MyID Enterprise

Version 12.9

Self-Service App

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

Document reference: IMP1852-12.9.0-Enterprise

December 2023

Copyright

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.

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 describes the installation and configuration of the MyID® Self-Service App.

This application allows users to collect, activate and update their security devices (for example, smart cards, virtual smart cards, and Windows Hello for Business credentials) without requiring operator access to MyID Desktop. Users can also use the Self-Service App to reset or change the PIN on their own card.

The Self-Service App operates in the following ways:

- Interactive mode – The user runs the app as a normal desktop application, and can use the app to reset or change their PIN, or to check if there are any collection, activation, or update tasks available for their security device.

  See section *3.1*, *Self-Service App interactive mode* for details.

- Wizard mode – The app runs in the background, and pops up a notification if there are any tasks available for the user's security devices.

  See section *3.2*, *Self-Service App wizard mode* for details.

- Automation mode – The app runs in the background, and if there is a VSC PIN lock or unlock task available for the user, the app processes the task automatically without any user interaction.

  See section *3.3*, *Self-Service App automation mode* for details.

The user interface is designed to guide users through the process without requiring extensive training or documentation. The app is recommended for end-user credential management, as it is easy to use, has a friendly user interface, and is easy to deploy and configure.

2 Installing the Self-Service App

This chapter contains information on prerequisites and installation of the Self-Service App.

See:

- section *2.1*, *Prerequisites*.
- section *2.2*, *Running the installation program*.

2.1 Prerequisites

The Self-Service App requires the following software:

- .NET Framework 4.8

  You must have .NET 4.8 installed on your client PCs. MyID is developed and tested using .NET Framework 4.8; if you need to use a later version of the .NET Framework, contact customer support quoting reference SUP-283.

2.1.1 Communication between the Self-Service App and MyID

To allow your clients to communicate with the MyID server, your PC must be able to communicate with the URLs of the MyID 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.

You must add the address of the MyID server to the **Trusted Sites** list in **Internet Options** on the client PC.

You are recommended to set up SSL/TLS on the connection between the Self-Service application clients and the MyID web services. See section *4.3*, *Setting up SSL/TLS* for details.

2.1.2 Minimum client PC specifications

Your client PC must meet the following minimum specifications:

- 1 GHz 32-bit (x86) or 64-bit (x64) processor
- 1 GB RAM (32-bit) or 2 GB RAM (64-bit)
- 2 GB hard disk free space
- Screen resolution of 1024x768 (at 100% text size)
- Network access

2.1.3 Supported operating systems

The Self-Service App is supported on the following client operating systems:

- Windows 10 Anniversary Update (build 1607) or later – 32-bit or 64-bit.
- Windows 11.

2.1.4 Microsoft WebView2 Runtime

You must have the Microsoft WebView2 Runtime installed on each client PC where you want to use HTML-based Terms and Conditions or mailing documents. This component allows the client software (MyID Client Service, MyID Desktop, Self-Service App, and Self-Service Kiosk) to display or print the HTML content.

The minimum version supported of the WebView2 Runtime is 109.0.1518.46; however, you are recommended to use the Evergreen installer, which allows Windows to keep the component up to date with the latest features and security updates.

See the Microsoft website for details and to obtain an installer:

- ***developer.microsoft.com/en-us/microsoft-edge/webview2/***

2.2 Running the installation program

The installation program for the Self-Service App is provided in the following location within the MyID release:

```
MyID Clients\Self Service App\
```

Note: You do not need to have administrative privileges to install the Self-Service App. However, you must make sure that you have the correct permissions to install software to the program folder; for example, your system administrator may not permit you to install software to the default `Program Files (x86)\Intercede\` folder. In this case, choose a different destination location during the installation process.

Note: Intercede also provides MSIX versions of the installation programs for MyID Desktop, the Self-Service App, and the MyID Client Service. These are intended for an administrator to create an installation package that combines all of the necessary client software and administrator-controlled configuration. See the ***MyID Client MSIX Installation Guide*** for details of working with these installation programs.

- **IKB-359 – Issue when upgrading the Self-Service App from a previous version**

  When you install the Self-Service App over a previous version, the application may fail to launch; no error is displayed. Windows system events may be recorded on the client workstation (event ID 1000 or 1026, relating to `MyIDApp.exe`). This issue is caused when the installation program cannot replace files in the program folder. As a workaround, uninstall the previous version of the application, delete the program folder (by default, `C:\Program Files (x86)\Intercede\MyIDApp\`) and then install the new version.

To install the Self-Service App on your client PC:

1. Copy the installation program to a local drive.

   If you do not run the installation program from a local drive, you may experience problems with the App running slowly due to certificate checks.

2. Run the `.msi` installer.

3. Click **Next**.

4. Select the destination location.

   By default, the Self-Service App installs to:

   ```
   C:\Program Files (x86)\Intercede\MyIDApp\
   ```

   On a 32-bit system, this is:

   ```
   C:\Program Files\Intercede\MyIDApp\
   ```

5. Click **Next**.
6. In the **Server URL** box, type the location of the server on which the MyID Web Services are installed.

   For example:

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

   Note: Make sure you use the correct protocol: http or https.
7. In the **SSL Certificate Issuer DN** box, type the Issuer DN of the client-side certificate used to authenticate the client to the server for two-way SSL/TLS.

   This is optional.
8. Click **Next**.
9. Click **Install**.
10. When the installer has completed, click **Finish**.

Note: The shortcut to the Self-Service App is created only for the current user.

2.2.1 Installing the Self-Service App silently

To install silently on a client PC, you can use the `.msi` installer with the following command-line parameters:

```
msiexec /i "<msi path>" /lv <LogFile> /q SSA_SERVERNAME=<ServerURL>
SSLCERTIFICATEDN=<sslcertdn> INSTALLDIR=<InstallationFolder>
```

where:

- `<msi path>` is the path to the `.msi` file.
- `<LogFile>` is the name of the file to which you want to write a verbose log. This is optional.
- `<server url>` is the URL of the server on which the MyID Web Services are installed.

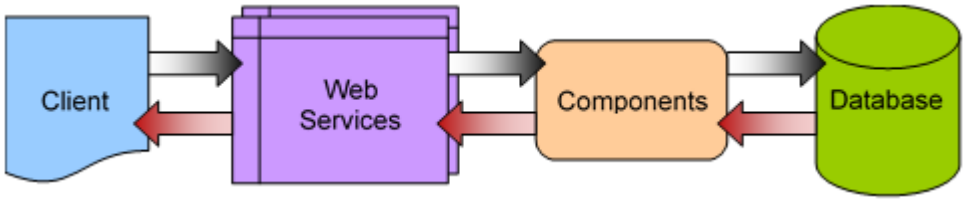
  Note: Make sure you use the correct protocol: `http` or `https`. Also, you must include the trailing slash (`/`).
- `<sslcertdn>` is the Issuer DN of the client certificate used to authenticate the client to the server for two-way SSL/TLS. This is optional.
- `<InstallationFolder>` is the name of the folder to which you want to install the application. This is optional.

Note: Do not put a space character on either side of the `=` sign in the command line.

For example:

```
msiexec /i "C:\install\<installer>.msi" /lv msilog.txt /q SSA_
SERVERNAME=https://myserver/ INSTALLDIR="C:\temp\ssa"
```

3 Overview

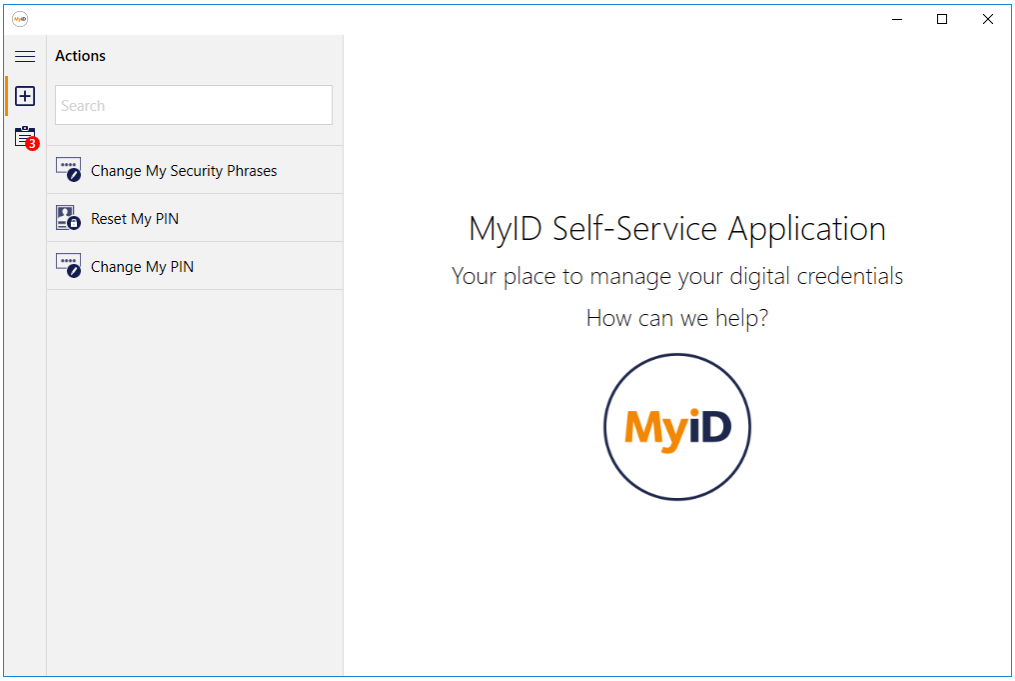


The Self-Service App passes requests through HTTP or HTTPS to the MyID Data Source and MyID Process Driver web services. The web services communicate using DCOM with the MyID components on the application server; these components provide the business logic and communicate with the MyID database. Responses are returned to the client through the MyID web services.

The web services, components and database may be on separate servers, or on the same server.

3.1 Self-Service App interactive mode

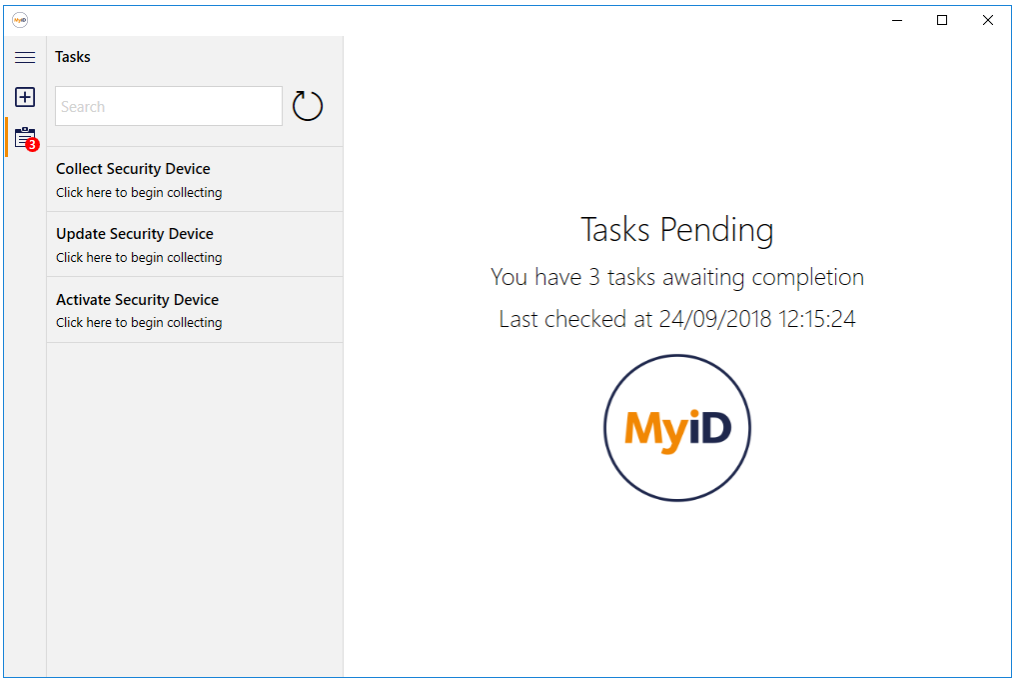
The Self-Service App interactive mode allows the user to run the app as a desktop application.



The app displays a list of the Actions available. Actions are processes that are initiated by the user – for example, the user can reset the PIN of their security device, or change the PIN to a different value.

The Tasks icon also displays a notification if there are any tasks available for the user:

Tasks are processes that are initiated by the system or an operator on behalf of the user – for example, collecting or activating a smart card. Click the Task icon to display the available tasks.

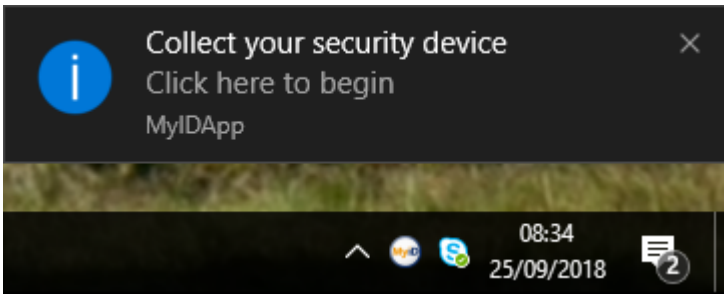


Click the Refresh button to refresh the list of tasks from the server.

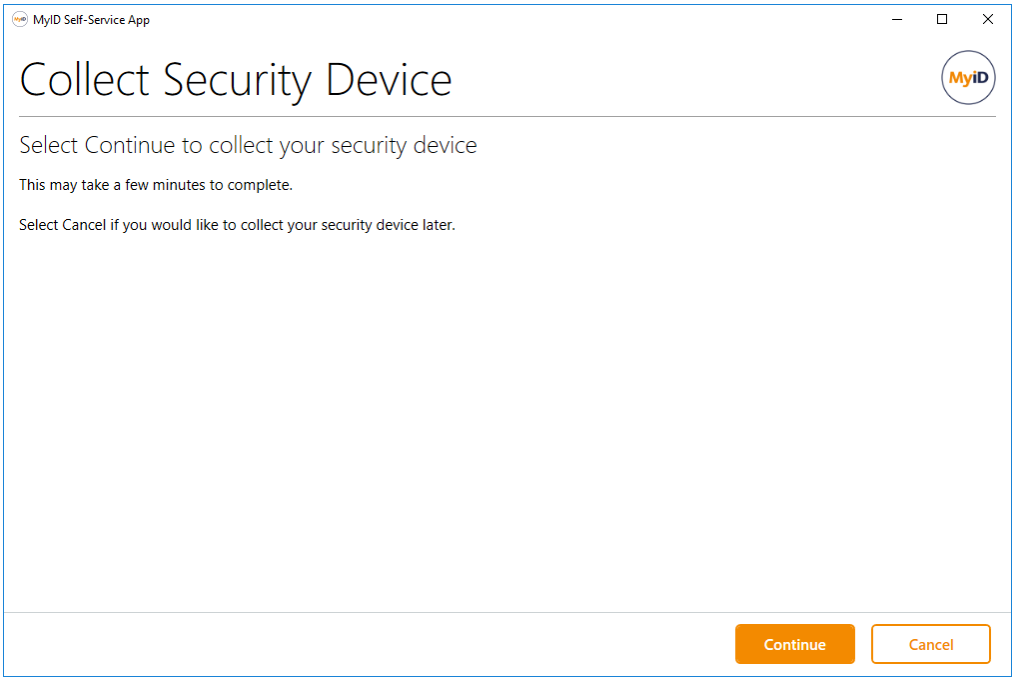
Both the Action list and the Task list are searchable.

3.2 Self-Service App wizard mode

The Self-Service App wizard mode is designed to run on the user's own PC and to provide notifications. If the user has a task available – for example, the collection of a security device – the Self-Service App displays a Windows notification.



The user then clicks the notification, and the Self-Service App guides the user through the task.



You can set up the Self-Service App to be run manually by the user to check for updates, to run automatically at Windows logon, to run periodically as a scheduled task, and so on. The app does not monitor available tasks continuously – it checks for tasks only when it is run, and shuts down if there are no current tasks available.

A simple scenario would be:

1. The user logs on to their own PC.
2. The Self-Service App starts automatically.
3. The Self-Service App checks the MyID server for pending tasks for the user.

4. If there are any tasks outstanding – for example, the collection of a new smart card – the Self-Service App pops up a notification in the Windows system tray.

   Note: If the user is collecting a task that supports both contact and virtual smart cards, the Self-Service App gives preference to the collection of a virtual smart card over a contact card.

5. The user clicks the notification bubble and the Self-Service App window appears.
6. The Self-Service App guides the user through the task.

   If there are no tasks available, the Self-Service App shuts down without alerting the user. This ensures that the user is presented with information only when they need to make a decision.

3.3 Self-Service App automation mode

The Self-Service App automation mode is designed to run on the user's own PC and to carry out VSC lock operations without user interaction.

You can set up the Self-Service App automation mode to run automatically at Windows logon, to run periodically as a scheduled task, and so on.

A simple scenario for a VSC PIN lock would be:

1. The user logs on to their own PC.
2. The Self-Service App automation mode starts automatically.
3. The Self-Service App checks the MyID server for requests for VSC PIN locks or unlocks.
4. If there is a pending request for a PIN lock or unlock, the Self-Service App processes the request.

No user interaction is required.

3.4 Authentication

When the user runs the Self-Service App, whether in interactive mode or wizard mode, they do not have to authenticate themselves until they actually start to carry out an action or a task. By default, the app uses the Windows account name of the logged-in user, and matches this against the `SAMAccountName` stored in the MyID database to return the list of available tasks.

Depending on how MyID is configured, the user can authenticate to the MyID server using the following methods:

- Smart card logon (not available for automation mode)
- Security phrase logon
- Windows authentication
- Authentication codes (not available for automation mode)

The user can also use various combinations; for example, smart card logon backed up by authentication codes.

See the *Logon mechanisms* section in the ***Administration Guide*** for details of setting up the various types of authentication. The Self-Service App uses the same configuration for authentication as MyID Desktop.

The Self-Service App supports an additional authentication feature – you can specify the order of authentication methods that are presented to the user. See the *Logon Priority page (Security Settings)* section in the ***Administration Guide*** for details.

Note: Some actions have specific requirements.

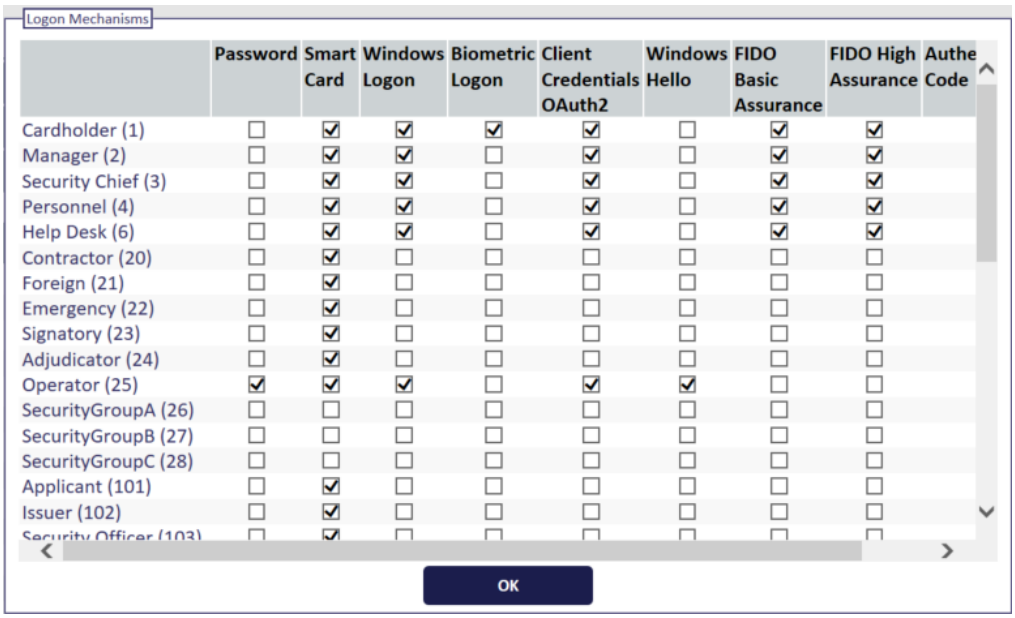
- **Change My Security Phrases** – you can specify the logon priority.
- **Reset My PIN** – you must use the mechanisms configured in the credential profile or the global configuration options.
- **Change My PIN** – you must authenticate using your smart card, as you must log on to your card with the existing PIN to change it.

Carrying out tasks requires the appropriate authentication mechanism:

- Collecting updates, reprovision jobs, or certificate renewals requires the device PIN.
- Collecting new or replacement devices or activating devices requires the authentication set up in the credential profile and in the MyID system configuration.

When you set up the roles for access to particular workflows, you must make sure that the role has the correct logon methods; for example, if you add all the workflows to the Applicant role, and are using security phrase logon, you must set the Applicant role to have access to the Password logon mechanism.

You can configure this using the Logon Mechanisms dialog in the **Edit Roles** workflow; see the *Assigning logon mechanisms* section in the ***Administration Guide***.



3.5 Self-Service App features

Control over which actions and tasks are available to Self-Service App users is maintained within MyID by using the standard roles mechanism. The user must be granted a role that has access to the correct workflows.

Use the **Edit Roles** workflow to specify which actions and task types are available.

3.5.1 Controlling which actions are available

The **Default SSA User** role determines which actions are visible in the Self-Service App's list of actions. Note, however, that once you have selected an action, you must authenticate to MyID, at which point if your *own* roles do not allow access to the appropriate workflow, you will not be allowed to progress with the action.

The Self-Service App can carry out the following actions:

- **Change My Security Phrases**

  Allows you to change your security phrases. Requires that the user's role has access to the **Change My Security Phrases** workflow.

- **Reset My PIN**

  Allows you to reset a locked card PIN. Requires that the user's role has access to the **Unlock My Card** workflow.

- **Change My PIN**

  Allows you to change your card PIN. Requires that the user's role has access to the **Change PIN** workflow.

  Note: This action requires a card that has been issued with MyID Logon capabilities; the user must also be permitted to log on with a smart card.

- **Update My Device**

  Allows you to update your card. Requires that the user's role has access to the **Collect My Updates** workflow.

  See section *4.12*, *Self-service device update* for details.

Note: You can also launch these operations 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.

When MyID is installed, the **Default SSA User** role has access to the **Change PIN**, **Unlock My Card**, and **Change My Security Phrases** workflows.

Note: If you have upgraded from an earlier version of MyID, the **Default SSA User** role may not have all of the required workflows; for example, MyID versions earlier than 11.4 did not include the **Change My Security Phrases** workflow for this role by default, and the upgrade process does not change the assigned workflows for the role. Use the **Edit Roles** workflow to make sure that the role has the correct workflow permissions.

3.5.2 Controlling which actions are available using the registry

You can use the Self-Service App action block list to hide particular actions from the Self-Service App on a per-machine or per-user basis.

This is not a security feature (anyone with access to the registry can make these changes) but a usability feature; because the Self-Service App displays the list of all actions available to the Default SSA User role, and only checks whether the user can carry those actions based on the user's own roles *after* the action is selected, you may want to be able to hide the unavailable actions on some PCs or for some users.

You specify the actions based on their numeric operation ID. You can use the following IDs:

- `110` – **Change My Security Phrases**
- `255` – **Reset My PIN**
- `202` – **Change My PIN**

To block actions on a per-user basis:

1. On the client PC, open the Registry Editor.
2. Open the following key:

   `HKEY_CURRENT_USER\Software\Intercede\SsaActionBlacklist`

   If the key does not exist, create it.
3. Within this key, create a String value with the name of the operation ID you want to hide.

   For example, create a String value with the name `110` to hide the **Change My Security Phrases** operation.

   Note: You do not need to add any data to the String value. The Self-Service App checks whether the String value is present.

Note: If you override the username being used for SSA using either the `/un` command line argument or the `MYID_USERNAME` environment variable, the per-user block list is ignored. You can still use the per-machine block list.

To block actions on a per-machine basis:

1. On the client PC, open the Registry Editor.
2. Open the following key:

   `HKEY_LOCAL_MACHINE\Software\Intercede\SsaActionBlacklist`

   On a 64-bit system, create the following key instead:

   `HKEY_LOCAL_MACHINE\Software\Wow6432Node\Intercede\SsaActionBlacklist`

   If the key does not exist, create it.
3. Within this key, create a String value with the name of the operation ID you want to hide.

   For example, create a String value with the name `110` to hide the **Change My Security Phrases** operation.

   Note: You do not need to add any data to the String value. The Self-Service App checks whether the String value is present.

3.5.3 Controlling which tasks are available

The Self-Service App can carry out the following types of task:

- Collect a card.

  Requires access to the **Collect My Card** workflow.
- Activate a card.

  Requires access to the **Activate Card** workflow.
- Update a card.

  Requires access to the **Collect My Updates** workflow.

Note: This task requires a card that has been issued with MyID Logon capabilities; the user must also be permitted to log on with a smart card.

- Collect a replacement card.

  Requires access to the **Collect My Card** workflow.

- Collect a certificate renewal.

  Requires access to the **Collect My Certificates** workflow.

  Note: This task requires a card that has been issued with MyID Logon capabilities; the user must also be permitted to log on with a smart card.

- Lock or unlock a VSC (in automation mode only).

  Requires access to the **Update VSC** workflow.

  If you launch the Self-Service App automation mode using a MyID username and password, you are strongly recommended to use a specially-created MyID user that has access only to the required workflow. Create a new role, grant it access only to the **Update VSC** workflow, create a new user with access only to that role, and set the user's security phrases. The MyID user must also have sufficient scope to carry out operations on behalf of the end user.

  See the *Requesting VSC locks* section in the ***Microsoft VSC Integration Guide*** for details of requesting PIN locks for VSCs.

4 Configuring the Self-Service App

This chapter contains information on configuring and running the Self-Service App.

4.1 Server location

The Self-Service App is configured to communicate with the MyID Web Services server when you install the application. If you want to change the server, you can edit the configuration file.

To edit the configuration file:

1. On the client PC, shut down the Self-Service App.
2. Back up the `MyIDApp.exe.config` file.

   By default, this is in the following folder:

   ```
   C:\Program Files (x86)\Intercede\MyIDApp\Self Service Application\
   ```
3. Using a text editor, open the config file.

   Note: Make the changes to the config file exactly as shown. The case is important.
4. Edit the `value` parameter in the following line:

   ```
   <add key="Server" value="http://myserver.example.com/"></add>
   ```

   For example:

   ```
   <add key="Server" value="http://myserver2.example.com/"></add>
   ```
5. Save the configuration file.

4.2 Timeout

The Self-Service App is configured to time out after 30 seconds on some stages. This ends the current activity after that period of inactivity.

The default timeout for MyID Windows clients (MyID Desktop, the MyID Self-Service Kiosk, and the MyID Self-Service App) is controlled by the **Page Timeout for Windows Clients** configuration option (on the **General** page of the **Operation Settings** workflow).

If you want to change the timeout for a particular installation of the Self-Service App, you can edit the configuration file. This overrides the configuration option in **Operation Settings**.

To edit the configuration file:

1. On the client PC, shut down the Self-Service App.
2. Back up the `MyIDApp.exe.config` file. By default, this is in the following folder:

   ```
   C:\Program Files (x86)\Intercede\MyIDApp\Self Service Application\
   ```
3. Using a text editor, open the `MyIDApp.exe.config` file.

   Note: Make the changes to the config file exactly as shown. The case is important.
4. Edit the `value` parameter in the following line:

   ```
   <add key="PageTimeoutSeconds" value="30"></add>
   ```

   If this line does not exist, you can add it to the `<appSettings>` section.

   For example:

   ```
   <add key="PageTimeoutSeconds" value="60"></add>
   ```

   This increases the timeout to 60 seconds.
5. Save the configuration file.
6. Restart the Self Service App.

4.3 Setting up SSL/TLS

4.3.1 One-way SSL/TLS

If you want to configure the Self-Service App to use one-way SSL/TLS for its communications with the MyID Web Services server, you must install the server's certificate under the Trusted Root Certification Authorities in the user's certificate store.

4.3.2 Two-way SSL/TLS

Note: If your server is set up to use two-way SSL/TLS, you must set up your client to use two-way SSL/TLS. If you do not specify SSL on the client, an error is displayed.

Note: The Self-Service App does not support two-way SSL/TLS using a certificate stored on a smart card.

To use two-way SSL/TLS using a specific certificate:

1. Install the client certificate in the user's personal store.

   The client certificate must have the Client Authentication application policy – this has the following OID:

   ```
   1.3.6.1.5.5.7.3.2
   ```

2. Find the client certificate's serial number:
   a. Run the `CertMgr.msc` snap-in.
   b. Expand **Personal > Certificates**.
   c. Double-click the client certificate.
   d. Click the **Details** tab.
3. Run the application using the following command line:

   `myidapp.exe /ssl /sslsn:<serial number>`

   where:

   `<serialnumber>` – the serial number of the client certificate. Enter the serial number without spaces. For example, if the serial number is:

   `62 00 00 00 34 fe 3c a9 a8 1c 98 6a f1 00 00 00 00 00 34`

   use the following command line

   `myidapp.exe /ssl /sslsn:6200000034fe3ca9a81c986af1000000000034`

If you run the application with the `/ssl` command line option but omit the `/sslsn` option, the application carries out the following:

1. The application checks the application settings file for the details of the last certificate that was successfully used to log on.
2. If no details are found, if the certificate is no longer in the personal store, or the server rejects the certificate, the application searches the personal store for certificates that match the issuer DN (optionally set up when you install the application) and have the Client Authentication policy.
3. If more than one certificate is found, the application displays a list of certificates for the user to select.

When the application has successfully logged on to the server using a certificate, the certificate's details are stored in the user's application settings file.

4.3.2.1 Specifying two-way SSL in the configuration file

An alternative to specifying `/ssl and /sslsn` on the command line each time you run the Self-Service App (for example, if you are launching the Self-Service App using a hyperlink, and therefore cannot specify `/ssl` or `/sslsn`) you can add settings to the `<appSettings>` node of the configuration file.

To edit the configuration file:

1. On the client PC, shut down the Self-Service App.
2. Back up the `MyIDApp.exe.config` file.

   By default, this is in the following folder:

   `C:\Program Files (x86)\Intercede\MyIDApp\Self Service Application\`
3. Using a text editor, open the config file.

   Note: Make the changes to the config file exactly as shown. The case is important.
4. In the `<appSettings>` node, add the following lines:

```
<add key="TwoWaySSL" value="true"/>
<add key="SSLCertificateSN" value="YourCertificateSerialNumber"/>
```

If you want to stop using two-way SSL, you can set the `TwoWaySSL` value to `false` or remove the line. If you set this value to false, but include `/ssl` on the command line, the command line takes precedence, and the Self-Service App attempts to use SSL. If you specify a value for `SSLCertificateSN` but also include `/sslsn` on the command line, the command line takes precedence.

5. Save the configuration file.

4.4 Integrated Windows Logon

If you set up the MyID server to use Integrated Windows Logon, the Self-Service App can use the user's currently logged-on Windows identity to authenticate to MyID without having to enter passphrases or use a smart card.

Note: Integrated Windows Logon is available for tasks only – actions do not support Integrated Windows Logon.

To set up integrated Windows logon:

1. In MyID Desktop, from the **Configuration** category, select **Security Settings**.
   a. On the **Logon Mechanisms** tab, make sure that **Integrated Windows Logon** is set to Yes.
   b. Click **Save changes**, then click **Save** to confirm your changes.
2. From the **Configuration** category, select the **Directory Management** workflow and set up a configuration-only directory for MyID.
   a. Click **New** and enter a new name – this can be any value.
   b. Select the **Retrieve Base DN** option.

      MyID attempts to connect to the directory and, if successful, displays a list of possible DNs. Select one of the DNs from the list.

      In most cases, you must select the DN that begins `CN=Configuration`.
   c. Click **Save**.

   See *Setting up a configuration-only directory* section in the ***Administration Guide*** for more information.
3. Edit the roles within MyID.
   a. From the **Configuration** category, select **Edit Roles**.
   b. Click the **Logon Methods** option, and select **Windows Logon** for each role you want to be able to log on with Integrated Windows Logon.
   c. Click **OK**.
   d. Click **Save Changes**.

Note: The fields `SAMAccountName` and `Domain` must be stored in MyID when using Integrated Windows Logon. The `Domain` must contain the NetBIOS domain name and not the DNS format.

You must also carry out additional configuration on the web services for Integrated Windows Logon; see the *Configuring the MyID web services for Integrated Windows Logon* section in the ***Web Service Architecture*** guide for details.

4.5 Running the Self-Service App

The installation program creates a shortcut for the Self-Service App, but you can also run the app in the following ways:

- Using a Windows Logon script.
- Using third-party software.
- By putting a shortcut in the Startup program group.
- Using the Windows Scheduler.
- From a hyperlink.

To run the Self-Service App from the command line:

1. Open a command prompt and change to the `MyIDApp` folder. By default, this is:

   ```
   C:\Program Files (x86)\Intercede\MyIDApp\Self Service Application\
   ```

2. Type the following, and press Enter:

   ```
   MyIDApp.exe
   ```

By default, the Self-Service App runs in interactive mode. To run in wizard mode or in automation mode, you must specify the appropriate command-line parameters.

For information about the additional command-line parameters, see section *5*, *Command line arguments*.

4.5.1 Launching the Self-Service App from a hyperlink

When you install the Self-Service App, it registers the `myidssa:` protocol – this means that you can click on hyperlinks on web pages and email messages to launch the Self-Service App. This allows you to create tailored email notifications from within MyID; for example, to send to a user when there is a new security device to collect.

You can use the same command line options as those available at the Windows command prompt – see section *5*, *Command line arguments* for details. If you need to use more than one command line option, use the `+` symbol to combine them.

Using the hyperlink mechanism, you can specify the following:

- Start the Self-Service App in interactive mode.

  ```
  myidssa://
  ```

- Start the Self-Service App in wizard mode.

  ```
  myidssa:///w
  ```

- Start the Self-Service App in wizard mode with no pop-ups.

  ```
  myidssa:///w+/nopopup
  ```

  Using the bare `myidssa:///w` link provides no feedback to the end user if there are no jobs to collect – you are recommended to use the `myidssa:///w+/nopopup` link so that the user can see that the Self-Service App has started and has checked for outstanding jobs.

- Start the Self-Service App to collect a specific task.

  You can use the `%jobid` placeholder in a MyID email template; this will be substituted with the appropriate job ID when the email message is created.

  For example, if your email template includes the following:

  ```
  Click <a href="myidssa:///jobid:%jobid">Self-Service App</a>
  ```

  when the email message is created, it would become something similar to:

  ```
  Click <a href="myidssa:///jobid:256">Self-Service App</a>
  ```

- Start the Self-Service App to collect a specific task for a specific user.

  To make sure that usernames with spaces are dealt with correctly, you must replace the spaces with `+` signs. For URLs created from email templates, MyID can do this automatically if you use the correct syntax. For example, if your email template includes the following:

  ```
  Click <a href="myidssa:///jobid:%jobid+/un:{%logonName:URI}">Self-Service App</a>
  ```

  when the email message is created, it would become something similar to:

  ```
  Click <a href="myidssa:///jobid:256+/un:Jane+Smith">Self-Service App</a>
  ```

When you click a link in another application (for example, in a browser, in an email, or within a document) a warning message may be displayed. Click **Allow** or **Yes** (depending on the application) to open the link. You may also be able to deselect the **Always ask before opening this type of address** to prevent the warning message from appearing again.

Note: The installation program adds protocol registry entries for the current user only when installed by a non-administrator (`HKEY_CURRENT_USER`), or for all users when installed as an administrator (`HKEY_LOCAL_MACHINE`). The current user entry takes precedence if both are available. This may cause an issue if different users update the software to different versions in different locations. The registry locations are:

- `HKEY_CURRENT_USER\Software\Classes\myidssa`
- `HKEY_LOCAL_MACHINE\Software\Classes\myidssa`

4.6 Translating the user interface

The Self-Service App supports translation of the on-screen text to change the terminology used or to change the language completely.

Contact Intercede customer support for details, quoting reference SUP-71.

4.7 Logging

You can set up your Self-Service App to write debug information to a log file. You may need to provide this information to Intercede customer support.

See the *Windows clients* section in the ***Configuring Logging*** guide for details.

4.8 Job filtering

You may not want every client application to handle every job that is available for the user. For example, you may want your Self-Service Apps to handle only activation tasks. You can set up the web service to provide a customized list of jobs.

See the *Job filtering* section in the ***Web Service Architecture*** guide for details of setting up job filtering.

4.9 Specifying the target user

By default, the user identifier for the Self-Service App that is passed to the MyID server is based on the Windows logon name of the user. This is then matched against the SAM Account Name stored for the user in the MyID database.

The Self-Service App also passes the User Principal Name from the client and attempts to match this against the UPN stored for the user in the MyID database.

For information on changing how MyID behaves, see the *Specifying the target user* section in the ***Web Service Architecture*** guide for details.

4.10 Multiple instances

When the Self-Service App is launched, it first checks if another instance is already running and if there is, before progressing further, the newly launched instance requests that the original instance closes. The state of the original instance will determine whether or not the new instance will be allowed to progress:

- If the Self-Service App is running in the system tray, it will close without operator interaction and allow the new instance to progress.
- If the Self-Service App is displaying a window, in either wizard or interactive mode, the operator is prompted to confirm that they want to allow the current instance to close. If the operator confirms, the Self-Service App attempts to cancel any operations that are currently in progress before closing; if the operator does not confirm that they want to close the current instance, the new instance will not be allowed to progress.

  Exceptions to this behavior are:

  - If operator cancellation is disabled through the use of the `/hidecancel` argument in wizard mode, the operator will not be prompted and any new instances will not be allowed to progress.
  - If the Self-Service App is currently displaying an active dialog, the operator will not be prompted and any new instances will not be allowed to progress.

4.11 Signature validation

The Self-Service App performs signature validation at startup to ensure that all components are properly signed by Intercede and have not been tampered with. These checks are performed using the native Windows APIs, and may require the client to connect to the Internet to retrieve the latest Certificate Revocation Lists (CRLs) for revocation checks of the Intercede signing certificate. If the client is permanently running in an isolated environment without access to the Internet, the CRLs cannot be retrieved, which can cause signature verification to fail. Under these circumstances, you may see an error similar to the following:

```
Failed to verify signature for running application. Error Code: 128
```

This error usually indicates that the client is unable to perform its revocation checks; to continue, you must disable these checks by adding an option to the application configuration file.

To edit the configuration file:

1. On the client PC, shut down the Self-Service App.
2. Back up the `MyIDApp.exe.config` file. By default, this is in the following folder:

   ```
   C:\Program Files (x86)\Intercede\MyIDApp\Self Service Application\
   ```
3. Using a text editor, open the `MyIDApp.exe.config` file.

   Note: Make the changes to the config file exactly as shown. The case is important.
4. Edit the `value` parameter in the following line:

   ```
   <add key="ComponentVerificationSkipRevocationChecks"
   value="TRUE"></add>
   ```

   If this line does not exist, you can add it to the `<appSettings>` section.
5. Save the configuration file.
6. Restart the Self Service App.

When this option is enabled, the client performs all of its normal validation, but does not perform the revocation check. As the client does not need to retrieve the CRLs, it does not need to be connected to the Internet.

Note: This reduces the integrity of the signature validation, as the client will unable to determine if any of the certificates in the chain have been revoked since signing occurred – as such, you should ensure that the client's configuration file is modifiable only by users with administrative privileges.

4.11.1 Installing the required certificates for offline operation

If you have disabled the revocation check, you must also ensure that you have the appropriate certificates in the store of the client PC. On a PC with internet access, these certificates are obtained automatically; on a PC without internet access, you must obtain and install these manually.

To determine which certificates are required:

1. On a PC with internet access and the Self-Service App installed, locate the Self-Service App program file in Windows Explorer.

By default, this is:

```
C:\Program Files (x86)\Intercede\MyIDApp\Self Service
Application\MyIDApp.exe
```

2. Right-click the file, and from the pop-up menu select **Properties**.
3. Click the **Digital Signatures** tab, then select the **Intercede Ltd** item in the **Signature list** and click **Details**.

   This is the code signing certificate.
4. Click **View Certificate**, then click **Certification Path**.
   - The certificate at the top is the root certificate – you must ensure that this certificate is in the trusted root store of the client PC. This is normally carried out by Windows Update, but a completely unpatched and disconnected PC may not have this certificate.
   - In the certificate chain, all certificates except the top and bottom certificates are intermediate certificates – you must ensure that these certificates are present in the intermediate certificate store on the client PC. These certificates are normally downloaded on demand, but may not be available on a disconnected PC.
5. Close the certificate dialog, then select the timestamp certificate in the **Countersignatures** list and click **Details**.
6. Obtain the root and intermediate certificates for this certificate, as you did for the code signing certificate.

4.12 Self-service device update

The Self-Service App provides you with a mechanism to provide self-service device updates for your cardholders.

You can use this optional feature in the following situations:

- Rolling out a new certificate policy to end users (change credential profile, instruct users to use the feature to update their device).
- Letting end users recover additional encryption certificates that have been issued to other devices (configure the credential profile rules to allow automatic recovery of key history, instruct users to use the feature).
- Troubleshooting incorrect revocation of certificates (if a certificate has been incorrectly revoked, instruct users to use the feature to replace those certificates).
- Carrying out full reprovisioning of the device to erase and rewrite the device content.

4.12.1 Overview

When enabled, this feature works in the following way.

Once authenticated, MyID creates an update request for the authenticated device and applies it. This is a standard update request type that looks for differences in certificate assignment; where those differences are found, it applies the required changes. The update request is the same as would be applied if the request was generated by an operator or through the Lifecycle API.

If no differences are found, no actions are taken against the device. The attempt is still included in audit reports.

The feature applies to any supported device type on a Windows PC (smart card, USB token, VSC, Windows Hello).

Use cases – with the latest version of the credential profile assigned to the device as a reference, the self-service update feature:

- Replaces certificates that are revoked in MyID.
- Collects any new certificate policies that are available.
- Collect new additional identity certificate policies linked to the user in MyID.
- Removes any additional identity certificate policies that are no longer linked to the user in MyID.
- Collects key history updates based on the requirements of the credential profile.

Alternatively, you can configure the **Update My Device** feature to carry out a full reprovision of the device instead of a simple update; this erases and rewrites the device content, using the latest version of the credential profile. The device retains the same expiry date; new certificates may have longer expiration times than the original certificates, but these will not exceed the lifetime of the device itself.

4.12.2 Configuring MyID to allow self-service device updates

Important: This feature is not enabled by default. It may not be appropriate for all environments.

Configuration requires that two roles are configured to control access to the feature. The first makes the **Update My Device** option available before user authentication (to allow it to be easily located by self-service users) with the second used to control any further role-based restrictions on who is allowed to use the feature. If you require this feature to be available to any user of the Self-Service App, you are recommended to apply this to the Cardholder role, which is typically given to all users. Review which roles are allocated by default in your installation to determine if a more suitable role exists.

An external system configuration is also required. This step enables the UserSync process that creates the update request, and also allows you to determine whether the update performs a simple update or a full reprovision; however, note that all processing takes place on the MyID server when using the **UserSync_UpdateCardToLatest** or **UserSync_ReprovisionCard** mapping file.

To set up the role permissions for self-service device updates:

1. In MyID Desktop, from the **Configuration** category, select **Edit Roles**.
2. Add the **Update My Device** option from the **Cards** section to the **Default SSA User (981)** role.

   This allows the operation to be visible in the Self-Service App before the user authenticates.
3. Add the **Update My Device** option to the roles you want to be able to use the feature.

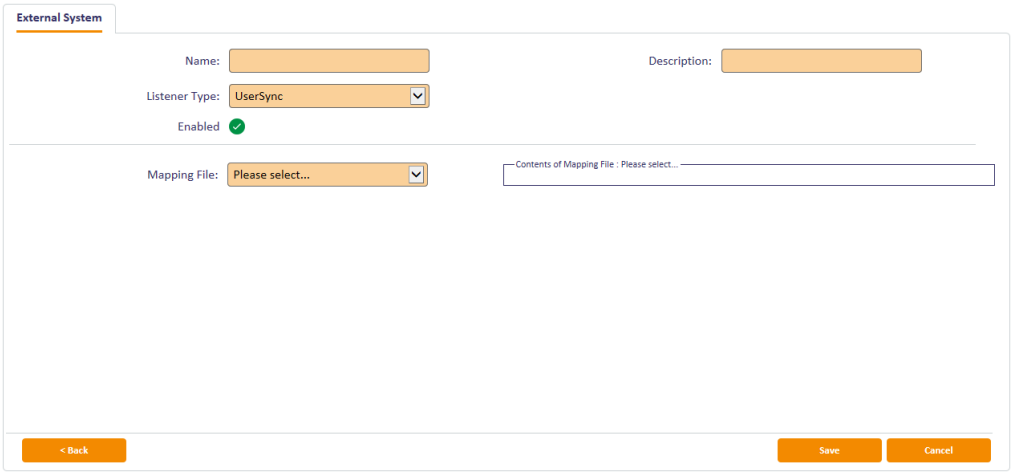
   For example, add the option to the **Cardholder** role.
4. Additionally, add the **Collect My Updates** option from the **Cards** section to the roles you want to be able to use the feature.

   For example, add the option to the **Cardholder** role.
5. Click **Save Changes**.

To configure the external system for the self-service device update feature:

1. In MyID Desktop, from the **Configuration** category, select **External Systems**.
2. Click **New**.
3. From the **Listener Type** drop-down list, select **UserSync**.

   The configuration details for the self-service device update feature appear.

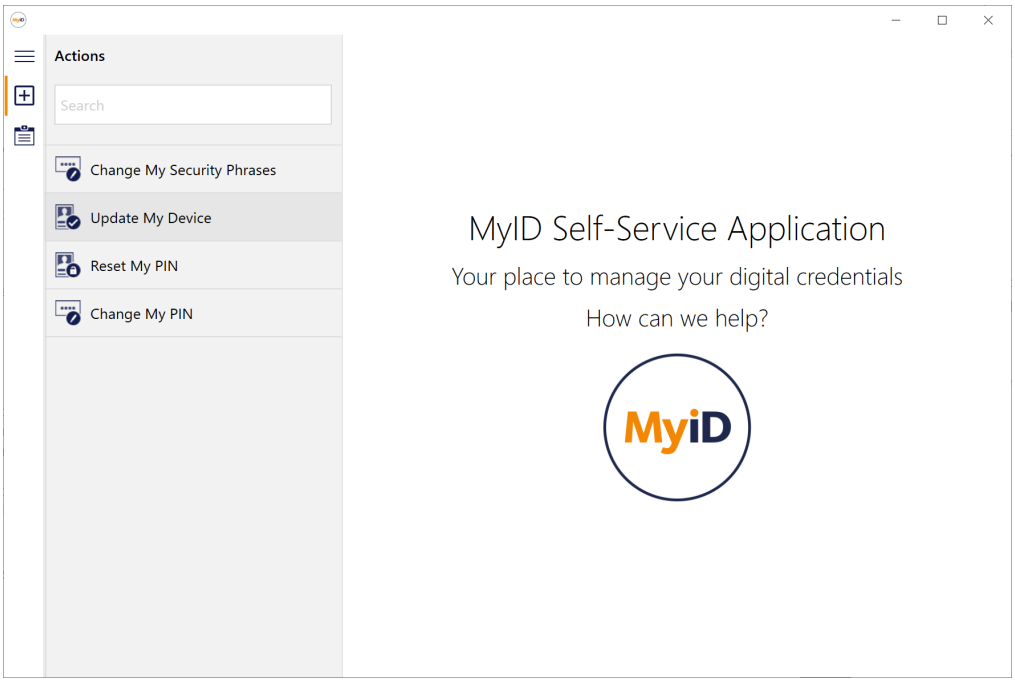


4. Type a **Name** and **Description** for the external system.
5. From the **Mapping File** drop-down list, select one of the following:
   - **UserSync_UpdateCardToLatest** – all self-service updates through the **Update My Device** option in the MyID Self-Service App carry out an update of the device to the latest version of the credential profile.
   - **UserSync_ReprovisionCard** – all self-service updates through the **Update My Device** option in the MyID Self-Service App carry out a full reprovision of the device.

   The mapping file contents are displayed in the Contents pane.
6. Click **Save**.

4.12.3 Running self-service device updates

You must deploy the Self-Service App to your users' workstations. Once installed, users can run the Self-Service App and select the **Update My Device** option:



Alternatively, you can provide a shortcut directly to the **Update My Device** option (which has ID 5013).

You can use the command line; for example:

```
MyIDApp.exe /opid:5013
```

or a hyperlink; for example:

```
myidssa:///opid:5013
```

4.12.4 Troubleshooting

If the user cannot authenticate to MyID (for example, if the certificate used for signing on the current credential profile is revoked or not set up, you cannot use this feature. This also applies if PIN authentication cannot be achieved; for example, if the PIN is locked.

The feature does not override or conflict with update requests created using other methods; for example, operator request, API request, or other MyID process such as certificate renewal. Those mechanisms are still valid and continue to work as before.

4.13 Filtering jobs for absent virtual devices

An end user may have active jobs that target virtual devices (for example, VSCs, Windows Hello) that do not exist on their current PC. You can configure the Self-Service App to hide these jobs; when you set this configuration flag, the Self-Service App checks through the list of available virtual devices, and if it cannot find the serial number that the job is targeting, it hides the job from the list.

To edit the configuration file:

1. On the client PC, shut down the Self-Service App.
2. Back up the `MyIDApp.exe.config` file. By default, this is in the following folder:

   `C:\Program Files (x86)\Intercede\MyIDApp\Self Service Application\`
3. Using a text editor, open the `MyIDApp.exe.config` file.

   Note: Make the changes to the config file exactly as shown. The case is important.
4. Edit the `value` parameter in the following line:

   `<add key="FilterJobsForAbsentVirtualDevices" value="TRUE"></add>`

   If this line does not exist, you can add it to the `<appSettings>` section.
5. Save the configuration file.
6. Restart the Self Service App.

5 Command line arguments

The Self-Service App command-line arguments determine the mode of the app, the authentication used, and so on.

For basic information about starting the app in its various modes, see section *4.5, Running the Self-Service App*.

This section contains a reference for the command line arguments.

You can specify the arguments at the Windows command prompt, as part of a Windows shortcut, or as part of a hyperlink.

5.1 Command line reference

Syntax:

```
MyIDApp.exe <display-mode> <credentials> <additional>
```

where:

- `<display-mode>` is one of the following:
  - `/h` – Displays the on-screen help text.
  - `/?` – Displays the on-screen help text.
  - `/w` – Starts the Self-Service App in wizard mode.
  - `/a` – Starts the Self-Service App in automation mode. You must also include a logon mechanism – `/lp` or `/lw` – for automation mode.
  - `/err` – Displays the list of exit codes.

  Note: You can specify only one display mode.
- `<credentials>` may include the following:
  - `/ssl` and `/sslsn:<value>` – For two-way SSL/TLS. See section *4.3.2, Two-way SSL/TLS*.
  - `/un:<value>` – The username for SSA to use.
  - `/lp` (automation mode only) – log on with passphrases. You must also include `/un` and `/pw` in the arguments.
  - `/lw` (automation mode only) – log on with Integrated Windows Logon. You cannot use `/lw` at the same time as `/lp`.
  - `/pw:<value>` (automation or wizard mode) – A security phrase or logon code to use for authentication. If you have multiple security phrases, you can specify `/pw` multiple times.
- `<additional>` may include the following:
  - `/jobid:<value>` (interactive and wizard mode only) – Launch a task by its MyID job ID. You can specify only one task. You cannot specify a job ID in automation mode.
  - `/opid:<value>` (interactive mode only) – Launch an Action by its MyID operation ID. You can currently use one of the following IDs:
    - `110` – Change My Security Phrases

  - `255` – Reset My PIN
  - `202` – Change My PIN
- `/vsconly` (wizard mode only) – Only virtual smart card jobs will be detected. All other jobs will be ignored.
- `/nopopup` (wizard mode only) – The Self-Service App runs with no pop-up notification balloon messages. All messages appear within the main window.
- `/hidenojobs` (wizard mode only) – Prevents the No Jobs Found page from being displayed.
- `/hidecancel` (wizard mode only) – Removes the Cancel button from any page that displays it, and removes the minimize, maximize, and close buttons. This allows you to prevent users from canceling operations.

  When you use `/hidecancel`, you must also use `/nopopup`.

  Note: The one exception where a cancel option still appears is if your system is not completely configured for secure issuance (see the **Device Security** page on the **Security Settings** workflow). Under these circumstances, the Self-Service App displays a warning that the system is not fully configured for this type of device; at this stage, it is possible to continue or cancel.
- `/hidewindow` (automation mode only) – Hides the Self-Service App window so that no user interface is displayed.
- `/processalljobs` (automation mode only) – Process all of the available jobs. Outputs a list of all jobs processed to the console.

5.2 Command line examples

Showing the on-screen help text:

```
MyIDApp.exe /?
```

or:

```
MyIDApp.exe /h
```

Showing the exit codes:

```
MyIDApp.exe /err
```

Starting the **Reset My PIN** action in interactive mode:

```
MyIDApp.exe /opid:255
```

Starting in wizard mode to collect a specific task (with job ID 123) with cancellation disabled:

```
MyIDApp.exe /w /jobid:123 /hidecancel /nopopup
```

Starting the Self-Service App in automation mode using security phrase authentication:

```
MyIDApp.exe /lp /un:user1 /pw:password1 /pw:password2 /a
```

Starting the Self-Service App in automation mode using integrated Windows logon:

```
MyIDapp.exe /lw /a
```

If usernames or security phrases contain spaces, surround them in double quotes:

```
MyIDApp.exe /w /un:"user 1" /pw:"pass word1" /pw:password2
```

Alternatively, for user names, you can replace spaces with + signs:

```
MyIDApp.exe /w /un:user+1 /pw:"pass word1" /pw:password2
```

6 Application exit codes

When the Self-Service App stops running, it sets an exit code that you can use to determine what happened in the last run of the application; for example, if the task succeeded or failed.

This section provides a reference for the possible exit codes.

6.1 Success exit codes

Success exit codes represent states where the application did not encounter any unforeseen problems.

Code	Description	Explanation
0	OK	Task completed, no errors reported.
1	No jobs available	No jobs were found for the current user when their details were submitted to the web service. Wizard mode only.
2	Closed by other instance	The Self-Service App was shut down by another copy of the Self-Service App.

6.2 Failure exit codes

Failure exit codes represent states where the application encountered problems or needs to provide details to the user for alterations to the startup requirements of the application.

Code	Description	Explanation
100	Abort	Either the server issued an abort command, in which case the client terminated its operation, or the client encountered an error which caused it to terminate its operation early. If logging is enabled, evidence of the operation should be discoverable near the end of the log file.
101	An error occurred, please refer to your documentation to enable logging	Either the server raised an error command, in which case the client terminated its operation, or the client encountered an error which caused it to terminate its operation early. If logging is enabled, evidence of the operation should be discoverable near the end of the log file.
102	Device is already issued	When details of the current device were submitted to the web service the determination was that the device was already issued. Therefore it is not possible to issue again to that device until it is canceled. Wizard mode only.
107	Unable to connect to the MyID Web Service	There was an error attempting to connect to the MyID Web Service. This could be due to network connectivity issues or server errors.

Code	Description	Explanation
108	An unexpected error occurred, please refer to your documentation to enable logging	An unknown error was encountered. If logging is enabled, evidence of the issue should be discoverable in the log.
109	The specified command switches are invalid	A command line argument was provided which is not part of the set of recognized command line arguments.
110	Authentication failed	Using the authentication details provided, the application could not authenticate the user against the MyID Web Service.
112	User terminated the process	The user elected to terminate the MyID Self-Service app. Wizard mode only.
113	Client Components are not installed	Detection of the required MyID client components failed. Reinstall the Self-Service App.
115	Incorrect or duplicate command switch	A command line argument was provided but was either incorrect or a duplicate of an existing command line argument.
116	Component Verification Check Failed	The Self-Service App's components are not registered correctly or are not present.
117	Out Of Date Application	The version of the Self-Service App you are using is out of date.
118	Application Already Running	You have attempted to run the Self-Service App when it is already running.
120	Two-way SSL/TLS has been requested but the server URL is unsecured.	You have specified `/ssl` on the command line, but the server does not use HTTPS.
124	When using automation mode, the card that is needed to perform the action is unavailable.	Make sure the card is available.
125	When collecting multiple jobs using `/processalljobs` an error has occurred.	Check the console for errors.

6.3 Retrieving application exit codes

6.3.1 Displaying exit codes

You can display the list of exit codes by running the following at the command line:

```
MyIDApp.exe /err
```

Note: Not all codes listed are currently used. Some codes relating to previous versions of the Self-Service App will not appear.

To view the exit code from the last run of the application, in a batch file include the following:

```
echo %errorlevel%
```

In PowerShell, you can use `$LASTEXITCODE`.

6.3.2 Batch files

If you execute the app from a Windows batch file, you can capture the application exit codes and display them to the user.

For example, copy the following text into a `.bat` file:

```
@echo off
start /wait MyIDApp.exe /w /un:"Simple User" /pw:"pass1" /pw:"pass2"
echo Exit Code: %errorlevel%
```

You can use the tables in section *6.1, Success exit codes* and section *6.2, Failure exit codes* to determine the meaning of the code and display this to the end user; for example:

```
@echo off
start /wait MyIDApp.exe /un:"Simple User" /pw:"pass1" /pw:"pass2"
IF %errorlevel% EQU 0 (echo Exit code 0 OK )
IF %errorlevel% EQU 1 (echo Exit code 1 No jobs available )
IF %errorlevel% EQU 2 (echo Exit code 2 Closed by other instance )
IF %errorlevel% EQU 101 (echo Exit code 101 An error occurred, please refer
to your documentation to enable logging )
IF %errorlevel% EQU 102 (echo Exit code 102 Device is already issued )
```

6.4 Console output

When passing the `/processAllJobs` command line, the Self-Service App outputs the status of the processed jobs to the console. For example, if there were three jobs processed it would appear as:

```
Self Service Application Automation Mode
Job <ID> - <Application Return Code>
Job <ID> - <Application Return Code>
Job <ID> - <Application Return Code>
```

In this example:

- `<ID>` would be the ID of the job.
- `<Application Return Code>` would represent the status of this job.

The application exit code would be either:

- `0` if all jobs have been collected successfully.
- `125` if one of the jobs have failed.

Example:

```
Self Service Application Automation Mode
Job 256 - 0
Job 267 - 101
Job 278 - 0
```

If there was a terminal error that stopped processing of the jobs it would appear as:

```
Self Service Application Automation Mode
Application terminated with <Application Return Code>
```

- `<Application Return Code>` would be the numeric value of the exit code. The application exit code would be the same value.

The following would be output if no jobs are found:

```
Self Service Application Automation Mode
0 Jobs found
```

- The application exit code would be `1` to state that no jobs have been found.

7 Accessibility

The Self-Service App has been designed with accessibility in mind, and includes support for both Display Scaling and High Contrast Mode in Windows.

Improvements for operators using screen readers (for example, Microsoft Narrator) have been added, including support for live regions – this feature allows an operator to be notified when important regions of the user interface are updated.

Note: The live regions feature must be supported by the version of the screen reader software you are using.

Additionally, you can configure the Self-Service App so that controls will not be focused automatically when a screen changes, meaning that screen readers will not be interrupted by an unexpected focus change.

To disable automatic focusing of controls:

1. On the client PC, back up the `MyIDApp.exe.config` file. By default, this is in the following folder:

   `C:\Program Files (x86)\Intercede\MyIDApp\Self Service Application\`

2. Using a text editor, open the config file.

   Note: Make the changes to the config file exactly as shown. The case is important.

3. Inside the `<appSettings>` node, add the following:

   ```
   <add key="DisableAutoFocus" value="TRUE"></add>
   ```

4. Save the configuration file.

When using a screen reader, you may also want to increase the timeout on the Self-Service App to allow the screen reader software more time to read out the text on the screen; for example, you may want to set the timeout to 120 seconds. See section *4.2, Timeout* for details of changing the default timeout value.

8 Troubleshooting and known issues

This section contains information about troubleshooting and known issues.

8.1 Troubleshooting

You may experience the following error, which does not present an error code:

- **Problems collecting or updating cards**

  You may experience problems when collecting or updating cards, which may result in a message similar to:

  ```
  One of the certificates that have been requested for you has failed to issue. Please contact your administrator.
  ```

  Note that the certificate may have issued correctly even though the card update has failed.

  Try increasing the **Certificate Refresh Threshold** option on the **Certificates** tab of the **Operation Settings** workflow to a higher value; for example, `45`.

8.2 Known issues

- **Cannot run the Self-Service App if another MyID application is running**

  You cannot run the Self-Service App if you have MyID Desktop or the Self-Service Kiosk running, or if you have an Internet Explorer window open at the MyID web page. This is because both the Self-Service App and other MyID applications conflict with each other over card transaction locking.

- **Cannot choose a card in the Self-Service App**

  When performing a task, if you have more than one card reader, the Self-Service App does not allow you to select which reader to use. If you have unissued smart cards in more than one reader, and are collecting a card issuance job, the Self-Service App will select one of the cards without any user intervention.

  Note: You can select the card you want to use when carrying out actions.

- **Capturing command-line output**

  Running the Self-Service application with a parameter such as `/h/?` or `/err` runs on a new line in the command window. If you are trying to capture the output, this may cause problems. As a workaround, you can use the start command. For example, to capture the help text to a text file called `output.txt`:

  ```
  start /wait myidapp.exe /h > output.txt
  ```

- **MyID icon still visible in system tray when application is closed**

  When the Self-Service application has finished running, the application icon may still be visible in the Windows system tray. This is due to a known issue in Windows. Moving the mouse pointer over the icon causes the icon to disappear.

- **Enter PIN twice for card update and certificate renewal jobs**

  If you have terms and conditions enabled for the credential profile, and the **Terms and Conditions During Device Update** configuration option is set to `Yes`, you are required

to enter the PIN both before the terms and conditions are displayed and after accepting the terms and conditions.