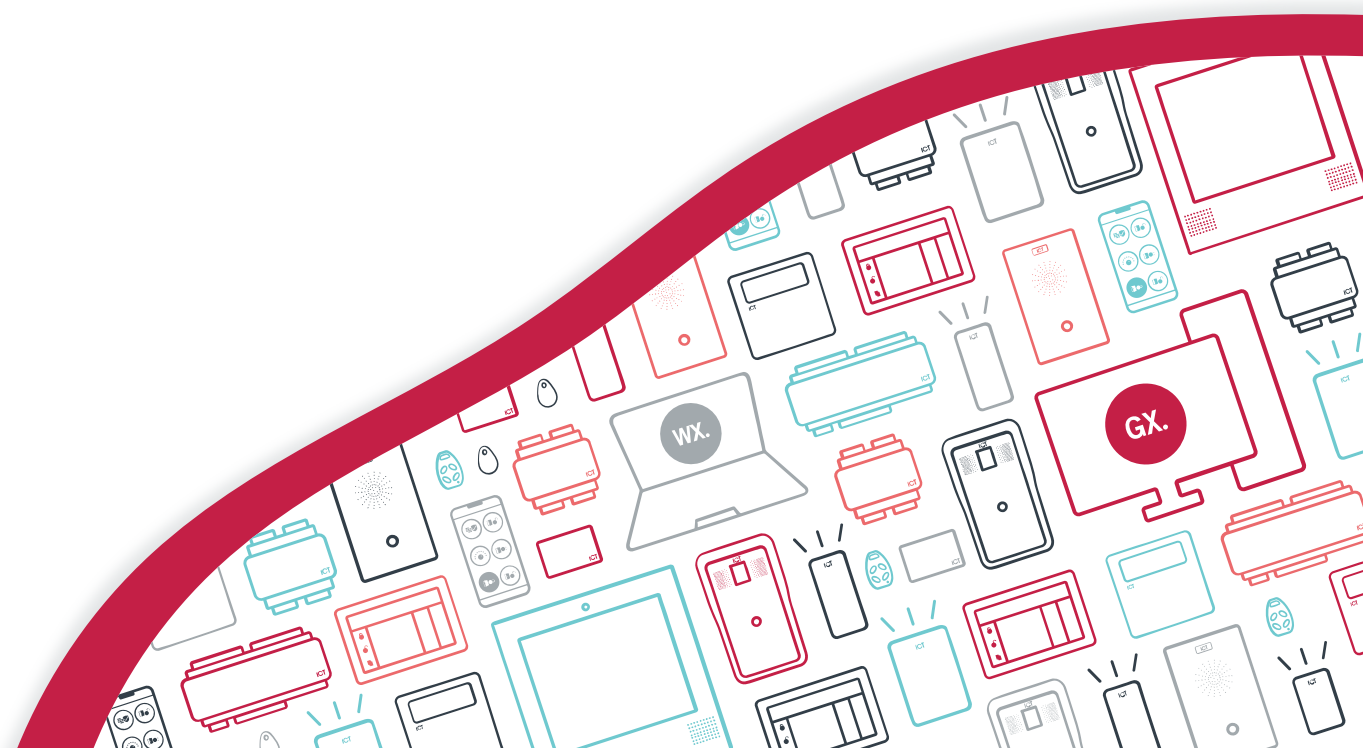
AN-196

Protege GX Schindler HLI Integration

Application Note



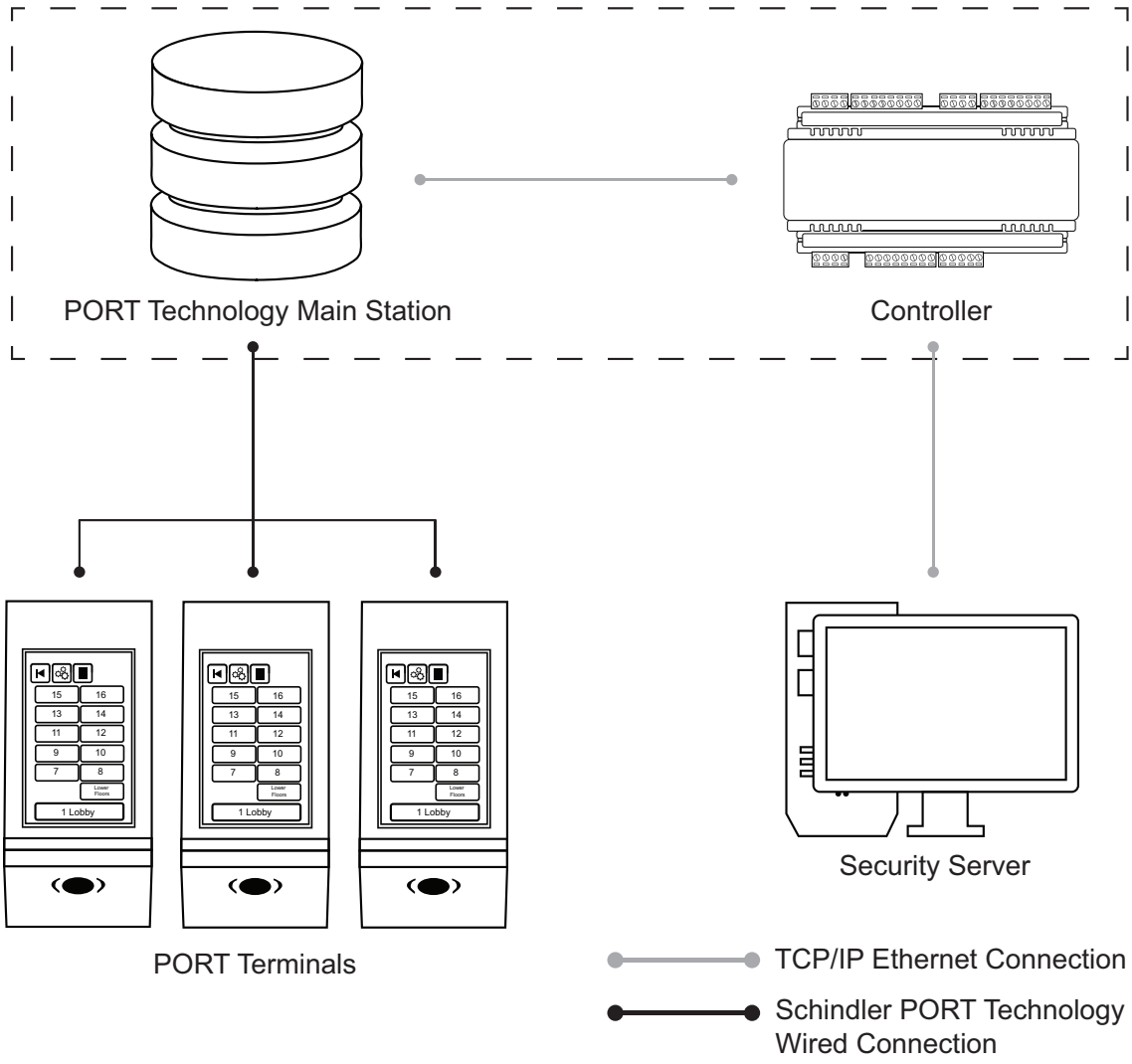
Last Published: 02-Nov-23 9:42 AM

Contents

Introduction

High level integration between Protege GX and the Schindler PORT Technology Elevator Security Management System enables you to take full advantage of a complete destination dispatch solution while providing access control and intruder detection.

Users and access credentials are configured within Protege GX and transferred to the Schindler system. When a user presents their access card at a Schindler PORT terminal or ICT card reader, the Schindler system verifies their credentials using the information supplied by Protege GX. Once access has been verified, the Schindler system automatically calls an elevator to transport the user to the selected floor.



The Protege GX Schindler HLI elevator integration supports up to 128 floors.

This integration is a licensed feature.

Prerequisites

Software Requirements

The following software must be installed and operational to configure this integration.

Software	Version	Notes
Protege GX	4.1.180 or higher	
Schindler PORT Elevator System	1.3.369.1	This is the **only** tested and supported version for this integration.

Hardware Requirements

The following Protege GX controllers support this integration.

Product Code	Firmware Version
Protege GX Controller	• **Basic operation (64 floors)**: 2.08.919 or higher • **Basic operation (128 floors)**: 2.08.1158 or higher • **Home floor**: 2.08.1068 or higher

Licensing Requirements

The following license is required for this integration.

License	Order Code	Notes
Protege GX Schindler Elevator High Level Interface License	PRT-GX-ELV-HLI-SC	1 license per controller used for this integration

ICT Card Readers for Elevator Access Control

In order to use ICT card readers for elevator access control alongside the Schindler system, the following additional prerequisites are required.

Hardware Requirements

Product Code	Firmware Version
Protege GX Controller	• **Basic operation (64 floors)**: 2.08.937 or higher • **Basic operation (128 floors)**: 2.08.1158 or higher • **Antipassback**: 2.08.1297 or higher • **Home floor**: 2.08.1404 or higher

Licensing Requirements

License	Order Code	Notes
Protege GX Door License	PRT-GX-DOR-1	1 license per Schindler Destination Terminal used in the integration
	PRT-GX-DOR-10	
	PRT-GX-DOR-50	

It is the responsibility of the installation professional to verify the version of the proposed third-party system and supported components with the version listed in this document. ICT will not accept responsibility for the failure to verify integrated system versions and requirements.

Floor Mapping

Before beginning to program the elevator integration, it is important to correctly map the layout of the elevator system. All elevator-accessible floors need to be identified and mapped in sequential order from bottom to top. You need to identify which (if any) floors have rear doors, and which (if any) floors are considered 'below ground' in the elevator system programming. This is essential for Protege GX to correctly map floors for event reporting and 'home floor' functionality.

Floor Relay Numbering

Each floor accessed by the elevator system needs to be programmed in Protege GX, with a unique **Floor relay** number assigned. The floor relay number tells the controller where the physical floor is located, creating a map in Protege GX of accessible floors.

Floor relay numbers must be **unique**, programmed in **numerical order** (starting at 1), and beginning at the **lowest accessible floor**, including any basement floors.

The lowest accessible floor must be assigned a floor relay of **1**, as shown in the example below.

Physical Floor	Floor Relay Number
Basement 2	1
Basement 1	2
Ground Floor	3
Level 1	4

Skipped Floor Numbers

Numbers are commonly skipped in floor labels due to superstitions surrounding specific numbers, and technicians may also skip these in the elevator controller programming. Where this occurs a placeholder floor record must be created in Protege GX to maintain the required sequential floor relay numbering, as in the example below.

Physical Floor	Elevator System Floor Relay	Floor Relay Number
Ground	0	1
Level 1	1	2
...	...	...
Level 12	12	13
N/A	Not programmed	14 (Placeholder)
Level 14	14	15

In this example a placeholder floor record is added with relay number 14 to maintain the sequential numbering and represent the skipped programming in the elevator system. Had the sequence been maintained in the elevator system (i.e. Level 14 programmed with elevator system floor relay 13) no placeholder floor would be required.

Lowest Basement Floor

The Protege GX floor programming must match the elevator system setup, or the integration will not function correctly. To do this you need to identify which floor is programmed as the first aboveground floor in the elevator system configuration, then count the number of elevator-accessible floors below this one. This number becomes the controller's **Lowest basement floor** setting in the controller configuration.

Depending on the building layout and elevator system configuration, the first aboveground floor could be the ground floor, the first floor, a lobby, or even a carpark. The labels and numbering of the floors is not important, only the configuration of the relay numbering in the elevator system.

The floor programmed with relay '0' in the elevator system is the first aboveground floor. Any floor programmed with a negative relay number is treated as a basement floor.

Examples are provided to illustrate the Protege GX programming required for different elevator system scenarios.

Lowest Basement Floor Examples

1. In the first scenario the building has no basement floors, with standard relay programming beginning from '0'. As there are no belowground floors the **Lowest basement floor** setting is 0.

Physical Floor	Elevator System Floor Relay	Floor Relay Number
Ground Floor	0	1
Level 1	1	2
Level 2	2	3
Level 3	3	4

2. In the next scenario the building has two basement floors, with relay programming sequentially numbered. With two belowground floors the **Lowest basement floor** setting is 2.

Physical Floor	Elevator System Floor Relay	Floor Relay Number
Basement 2	-2	1
Basement 1	-1	2
Ground Floor	0	3
Level 1	1	4

3. In the following scenario the building has two basement floors, however the ground floor is programmed as relay '1' instead of '0'. A placeholder floor record is required in Protege GX to maintain the relay sequence, and the **Lowest basement floor** setting is 2.

Physical Floor	Elevator System Floor Relay	Floor Relay Number
Lower Carpark	-2	1
Upper Carpark	-1	2
	Not programmed	3 (Placeholder)
Ground Floor	1	4

Floors with Rear Doors

For some floors it may be possible to exit the elevator at the rear as well as the front. Each 'rear floor' must also be programmed in Protege GX.

The rules for programming rear floors are as follows:

- The **Floor relay** number of the rear floor must be equal to that of the corresponding front floor.
- Each rear floor must be programmed with the command `Rear`.
- The controller must be programmed with the command `HLI_128_FLOORS = true`.

The `Rear` and `HLI_128_FLOORS = true` commands are supported in controller firmware version 2.08.1158 or higher. For earlier firmware versions, rear doors must be programmed with relay numbers from 65-128.

The table below demonstrates front and rear floor relays in a building with 50 aboveground floors and two basement floors.

Physical Floor	Front Floor Relay	Rear Floor Relay
Basement 2	1	1
Basement 1	2	2
Ground Floor (no rear access)	3	-
Level 1 (no rear access)	4	-
Level 2	5	5
...	...	...
Level 48	51	51
Level 49	52	52

It is not necessary to program rear floor records for any floors that do not have rear access, but you may want to do this to keep the programming tidy.

Rear Floor Relays 65-128

For sites with up to 64 floors the rear floor relays can be programmed from 65-128 without using the `Rear` command. To calculate the rear floor relay, add 64 to the front floor relay.

This is a legacy programming option which provides backward compatibility for existing installations and controller firmware versions prior to 2.08.1158, but is not recommended for new installations. It supports a maximum of 64 floors and is not available if the controller has been configured to support 128 floors.

The table below demonstrates front and rear floor relays in a building with 50 aboveground floors and two basement floors where the **Rear** command is not used and rear floor relays start at 65.

Physical Floor	Front Floor Relay	Rear Floor Relay
Basement 2	1	65
Basement 1	2	66
Ground Floor (no rear access)	3	-
Level 1 (no rear access)	4	-
Level 2	5	69
...	...	...
Level 48	51	115
Level 49	52	116

Landing Floors

Occasionally there may be situations where different elevators have a different schedule for open access to a particular floor. This most typically occurs where banks of elevators (or risers) service particular sections of a building (such as lower, mid and upper floors) and share common access to landing floors which transition between the sections. It can also occur where different wings of a building or separate elevators for staff and customers have different open access to a floor.

Any time a floor has more than one schedule for open access, Protege GX requires a unique floor record, with a unique floor relay number, for each access configuration.

Same Schedule

Below is a very simplified and compressed example of a building where four elevators have access to specific floors. The floor relays are numbered in sequence as normal, with the common landing floors highlighted.

In this initial scenario all elevators will have the same open schedule access to the landing floors. As there is only one open access schedule there is no need for additional floor records and relay numbers.

Physical Floor	Basement	Lower Floors	Mid Floors	Upper Floors
Basement 2	1			
Basement 1	2			
Ground Floor	**3**			
Level 1		4		
Level 2		5		
Level 3		**6**		
Level 4			7	
Level 5			**8**	
Level 7				9
Level 8				10

Different Schedules

Now we will change the scenario so that all elevators have a different schedule for open access to each of the common landing floors. Remember that for each different access schedule, Protege GX requires an additional floor record with a unique floor relay number.

- All floor relay numbers must still be programmed in sequential order with standard base programming.
- Additional floor records must be added in sequence at the end of the initial base programming.

After the standard floor relays have been mapped, we add the floor relay numbers for the additional floor records.

Physical Floor	Basement	Lower Floors	Mid Floors	Upper Floors
Basement 2	1			
Basement 1	2			
Ground Floor	3	11	12	14
Level 1		4		
Level 2		5		
Level 3		6	13	
Level 4			7	
Level 5			8	15
Level 7				9
Level 8				10

Remember to name the additional floor records in a consistent way that follows your standard naming conventions and clearly identifies the specific floor access.

The Schindler system reports only the first relay number for the floor. In our example, travel from any elevator to the ground floor will always be reported as relay 3, with events indicating travel via the basement elevator.

Programming Steps

The following instructions outline the steps required to configure the integration within Protege GX. These include:

- Enabling controller HLI and configuring connection with the Schindler system
- Adding the required floors
- Creating the default floor group
- Assigning Schindler user profile templates to access levels
- Configuring access control:

  If using Schindler Card Readers:
  - Configuring user credentials and formats to send to Schindler

  If using ICT Card Readers:
  - Configuring door records as Schindler Destination Terminals
- Configuring outputs to trigger SOMs (Special Operating Modes) from within Protege GX

Configuring the Controller

1. Navigate to **Sites | Controllers** and select the controller to be used for the integration.
2. Open the **Configuration** tab and scroll down to the **Elevator HLI** section.
3. Set the **Elevator HLI type** to Schindler, then configure the required options:
   - **Network adaptor**: Only Cable is supported for this integration.
   - **Port system primary IP**: The primary IP address of the Schindler server.
   - **Port system secondary IP**: The secondary IP address of the Schindler server (backwards compatible configuration).
   - **Online database port**: The TCP port of the Schindler online database interface.
   - **Call interface port**: The TCP port of the Schindler call interface.
   - **Life reporting interface port**: The TCP port of the Schindler life reporting interface.
   - **Lowest basement floor**: The lowest physical underground floor accessible by an elevator. For example, if there are five underground floors the value should be 5. If there are no underground floors, set to 0.

     What is considered an underground floor is determined by the elevator system configuration.
   - **Default floor group**: The floor group containing all accessible floors and the schedules used to control when each floor can be freely accessed.

     This floor group will be created later in the programming (see page 13), and must be assigned to the controller in order for the integration to function.
   - **Enable call interface**: Enables the Schindler call interface.
   - **Enable life reporting interface**: Enables the Schindler life reporting interface.
   - **Enable elevator HLI debug:** When this option is enabled, system debug messages will be logged in the event log for troubleshooting.

     This option may be useful for initial configuration and troubleshooting but should be disabled during normal operation to save event storage.
   - **Site code formats**: Defines the facility numbers and formats of user credentials which will be sent to the Schindler system. Site code formats are only necessary if Schindler readers and credentials are being used for this integration

     These will be configured later in the programming (see page 15).

Configuring up to 128 Floors

By default the controller supports up to 64 floors for this integration. However, with some additional configuration the integration can support up to a maximum of 128 floors.

This feature requires controller firmware version 2.08.1158 or higher.

1. Navigate to **Sites | Controllers** and select the controller to be used for the integration.
2. Expand the **Commands** field and enter the following commands:
   - `HLI_MAX_FLOORS = X`

     Where **X** is the total number of floors in the building. This can be a value from 1-128.
   - `HLI_128_FLOORS = true`

     This command is required to enable the controller to support using the `Rear` command for rear floors.
3. Click **Save**.

Adding Floors

Each floor accessed by the elevator system needs to be configured in Protege GX with a floor relay number assigned. This tells the controller where the physical floor is located, creating a map of accessible floors.

1. Navigate to **Programming | Floors**.
2. Add a floor record for each elevator-accessible physical floor, assigning the **Floor relay** for each floor as explained in the Floor Mapping section (see page 6).
3. The **Elevator HLI options** define the commands that Protege GX will send to the Schindler system when floor schedules become valid or invalid. These are used to instruct the Schindler system to enable or disable open access for each floor. Set the following for each floor:
   - **Schindler schedule valid time pattern**: Defines the message that is sent to the Schindler Call Interface when the schedule assigned to the floor becomes valid.
   - **Schindler schedule invalid time pattern**: Defines the message that is sent to the Schindler Call Interface when the schedule assigned to the floor becomes invalid.

     The above commands will be specific to your Schindler system.
   - **Schindler primary terminal ID**: Defines the ID of the Schindler terminal that the schedule valid/invalid message is sent to when the controller is communicating via the primary IP address.
   - **Schindler secondary terminal ID**: Defines the ID of the Schindler terminal that the schedule valid/invalid message is sent to when the controller is communicating via the secondary IP address.
4. If the Schindler home floor feature is in use, enter the following command in the **Commands** field on the **General** tab for each floor:

   `FloorZone = X`

   where **X** is the **Zone number** for the floor in the Schindler system. These commands must match the zone numbers configured in Schindler.
5. If the floor is a 'rear floor' add the `Rear` command in the **Commands** field.

   The `Rear` command is supported in controller firmware version 2.08.1158 and above. The controller must also be programmed with the `HLI_128_FLOORS = true` command.
6. Click **Save**.

Adding the Default Floor Group

This floor group must contain all of the floors that the Schindler system is able to access. It determines the default schedule for free access of all floors in the Schindler system. Once a floor is added to the floor group, a schedule can be applied to the floor to define when it is freely accessible without credentials.

1. To create a floor group for the floors that are accessible to the Schindler system, navigate to the **Groups | Floor groups** menu, click **Add** and **Name** the floor group, (e.g. 'All Schindler Floors').
2. In the **Floors** window, click **Add** to select the floors to be included in the floor group. Drag and drop items into the main window or select every Schindler floor and click **Ok**.
3. In the **Schedule** column, apply a schedule to each floor:
   - If the Always option is selected, the floor will be freely accessible without credentials at all times.
   - If a specific schedule is selected, the floor will be freely accessible when the schedule is valid, and will require valid credentials to access when the schedule is invalid.
4. Click **Save**.
5. Return to **Sites | Controllers | Configuration** tab. Set the **Default floor group** of the relevant controller to the floor group created above.

Assigning Schindler Templates to Access Levels

Access levels are required to link each Protege GX user with a Schindler user profile template. User profile templates define which floors users have access to and when, within the Schindler system. There are two methods available for assigning Schindler user profile templates to access levels, depending on which version of the Protege GX software you are using.

Users may have multiple access levels within Protege GX, but each user can only be associated with a single User Profile Template in Schindler. Therefore, only the assigned Schindler template or the first access level with a valid floor group assigned to a user will be sent as the user profile template for that user.

Method 1: Schindler Template

This method does not support life reporting events in Protege GX. Life reporting events are only generated when Method 2: Floor Groups is used (see below).

This method is only available with Protege GX version 4.2.208.0 or later.

1. Navigate to **Users | Access levels** and select or create a relevant access level.
2. In the **General** tab, expand the **Elevator HLI** section.
3. In the **Schindler template** field, enter the name of the required user profile template (as it is defined in the Schindler system).
4. Assign the access level to users who require that user profile template.

   Note: Protege GX will prevent you from assigning more than one access level with the **Schindler template** field set to each user.

Method 2: Floor Groups

This method is available with all versions of Protege GX which support this integration.

1. Navigate to **Groups | Floor groups** and create a new floor group for each Schindler user profile template.
2. Give each floor group a **Name** which is the same as a user profile template (as it is defined in the Schindler system)
3. Click **Add** and add at least one floor to each floor group. You are not required to add the floors which the user profile template allows access to.

   The floors that can be accessed are determined by the user profile template, not by the floor group itself. You can assign any floor to the floor group, but you must include at least one floor to ensure that the Protege GX system recognizes the group.

4. Navigate to **Users | Access levels** and select or create a relevant access level.
5. In the **Floor groups** tab, add the floor group created above. Repeat for any other access levels.

   Only one floor group per user will be sent to the Schindler system. This will be the first floor group in the user's first access level that has a valid floor group.

6. Assign the access level to users, ensuring that it is the first access level with a floor group for each user.

Assigning Home Floors to Access Levels

Setting a home floor on an access level allows users with that access level to automatically travel to that floor when they call an elevator.

For this feature to function, you must set the **Zone number** for each floor used in this integration (see page 12).

To assign a home floor to an access level:

1. Select the desired access level in **Users | Access levels**.
2. In the **General** tab, expand the **Elevator HLI** section.
3. Set the **Elevator destination floor** to the desired home floor for that access level.
4. Click **Save**.

Configuring Access Control for Schindler Readers

If you are using Schindler readers and credentials for access control in this integration, follow the steps below.

Sending Credentials to Schindler

Protege GX must send user credential data to the Schindler system to allow Schindler readers to recognize those credentials. Site code formats define the format in which user credentials will be sent to the Schindler system.

Return to **Sites | Controllers | Configuration** tab, under **Elevator HLI**. In the **Site code formats** section, click **Add** to define a format for a range of credentials:

- **Site code**: The site code or facility number of the user credentials that will be formatted.
- **Format**: All credentials that match the **Site code** defined above will be converted to this format and sent on to the Schindler system. See the list of supported Schindler formats below.
- **Subformat**: Set to 0 by default. Only relevant when the **Format** is set to Unknown Wiegand (see below).

You can define up to 32 site code formats.

Protege GX will only transmit a maximum of three facility/card numbers to the Schindler system, as that is the number that Schindler can assign to one user. Protege GX will convert each credential to a single credential string based on the Schindler **Format**.

For firmware versions before 2.08.1068, the controller will only transmit the first three facility/card numbers assigned to each user, if they match a defined site code format. Any Schindler cards must be programmed in the first three slots for each Protege GX user.

For firmware versions after 2.08.1068, the controller will check all of the user's facility/card numbers in order, then all assigned credential types. It will send the first three credentials that match a defined site code format.

If a facility/card number or credential type is disabled, the data will not be sent to Schindler. However, this credential will still count towards the maximum of three that can be sent. To free up a slot for a new credential, ensure that you delete the old one.

Supported Schindler Credential Formats

The following Schindler credential formats are supported when integrating with Schindler readers:

- Hitag 1 (Schindler Type = 0)
- HID (Schindler Type = 10)
- HID Corporate 1000 (Schindler Type = 15)
- Generic Wiegand (Schindler Type = 38)
- Unknown Wiegand (Schindler Type = 39)
  - 34 Bit Wiegand (Protege GX Subformat = 0)
  - 26 Bit Wiegand (Protege GX Subformat = 1)

Pure Wiegand Format

Pure Wiegand (Schindler Type = 25) is available as a supported Schindler credential format, but requires additional configuration. Note that configuring Pure Wiegand format via commands will override any existing Schindler credential formats configured in the **Site code formats** section.

This feature is only available with controller firmware version 2.08.1068 or higher.

1. Create a credential type in **Sites | Credential types** with the required Wiegand format.

   The **Wiegand or TLV format** strings for Schindler credential formats must contain only single underscores. A double underscore in the string will prevent the card data from being transferred in the correct format.

   For more information on configuring custom Wiegand credential formats, see Application Note 276: Configuring Credential Types in Protege GX.

2. Navigate to **Sites | Controllers** and select the controller with this integration enabled.
3. Enter the following command in the **General** tab:

   `SCH_SC = X,37,Y`

   - **X** is the facility/site code that will be formatted.
   - **37** is the **Schindler type** (in this case, Pure Wiegand).
   - **Y** is the Database ID of the credential type created above.

Configuring Access Control for ICT Readers

It is possible to use ICT card readers with the Schindler integration, alongside Schindler destination terminals located on each floor. When a user badges at a card reader the Protege GX controller will process the credential and send the user data to the Schindler system, which will then process floor access as normal.

If only ICT readers are used for this integration, there is no need to configure **Site code formats** in the controller programming, as Schindler is no longer directly processing user credentials. However, it is possible to use a mixture of both kinds of readers by completing the programming for each.

Configuring Schindler Destination Terminals

In this version of the integration, Schindler destination terminals are represented by door records within the Protege GX system. Each terminal on each floor requires a record to be created in Protege GX. The reader expanders controlling those doors must also be configured for elevator control.

Note: Each destination terminal programmed in this integration will consume a single door license. Ensure that you have enough door licenses available for this integration.

1. Navigate to **Expanders | Readers expanders** and select the reader expanders used for this integration.
2. In the **Reader 1** and/or **Reader 2** tabs, ensure that the **Reader 1/2 door** is set to a door that will represent a Schindler destination terminal. Then set the **Reader 1/2 mode** to Elevator so that the 'door' is treated as part of the elevator system.
3. Each door used for elevator access must then be configured using commands. Navigate to **Programming | Doors** and select the doors connected to the above reader expanders.
4. For each door record, enter a **Name** that describes the Schindler destination terminal and where it is located, e.g. 'Schindler Terminal FL5'.
5. In the **General** tab, enter the following **Commands**:
   - `DOP = true`
   - `DOPFloor = X`, where **X** is the **Database ID** of the floor on which the destination terminal is located.

Enabling Home Floor Operation

If your site uses home floor operation, some additional configuration is required for terminals controlled by ICT readers.

1. Navigate to **Programming | Doors** and select the door record which represents the destination terminal.
2. Enable **Door used for elevator HLI**.
3. Select the **Controller** used for this integration.
4. Set the **Operator panel type** to DOP.
5. Fill in the following settings:
   - **DOP ID**: Enter the unique ID of the destination terminal in the Schindler system.
   - **Floor group**: Not required.
   - **Floor**: Select the floor the DOP is located on.
   - **DOP sends elevator call**: Enabled.
   - **Elevator group**: Not required.
   - **Group number**: Not required.
   - **DEC operation mode**: Not required.
6. Click **Save**.
7. Repeat for each terminal that needs to send a home floor call.
8. Ensure that the other required programming for home floors is complete:

- Each floor has the `FloorZone` command programmed (see page 12).
- Each access level has the required **Elevator destination floor** programmed (see page 14).

Programming Antipassback

Optionally, you can use antipassback with operating panels in HLI elevator integrations. For example, you could enable hard antipassback on a turnstile to prevent users from entering the turnstile then passing their card back to someone on the other side.

The operation is the same as standard door antipassback: the system tracks which area each user is in based on their access activity, and reports an antipassback violation if the user is not recorded in the correct area required to enter or exit the door.

Requirements for Elevator HLI Antipassback

- This feature is available in controller firmware version **2.08.1297 or higher**.
- Antipassback should only be used on turnstiles, security gates or similar operating panels with **entry and exit readers** to record movement into and out of the area. When the user enters the turnstile, the system will update their user area and call an elevator based on their home floor or selection. When the user exits the turnstile, it will update their user area without calling an elevator.
- User area tracking is based on the **inside and outside area** programmed in the door record, **not** the floor that the user has selected. For example, the outside area might be the ground floor reception. When a user accesses the turnstile, it grants entry to the elevator lobby and calls an elevator.

  Antipassback should not be enabled for operating panels which are only used to select a floor (such as car operating panels), as the user could be going to any floor instead of entering a specific physical area such as a lobby.
- User credentials must be processed by the Protege GX controller, not the elevator system.
- The **Reader 1/2 elevator** in the reader expander programming must be <not set>.

Programming Antipassback for a Turnstile

1. Navigate to **Programming | Door types** and add a new door type.
2. Set the **Entry passback mode** to control entry through the turnstile and **Exit passback mode** to control exit. The options are:
   - **Hard passback**: Access will be denied if a user attempts to enter/exit the turnstile from the wrong area.
   - **Soft passback**: Access will be granted even if a user attempts to enter/exit the turnstile from the wrong area, but a 'Soft Passback Violation' event will be logged.
3. Optionally, enable **Entry/Exit passback is qualified with door opening**. This will prevent the user's antipassback status from being updated unless they actually pass through the turnstile after badging.
4. Program any other settings required for this door type such as the **Entry/Exit reading mode**, then click **Save**.
5. Navigate to **Programming | Doors** and select a door record which represents a turnstile, security gate or other operating panel.
6. Set the **Door type** to the one programmed above.
7. Set the **Area inside door** and **Area outside door** for the turnstile.
8. Click **Save**.

Configuring Schindler SOMs

SOMs (Special Operating Modes) are used within the Schindler system to perform specific functions such as enabling VIP service when a certain passenger badges their card, or stopping all cars at the nearest floor and opening the doors when a fire alarm is triggered.

Configuring a SOM as an output enables you to trigger it from Protege GX using a variety of methods, such as manually from a status page or floor plan. When the output is activated, the controller sends a message to the Schindler system which then takes the relevant action, such as dispatching an express elevator, locking down all elevators or releasing trapped passengers.

Create Virtual Outputs

For this application, it is convenient to use virtual outputs which do not represent any physical hardware. However, a virtual output must still be associated with an addressed module, a virtual output expander.

1. In **Expanders | Output expanders**, add an output expander with the **Virtual module** option enabled. This option prevents the expander from generating health status messages.
2. Set the **Physical address** to a value above those used by physical expanders (e.g. beginning at 33) then click **Save**.
3. Select the number of outputs to be added (up to a maximum of 16) then click **Add now**.

This automatically creates and addresses the first 16 virtual outputs ready to be configured. If you need additional outputs, they can be added manually.

You can assign up to 255 outputs to one virtual output expander. If there are more than 255 SOMs required, simply add another virtual module.

Configure the Virtual Outputs (SOMs)

1. Navigate to **Programming | Outputs**.
2. Select one virtual output for each SOM and set the **Name** as desired for reference within Protege GX.
3. Set the **Keypad display name** to match the name of the SOM in the Schindler system.
4. Enable the **Output used for elevator HLI** option, then set the required options:
   - **Controller**: The controller used for Schindler integration.
   - **SOM activation mode**: Sends a message to the Schindler system when the output changes state. Choose from On, Off and On Change. If using a single output to activate and deactivate a SOM, select the On Change SOM activation mode.
   - **Append output state to SOM message**: When selected, the state of the output is added to the end of the SOM message. Enable this option if the SOM activation mode is set to On Change.
   - **SOM primary terminal ID**: Defines the ID of the terminal that the schedule on/off message is sent to when the controller is communicating via the primary IP address.
   - **SOM secondary terminal ID**: Defines the ID of the terminal that the schedule on/off message is sent to when the controller is communicating via the secondary IP address.

Configuring Multiple Ports for Call Interface

Multiple connections can be configured for the call interface using the same IP address.

Up to a total of 32 connections can be configured for the call interface. This can be configured via **Sites | Controllers | General** tab. In the **Commands** section, include the following setting:

```
SCH_Call_Port = 5050,5051
```

This command sets the ports that will be used to connect to the call interface, separated by commas.

Configuring Multiple Ports for Life Reporting Interface

Multiple loggers can be configured for the life reporting interface using the same IP address.

Up to a total of 32 loggers can be configured for the life reporting interface. This can be configured via **Sites | Controllers | General** tab. In the **Commands** section, include the following setting:

```
SCH_Life_Port = 6060,6061,6062
```

This command configures the ports that will be used to connect to the various life reporting interface loggers, separated by commas.

Restarting the HLI

When the HLI service is first started, all user details are sent to the Schindler system. After this, changes are only pushed out when certain user details are modified within Protege GX. For more information, see Appendix: User Information sent to the Schindler System (next page).

In the event that the Schindler server fails, the HLI service must be restarted. This resynchronizes the systems by importing all user details into the Schindler online database again.

1. Navigate to **Sites | Controllers**, select the appropriate controller and open the **Configuration** tab.
2. Click on the **Restart HLI** button:

Appendix: User Information sent to the Schindler System

The following information for each user is sent from Protege GX to the Schindler system. Updating these details in Protege GX will result in updated user information being sent to Schindler.

- Database ID

  From firmware version 2.08.1068, Protege GX also sends the Database ID instead of the family name and first name of each user, for improved protection of personal information.

- Company (currently not configurable)
- Enterprise (currently not configurable)
- Department (currently not configurable)
- Schindler user profile template name

  This is drawn from the **Schindler Template** field in **Users | Access levels | General**. If this field is not available or not configured, the name of the floor group assigned to the user's first access level with a valid floor group is sent.

- Three facility/card numbers, each converted to a single credential string
  - Below firmware version 2.08.1068, the first three programmed facility/card numbers are sent
  - From firmware version 2.08.1068, the first three facility/card numbers or credential types that match a programmed site code format are sent

  If a card is disabled, a blank credential will be sent. If a card does not match a configured site code format, the credential will be sent to Schindler as unformatted data.

- Entry date (user expiry start date)
- Exit date (user expiry end date)
- Phone number (currently not configurable)
- Home floor (only available with firmware version 2.08.1068 or later)

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