AN-306

User PIN Encryption and Advanced PIN Management in Protege GX

Application Note

The specifications and descriptions of products and services contained in this document were correct at the time of printing. Integrated Control Technology Limited reserves the right to change specifications or withdraw products without notice. No part of this document may be reproduced, photocopied, or transmitted in any form or by any means (electronic or mechanical), for any purpose, without the express written permission of Integrated Control Technology Limited. Designed and manufactured by Integrated Control Technology Limited, Protege® and the Protege® Logo are registered trademarks of Integrated Control Technology Limited. All other brand or product names are trademarks or registered trademarks of their respective holders.

Last Published: 08-Sep-21 11:39 AM

Contents

Introduction

The Protege GX database, by default, stores user PIN codes in plain text. This allows for easy reading and display of PINs for operators that are configured with the **Show PIN numbers for Users** privilege.

However at high security sites this visibility may be seen as a risk, so Protege GX provides the ability to encrypt user PINs in the database, removing open visibility and strengthening security protocols.

User PIN encryption in the Protege GX database utilizes the Always Encrypted feature built into Microsoft SQL Server, and provides a high level of encryption using a deterministic **AEAD_AES_256_CBC_HMAC_SHA_256** algorithm.

Prerequisites

User PIN encryption in Protege GX requires:

Software	Version	Notes
Protege GX software	4.3.291.6 or higher	
Microsoft SQL Server	2016 Service Pack 1, or a later edition that supports the Always Encrypted feature.	For the SQL server editions that support this feature, see the Microsoft Help. For the SQL server editions supported by Protege GX, see the Protege GX Server Installation Manual.
Protege GX Controller	**Firmware Version**	**Notes**
PRT-CTRL-DIN	2.08.1045 or higher	
PRT-CTRL-DIN-1D		

Enabling user PIN encryption is **irreversible**. It is therefore strongly recommended that a full database backup is taken prior to implementation of this feature. Restoring this backup will be the only way to remove user PIN encryption and advanced PIN security features.

Minimum Service Permissions

If the Protege GX services do not have administrative permissions, some additional permissions are required to set up and use PIN encryption.

- To **set up** user PIN encryption, the following permissions are required:
  - VIEW ANY COLUMN MASTER KEY DEFINITION
  - VIEW ANY COLUMN ENCRYPTION KEY DEFINITION
  - ALTER ANY COLUMN MASTER KEY
  - ALTER ANY COLUMN ENCRYPTION KEY
  - Read and write access to the Local machine > Personal certificate store
  - Read and write access to the Local machine > Trusted Root Certification Authorities certificate store
- To **use** user PIN encryption, the following permissions are required:
  - VIEW ANY COLUMN MASTER KEY DEFINITION
  - VIEW ANY COLUMN ENCRYPTION KEY DEFINITION
  - Read access to the Local machine > Personal certificate store

The permissions that are not required to use PIN encryption may be disabled after initial configuration.

Configuring User PIN Encryption

Installations operating the Protege GX **Single Record Download Service** or **Secondary Download Servers** have additional programming requirements for user PIN encryption. If these server configurations apply to your installation you should review the steps on the following page before enabling user PIN encryption.

Below are the steps required to enable user PIN encryption.

1. Navigate to **Global | Global Settings**.
2. In the **Main Database** section, check the **Encrypt User PINs** checkbox.

   This option will only be visible when Protege GX and SQL Server meet the minimum version requirements.
3. A popup warning notification will advise 'This action cannot be undone, and PINs will no longer be visible to ALL operators, do you wish to continue?'
4. Click **Yes** to continue.
5. While encryption is being performed, the progress spinner will display 'Encrypting'.
6. When encryption is complete a popup message will display 'Encryption Succeeded'.
7. Click **OK** to finalize.
8. Click **Save**.

Encryption Process

During the encryption process the following steps are completed in the background.

1. A **Self-Signed Certificate for Encryption** is created on the local machine if one does not already exist.

   This is given the friendly name Data Service Encryption Certificate.
2. A **Column Master Key** (CMK) is created using the above certificate.
3. A **Column Encryption Key** (CEK) is created using the above CMK.
4. The **PINNumber** column in the database is renamed to **OLDPINNumbersBE**.
5. A new encrypted **PINNumber** column is created.
6. The plain text PIN codes in the OLDPINNumbersBE column are encrypted and copied to the new **PINNumber** column.
7. The **OLDPINNumbersBE** column is deleted from the database.

Single Record Download Service

For sites operating the Protege GX Single Record Download Service, some additional configuration is required for the user PIN encryption feature to operate alongside the single record download service. Until this change is completed the single record download service will not be able to read encrypted user PINs.

1. Stop the Protege GX Single Record Download Service in the Windows Services Manager.
2. In the File Explorer, navigate to the installation directory of the single record download service.
   The default installation directory is: C:\Program Files (x86)\Integrated Control Technology\Protege GX.
3. Open GXSV2B.pack.exe.config.

   > Files in this directory require administrator permissions to edit. You may need to open the file as an administrator using an application like Notepad++, or make a copy in a different directory to edit and replace the original.

4. Locate the following connection string:
   ```
   <add name="Main" connectionString="Trusted_Connection=yes;
   TrustServerCertificate=yes; Encrypt=yes; Server=[DatabaseServer];
   Database=[MainDatabase]; max pool size=2000;" />
   ```
5. Add the text in bold:
   ```
   <add name="Main" connectionString="Trusted_Connection=yes;
   TrustServerCertificate=yes; Encrypt=yes; Server=[DatabaseServer];
   Database=[MainDatabase]; max pool size=2000; Column Encryption
   Setting=Enabled;" />
   ```
6. Save the config file.
7. Start the Protege GX Single Record Download Service in the Windows Services Manager.

Secondary Download Servers

The self-signed certificate created during the encryption process is the key to encrypting and decrypting PIN data in the database. For sites operating multiple Protege GX download servers the certificate must be installed on all secondary download servers in order for them to apply and recognize the same PIN encryption.

Once the encryption process has completed on the primary server, secondary servers will no longer be able to read user PINs until the certificate has been imported.

1. Stop the Protege GX Download Service on all secondary servers before beginning the encryption process.
2. Configure user PIN encryption on the primary server (see previous page).
3. Export the self-signed certificate from the primary server.

   > Refer to the Data Service Encryption Certificate Export instructions (see page 9).

4. Import the certificate to all Protege GX secondary download servers.

   > Refer to the Data Service Encryption Certificate Import instructions (see page 10).

5. Start the Protege GX Download Service on all secondary servers.

MS Azure Support

There are no specific settings to add to an Azure virtual machine during the default Protege GX deployment/installation process. The required ODBC drivers are installed by the Protege GX installer, and the self-signed certificate is generated when the operator selects the **Encrypt User PINs** check box.

Azure SQL Database

If the database is held in an Azure SQL DB, the connection strings for the **Data**, **Event** and **Download** services need to be updated as below:

Data Service:

- `<add name="MainConnection" connectionString="Trusted_Connection=`**`no`**`; TrustServerCertificate=yes; Encrypt=yes; Column Encryption Setting=Enabled; Server=tcp:`**`SQLSERVERNAME`**`.database.windows.net,1433; User ID=`**`SQLUSERNAME`**`; Password=`**`SQLPASSWORD`**`; `**`Persist Security Info=True`**`; Database=ProtegeGX;"></add>`
- `<add name="EventConnection" connectionString="Trusted_Connection=`**`no`**`; TrustServerCertificate=yes; Encrypt=yes; Column Encryption Setting=Enabled; Server=tcp:`**`SQLSERVERNAME`**`.database.windows.net,1433; User ID=`**`SQLUSERNAME`**`; Password=`**`SQLPASSWORD`**`; `**`Persist Security Info=True`**`; Database=ProtegeGXEvents; MultipleActiveResultSets=True;"></add>`

Download Service:

- `<add name="MainConnection" connectionString="Provider=MSDASQL; Extended Properties=&quot; Driver={ODBC Driver 17 for SQL Server}; Server=tcp:tcp:`**`SQLSERVERNAME`**`.database.windows.net,1433; Persist Security Info=True; Database=ProtegeGX; Trusted_Connection=`**`no`**`; ColumnEncryption=Enabled;&quot;;UID=`**`SQLUSERNAME`**`; PWD=`**`SQLPASSWORD`**`;&quot;"/>`
- `<add name="EventConnection" connectionString="Provider=MSDASQL; Extended Properties=&quot; Driver={ODBC Driver 17 for SQL Server}; Server=tcp:tcp:`**`SQLSERVERNAME`**`.database.windows.net,1433; Database=ProtegeGXEvents; Trusted_Connection=`**`no`**`; ColumnEncryption=Enabled;&quot;;UID=`**`SQLUSERNAME`**`; PWD=`**`SQLPASSWORD`**`;&quot;"/>`

Event Service:

- `<add name="MainConnection" connectionString="Provider=MSDASQL; Extended Properties=&quot; Driver={ODBC Driver 17 for SQL Server}; Server=tcp:`**`SQLSERVERNAME`**`.database.windows.net,1433; Database=ProtegeGX; Trusted_Connection=`**`no`**`; Persist Security Info=True; ColumnEncryption=Enabled;&quot;;UID=`**`SQLUSERNAME`**`; PWD=`**`SQLPASSWORD`**`;&quot;"/>`
- `<add name="EventConnection" connectionString="Provider=MSDASQL; Extended Properties=&quot; Driver={ODBC Driver 17 for SQL Server}; Server=tcp:`**`SQLSERVERNAME`**`.database.windows.net,1433; Database=ProtegeGXEvents; Trusted_Connection=`**`no`**`; ColumnEncryption=Enabled;&quot;;UID=`**`SQLUSERNAME`**`; PWD=`**`SQLPASSWORD`**`;&quot;"/>`

Advanced PIN Security Management

When user PIN encryption is implemented, the following advanced PIN security management features are also enabled in the Protege GX user interface.

- All operators can no longer view user PINs, regardless of the **Show PIN numbers for Users** setting. All user PINs are now starred out for all operators.
- Operators can no longer manually enter a new PIN for a user. New user PINs can only be created using the **4**, **5** or **6** digit random PIN generation buttons, or the **Reset PIN** button.

  It is possible to restrict individual operators from generating new user PINs using the **Generate PINs** option in **Sites | Security Levels**.

- Any new user PIN created in the Protege GX user interface will immediately expire (regardless of expiry settings), effectively making it a single use PIN only. When the user next logs in to a keypad they will be prompted to change this PIN and select a new one for themselves. This ensures that only the individual user knows their permanent PIN.

  The **PIN Expiry Time** setting will then apply as normal to the permanent PIN selected by the user.

- User PIN numbers can no longer be imported and added to user records via the SOAP Service, ICT Data Sync Service or by the CSV user import feature. All user PINs must be created in the user interface or web client using the random PIN generation buttons, or by a user at the keypad.

  With SOAP Service version 1.6.0.3 or higher, it is possible to randomly generate a user PIN and request the record to view the newly updated PIN. Only newly updated PINs can be viewed this way. For more information, see the Protege GX SOAP Service API Specification.

Important Notes

- The **Encrypt User PINs** option will only be displayed in Protege GX if the installed SQL Server version detected is 2016 SP1 or higher.
- Database backups created prior to enabling PIN encryption are able to be restored. However, PIN encryption will need to be re-enabled after the restore is complete.
- Database backups created after PIN encryption is enabled are able to be restored with no further configuration required. However, PIN encryption cannot be reversed in these databases.
- If the Column Master Key (CMK) and/or the Column Encryption Key (CEK) are deleted from within SQL the PINNumber column will become unreadable. The software will then start to produce exception errors.

  If this occurs the only way to recover the PINs is to restore a database backup.

Certificate Backup and Recovery

The self-signed certificate created during the encryption process is the key to encrypting and decrypting PIN data. If the certificate is damaged or deleted the encrypted data will be irrecoverably lost, and all user PIN codes would need to be reset. The self-signed certificate for encryption should be backed up and stored securely, so that it is available for recovery should the need arise.

Data Service Encryption Certificate Export

To create a backup of the certificate you will need to access the **Certificate Manager** tool on the machine where the certificate exists, to create an **export** of the Data Service Encryption Certificate.

> The certificate is created on the machine where the data service is installed, which may not be the same machine as the SQL server installation.

1. To open the Certificate Manager tool, press the **Windows + R** keys, then type **certlm.msc** into the search bar and press **Enter**.
2. The tool directory will display Certificates - Local Computer.
3. Open the **Personal** folder, then click the **Certificates** sub-folder.
4. In the window displaying the certificates, scroll across to the **Friendly Name** column and locate the certificate with the assigned friendly name Data Service Encryption Certificate.
5. Right click the certificate and navigate to **All Tasks**, then select **Export**.
6. The **Certificate Export Wizard** will open. Click **Next**.
7. You must select the **Yes, export the private key** option.

   > The private key is the critical component in decryption. If you do not export the private key, when the certificate is imported it will not be able to decrypt the encrypted data.

   Then click **Next**.
8. Ensure that the following **Export File Format** options are selected:
   - Include all certificates in the certification path if possible
   - Enable certificate privacy

   > The Delete the private key if the export is successful option **must be disabled**.

   Then click **Next**.
9. On the **Security** page, enter and confirm a strong **Password**.

   > Ensure the password is recorded and stored securely with important site information.
10. Ensure that **Encryption** is set to AES256-SHA256, then click **Next**.
11. Specify an export **File name** and path, then click **Next**.
12. Click **Finish** to complete the certificate export.
13. The export wizard should verify that 'The export was successful'.
14. Confirm that the certificate backup .pfx file has been exported to the file path as specified.
15. The file should be stored securely in an alternate location to ensure that it is available if required.

Data Service Encryption Certificate Import

The self-signed certificate may need to be recovered in the event that it is lost or damaged. It is also necessary to import the certificate to all secondary download servers for sites operating multiple download servers.

To recover or import the self-signed certificate you will need to access the **Certificate Manager** tool and **import** the **.pfx** backup of the Data Service Encryption Certificate.

1. Ensure that the .pfx backup file is accessible from the local PC.
2. Stop all Protege GX services before initiating the import.
3. To open the Certificate Manager tool, press the **Windows + R** keys, then type **certlm.msc** into the search bar and press **Enter**.
4. The tool directory will display Certificates - Local Computer.
5. Open the **Personal** folder.
6. Right click the **Certificates** sub-folder and navigate to **All Tasks**, then select **Import**.
7. The **Certificate Import Wizard** will open. Click **Next**.
8. Click **Browse** and locate the .pfx backup file to import, then click **Next**.

   You will need to change the file type dropdown selection to Personal Information Exchange (*.pfx;*.p12).

9. Enter the **Password** that was created during the export process.
10. Import Options:
    - Mark this key as exportable. This will allow you to back up or transport your keys at a later time.
      - This option must be selected if you want to be able to export/backup the private key with this certificate in the future. This option is slightly less secure.
      - The key is more secure if this option is not selected, however you will not be able to export the private key with the certificate in the future if you lose your current .pfx backup file.
    - Ensure that Include all extended properties is selected.
11. Click **Next**.
12. Ensure the **Certificate store** is set to Personal, then click **Next**.
13. Click **Finish** to complete the certificate import.
14. The import wizard should verify that 'The import was successful'.
15. Close the Certificate Manager tool.
16. Restart the Protege GX services.

Designers & manufacturers of integrated electronic access control, security and automation products.
Designed & manufactured by Integrated Control Technology Ltd.

Disclaimer: Whilst every effort has been made to ensure accuracy in the representation of this product, neither Integrated Control Technology Ltd nor its employees shall be liable under any circumstances to any party in respect of decisions or actions they may make as a result of using this information. In accordance with the ICT policy of enhanced development, design and specifications are subject to change without notice.