MyID PIV

Version 12.11

Configuring Logging

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

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. ---

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 how to set up logging for various MyID® systems, including:

- MyID Desktop, Self-Service App, Self-Service Kiosk and the MyID Client Service – see section *2*, *Windows clients*.
- MyID Image Capture – see section *3*, *MyID Image Capture*.
- MyID Windows Integration Service (WSVC) – see section *4*, *MyID Windows Integration Service*.
- MyID Identity Agent – see section *5*, *MyID Identity Agent*.
- MyID Client Components – see section *6*, *MyID Client Components*.
- MyID Web Service Architecture – see section *7*, *MyID Web Services*.
- MyID Notifications Service – see section *7*, *MyID Notifications Service*.
- MyID REST and Authentication Web Services – see section *8*, *MyID REST and authentication web services*.
- Other MyID web services – see section *9*, *Other MyID web services*.
- MyID server components – see section *10*, *Server components*.

Important: Use this document only in conjunction with advice from customer support. Log files are not intended to be readable by customers, and may require expert analysis. Do not leave logging switched on when you do not need to; the files may become very large and may impact performance. Log files may also contain sensitive information. Always back up your system before making any changes; switching on logging may require the manual editing of configuration files or the system registry.

2 Windows clients

This chapter contains information on configuring logging for the MyID Windows clients:

- MyID Desktop.
- MyID Client Service.
- Self-Service App.
- Self-Service Kiosk.
- MyID Client WebSocket Service.

  Note: The MyID Client WebSocket Service uses a different method of configuring logging to the other Windows clients; see section *2.1*, *MyID Client WebSocket Service* for details.

To enable logging for these client applications, you must edit the application's configuration file.

- For MyID Desktop, the configuration file is:

  `MyIDDesktop.exe.config`

  and is located in the following folder by default:

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

- For the MyID Client Service, the configuration file is:

  `MyIDClientService.dll.config`

  and is located in the following folder by default:

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

- For the Self-Service App, the configuration file is:

  `MyIDApp.exe.config`

  and is located in the following folder by default:

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

- For the Self-Service Kiosk, the configuration file is:

  `MyIDKiosk.exe.config`

  and is located in the following folder by default:

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

To enable logging:

1. On the client PC, back up the configuration file.
2. Open the configuration file in a text editor.
3. Add the following to the `<appsettings>` section of the configuration file:

   ```
   <add key="EnableLogging" value="true"/>
   ```

   Set the value to true to enable logging, or false to disable logging.

   By default, the log is written to the following folder:

   ```
   %LocalAppData%\Intercede\Logs
   ```

   If you want to specify a different location, add the following to the `<appsettings>` section of the configuration file:

   ```
   <add key="LogDirectory" value="C:\Logs"/>
   ```

   Set the value to the folder where you want to write the logs.
4. Save the configuration file.
5. Restart the application.

Note: If you are using the MSIX installer for your client applications, by default the logs are written to the following folder:

```
%LocalAppData%\Intercede
```

If the logs are not displayed in this folder as expected, use the `LogDirectory` setting to specify a different folder.

2.1 MyID Client WebSocket Service

You can configure logging for the MyID Client WebSocket Service by editing the service's `appsettings.json` file. By default, this file is installed in the following location:

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

```
{
  "Logging":  {
    "EventLog":  {
      "LogLevel":  {
        "Default":  "None"
      }
    }
  },
  ...
}
```

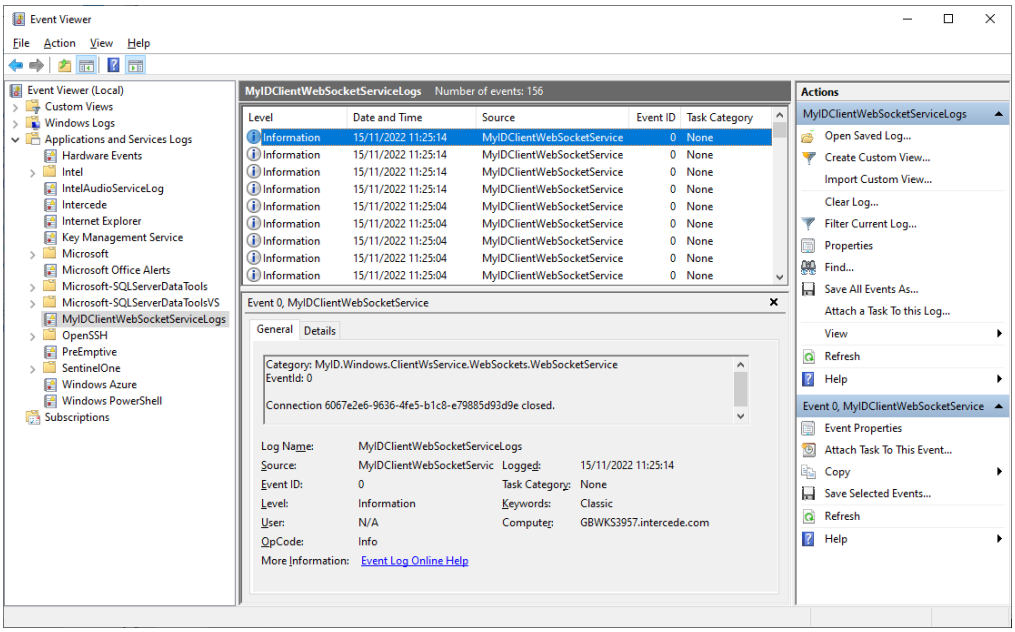
To configure logging, set the `Logging > EventLog > LogLevel > Default` value to one of the following:

- `None`
- `Information`
- `Warning`
- `Error`
- `Critical`
- `Debug`
- `Trace`

The options are listed in ascending level of detail.

Note: The `Trace` level causes all WebSocket messages and session ID registrations to be included in the logs, which may include sensitive information; do not use this level in a production environment.

Log information is displayed in the Windows Event Viewer, under **Applications and Services Logs > MyIDClientWebSocketServiceLogs**:



3 MyID Image Capture

To set up logging for the MyID Image Capture component, create a text file called `LogConfig.xml` and add the following to it:

```
<log4net>
  <root>
    <level value="ALL" />
    <appender-ref ref="RollingFileAppender" />
  </root>
  <appender name="RollingFileAppender" type="log4net.Appender.RollingFileAppender">
    <file type="log4net.Util.PatternString" value="%property {LogFile}"/>
    <appendToFile value="true"/>
    <datePattern value="yyyyMMdd"/>
    <rollingStyle value="Size"/>
    <maxSizeRollBackups value="10"/>
    <maximumFileSize value="10000KB"/>
    <layout type="MyIDApp.Utility.XmlLayoutSchemaLog4j">
      <locationInfo value="true"/>
    </layout>
  </appender>
</log4net>
```

Copy this file into the following folder on the client PC:

`%UserProfile%\AppData\LocalLow\Intercede\ImageCapture\`

Note: This folder is created the first time you run the Image Capture component.

The Image Capture component creates the following log files:

- `MyIdImageCaptureActiveXLog.xml`

  This corresponds to the ActiveX control.

- `MyIdImageCaptureComLog.xml`

  This corresponds to the COM object that the ActiveX control invokes to launch the Image Capture control in its own process.

- `MyIdImageCaptureLog.xml`

  This corresponds to the WPF control itself, which is invoked by the COM object.

4 MyID Windows Integration Service

To set up logging for the Windows Integration Service (WSVC), open the `Log.config` file, which is installed to the following folder by default:

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

Set the following options in the file:

- `configuration/log4net/appender/file` – set the `value` to the path and filename of the log file you want to use.

  For example:

  ```
  <file value="C:\logs\MyID.log" />
  ```

- `configuration/log4net/root/level` – set the `value` to `DEBUG`.

  For example:

  ```
  <level value="DEBUG" />
  ```

  To switch off logging, set this value to OFF.

  For example:

  ```
  <level value="OFF" />
  ```

5 MyID Identity Agent

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`.

6 MyID Client Components

The MyID Client Components provide logging for a variety of the components in the UMC package.

You can set up logging for the following components individually:

- ApduScript
- CanonCapture
- ClientVersion
- CSP COM
- CSPCertEnroll
- DataExchange
- DirectAPISmartCard
- ECardPrintX
- Edefice_OCR
- EdeficeSmartCard
- Envelope COM
- eSCardCOM
- FileUtils
- MifareCom
- ScannerCapture
- SmartcardKeypair
- WHfB

Note: Not all of these components may be available on your system, depending on which edition of MyID you are using.

To set up logging for a component:

1. Set the following in the client PC's registry:

   `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Intercede\Edefice\Trace`

   If the `Trace` key does not exist, you must create it.

2. In the `Trace` key, create a DWORD value with the name of the component from the list above; for example, `EdeficeSmartCard`. Set the value to `1` to enable logging, and `0` to disable logging.

   Note: For the `WHfB` (Windows Hello for Business) component, you must set the value to `9` to enable logging, as this component supports only parameter-level tracing.

3. In the `Trace` key, create a key with the name of the component; for example, `EdeficeSmartCard`. Within this key, create a string value called Location and set this to the full path of the file to which you want to send the log information.

Note: If you are on the server, you must ensure that the MyID named COM user has the necessary permissions to create and write to the log file. You can create a file then give the user write permissions if you prefer not to give the user create permissions.

Note: You must ensure that all users can write to the location; set the permissions of this folder to be **Everyone - Full control**.

Important: Disable the logging when you have completed diagnosing the issues, as the log file may become very large.

7 MyID Web Services

You can set up logging for the following web services, included as part of the MyID Web Service Architecture:

- `MyIDDataSource`
- `MyIDProcessDriver`

To set up logging:

1. In a text editor, open the `Log.config` file for the component you want to log.

   For `MyIDDataSource`, this is:

   ```
   C:\Program Files\Intercede\MyID\SSP\MyIDDataSource\Log.config
   ```

   For `MyIDProcessDriver`, this is:

   ```
   C:\Program Files\Intercede\MyID\SSP\MyIDProcessDriver\Log.config
   ```

2. Set the value of the `file` node to the output location; for example:

   ```
   <file value="C:\logs\MyIDDataSource.log" />
   ```

3. Replace the following line:

   ```
   <level value="OFF" />
   ```

   with:

   ```
   <level value="All" />
   ```

4. Save the file.

Note: You must ensure that the MyID web service user has the necessary permissions to create and write to the log file. You can create a file then give the user write permissions if you prefer not to give the user create permissions.

The log is set to a maximum of 60MB, split over six rolling files.

Important: The log files may contain personal data, including names and addresses. Make sure you delete these logs as soon as possible.

7 MyID Notifications Service

You can configure logging for the MyID Notifications Service using the `Log.config` file in the Notifications folder.

To set up logging:

1. In a text editor, open the `Log.config` file for the notifications component.

   By default, this file is in the following location:

   ```
   C:\Program Files\Intercede\MyID\Components\MyID.Notifications.Net
   ```

2. Set the value of the `file` node to the output location; for example:

   ```
   <file value="C:\logs\NotificationsService.log" />
   ```

3. Set the value of the `level` node to the level of detail that you want to be logged. In order of least to most information saved to the log file, the options for this value are:

   ```
   OFF
   FATAL
   ERROR
   WARN
   INFO
   DEBUG
   ALL
   ```

   For example:

   ```
   <level value="ERROR" />
   ```

4. Save the file.

Note: You must ensure that the MyID web service user has the necessary permissions to create and write to the log file. You can create a file then give the user write permissions if you prefer not to give the user create permissions.

Important: The log files may contain personal data. Make sure you delete these logs as soon as possible.

8 MyID REST and authentication web services

MyID provides web services for the MyID Operator Client to communicate with and authenticate to the web server, as well as external authentication services that are used, for example, for AD FS authentication.

You can set up logging for the following web services:

- `rest.core`
- `web.oauth2`
- `web.oauth2.ext`
- `AdfsAuth`

To set up logging:

1. In a text editor, open the `Log.config` file for the web service you want to log:

   `C:\Program Files\Intercede\MyID\<web service name>\Log.config`

   where `<web service name>` is the name of the web service.
2. Set the value of the `file` node to the output location; for example:

   `<file value="C:\logs\rest.core.log" />`
3. Edit the following line:

   `<level value="OFF" />`

   and replace the `OFF` value with one of the following:

   `ALL`

   `DEBUG`

   `INFO`

   `WARN`

   `ERROR`

   `FATAL`

   These error levels generate different levels of detail in the log, from most (`ALL`) to least (`FATAL`). To switch logging off altogether, set the value back to `OFF`. For diagnosing issues, you are recommended to set the level to `ERROR`; this level provides useful information without providing too much additional detail that can mask the information you need.

   Important: Log levels `ALL` and `DEBUG` log all COM calls including parameters sent to and from the MyID application server. This produces a high volume of log information and may contain personal data. Reduce the log level, or set it to `OFF`, as soon as possible once you have obtained the relevant logging details.
4. Save the file.

Note: You must ensure that the MyID web service user has the necessary permissions to create and write to the log file. You can create a file then give the user write permissions if you prefer not to give the user create permissions.

The log is set to a maximum of 60MB, split over six rolling files.

Important: The log files may contain personal data, including names and addresses. Make sure you delete these logs as soon as possible.

In addition to this logging, you can also set up logging for the Microsoft components used by these web services; see section *8.1*, *Logging Microsoft components* for details.

8.1 Logging Microsoft components

The logging for the REST and authentication services controlled by the `Log.config` file reports activity within MyID components; however, there may be instances within those logs where errors are reported from Microsoft components. If you require further information about these Microsoft errors, you can add extra logging for these components to provide information on issues deeper in the stack (for example, JWT validation failures, or ASP.net infrastructure issues).

To enable this logging:

1. In a text editor, open the `appsettings.Production.json` file for the web service.

   `C:\Program Files\Intercede\MyID\<web service name>\appsettings.Production.json`

   where `<web service name>` is the name of the web service.

   These files are the override configuration files for the `appsettings.json` files for the web services. If these files do not already exist, you must create them in the same folder as the `appsettings.json` files.

2. Add an entry for logging Microsoft components.

   For example:

   ```
   {
       "Logging":  {
           "LogLevel":  {
               "Microsoft": "Error"
           }
       }
   }
   ```

   Note: If you already have `appsettings.Production.json` files, add the `Logging:LogLevel:Microsoft` section to the existing file. The above example assumes that there are no other entries in the file.

   This example adds logging information from all Microsoft components at `Error` level. You are recommended to use this level for diagnosing issues; this level provides useful information without providing too much additional detail that can mask the information you need.

   The supported log levels are different from the values in the `Log.config` file. From most detail to least, the options are:

   `Trace`

   `Debug`

   `Information`

   `Warning`

   `Error`

```
Critical

None
```

Note: You must make sure that the `Log.config` file is configured to produce a log (see section *8*, *MyID REST and authentication web services*), or the additional Microsoft logging information will not be logged.

For more information, search for the *Logging in .NET Core and ASP.NET Core* article on the Microsoft website.

9 Other MyID web services

You can set up logging for the following web services:

- Credential Web Service (CWS) – see section *9.1*, *Logging for the Credential Web Service*.
- Device Management API (DWS) – see section *9.2*, *Logging for the Device Management API*.
- Lifecycle API (MyIDEnroll) – see section *9.3*, *Logging for the Lifecycle API*.

These web services use the same method of configuring logging.

Note: For each web service, you must ensure that the MyID web service user has the necessary permissions to create and write to the log file. You can create a file then give the user write permissions if you prefer not to give the user create permissions.

9.1 Logging for the Credential Web Service

To set up logging for the Credential Web Service (CWS), copy the following into a text file called `Log.Config` in the following folder:

```
C:\Program Files\Intercede\MyID\SSP\CredentialWebService
```

```
<configuration>
  <configSections>
    <section name="log4net"
type="log4net.Config.Log4NetConfigurationSectionHandler, log4net" />
  </configSections>
  <log4net>
    <appender name="RollingFileAppender"
type="log4net.Appender.RollingFileAppender">
      <file value="CredentialWebService.log" />
      <appendToFile value="true" />
      <rollingStyle value="Size" />
      <maxSizeRollBackups value="5" />
      <maximumFileSize value="10MB" />
      <staticLogFileName value="true" />
       <layout type="CredentialWebService.Web.MyXmlLayout" />
    </appender>
    <root>
      <level value="ALL" />
      <appender-ref ref="RollingFileAppender" />
    </root>
  </log4net>
</configuration>
```

9.2 Logging for the Device Management API

To set up logging for the Device Management API web service (DWS), copy the following into a text file called `Log.Config` in the following folder:

```
C:\Program Files\Intercede\MyID\SSP\DeviceManagementAPI
```

```
<configuration>
  <configSections>
    <section name="log4net"
type="log4net.Config.Log4NetConfigurationSectionHandler, log4net" />
  </configSections>
  <log4net>
    <!-- ConsoleAppender -->
    <appender name="ConsoleAppender"
type="log4net.Appender.ConsoleAppender">
      <layout type="log4net.Layout.PatternLayout">
        <conversionPattern value="%-4timestamp [%thread] %-5level -
%message%newline%newline" />
      </layout>
    </appender>
    <!-- Rolling file appender to ProcessDriver.log-->
    <appender name="RollingFileAppender"
type="log4net.Appender.RollingFileAppender">
      <file value="DeviceManagementAPI.log" />
      <appendToFile value="true" />
      <rollingStyle value="Size" />
      <maxSizeRollBackups value="5" />
      <maximumFileSize value="10MB" />
      <staticLogFileName value="true" />
      <!--<layout type="log4net.Layout.PatternLayout">
        <conversionPattern value="%-4timestamp [%thread] %-5level -
%message%newline%newline" />
      </layout>-->
      <layout type="DeviceManagementAPI.MyXmlLayout" />
    </appender>
    <!-- Set root logger level to INFO and its only appender to A1 -->
    <root>
      <level value="ALL" />
      <!-- uncomment to see logging to output window -->
      <!-- <appender-ref ref="ConsoleAppender" />-->
      <appender-ref ref="RollingFileAppender" />
    </root>
  </log4net>
</configuration>
```

9.3 Logging for the Lifecycle API

To set up logging for the Lifecycle API web service (MyIDEnroll), copy the following into a text file called `Log.Config` in the following folder:

```
C:\Program Files\Intercede\MyID\Web\MyIDEnroll
```

```
<configuration>
  <configSections>
    <section name="log4net"
type="log4net.Config.Log4NetConfigurationSectionHandler, log4net" />
  </configSections>
  <log4net>
    <appender name="RollingFileAppender"
type="log4net.Appender.RollingFileAppender">
      <layout type="MyIDEnroll.LogLayout" />
      <file value="MyIDEnroll.log" />
      <appendToFile value="true" />
      <rollingStyle value="Size" />
      <maxSizeRollBackups value="5" />
      <maximumFileSize value="10MB" />
      <staticLogFileName value="true" />
    </appender>
    <root>
      <level value="ALL" />
      <appender-ref ref="RollingFileAppender" />
    </root>
  </log4net>
</configuration>
```

10 Server components

You can set up logging for a variety of server components. The method for configuring logging depends on the component you want to log.

- Logging using the registry method – see section *10.1*, *Registry logging*.
- Logging using the Log4Net method – see section *10.2*, *Log4Net*.
- Logging for Entrust JTK – see section *10.3*, *Entrust JTK logging*.
- Logging for the Dal4Net component – see section *10.4*, *Dal4Net logging*.

10.1 Registry logging

You can use the registry method of configuring logging for the following components:

- `AccessProfileImport`
- `ADDeletionSync`
- `ADDeletionSync`
- `AdjudicationEquifax`
- `AdjudicationOPM`
- `AMAGPACSConnector`
- `ASyncImport`
- `BOL_Authentication`
- `BOL_Certificates`
- `BOL_Core`
- `BOL_DeviceManagement`
- `BOL_DevicePolicy`
- `BOL_Devices`
- `BOL_ImportFromCard`
- `BOL_Jobs`
- `BOL_LDAP`
- `BOL_Notifications`
- `BOL_People`
- `CardScriptExtensions`
- `CBPACSConnector`
- `CertificateRevocationConnector`
- `CertificateSrv`
- `eActivIDSDSProcessor`
- `eBureauSrv`
- `ECardPrintX`

- eConfiguration
- eCS
- Edefice_CS
- Edefice_DAL
- EdeficeBOL_PKI
- EdeficeSmartCard
- eDirectory
- eEMVDataProcessor
- eJobMaintenanceProcessor
- eJobServer
- eKeySrv
- eKeySrvPool
- Entrust_Admin
- EntrustJTKConnector
- ePkiConfig
- eStaleJobProcessor
- GEFCPACSConnector
- GEPACSConnector
- GPOBureauMessage
- HSMTestUtility
- ImportProcessor
- JobBatch
- LUNAKeySrv
- MicrosoftConnector
- MicrosoftKeyStore
- MifareCom
- NCKeySrv
- OfflineRevocationConnector
- OpenPlatformSecurity
- PivDataProcessor
- PivTransport
- PreciseConnector
- ResyncByCounter
- SecugenConnector

- SymantecLH
- SymantecMPKIHelper
- SunOne
- THNGHooks
- Unicert

Note: Not all of these components may be available on your system, depending on which edition of MyID you are using.

You can also set up logging for any component that ends BureauTransport; for example, GenBureauTransport.

You can set up logging for the Notifications component, but this is the older component – for the current Notifications system, see section *10.2*, *Log4Net*.

You can set up logging for the eSCardCOM component, but only after installing a debug version of the DLL. For example, for MyID PIV 9.0 SP1, the diagnostic patch D901MP316 is available.

To set up logging for a component:

1. Set the following in the MyID application server's registry:

   HKEY_LOCAL_MACHINE\SOFTWARE\Intercede\Edefice\Trace

   If the Trace key does not exist, you must create it.
2. In the Trace key, create a DWORD value with the name of the component from the list above; for example, TPMManager. Set the value to 1 to enable logging, and 0 to disable logging.
3. In the Trace key, create a key with the name of the component; for example, TPMManager. Within this key, create a string value called Location and set this to the full path of the file to which you want to send the log information.

Note: You must ensure that the MyID named COM user has the necessary permissions to create and write to the log file. You can create a file then give the user write permissions if you prefer not to give the user create permissions.

Important: Disable the logging when you have completed diagnosing the issues, as the log file may become very large. Alternatively, you can configure a maximum file size and a backup location for your log files; see section *10.1.1*, *Maximum log size and backups*.

10.1.1 Maximum log size and backups

You can configure a maximum log file size. When MyID attempts to write to the log file, if the current file size exceeds the maximum configured, MyID clears the log and starts again. Optionally, you can configure MyID to back up the old log file before clearing it.

Note: These settings are applied for all modules that use this logging method.

To configure a maximum log size and backups:

1. Within the `Trace` key in the MyID application server's registry, create a DWORD value named `LogFileSize`.

   Set the value to the maximum size (in KB) of the log file.

2. Within the `Trace` key in the MyID application server's registry, create a DWORD value named `CreateBackups`.

   Set the value to `1` to enable backups, and `0` to disable backups.

3. Within the `Trace` key in the MyID application server's registry, create a String value named `BackupLocation`.

   Set the value to the name of the folder to which you want to copy the backup log files.

You can specify a backup location on a file server rather than on the local application server; however, you must ensure that the MyID COM user has write access to this folder. Backup log files are copied to this folder with an appended timestamp in their filenames.

10.1.2 Bureau logging

Logging for the Bureau server components is a variation on the standard registry method.

To set up logging for the bureau components:

1. Set the following in the MyID application server's registry:

   `HKEY_LOCAL_MACHINE\SOFTWARE\Intercede\Edefice\Trace`

   If the `Trace` key does not exist, you must create it.

2. In the `Trace` key, create the following keys:

   - `eBureauSrv`
   - `Boewe`

3. Inside each of the above keys, create a string value called `Logfile` and set this to the full path of the file to which you want to send the log information.

Note: You must ensure that the MyID named COM user has the necessary permissions to create and write to the log file. You can create a file then give the user write permissions if you prefer not to give the user create permissions.

Important: Disable the logging when you have completed diagnosing the issues, as the log file may become very large.

10.2 Log4Net

You can use the Log4Net method of configuring logging for the following components:

- EJBCA connector
- SymantecMPKI connector
- DigiCert ONE connector
- MyIDMailer

When you switch on logging, it generates log information for all of the above components. You cannot decide to log individual components.

To set up logging for these components, copy the following into a text file called `Log.Config`:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <configSections>
    <section name="log4net"
type="log4net.Config.Log4NetConfigurationSectionHandler, log4net" />
  </configSections>
  <log4net>
    <appender name="MyIdLogFile"
type="log4net.Appender.RollingFileAppender">
      <file value="c:\Logs\log.txt" />
      <appendToFile value="true" />
      <lockingModel type ="log4net.Appender.FileAppender+MinimalLock" />
      <maxSizeRollBackups value="10" />
      <maximumFileSize value="32Mb" />
      <rollingStyle value="Size" />
      <staticLogFileName value="true" />
      <layout type="log4net.Layout.PatternLayout">
        <header value="[Header] "/>
        <footer value="[Footer] "/>
        <conversionPattern value="%date [%thread] %-5level %logger -
%message %newline" />
      </layout>
    </appender>
    <root>
      <level value="ALL" />
      <appender-ref ref="MyIdLogFile" />
    </root>
  </log4net>
</configuration>
```

Copy the file to the Windows `System32` folder on the MyID application server.

To change the path of the log file, edit the `Log.Config` file in an text editor and update the following line:

```
<file value="c:\Logs\log.txt" />
```

Note: You must ensure that the MyID named COM user has the necessary permissions to create and write to the log file. You can create a file then give the user write permissions if you prefer not to give the user create permissions.

To disable logging, delete the `Log.Config` file from your Windows `System32` folder.

Note: Switch off logging when it is no longer needed, or you could end up with a large amount of files. The `maximumFileSize` option determines the maximum file size, but the logging will create additional files when this limit is reached.

It is important to note that this logging generates entries from all MyID components that use this form of logging.

10.3 Entrust JTK logging

You can enable logging for the Entrust JTK component. On the MyID application server, open regedit and browse to the registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Intercede\Edefice\Connector\EntrustJTK
```

This key contains the following values:

- `JavaLocation` – an existing value containing the path to the MyID Java components.
- `LogLevel` – a DWORD value containing the logging level to use.
- `LogFile` – a String value containing the path of the JTK log file.

If the `LogLevel` or `LogFile` entries do not exist, you can create them.

For example:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Intercede\Edefice\Connector\EntrustJTK]
"JavaLocation"="C:\\Program Files\\Intercede\\MyID\\Components\\Java"
"LogFile"="c:\\logs\\jtklog.log"
"LogLevel"=dword:00000004
```

In this example, the `LogFile` has been set to the logs folder on drive C:, and in a file named `jtklog.log`.

Note: Do not use the same log file as you are using for any other logging.

The logging level is set to `4`. According to the Oracle documentation for logging, the available logging levels are:

- `0` – off
- `1` – basic
- `2` – network, cache, and basic
- `3` – security, network and basic
- `4` – extension, security, network and basic
- `5` – LiveConnect, extension, security, network, temp, basic, and Deployment Rule Set

The above example will log extension, security, network, and basic calls.

To disable logging, you can set the `LogLevel` to `0`, or remove the `LogFile` entry.

For example:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Intercede\Edefice\Connector\EntrustJTK]
"JavaLocation"="C:\\Program Files\\Intercede\\MyID\\Components\\Java"
"LogFile"="c:\\logs\\jtklog.log"
"LogLevel"=dword:00000000
```

or:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Intercede\Edefice\Connector\EntrustJTK]
"JavaLocation"="C:\\Program Files\\Intercede\\MyID\\Components\\Java"
```

Note: The difference between providing no values and a `LogLevel` setting of `0` is that the Java tracing will create or reset the existing log file to a file of length 0, and not produce any logging.

Note: Issuing a single certificate with a `LogLevel` of `4` produces a file over 500 KB; leaving the diagnostic running has implications for disk space.

10.4 Dal4Net logging

You can configure logging on the MyID Dal4Net component. If your system uses Dal4Net – for example, for systems using SQL Authentication – this logs every SQL query that MyID sends to the database (but not the results of those queries).

To set up Dal4Net logging:

1. On the MyID application server, open the `Dal4Net.dll.config` file in a text editor.

   By default, this is in the following folder:

   ```
   C:\Program Files\Intercede\MyID\Components\Dal4Net\
   ```

2. Uncomment the `<log4net>` node.
3. Update the following line to specify the location of the log file:

   ```
   <file value="Dal4Net.log"/>
   ```

Note: You must ensure that the MyID named COM user has the necessary permissions to create and write to the log file. You can create a file then give the user write permissions if you prefer not to give the user create permissions.

Important: Disable the logging when you have completed diagnosing the issues, as the log file may become very large.

11 Known issues

This section contains any known issues that may occur when logging the MyID components.

- **Performance issues with antivirus scanning software**

  If you have logging switched on, MyID writes a great deal of frequently-updated data to the log file folder. With some antivirus software, this may cause a problem – under heavy load, the antivirus software checks the frequently-updated log files over and over, which may have a significant effect on the performance of your PC.

  To prevent issues occurring with your antivirus software, you are recommended to exclude the log file folder from the antivirus scanning software.