileNumber` – the cell/mobile phone number from the user's MyID record.
- `userRS` – reserved for future use.

The function returns the response from the SMS gateway.

You can implement your system in any way. You are required only to send the body contained in `message` to the phone number in `mobileNumber`, and `return` the response from the gateway.

Note: You must keep a backup of this file once you have customized it.

3.4.4 Configuring SMS and email certificate renewal notifications

This section is relevant for certificate renewal notifications whether you issued the mobile device through the MyID Operator Client on through MyID Desktop.

You can decide whether to send renewal messages through email, through SMS, or through both email and SMS.

To allow MyID to send SMS messages, set the **SMS email notifications** on the **General** tab of the **Operation Settings** workflow to `Yes`.

By default, SMS messages are sent to an Email to SMS gateway, in the format `<cellnumber>@<gateway>`, where:

- `<cellnumber>` – the cell phone number from the user's record.
- `<gateway>` – the URL from the **SMS gateway URL for notifications** option on the **General** tab of the **Operation Settings** workflow.

For example: `00447700900123@smsgateway.com`

If this is not suitable, you can customize the `sp_CustomPrepareSMS` stored procedure in the MyID database.

You can use different content for email and SMS certificate renewal messages, and different content for different kinds of device – mobile or card, for example. Six additional renewal messages are provided – three messages for SMS to mobile devices, and three messages for email to mobile devices. Use the **Email Templates** workflow to edit the content of these messages.

Note: If you have upgraded your MyID system, MyID does not update your stored procedure to prevent the overwriting of your changes. The base stored procedure has been updated, and you must edit the `sp_CustomPrepareSMS` stored procedure to allow it to work. If your stored procedure contains the following:

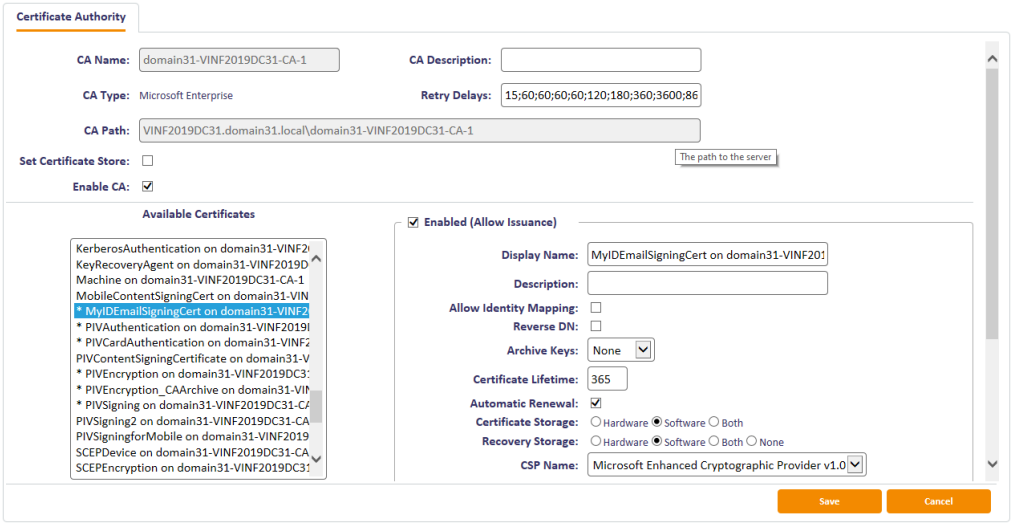
```
-- Output an XML string
@xml NVARCHAR(MAX) OUT
```

change it to:

```
-- Output an XML string
@xml NVARCHAR(4000) OUT
```

3.5 Configuring the certificate authority

You must configure the certificate template to set the options for storing the certificate on your mobile device.



For the **Certificate Storage** and **Recovery Storage** options, select the following:

- **Software** – the certificate is stored on the mobile device local key store. You cannot select **Hardware** for mobile devices.

3.6 Registering mobile devices

Note: If you are working with a mobile device management system, it is recommended that you use the feature to check validity of the device against the MDM instead of the **Mobile Device Restrictions** feature; see section *3.11*, *Setting up your MDM system*.

You can use the **Mobile Device Restrictions** option to set up your credential profiles to issue mobile identities only to those mobile devices that have been registered with MyID.

You can obtain the serial number from the app on the mobile device.

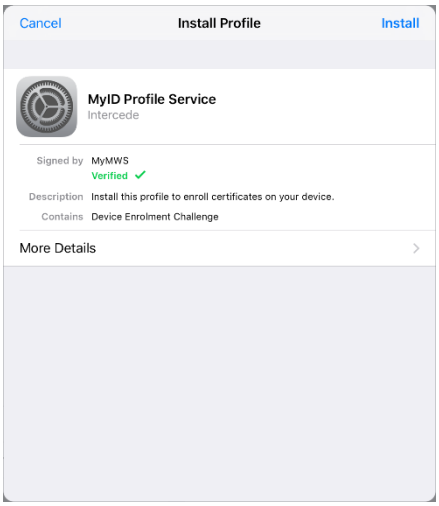
Once you have the serial number, you can use the `AddDevice` method of the Device Management API to register the device with MyID.

See the ***Device Management API*** guide for details.

3.7 Setting up iOS OTA provisioning

Note: This feature requires a mobile app that has been configured to use iOS OTA provisioning. For further information, contact Intercede quoting reference SUP-392.

You can configure MyID to enroll a certificate on your iOS device using Over the Air (OTA) provisioning. The update appears on the device as a profile to be installed when you are issuing a mobile identity.



This feature requires the following additional web service modules to be installed and configured on your MyID server:

- **SCEP API** – Simple Certificate Enrollment Protocol (SCEP) device identities.

  You must follow the instructions in the *Managing devices* section of the ***Administration Guide*** for setting up your SCEP server before setting up iOS OTA provisioning. You do not need to request or collect any SCEP device identities.
- **Mobile iOS OTA** – OTA (Over The Air) provisioning of certificates to iOS.

3.7.1 Setting up the application pool for iOS OTA

After you have installed MyID with the **Mobile iOS OTA** module, you must update the Load User Profile option for the application pool used by the service:

1. In Internet Information Services (IIS) Manager, select **View Application Pools**.
2. Right-click the **MyIDiOSOTA_Pool**, then from the pop-up menu select **Advanced Settings**.
3. Set the **Load User Profile** option to **True**.
4. Click **OK**.
5. Right-click the **MyIDiOSOTA_Pool**, then from the pop-up menu select **Recycle**.

3.7.2 Setting up a signing certificate for iOS OTA

The web services have to be able to sign the information being sent to the phone and so require a signing certificate to be issued. No specific attributes are required: any certificate that can be used for signing data is suitable.

1. On the MyID web server, log on using the MyID web service user.
2. Request a certificate that will be placed in the CAPI store. You can issue a certificate from any certificate authority as long as it is available to CAPI.

   Note: Do not enable strong private key protection on the certificate, as this will prevent processing of the request by the MyID account.
3. Once the certificate has been generated, copy its thumbprint data into the `SigningCertThumbprint` value in the `Web.config` file for the `MyIDiOSOTA` web service.

   By default, this is in the following folder:

   `C:\Program Files\Intercede\MyID\SSP\MyIDiOSOTA\`
4. Copy the same certificate thumbprint data into the `iOSOTA:SigningCertThumbprint` section of the `appsettings.Production.json` file of the `rest.provision` web service:
   a. As an administrator, open the `appsettings.Production.json` file in a text editor.

      By default, this is:

      `C:\Program Files\Intercede\MyID\rest.provision\appsettings.Production.json`

      This file is the override configuration file for the `appsettings.json` file for the web service. If this file does not already exist, you must create it in the same folder as the `appsettings.json` file.
   b. Edit the file to include the following:

      ```
      "iOSOTA": {
          "SigningCertThumbprint": "<certificate thumbprint>"
          }
      ```

      where `<certificate thumbprint>` is the thumbprint of the signing certificate.
   c. Save the `appsettings.Production.json` file.
   d. Recycle the web service app pool:
      a. On the MyID web server, in Internet Information Services (IIS) Manager, select **Application Pools**.
      b. Right-click the **myid.rest.provision.pool** application pool, then from the pop-up menu click **Recycle**.

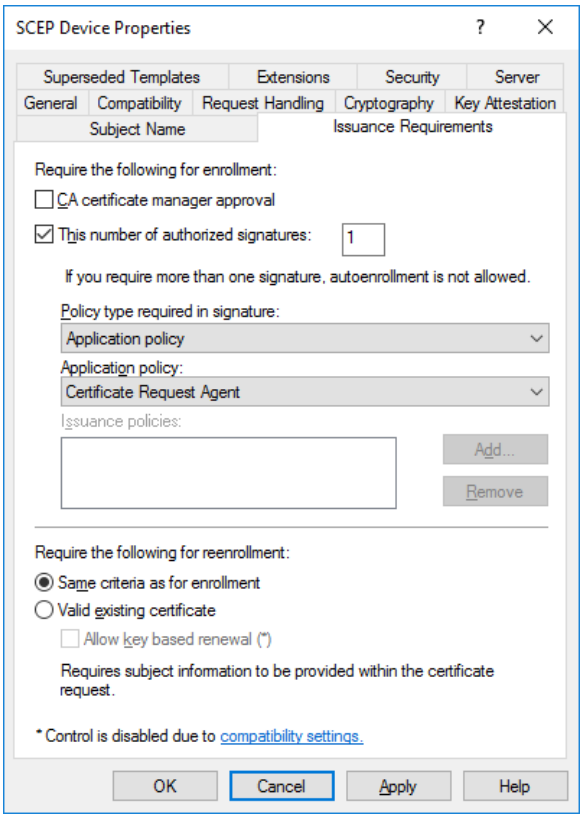
      This ensures that the web service has picked up the changes to the configuration file.

3.7.3 iOS OTA certificate requirements

This section contains some specific issuance requirements for the certificate template for a Microsoft Certificate Authority for iOS OTA issuance.

- The certificate you use for iOS OTA issuance must have the **CA certificate manager approval** option deselected.

- Set the **Policy type required in signature** drop-down list to **Application policy**.
- Set the **Application policy** drop-down list to **Certificate Request Agent**.



If you see a message in the "Failed requests" section of the CA similar to:

```
One or more signatures did not include the required application or issuance
policies. The request is missing one or more required valid signatures.
```

this means that the **Application policy** option is set to **Any Purpose** instead of **Certificate Request Agent**.

3.7.4 Configuring MyID for OTA provisioning

To configure MyID for OTA provisioning:

1. Create an **Identity Agent** credential profile that uses the following:
   - A **Card Format** of **Mobile**.
   - One or more certificates that uses the **System Store** container.

   See section *3.8*, *Setting up the Identity Agent credential profiles* for details.
2. Create a **Device Identity (Only)** credential profile that uses the following:
   - **Require Challenge** option selected.

   See the *Setting up a credential profile to use to issue device identities* section in the ***Administration Guide*** for details of completing the credential profile.

   See also section *3.7.3*, *iOS OTA certificate requirements* for details of the requirements for the device certificate.
3. From the **Configuration** category, select **Operation Settings**.

4. Click the **Certificates** tab.
5. Set the following options:
   - **iOS OTA Credential Profile** – set this option to the name of the Device Identity credential profile.
   - **iOS OTA Organization** – set this option to the name of your organization. This appears on the OTA provisioning message on the mobile device.
   - **iOS OTA Display Name** – set this option to a name for the OTA update. This appears on the OTA provisioning message on the mobile device.
   - **iOS OTA Description** – set this option to the description for the OTA update. This appears on the OTA provisioning message on the mobile device.
6. If required, you can customize the transform on the web services server that is used to display the intermediate web page that presents a link to the CA root certificate and the Enroll page used to provision the certificates.

   See the *iOS OTA web page* section in the ***Web Service Architecture*** guide for details.
7. Click **Save changes**.

3.8 Setting up the Identity Agent credential profiles

You must create at least one new credential profile for issuing mobile IDs to mobile devices.

The credential profile contains the certificates that you want to issue to mobile users. You may create as many of these credential profiles as you need.

Note: The term Identity Agent refers to mobile apps that have been built using the MyID Identity Agent Framework; this is an SDK supplied to mobile app developers that enables integration with MyID CMS. For example, this framework is built into the mobile apps provided with supported MDMs.

3.8.1 Creating the Identity Agent credential profile

To create a credential profile for issuing mobile identities:

1. From the **Configuration** category, select **Credential profiles**.
2. Click **New**.
3. Type a **Name** for the credential profile.
4. In **Card Encoding**, select **Identity Agent**.

5. In **Issuance Settings**, set the following options:

- **Require Fingerprints at Issuance** – set to **Never required**.
- **Require Facial Biometrics** – set to **Never required**.
- **Notification Scheme** – select one of the following:
  - **Default** – MyID sends the collection URL as an email, the authentication code as a separate email, and the authentication code as an SMS.
  - **None** – MyID does not send any notifications. You must use the **Request Mobile (View Auth Code)** option in the MyID Operator Client to display the collection URL and authentication code on screen.
  - **Mobile Only – Auth Code Via Email** – MyID sends the collection URL as an email, and the authentication code as a separate email.
  - **Mobile Only – Auth Code Via SMS** – MyID sends the collection URL as an email, and the authentication code as an SMS.

  Note: Notification schemes are relevant only for mobile devices requested through the MyID Operator Client or the MyID Core API. They do not affect the notifications sent when you request mobile devices through MyID Desktop or the Credential Web Service API.

  See section *3.4.1*, *Configuring SMS and email notifications for the MyID Operator Client*.

  The complexity of the authentication codes is determined by the **Certificate Recovery Password Complexity** configuration option (on the **Certificates** page of the **Operation Settings** workflow). See section *3.2.2*, *Setting the authentication code complexity* for details.

6. In the **Mobile Device Restrictions** drop-down list, select one of the following:
   - **Any** – The mobile identity can be loaded onto any mobile.
   - **Known Mobiles** – The mobile identity can be loaded onto any mobile that has already been registered with MyID. See section *3.6*, *Registering mobile devices* for details.
   - **My Mobiles Only** – The mobile identity can be loaded only onto mobiles associated with the user's account.
7. In **Device Profiles**, set the following from the **Card Format** drop-down list:
   - To issue certificates to the iOS or Android System Store, select the generic **Mobile** card format with the **System Store** container. MyID will detect the type of mobile device when the mobile identity is issued and issue certificates to the appropriate system store, iOS or Android.
   - For Microsoft Intune and VMware Workspace ONE enabled mobile devices, make sure that **None** is selected.
   - For all other mobile devices, make sure that **None** is selected.

   Note: If you attempt to issue a mobile device using a credential profile that includes support for certificates stored in the iOS System Store, but the mobile device does not support these certificate stores, the issuance will succeed; however, any certificates specified by the credential profile to be installed to containers that the mobile device does not support will be ignored.

For example, if your credential profile contains an iOS Signing certificate, an iOS Encryption certificate, and a certificate with no container specified, an iOS-enabled mobile device will receive all three certificates, while a mobile device that is not iOS-enabled will receive only the certificate with no container specified.

8. Click **Next**.
9. Select the certificates you want to make available.
   - If you are issuing multiple certificates to the iOS System Store, make sure that all of the certificates have the same expiry date; if the certificates do not have the same expiry date, you will not be able to renew them, as all of the certificates are added to the same iOS security profile in the system keystore.

     You can also select the **System Store** for one or more certificates. See section *3.7*, *Setting up iOS OTA provisioning* for details of provisioning certificates to the iOS System Store.
   - For credential profiles that use the Mobile data model, you can select the **System Store** for one or more archive certificates.
   - For Microsoft Intune and VMware Workspace ONE enabled mobile devices, do not select any containers.
   - For all other types of credential profiles, do not select any containers.

   All of the certificates you select here will be issued to your mobile device.

   You can select the archived and historic certificate options on this screen. See the *Selecting certificates* section in the ***Administration Guide*** for details of the **Issue new**, **Use existing**, and **Historic Only** options.

   If you want to distribute certificates that were not issued through MyID, you can import a PFX file then select the **Unmanaged** certificate option to specify it for distribution to the mobile device. See the *Import and distribute certificates to devices* section in the ***Administration Guide*** for details of setting up your credential profile and using the **Upload PFX Certificates** workflow.
10. Click **Next** and proceed to the Select Roles screen.
11. Select the roles you want to be able to issue and receive this credential profile.
    - The **Can Receive** option determines which roles can receive credentials issued using this credential profile.
    - The **Can Request** option determines which roles can request credentials using this credential profile; for example, using **Request ID** for operator requests or **Request My ID** for self-service requests.
    - The **Can Validate** option determines which roles can validate requests for credentials using this credential profile using the **Validate Request** workflow.
    - The **Can Collect** option determines which roles can collect credentials using this credential profile; any user who is to receive a mobile identity must have both the **Can Receive** and the **Can Collect** options.
    - The **Can Unlock** option determines which roles can unlock mobile identities using the **Unlock Credential** workflow.

Note: Not all options may be available, depending on your system configuration. See the *Working with credential profiles* section in the ***Administration Guide*** for details.

Note: Any role you want to receive mobile identities must have the **Issue Device** option selected in the **Cards** category within the **Edit Roles** workflow.

12. Click **Next**.
13. Select the card layouts you want to make available to the mobile device.

    Badges based on these layouts will be transferred to the mobile device as part of the mobile ID. Note, however, that the reverse sides of the selected layouts (the `_back` layouts) will not be available on the mobile device.

    Note: Card layouts are optional, and are created only when using the Intercede key store and certificates are selected in the credential profile. Badge layouts are displayed only with some third-party apps; check with your app vendor when planning deployment.
14. Click **Next**.
15. Type your **Comments** and complete the workflow.

3.8.2 Configuring authentication types for Identity Agent credential profiles

In the **Credential Profiles** workflow, when you select a **Card Encoding** type of **Identity Agent**, the **Authentication Types** section becomes available; this allows you to specify the additional types of authentication that are available for the end user to use to access the Intercede keystore. If you do not select any additional authentication types, the user will be able access the Intercede keystore only using their PIN.

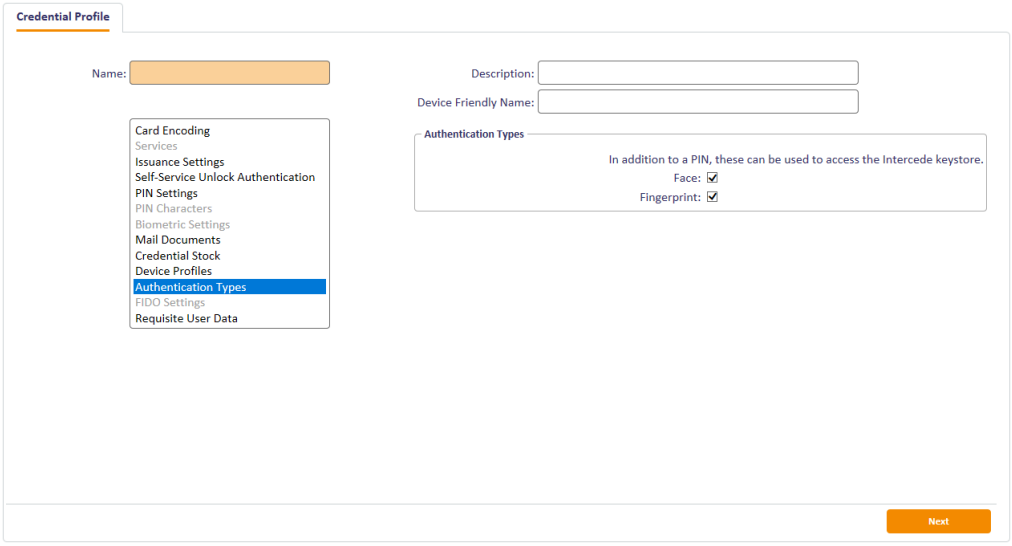
Authentication types are not used when using a mobile device management app. For support with third-party apps, check with the app vendor when planning deployment.

Note: A PIN is mandatory, as it provides a fallback option to the user in the event that they are unable to provide any of the other authentication types.

To set the authentication types:

1. From the **Configuration** category, select **Credential Profiles**.
2. Click **New**.
3. From the **Card Encoding** list, select **Identity Agent**.

4. Click **Authentication Types**.



5. Select the following:
   - **Face** – if the mobile device supports it, the user can use facial biometrics to access the Intercede keystore. Available on iOS devices that support facial ID only.
   - **Fingerprint** – if the mobile device supports it, the user can use fingerprint biometrics to access the Intercede keystore.
6. Click **Next** and complete the workflow.

3.9 Creating a custom mobile card format

MyID provides a selection of standard card formats to be used with mobile identities. If you need to customize your card model to have multiple containers, for example, you can create a custom card format file on the MyID application server.

Use of custom data models may require additional changes to the mobile apps that are used; check with the app vendor when planning deployment.

1. On the application server, create an XML file in the following folder:

   `C:\Program Files\Intercede\MyID\Components\CardServer\CardFormats\`

2. Copy the following into the XML file:

```
<CardDataModel>
  <identity>
    <name>Card data model name here</name>
    <description>Card data model description here</description>
  </identity>
  <Container>
    <Name>Container name here</Name>
    <ID>Container ID name here</ID>
    <Certificate/>
    <Explicit/>
    <Unique>1</Unique>
    <PreserveContainerForHistoric/>
  </Container>
```

```
</CardDataModel>
```

3. Provide the appropriate values in the following nodes:
   - `CardDataModel\identity\name`
   - `CardDataModel\identity\description`
   - `CardDataModel\Container\Name`
   - `CardDataModel\Container\ID`
4. If necessary, add more `CardDataModel\Container` nodes for additional containers.
5. Save the XML file.

   Note: Save the file as ANSI. If you save the file as UTF-8, you will experience errors when MyID tries to access the file.
6. Restart your MyID clients.

Note: You must back up your custom data model files before carrying out an upgrade. See the *Upgrading systems with customized card data models* section in the ***Installation and Configuration Guide***.

3.10 Setting up a custom PKCS #10 request

For requests made using the REST API for mobile credentials (rest.provision) you can customize the PKCS #10 certificate signing request where the subject name is provided in the request; you can create a DN from the information stored in the `vPeopleUserAccounts` view in the MyID database for the person for whom the request was made.

To configure the web service, you must edit the `appsettings.Production.json` file of the rest.provision web service:

1. As an administrator, open the `appsettings.Production.json` file in a text editor.

   By default, this is:

   `C:\Program Files\Intercede\MyID\rest.provision\appsettings.Production.json`

   This file is the override configuration file for the `appsettings.json` file for the web service. If this file does not already exist, you must create it in the same folder as the `appsettings.json` file.
2. In the `MyID` section, edit the `dnProcessor` section.

   If this section does not exist, you must add it.

   The format is:

```
{
   "MyID":{
      "dnProcessor":{
         "default":"default DN",
         "policy":[
            {
               "name":"Policy 1",
```

```
            "dn":"DN for policy 1"
         },
         {
            "name":"Policy 2",
            "dn":"DN for policy 2"
         },
         {
            "name":"Policy 3",
            "dn":"DN for policy 3"
         }
      ]
   }
}
```

Provide the following:

- `default` – provide the default DN to be used if the policy name does not match one of the specific policies in the configuration file.
- `policy` – provide a block of JSON that includes:
  - `name` – the friendly name of the certificate policy as listed in the `FriendlyName` column of the `CertPolicies` table in the MyID database.

    See section *3.10.1*, *Policy names*
  - `dn` – the DN to be used for this policy.

  You can include as many policy blocks as required, one for each policy.

The format of the DN requests may include:

- Simple text.

  For example:

  ```
  cn=Static DN
  ```

- Substitutions from the `vPeopleUserAccounts` view in the database.

  See section *3.10.2*, *Available fields for substitution* for details of which fields you can use.

  Enclose the codes for these fields in double square brackets; for example:

  ```
  cn=[[People.UserPrincipalName]], o=Users, c=uk
  ```

  The `People` element determines that the field is in the `vPeopleUserAccounts` view. This example takes the `UserPrincipalName` field for the user and inserts it into the `cn` component of the DN.

  If there are any special characters in the specified field, these are escaped with a slash `\` character.

- Raw substitutions from the `vPeopleUserAccounts` view in the database, without escaping.

  If the field in the database contains a full DN, you do not want to escape the special characters. To include the content of the field without substitutions, specify the field using triple square brackets; for example:

  ```
  [[[People.Xu55]]]
  ```

3. Save the `appsettings.Production.json` file.
4. Recycle the web service app pool:
   a. On the MyID web server, in Internet Information Services (IIS) Manager, select **Application Pools**.
   b. Right-click the **myid.rest.provision.pool** application pool, then from the pop-up menu click **Recycle**.

   This ensures that the web service has picked up the changes to the configuration file.

For example:

```
{
   "MyID":{
      "dnProcessor":{
         "default":"cn=Static DN",
         "policy":[
            {
               "name":"Smartcard Logon",
               "dn":"cn=[[People.UserPrincipalName]], o=Users, c=uk"
            },
            {
               "name":"Encryption",
               "dn":"[[[People.Xu55]]]"
            }
         ]
      }
   }
}
```

3.10.1 Policy names

The policy names you provide must match the `FriendlyName` column in the `CertPolicies` table in the database.

To obtain a list of these names, you can run the following SQL against the MyID database:

```
select FriendlyName from CertPolicies;
```

3.10.2 Available fields for substitution

The format to use when specifying the fields is:

```
[[<Entity>.<Field>]]
```

where:

- `<Entity>` is currently `People`.
- `<Field>` is the name of a field in the `vPeopleUserAccounts` table.

For example:

```
[[People.FullName]]
```

To obtain a list of the fields you can use, you can run the following SQL against the MyID database:

```
select COLUMN_NAME from INFORMATION_SCHEMA.COLUMNS where TABLE_NAME =
'vPeopleUserAccounts';
```

3.11 Setting up your MDM system

MyID allows you to work with Mobile Device Management (MDM) systems. See section *2.3*, *Supported Mobile Device Management integration* for details of which systems are supported.

Note: You can set up more than one MDM connector for each instance of MyID; you must specify which MDM connector to use in the credential profile. You can set up multiple MDM connectors of the same type, or multiple MDM connectors of mixed types; for example, you could set up two Intune connectors and one VMWare connector on the same MyID system.

3.11.1 Setting up an external system for Intune

For full details of how to use derived credentials in Microsoft Intune, see the Microsoft documentation:

docs.microsoft.com/en-us/intune/protect/derived-credentials

You must register MyID as an application in Entra:

- Take a note of the **Directory (tenant) ID** for your Entra system.
- Take a note of the **Application (client) ID** for your application.
- Within the **Certificates & secrets** section, add a new client secret or client certificate.
- When you are setting up the application, you do not need to set up a redirect URI, as MyID uses an OAuth2 client credential grant to authenticate with Entra.
- You can configure the application as Single tenant or Multitenant.
- The registered application must have the following Microsoft Graph permission:

  `DeviceManagementManagedDevices.Read.All`

See the following Microsoft guide for details of creating an application registration:

docs.microsoft.com/en-us/azure/active-directory/develop/quickstart-register-app

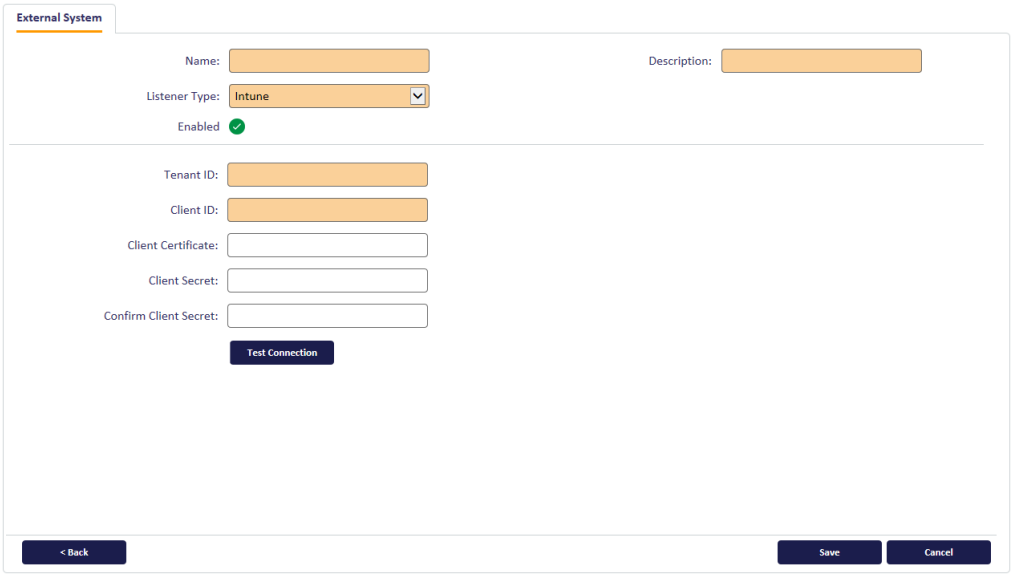
Once you have configured your application in Entra, you can configure MyID to communicate with the Entra server.

To set up an external system that allows you to connect to Microsoft Intune:

1. From the **Configuration** category, select **External Systems**.

   You can also launch this workflow from the **Connections and Notifications** section of the **More** category in the MyID Operator Client. See the *Using Connections and Notifications workflows* section in the ***MyID Operator Client*** guide for details.
2. Click **New**.
3. From the **Listener Type** drop-down list, select **Intune**.

   The Microsoft Intune options appear:



4. Set the following options:
   - **Name** – Type a name for the external system.
   - **Description** – Type a description for the external system.
   - **Enabled** – Set this option to enable the Intune connector. You can disable the connector and MyID will not attempt to communicate with the Intune system.
   - **Tenant ID** – Type the **Directory (tenant) ID** of the subscription in which you are running your Intune environment.
   - **Client ID** – Type the **Application (client) ID** of the registered application entry connected to your Entra subscription.
   - **Client Certificate** – You can authenticate using either a client certificate or a client secret.

     If you are authenticating using a client certificate, you must generate or import a certificate on the application server within the Personal certificate store of the MyID COM+ user account. Save the public certificate to a location on the application server, and register it as an authentication certificate against the API in the Azure portal.

     Type the path to the certificate file on the application server in the **Client Certificate** field.
   - **Client Secret** – If you are authenticating using a client secret, type the secret you have created in the Entra portal, then type it again in the **Confirm Client Secret** field.
5. Click **Test Connection** to check that you have entered the details correctly.
6. Click **Save**.

7. Restart the Edefice_BOL component to ensure that MyID is working with the updated configuration.

   To restart the component:

   a. On the MyID application server, open Windows Component Services.

   b. Expand **Component Services > Computers > My Computer > COM+ Applications**.

   c. Right-click **Edefice_BOL**, then from the pop-up menu click **Shut down**.

      The component will restart automatically the next time it is needed.

3.11.2 Setting up an external system for Workspace ONE

To set up an external system that allows you to connect to VMWare Workspace ONE:

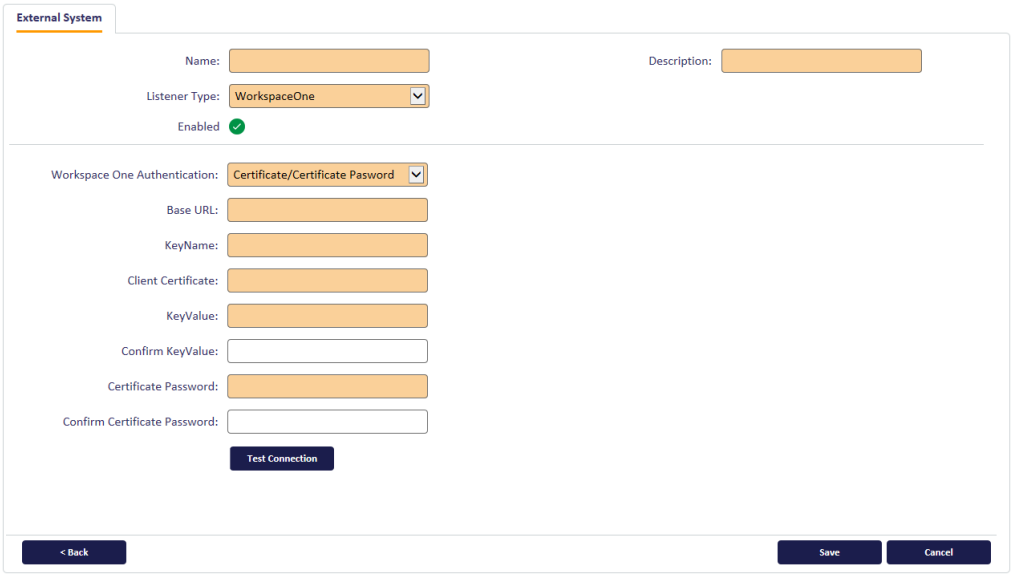
1. From the **Configuration** category, select **External Systems**.

   You can also launch this workflow from the **Connections and Notifications** section of the **More** category in the MyID Operator Client. See the *Using Connections and Notifications workflows* section in the ***MyID Operator Client*** guide for details.

2. Click **New**.

3. From the **Listener Type** drop-down list, select **WorkspaceOne**.

   The Workspace ONE options appear:



4. From the **Workspace One Authentication** drop-down list, select one of the following:

   - **Certificate/Certificate Password** – Select this option if you are using Certificate authentication for the Workspace ONE system. The **Client Certificate** and **Certificate Password** fields are now available.
   - **User/password authentication** – Select this option if you are using Basic authentication for the Workspace ONE system. The **Username** and **Password** fields are now available.

5. Set the following options:

   - **Name** – Type a name for the external system.
   - **Description** – Type a description for the external system.
   - **Enabled** – Set this option to enable the Workspace ONE connector. You can disable the connector and MyID will not attempt to communicate with the Workspace ONE system.
   - **Base URL** – Type the URL of the Workspace ONE installation. Use the following form:

     `https://*****.awmdm.com`

- **KeyName** – Type the name of the API key configured for the Workspace ONE REST API; this is usually:

  `aw-tenant-code`

- **KeyValue** – Type the value of the API key, then type it again in the **Confirm KeyValue** field.
- **Username** – If you selected **User/password authentication** from the **Workspace One Authentication** drop-down list, type the name of the configured account.
- **Password** – Type the password for the configured account, then type it again in the **Confirm Password** field.
- **Client Certificate** – If you selected **Certificate/Certificate Password** from the **Workspace One Authentication** drop-down list, provide the path and filename for the PFX file containing the client certificate generated for the admin account and exported from the Workspace ONE system. This must be present on the application server; for example:

  `C:\VMWare\Workspace ONE API\CN=23764_td.username.p12`

- **Certificate Password** – Type the certificate password, then type it again in the **Confirm Certificate Password** field.

6. Click **Test Connection** to check that you have entered the details correctly.
7. Click **Save**.
8. Restart the Edefice_BOL component to ensure that MyID is working with the updated configuration.

   To restart the component:

   a. On the MyID application server, open Windows Component Services.

   b. Expand **Component Services > Computers > My Computer > COM+ Applications**.

   c. Right-click **Edefice_BOL**, then from the pop-up menu click **Shut down**.

      The component will restart automatically the next time it is needed.

3.11.2.1 Configuring Workspace ONE for the PIV-D application

You must configure Workspace ONE for the PIV-D application.

You must configure the `ConnectorDeviceIdentifier` configuration key in the Application Configuration screen as follows:

Configuration Key	Value Type	Configuration Value
`ConnectorDeviceIdentifier`	String	`{DeviceUid}`

See your VMWare documentation for more information.

3.11.3 Setting up an external system for BlackBerry UEM

To set up an external system that allows you to connect to BlackBerry UEM:

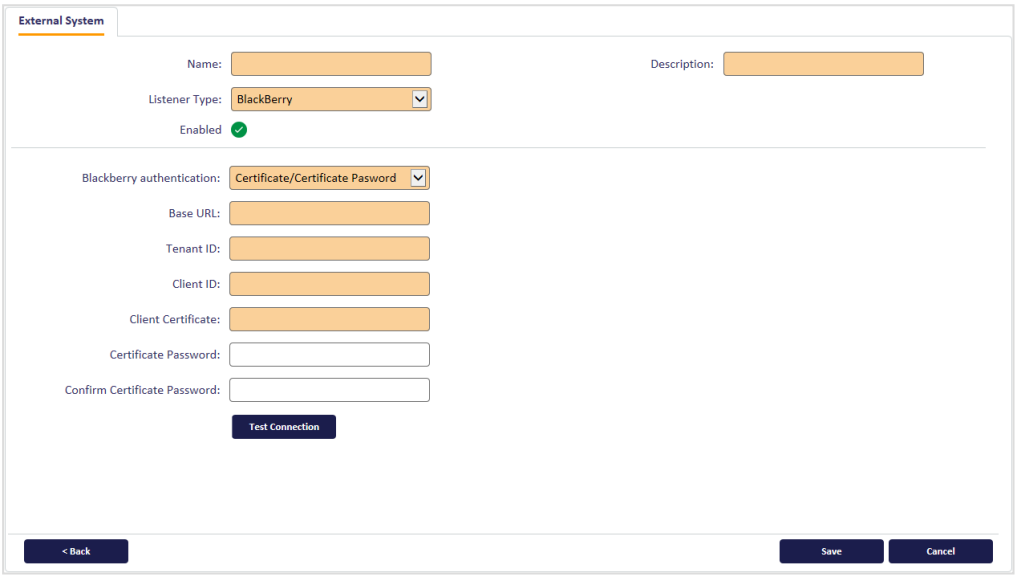
1. From the **Configuration** category, select **External Systems**.

   You can also launch this workflow from the **Connections and Notifications** section of the **More** category in the MyID Operator Client. See the *Using Connections and Notifications workflows* section in the ***MyID Operator Client*** guide for details.

2. Click **New**.

3. From the **Listener Type** drop-down list, select **BlackBerry**.

   The BlackBerry options appear:



4. From the **BlackBerry Authentication** drop-down list, select one of the following:

   - **Certificate/Certificate Password** – Select this option if you are using OAuth client credentials with a certificate's private key through the BlackBerry Enterprise Identity service. The **Client ID**, **Client Certificate** and **Certificate Password** fields are now available.

     See section *3.11.3.1*, *Configuring BlackBerry UEM*.

     Note: You can use certificate-based authentication for both on-premises UEM environments and cloud environments.

   - **User/password authentication** – Select this option if you are using basic authentication for the BlackBerry UEM system. The **Username** and **Password** fields are now available.

     Note: User/password authentication is available only for on-premises UEM environments; if you are using cloud environments, you must use certificate-based authentication.

5. Set the following options:

   - **Name** – Type a name for the external system.
   - **Description** – Type a description for the external system.
   - **Enabled** – Set this option to enable the BlackBerry UEM connector. You can disable the connector and MyID will not attempt to communicate with the BlackBerry UEM system.

- **Base URL** – Type the URL of the BlackBerry installation. Use the following form:

  `https://<server>:<port>`

  where:

  - `<server>` – the server
  - `<port>` – the port for the BlackBerry UEM instance.

    For on-premises environments, this is:

    `18084`

    For cloud environments, this is:

    `443`

- **Tenant ID** – The tenant ID of the BlackBerry UEM server. BlackBerry refers to this as the "Tenant GUID" or "SRPID". For example:

  `S12345678`

- **Client ID** – If you selected **Certificate/Certificate Password** from the **BlackBerry Authentication** drop-down list, provide the ID of the `client_credentials` client you set up with BlackBerry Enterprise Identity.

  See section *3.11.3.1*, *Configuring BlackBerry UEM*.

- **Client Certificate** – If you selected **Certificate/Certificate Password** from the **BlackBerry Authentication** drop-down list, provide the path and filename for the PFX file containing the private key corresponding to the public key provided to the `client_credentials` client through BlackBerry Enterprise Identity and exported from the BlackBerry system. This must be present on the application server in a folder accessible to the MyID COM+ user; for example:

  `C:\certs\client.pfx`

- **Certificate Password** – If the certificate file is password protected, type the certificate password, then type it again in the **Confirm Certificate Password** field.
- **Username** – If you selected **User/password authentication** from the **BlackBerry Authentication** drop-down list, type the name of the configured BlackBerry UEM admin account.
- **Password** – Type the password for the configured account, then type it again in the **Confirm Password** field.

6. Click **Test Connection** to check that you have entered the details correctly.
7. Click **Save**.
8. Restart the Edefice_BOL component to ensure that MyID is working with the updated configuration.

   To restart the component:

   a. On the MyID application server, open Windows Component Services.

   b. Expand **Component Services > Computers > My Computer > COM+ Applications**.

   c. Right-click **Edefice_BOL**, then from the pop-up menu click **Shut down**.

   The component will restart automatically the next time it is needed.

3.11.3.1 Configuring BlackBerry UEM

If you are using certificate-based authentication, you must configure your BlackBerry UEM system with an OAuth client credentials grant type with an enterprise app.

Configure the app resources in your BlackBerry Online Account, using the public key corresponding to the private key of the certificate you specify in the **Client Certificate** field in the **External Systems** workflow.

You must enable and authorize the app in BlackBerry UEM.

You must obtain the following:

- The client ID.
- The client certificate file.
- The certificate file password, if the file is password protected.

Note: While the documentation recommends OAuth with the `authorization_code` grant type for third-party applications, due to the nature of the MDM connector, you must use `client_credentials` instead.

3.11.4 Configuring credential profiles for MDM restrictions

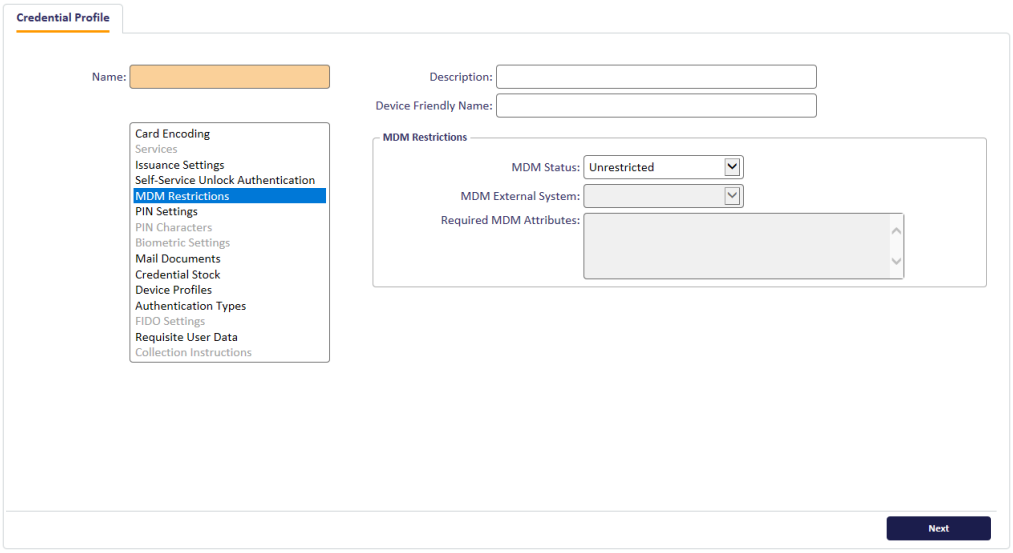
You can configure a credential profile to issue only to devices registered with the MDM, and you can require particular attributes of registered devices as stored in the MDM.

To set the MDM restrictions:

1. From the **Configuration** category, select **Credential Profiles**.

   You can also launch this workflow from the **Credential Configuration** section of the **More** category in the MyID Operator Client. See the *Using Credential Configuration workflows* section in the ***MyID Operator Client*** guide for details.
2. Create a new credential profile or modify an existing one.
3. In the **Card Encoding** section, make sure **Identity Agent** is selected.

   The **MDM Restrictions** section of the credential profile is available only when you have selected **Identity Agent** or **Mobile Identity Documents**.
4. Select the **MDM Restrictions** section.



5. Set the following options:

- **MDM Status** – Select one of the following:
  - **Unrestricted** – MyID does not carry out any checks against the MDM at collection.
  - **Must be registered** – The mobile device must:
    - Have an external mobile ID, *and*:

      Must be present in the connected MDM system.
  - **Must not be registered** – The device must either:
    - Have no external mobile ID registered in MyID, *or*:
    - Not be found in the connected MDM system.
- **MDM External System** – If you have set the **MDM Status** to **Must be registered** or **Must not be registered**, you must select an MDM external system from the drop-down list. If you have only one MDM external system configured, it is automatically selected.
- **Required MDM Attributes** – If you have set the **MDM Status** to **Must be registered**, you can specify any required attributes in the MDM, and MyID checks that the mobile device fulfills these requirements.

  Specify the required attributes as:

  ```
  [field]=[value]
  ```

  For example:

  ```
  jailBroken=False
  ```

  Important: The attributes and values are case sensitive.

  You can specify multiple conditions by separating them with a comma.

  Note: For nested JSON attributes, use a dot (.) to separate the components; for example:

  ```
  platform_info.platform_name=iOS
  ```

At collection, the MDM entry for the mobile device must meet all the required conditions.

6. Click **Next** and complete the workflow.

3.11.5 MDM validation

If you attempt to issue a mobile device with MDM restrictions, and the MDM system does not recognize the mobile device as valid (for example, it is not registered, or does not have the required MDM attributes) the following error occurs:

- REST007 – Unrecoverable error has occurred.

You can check the audit to see if there is any additional information about the validation error. For example, the identifier used to look up the device in the MDM is captured in the audit record; you can use this to correlate against your MDM system to identify why the device was rejected.

4 Requesting and approving mobile IDs

You can request a mobile ID for your own mobile device or for another user's mobile device.

The user for whom the mobile ID is requested must have the following:

- A cell/mobile phone number in their MyID record.
- An email address in their MyID record.

Collecting the mobile ID may take several minutes, depending on the complexity of the certificates and the speed of your network connection. If the collection fails due to network problems, you are recommended to use the **Cancel Credential** workflow to cancel the mobile ID, then request another mobile ID for the user.

When you are working with an MDM, it may require a specific process to be followed to collect certificates using the MDM app. See section *6.3*, *How do you request a derived credential?* for details.

4.1 Recovering archived certificates

To recover a certificate from an existing card, the user must have a certificate that:

- is issued to a current device.
- has archived keys.
- is issuable and recoverable to software.
- has a policy that is available on at least one credential profile available to the user that has **Identity Agent** selected.

4.2 Requesting a mobile ID for another user

You can request a mobile device for a person in the following ways:

- Using the **Request Mobile** or **Request Mobile (View Auth Code)** options in the MyID Operator Client.

  See the *Requesting a mobile device for a person* section in the ***MyID Operator Client*** guide.
- Using the MyID Core API.

  This uses the same mechanism and requires the same configuration as the **Request Mobile** or **Request Mobile (View Auth Code)** options in the MyID Operator Client. See the *Accessing the API documentation* section in the ***MyID Core API*** guide for details of accessing the API documentation, which contains details of the relevant methods.
- Using the **Request ID** workflow in MyID Desktop.

  See section *4.2.1*, *Requesting a mobile device in MyID Desktop*.
- Using the Credential Web Service API.

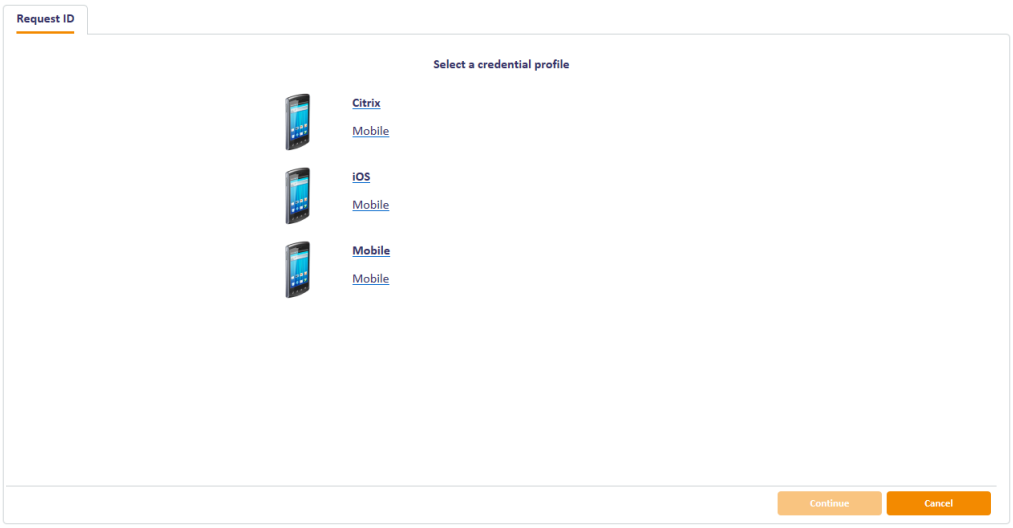
  This uses the same mechanism and requires the same configuration as the **Request ID** workflow in MyID Desktop. See the ***Credential Web Service*** guide.

4.2.1 Requesting a mobile device in MyID Desktop

Note: The **Request ID** workflow is not assigned to any roles by default. You must use the **Edit Roles** workflow to ensure that this workflow is assigned to the roles you want to be able to request mobile devices.

To request a mobile ID for another user:

1. From the **Mobile Devices** category, select **Request ID**.
2. Use the Find Person screen to select the appropriate person.
3. Select the credential profile you want to use.



4. Click **Continue**.
5. Check that the phone number or email address is correct.

   The phone number is taken from the **Cell** or **Mobile** (depending on the language setting) field in the user's MyID record.
6. If your system is not configured to send OTP authentication codes through SMS, take a note of the code on-screen.

   If your system is configured to send OTP authentication codes through SMS, this code is sent directly to the mobile device.

   This single-use code is required to install the mobile ID on the mobile device. If you have set the credential profile to require validation, the password does not appear on this screen; instead, you must use the **Validate Request** workflow.

   Note: The space in the password is optional when you enter the password on the mobile device.
7. Click **Send**.

   If both SMS and Email options are available, choose one of the methods to send the notification.

   MyID uses email or the SMS gateway to send a message. You can now collect the mobile ID on your mobile device.

4.3 Requesting a mobile ID for your own mobile device

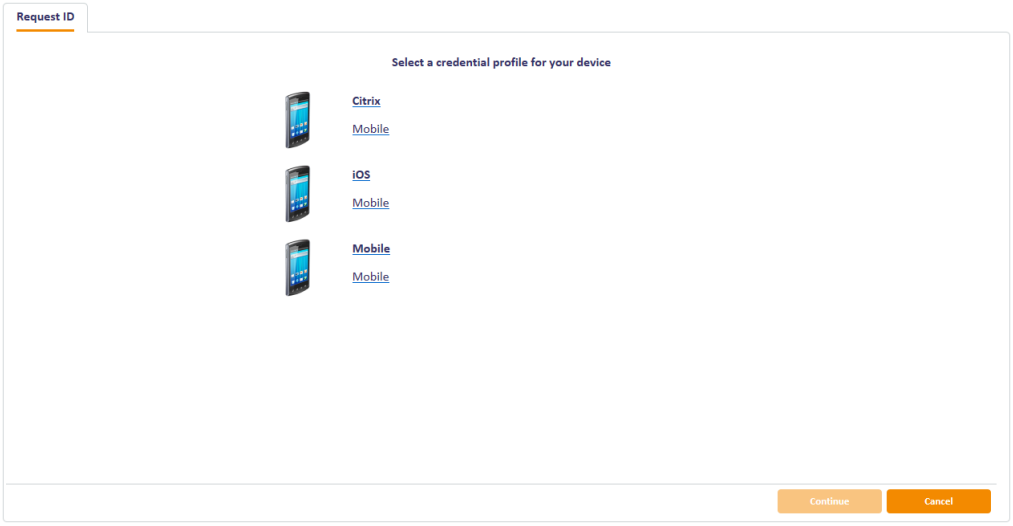
Note: The **Request My ID** workflow is not assigned to any roles by default. You must use the **Edit Roles** workflow to ensure that this workflow is assigned to the roles you want to be able to request mobile devices.

To request a mobile ID for your own mobile device:

1. From the **Mobile Devices** category, select **Request My ID**.

   Note: You can also launch this workflow from the self-service menu in the MyID Operator Client. See the *Launching self-service workflows* section in the ***MyID Operator Client*** guide for details.

2. Select the credential profile you want to use.



3. Click **Continue**.

4. Take a note of the password.

   This single-use code is required to install the mobile ID on the mobile device. If you have set the credential profile to require validation, the password does not appear on this screen; instead, you must use the **Validate Request** workflow.

   Note: The space in the password is optional when you enter the password on the mobile device.

5. Check that the phone number or email address is correct, then click **Send**.

   The phone number is taken from the **Cell** or **Mobile** (depending on the language setting) field of your MyID record.

   If you do not have an email address or mobile number set up on your account, MyID displays a QR code. Open the Identity Agent app on your phone and scan the QR code on screen, then click **Done**.

   Note: If you have an email address or mobile number set up, but prefer to use a QR code, click the **QR Code** button at the bottom of the screen. This option is not available if the credential profile has the **Validate Issuance** option set.

5 Working with mobile IDs

Once a user has been issued with a mobile ID, you can use MyID to manage the mobile IDs and their certificates.

Use of some of these lifecycle capabilities may depend on the apps you use to collect certificates. For example, unlocking mobile IDs may not be suitable for MDM-controlled apps.

This chapter contains information on:

- Support for mobile device lifecycle operations across different integrations.

  See section *5.1*, *Mobile device lifecycle operations*.
- Canceling mobile IDs.

  See section *5.2*, *Canceling mobile IDs*.
- Requesting replacement mobile IDs.

  See section *5.3*, *Requesting replacement mobile IDs*.
- Enabling and disabling mobile IDs.

  See section *5.4*, *Enabling and disabling mobile IDs*.
- Unlocking mobile IDs.

  See section *5.5*, *Unlocking mobile IDs*.
- Updating mobile IDs.

  See section *5.6*, *Updating mobile IDs*.
- Renewing mobile IDs.

  See section *5.7*, *Renewing mobile IDs*.

5.1 Mobile device lifecycle operations

MyID provides credential lifecycle operations that may be supported for your mobile devices, depending on the integration you have implemented.

Lifecycle operation	Microsoft InTune	VMWare Workspace ONE	BlackBerry UEM	Third-party app
Canceling mobile ID	✓	✓	✓	✓
Enabling and disabling mobile IDs	✓	✓	✓	✓
Requesting replacement mobile IDs	Repeat the issuance process on the new device.	Repeat the issuance process on the new device.	Repeat the issuance process on the new device.	Contact the app vendor.
Unlocking mobile IDs				Contact the app vendor.
Updating mobile IDs				Contact the app vendor.
Renewing mobile IDs	See the Microsoft documentation.	See the VMWare Workspace ONE documentation.	See the BlackBerry UEM documentation.	Contact the app vendor.

5.2 Canceling mobile IDs

If provisioning the mobile device fails, the mobile device is lost or stolen, or if the certificates expire and need to be replaced on the same device, you can cancel the mobile identities on the mobile device.

This does not affect the contents of the mobile device directly, but it revokes or suspends the certificates that were copied to the mobile device, and cancels the device in the MyID database. Any online check of the certificates or the mobile ID will fail, indicating that the mobile ID is no longer valid.

If you have multiple key stores on the same device, all key stores are canceled at the same time. Note, however, that any mobile identity documents on the same device are unaffected; you must cancel mobile identity documents separately.

You can cancel a mobile device in the following ways:

- Using the **Cancel Device** option in the MyID Operator Client.

  See the *Canceling a device* section in the ***MyID Operator Client*** guide.

- Using the MyID Core API.

  This uses the same mechanism and requires the same configuration as the **Cancel Device** option in the MyID Operator Client. See the *Accessing the API documentation* section in the ***MyID Core API*** guide for details of accessing the API documentation, which contains details of the relevant methods.

- Using the **Cancel Credential** workflow in MyID Desktop.

  See section *5.2.1*, *Canceling a mobile device in MyID Desktop*.

5.2.1 Canceling a mobile device in MyID Desktop

The **Cancel Credential** workflow allows you to cancel an issued ID and revoke its certificates.

See the *Canceling a credential* section in the ***Operator's Guide*** for details of using the **Cancel Credential** workflow.

In addition to the **Cancel Credential** workflow within MyID, you can also cancel a mobile ID from an external system using the `CancelDevice` method of the Device Management API. For more information, see the ***Device Management API*** guide.

5.2.2 Important information about canceling mobile IDs

Canceling a mobile ID from the mobile device removes the identity from the mobile device, but does not revoke the certificates. The recommended method is to use **Cancel Credential** within MyID to cancel the mobile identity in the MyID database and revoke its certificates, then cancel the mobile identity on the mobile device itself to clean up the security objects on the device.

5.3 Requesting replacement mobile IDs

You can request a replacement mobile device for a person in the following ways:

- Using the **Request Replacement Mobile** or **Request Replacement Mobile (View Auth Code)** options in the MyID Operator Client.

  See the *Requesting a replacement mobile device* section in the ***MyID Operator Client*** guide.

- Using the MyID Core API.

  This uses the same mechanism and requires the same configuration as the **Request Replacement Mobile** or **Request Replacement Mobile (View Auth Code)** options in the MyID Operator Client. See the *Accessing the API documentation* section in the ***MyID Core API*** guide for details of accessing the API documentation, which contains details of the relevant methods.

- Using the **Request Replacement ID** workflow in MyID Desktop.

  See section *5.3.1*, *Requesting a replacement mobile device in MyID Desktop*.

5.3.1 Requesting a replacement mobile device in MyID Desktop

The **Request Replacement ID** workflow allows you to replace a mobile ID that is missing or damaged.

1. From the **Mobile Devices** category, click **Request Replacement ID**.
2. Use the Find Person screen to select the person.

   The devices assigned to the person are listed.
3. Select the device you want to replace.
4. Select a reason and provide **Details** for the card replacement, then click **Next**.

The old mobile ID is canceled, and a job for a replacement mobile ID is created.

5.4 Enabling and disabling mobile IDs

You can enable or disable a mobile device in the following ways:

- Using the **Enable Device** and **Disable Device** options on the View Device screen in the MyID Operator Client.

  See the *Enabling and disabling devices* section in the ***MyID Operator Client*** guide.

- Using the MyID Core API.

  This uses the same mechanism and requires the same configuration as the **Enable Device** and **Disable Device** options in the MyID Operator Client. See the *Accessing the API documentation* section in the ***MyID Core API*** guide for details of accessing the API documentation, which contains details of the relevant methods.

- Using the **Enable / Disable ID** workflow in MyID Desktop.

  See section *5.4.1*, *Enabling or disabling a mobile device in MyID Desktop*.

5.4.1 Enabling or disabling a mobile device in MyID Desktop

The **Enable / Disable ID** workflow allows you to change the status of an issued ID and its certificates; you can disable an ID so that the certificates are suspended, or enable an ID so that the user can use its certificates again.

To enable or disable a mobile ID:

1. From the **Mobile Devices** category, click **Enable / Disable ID**.
2. Click **Search** then use the Find Person screen to find the cardholder, then select the device you want to enable or disable.
3. To disable a mobile ID, select the reason and type the details for disabling the mobile ID, then click **Disable**.

   To re-enable a mobile ID, click **Enable**.

5.5 Unlocking mobile IDs

You can unlock a mobile ID in the following ways:

- Using the **Unlock Credential** workflow in MyID Desktop.

  The **Unlock Credential** workflow allows you to retrieve an unlock code for an issued ID. The mobile device owner starts the unlock process on their mobile device, then contacts the helpdesk operator, who uses the **Unlock Credential** workflow to provide an unlocking code.

  See the *Unlocking a credential remotely* section in the ***Operator's Guide***.

  You can also launch the **Unlock Credential** workflow from the View Device screen of the MyID Operator Client. The **Unlock Credential** workflow appears in a MyID Desktop window with the device already selected. See the *Unlocking a device* section in the ***MyID Operator Client*** guide for details.

- Using the `RequestUnlockCodeForDevice` method of the Credential Web Service API. For more information, see the ***Credential Web Service*** guide.

Known issue:

- **IKB-179 – Cannot request an authentication code to unlock a mobile device**

  The **Unlock Credential** workflow can require an authentication code to confirm the user's identity before proceeding with the unlock process. For mobile credentials, it is not currently possible to request an authentication code for unlocking. Ensure that operators who perform unlock for mobile devices do not require authentication codes, or can bypass authentication where needed.

5.6 Updating mobile IDs

You can request an update for a mobile device using the **Request Update** option on the View Device screen in the MyID Operator Client.

See the *Requesting an update for a device* section in the ***MyID Operator Client*** guide for details.

You can also request an update for a device using the MyID Core API. This uses the same mechanism and requires the same configuration as the **Request Update** option in the MyID Operator Client. See the *Accessing the API documentation* section in the ***MyID Core API*** guide for details of accessing the API documentation, which contains details of the relevant methods.

5.7 Renewing mobile IDs

You can request a renewal for a mobile device using the **Request Device Renewal** option on the View Device screen in the MyID Operator Client.

See the *Renewing a device* section in the ***MyID Operator Client*** guide for details.

You can also request a renewal for a device using the MyID Core API. This uses the same mechanism and requires the same configuration as the **Request Device Renewal** option in the MyID Operator Client. See the *Accessing the API documentation* section in the ***MyID Core API*** guide for details of accessing the API documentation, which contains details of the relevant methods.

6 MDMs and derived credentials

You can use MyID derived credentials in conjunction with MDM systems such as Microsoft Intune and VMWare Workspace ONE.

See section *2.3*, *Supported Mobile Device Management integration* for details of which systems are supported, and section *3.11*, *Setting up your MDM system* for details of configuring the external systems for each type of MDM.

The objective is to use the trust placed in an issued certificate and use this to derive additional credentials to a mobile device managed by your MDM, to enable certificates to be used, depending on policy, for:

- App Authentication
- Email
- VPN
- S/MIME signing and encryption
- Wi-Fi authentication

Certificate issuance from MyID can be triggered by any process that displays a QR code to start a mobile provisioning process, for example Self Service Request Portal, PIV Derived Credential Kiosk or the **Request My ID** workflow in MyID Desktop.

The MDM providers have built the Intercede mobile SDK components into their apps to manage the provisioning process for certificates.

During the derived credential request process, MyID shows a QR code on screen. This is scanned by the mobile device which then triggers issuance of certificates from MyID to the mobile device. The MDM then takes control of these certificates to enable the certificate usage defined by MDM configuration.

6.1 Compliance with NIST guidelines for derived PIV credentials

If your organization is required to comply with the National Institute of Standards and Technology (NIST) guidelines for Derived Personal Identity Verification (PIV) credentials, you must use this feature with the PIV edition of MyID. A PIV card must be used to request the derived credentials.

Note: You must use the Self Service Request Portal or the PIV Derived Credential Kiosk to request the mobile identity; the **Request My ID** workflow in MyID Desktop does not follow the necessary process requirements for a PIV derived credential.

6.2 How do you configure MyID to issue derived credentials?

You must set up your MyID system to issue mobile identities, as described in section *3*, *Configuring the system*.

When you set up your system, make sure you configure the credential profile to use the Intercede Key Store; that is, in the **Card Format** drop-down list, select **None** to ensure that MyID uses the default key store. The Intercede key store is used as a temporary intermediate store for the certificates issued by MyID before the MDM system takes control of the certificates.

When you select the certificates in the credential profile, select whatever certificates your system requires; for example, you may need both a signing certificate and an encryption certificate, or just a signing certificate.

6.3 How do you request a derived credential?

To request a derived credential:

1. Register your mobile device with your MDM system.

   Your organization's portal will prompt you to request a mobile smart card.

2. Follow the instructions to use your existing smart card to request a derived credential.

   Depending on your system configuration, carry out one of the following:

   - Take your PIV card and insert it into the MyID Self-Service Kiosk.

     You must run the Kiosk with the `/dc` command-line parameter. See the *Running the Self-Service Kiosk* section in the ***Self-Service Kiosk*** guide for details.

     For this method, the PIV card does not need to have been issued by the MyID system you are using to issue derived credentials.

     This is the recommended method if you have an issued PIV card.

   - Take your PIV card and present it to the Self-Service Request Portal.

     See the ***Derived Credentials Self-Service Request Portal*** guide for details.

   - Use your MyID credentials to log on to MyID Desktop and use the **Request My ID** workflow.

     For this method, you must have credentials issued by the MyID system that is issuing the derived credentials.

     See section *4.3*, *Requesting a mobile ID for your own mobile device* for details.

   At the end of either process, you are presented with a QR code.

3. Use your organization's app to scan the QR code.

   MyID then issues certificates to your mobile device.

6.4 How do you manage derived certificates?

You can use MyID to manage the issued derived certificates using the standard MyID lifecycle management features; for example, you can revoke the certificates using the **Cancel Credential** workflow.

You can use the Derived Credentials Notification Listener to update MyID when the status of the original PIV card changes; for example, when the PIV card is canceled, you can use the API to inform the MyID system that the original card is no longer trusted, and MyID can revoke the derived credential. See the ***Derived Credentials Notifications Listener API*** guide for details.

For Intune, Microsoft recommend that replacement, renewed, or updated derived credentials are issued by canceling the existing derived credentials and repeating the process used to issue the credentials. For the certificates that are issued to your derived credentials, do not select a certificate policy that has the **Automatic Renewal** option set in the **Certificate Authorities** workflow.

7 Troubleshooting

This section provides information on the following:

- Limitations.
- Logging.
- Retry attempts.
- Configuration issues.

7.1 Limitations of mobile badge layouts

The mobile badge layouts displayed on your mobile devices do not support the full range of layout options that you can specify in the MyID Card Layout Editor.

Note: Badge layouts are displayed with only some third-party apps; check with your app vendor when planning deployment.

- Badges are displayed in portrait orientation only. The badge is scaled to fit the width of the screen.
- Only the Android or iOS system font is available.
- Only horizontal text is supported.

You are recommended to create specific card layouts for your mobile identities, and test them on your target devices before implementing your production system.

- **IKB-290 – Cannot use .jpeg extension**

  Mobile layouts can use PNG or JPEG image files; however, you must ensure that the files have the extension `.png` or `.jpg`. If you attempt to issue a mobile identity that contains a layout with an image that has a `.jpeg` extension, issuance will fail.

7.2 Setting up logging

You can configure the Identity Agent app to create a log file for debugging purposes. Customer support may ask you to set the log level and send the resulting log file to Intercede for analysis.

Note: The Identity Agent app uses the system default email app to send the log file. For iOS devices, this means that you must have Apple Mail configured with at least one email account.

To enable logging, use the following configuration options on the **Identity Agent Policy** page of the **Operation Settings** workflow:

- **Administrator email address** – Set this to the email address to which Identity Agent will send logs for troubleshooting purposes.
- **Log level** – Set this to the level of debug logging you want Identity Agent to produce. Higher levels result in more detail, but larger files.

  Set to one of the following:

  `0 – NONE`

  `1 – FATAL`

  `2 – ERROR`

  `3 – WARNING`

  `4 – INFO`

  `5 – DEBUG`

  `6 – VERBOSE`

  By default, the log level is set to level 2, `ERROR`.

  Note: This setting affects the level of *debug* logging only; the Identity Agent also logs all *messages* that occur between the client and the server. If you want to switch off logging altogether, set the **Maximum number of log files** to `0`.
- **Maximum log storage space** – The maximum amount of space (in MB) that log files will take up on a device. Once this limit is reached, log files will be deleted automatically, oldest first, to clear room for new files.
- **Maximum number of log files** – The maximum number of log files to be stored on a device. Once this limit is reached, log files will be deleted automatically, oldest first, to clear room for new files.

To allow as many files as will fit in the maximum log storage space, set this value to `-1`. This is the default setting.

To switch off logging, set this value to `0`.

7.3 Retry attempts

You can configure how Identity Agent handles attempts to reconnect to the server if the connection is lost during an operation.

Use the following configuration options on the **Identity Agent Policy** page of the **Operation Settings** workflow:

- **Maximum retry attempts**

  The maximum number of times Identity Agent should attempt to reconnect to the server if connection is lost during an operation. The default is 5 times.

- **Minimum retry delay**

  The minimum delay, in seconds, between each attempt to contact the server after connection has been lost. The default is 10 seconds.

7.4 Configuration issues

This section lists some issues that may occur if you have not configured the system correctly.

- None of the selected user's certificates are configured to be allowed to be put on a mobile phone.

  Make sure that you have setup the credential profiles correctly according to the instructions in this document. Make sure that the user has permission to receive the credential profile, and that the issuer can issue the credential profile.

  Make sure that the certificate policy is the correct one.

  Make sure that the certificate policy can be issued in software.

- The selected user has no certificates suitable for mobile devices and there are no credential profiles available for issuance.

  Make sure that the user has permission to receive the credential profile, and that the issuer can issue the credential profile.

  Make sure that the certificate policy is the correct one.

- The selected user has neither phone number nor email address registered and so is not suitable for mobile device activation.

  Make sure that the user has an entry in the MyID database for Mobile or Cell phone number or for email address. If you are using LDAP integration, and you do not have this field populated in the directory, synchronizing MyID with the directory may clear this field from the MyID database.

- The collection email does not contain the full URL.

  Make sure you have set the **Mobile Certificate Recovery Service URL** option to contain the server address; this is used as the start of the collection URL. This option is used for more operations than certificate recovery, despite the name.

- The issuance fails with an error similar to:

  ```
  Error: 0x8004600b : The user does not have sufficient privileges to
  carry out this action
  Info: Access denied to OperationID 103001 for user u10006,
  logonMechanism 0
  ```

This may occur when the user does not have access to the **Collect My Updates** and **Issue Device** workflows through a role that is set to allow **Password** as a **Logon Mechanism** – this logon mechanism is required for mobile issuance.