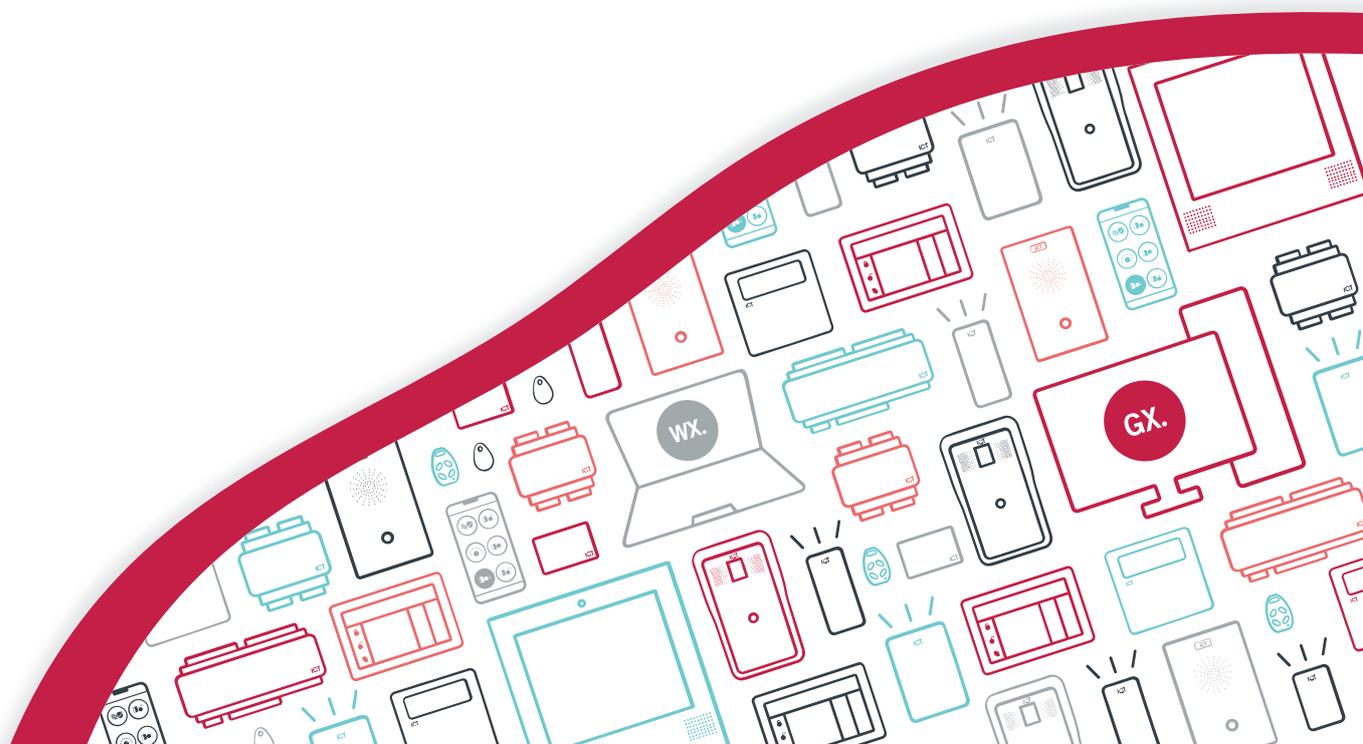




**AN-292**

# **PlateSmart LPR Integration with 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.

Copyright © Integrated Control Technology Limited 2003-2022. All rights reserved.

Last Published: 11-May-22 10:49 AM

# Contents

<b>Introduction</b>	<b>4</b>
Prerequisites	4
Programming Scenario	4
<b>Configuration in Protege GX</b>	<b>5</b>
Enabling the Controller's Onboard Reader Expander	5
Configuring the Reader Expander	5
Creating the Smart Readers	6
Creating the LPR Credential Type	6
Creating the Door Type	6
Creating the Doors	7
Configuring the Smart Readers	7
Assigning Access to Users	7
<b>Installing and Running the LPR Integration Service</b>	<b>9</b>
Installing the Service	9
Configuring the Service	9
Running the Service	10
Running as a Console Application	10
Running as a Windows Service	11
<b>Testing the Integration</b>	<b>12</b>
<b>Troubleshooting</b>	<b>14</b>

# Introduction

---

Protege GX offers integration with the PlateSmart system, combining automatic license plate recognition (LPR) with Protege GX's powerful and flexible access control system. This enables you to use cameras at carpark entry and exit points to grant access, providing enhanced convenience and security for your users.

When a user drives their car toward a carpark boom gate or roller door, the following procedure occurs:

- The camera sends live footage of the car to the PlateSmart system, which uses automatic LPR technology to interpret the license plate.
- PlateSmart sends the license plate data to the **Protege GX PlateSmart LPR Integration Service**, which is a middleware program installed on the Protege GX server or a connected workstation.
- The integration service sends the license plate data to a Protege GX controller.
- The controller recognizes the license plate as a valid credential and grants access, opening the gate to allow the user to drive in. If the license plate is not recognized as a valid credential, access is denied.
- The controller sends the relevant user access events to the Protege GX server, where they can be monitored on a status page or floor plan, or included in an event report.

This application note provides instructions on configuring Protege GX to recognize and grant access to license plate credentials. It also outlines how to install, configure and run the Protege GX PlateSmart LPR Integration Service, and test the integration.

This document only covers the programming that is relevant to Protege GX. Third-party applications and devices may have their own programming requirements.

## Prerequisites

Setting up PlateSmart LPR integration with Protege GX requires:

- An operational Protege GX system running version 4.2.180 or higher.
- A Protege GX controller running firmware version 2.08.0766 or higher.
- The Protege GX PlateSmart LPR Integration Service, version 1.0.0.0 or higher.
- One third party smart reader license (product code: PRT-GX-TPR-IF), applied to the relevant Protege GX SSN, for each camera that will operate in this integration.
- One Protege GX door license per door that is being controlled by this integration.
- An operational PlateSmart LPR system that is configured to send license plate data in XML format to the workstation on which you will install the integration service.

This integration has been tested and validated with PlateSmart version 2.7.19276.1. Other software versions may not be supported.

## Programming Scenario

This application note will use a basic programming scenario to demonstrate how to set up license plate recognition in the Protege GX system. In this scenario, there is a carpark area with two boom gates: one for entry and one for exit. There are two cameras (entry and exit), which will be used to grant users access based on their license plate details.

# Configuration in Protege GX

---

The following programming steps are required in Protege GX.

## Enabling the Controller's Onboard Reader Expander

In this integration, the controller must receive license plate data over the ethernet network. To achieve this, the controller's onboard reader expander must be addressed and it must have its ethernet port enabled.

The integration service can send data to multiple controllers. Each controller's onboard reader expander must be enabled and configured, and the service configured accordingly (see page 1).

If the onboard reader expander is not already enabled, do so as follows:

1. Navigate to **Sites | Controllers** and select the controller that will be used with this integration.
2. In the **Configuration** tab, set the **Register as Reader Expander** field to any address that is not currently being used by a reader expander.
3. Click **Save**.
4. Navigate to **Expanders | Reader Expanders**. Select the relevant **Controller** in the toolbar.
5. **Add** a new reader expander record (**Expanders | Reader Expanders**) and set the **Physical Address** to the address selected above. Click **Save**.
6. In the **Module Configuration** window, review whether you want to create the inputs, outputs, trouble inputs and doors associated with the onboard reader expander. This may not be necessary if the onboard reader expander is only being used to receive license plate data from the integration service. Click **Add Now**.

The doors used for this integration will be created at a later point.

## Configuring the Reader Expander

The controller's onboard reader expander must be configured to receive incoming data over the ethernet network. Each camera in the integration will send data to the controller over a specific TCP/IP port.

1. Select the controller's onboard reader expander and enable the ethernet port by setting the following:
  - **Ethernet network type:** Third party generic
  - **Ethernet port:** The first TCP/IP port that the controller will listen on to receive LPR data from the integration service. The command below will allow the controller to listen over a range of ports above this value. Therefore, you should select a range that will not interfere with any other communications used by the controller (e.g. reporting services).

For this scenario, we will set the **Ethernet port** to 9000.

2. Expand the **Commands** section and enter the following command:

```
SmartReaderPortOffset = true
```

This command allows the controller to receive ethernet data over a range of ports above the one set above. The controller will listen on every port equal to the **Ethernet port** + the **Configured address** of each smart reader associated with the ethernet port (see below).

This allows the data from each camera to be sent to the controller over a unique TCP/IP port so that the controller can distinguish between them.

These ports are used by the integration service to send data to the controller. There is no need to configure the cameras themselves to report LPR data over these ports.

3. Click **Save**. Wait for the programming to be downloaded to the controller, then right click on the reader expander record and click **Update module**.

## Creating the Smart Readers

In this integration, each camera is represented by a smart reader. A smart reader acts like a regular card reader, which receives credential data and sends it on the controller to validate access. In this case, the data is received from the integration service and sent to the controller over the ethernet connection.

For our scenario, it is first necessary to create two smart readers associated with the controller's ethernet port. At a later point these will be configured to control the carpark gates based on LPR credentials.

Smart reader records cannot be saved unless the appropriate licenses have been applied to the server SSN. One third-party reader license (product code: PRT-GX-TPR-IF) is required per smart reader.

1. Navigate to **Expanders | Smart readers** and **Add** a new smart reader record. We will call it Carpark Camera Entry.
2. Set the **Expander address** to the Physical address of the controller's onboard reader expander configured above.
3. Set the **Expander port** to Ethernet.
4. Set the **Configured address** to 1.  
This address will represent the first (entry) camera. The controller will expect to receive data from this camera over a TCP/IP port equal to the **Ethernet port** configured above plus this value. In this scenario, the first camera's data will be transmitted over port 9001.
5. Click **Save**.
6. Repeat the above to create the second smart reader (camera) with the following settings:
  - **Name:** Carpark Camera Exit
  - **Expander address:** The Physical address of the onboard reader expander
  - **Expander port:** Ethernet
  - **Configured address:** 2

The controller will expect to receive data from this camera over port 9002.

7. Click **Save**.

The **Reader** tab is configured at a later stage.

## Creating the LPR Credential Type

The integration service sends LPR data to the controller in ASCII format. First we need to create a credential type that allows the controller to interpret this data and use it for access decisions. The credential type must also be assigned to the smart readers so they can match the incoming data to the correct format.

1. Navigate to **Sites | Credential types**.
2. **Add** a new credential type. We will call it LPR.
3. Set the **Format** to ASCII.
4. Enable the **Unique value** option.
5. Click **Save**.

## Creating the Door Type

A door type must be used to set the LPR credential type as a valid credential for the boom gate doors.

1. Navigate to **Programming | Door types**.
2. **Add** a new door type. We will call this Carpark Boom Gate.

3. In the **Entry** section, set the following:
  - **Entry reading mode:** Custom
  - **Entry credential types:** Click **Add** and add the LPR credential type.
4. In the **Exit** section, set the following:
  - **Exit reading mode:** Custom
  - **Exit credential types:** Click **Add** and add the LPR credential type.
5. Click **Save**.

Optionally, a **Fallback door type** such as Card and PIN may be assigned to this door type to give users an alternative method of opening the boom gates. In this case, the door records created below may be assigned to standard reader expander ports as well as a smart reader so that they are associated with card readers.

## Creating the Doors

Now we can create two door records to represent the entry and exit boom gates for the carpark.

1. Navigate to **Programming | Doors**. Select the relevant **Controller** in the toolbar.
2. **Add** a new door record. We will call this door Carpark Gate Entry.
3. Set the **Door type** to Carpark Boom Gate (created above).
4. In the **Outputs** and **Inputs** tabs, enter the various outputs and inputs used by this boom gate. For example, automatic gates typically include a **Beam input** that ensures that the gate will not close when there is a car under the arm.
5. Configure any other settings required for this door. For example, the **Area inside door** should be set to the carpark area.
6. Click **Save**.
7. Repeat the above steps to create the Carpark Gate Exit door.

## Configuring the Smart Readers

Now the smart readers that were created above can be configured to interpret license plate (ASCII) credentials and control the carpark gates.

1. Navigate to **Expanders | Smart readers** and select the Carpark Camera Entry smart reader created above.
2. In the **Reader** tab, set the **Reader one door** to Carpark Gate Entry. Ensure that the **Reader one location** is set to Entry.
3. Scroll down to the **Reader credential match types** section. Click **Add** and select the LPR credential type. Click **Ok**.
4. Click **Save**.
5. Select the Carpark Camera Exit smart reader and configure the following settings in the **Reader** tab:
  - **Reader one door:** Carpark Gate Exit
  - **Reader one location:** Exit
  - **Reader credential match types:** LPR
6. Click **Save**.

## Assigning Access to Users

For users to access the carpark gates, the doors must be assigned to their access levels. In this case, it is also necessary to assign a license plate credential to each user, just as you would a card number or PIN.

To assign door access to users:

1. Navigate to **Groups | Door groups** and create a new door group called Carpark Gates.
2. Add both Carpark Gate Entry and Carpark Gate Exit to the door group. Click **Save**.
3. Navigate to **Users | Access levels** and assign the new door group to all access levels that require access to the carpark.

To assign license plate credentials to users:

1. Navigate to **Users | Users**.
2. Select a user who requires a license plate credential. Scroll down to the **Credentials** section.
3. Click **Add** to add a new credential and set the **Credential type** column to LPR.
4. Enter the user's license plate into the **Credential** field.
5. In the **Access levels** tab, ensure that the user is assigned an access level with access to the carpark gates, as configured above.
6. Click **Save**. Repeat for all required users.

The ICT Data Sync Service could be used to synchronize license plate data with a third-party system. For more information, see the [ICT Data Sync Service Integration Guide](#).

For our testing scenario below, we will grant carpark access to user Brett Lamb, who has the license plate ABC123.

# Installing and Running the LPR Integration Service

---

This section describes how to install, configure and run the LPR integration service. This service receives license plate data (XML format) from the LPR system and transmits it to the Protege GX controller over the specified TCP/IP ports (ASCII format).

In preparation for this section, ensure that the PlateSmart system is set up to send license plate data in XML format to the workstation that you will be installing the integration service on. Take note of the port that is being used for this data (by default, 399).

## Installing the Service

To install the Protege GX PlateSmart LPR Integration Service:

1. Run the supplied ProtegeGXPlateSmartIntegrationService.msi file to launch the install wizard and click **Next** to continue.
2. Select whether it is to be installed for everyone who uses the PC or just your user, and click **Next** to install to the default location (C:\Program Files (x86)\Integrated Control Technology\Protege GX PlateSmart LPR Integration Service).

If the service is to be installed in a different location, click **Browse** and navigate to the required directory, then click **Next** to install to the selected location.

3. Click **Next** to start the installation. Once the installation is complete, click **Close**.

## Configuring the Service

All configuration of the service is performed by editing the integration service config file. This is an XML file that is installed with the service application.

While modifying the config file, the Protege GX PlateSmart integration service must be stopped. Any modifications made will take effect when the service is restarted. The service does not start upon installation.

To configure the Protege GX PlateSmart LPR integration service:

1. Open the Windows File Explorer and navigate to the installation directory.

If you accepted the defaults during installation this will be C:\Program Files (x86)\Integrated Control Technology\Protege GX PlateSmart LPR Integration Service

2. Open the **LPRSVC.exe.config** file with a text editor.

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.

3. In the **<Client>** section, enter the following:

```
<Client>
  <add IP="192.168.1.7" Port="399" />
</Client>
```

- **IP:** The IP address of the machine that the LPR Integration Service is installed on.

This is not necessarily the IP address of either the LPR or Protege GX servers.

- **Port:** The port used by the service to communicate with PlateSmart. The default port is 399.
4. In the **<Controller>** section, enter an **<add>** element for each camera that will be used by the integration. For our scenario, it would be configured as follows:

```
<Controller>
  <add Camera="Carpark Camera
Entry" ControllerIP="192.168.1.113" EthernetPort="9001" />
  <add Camera="Carpark Camera
Exit" ControllerIP="192.168.1.113" EthernetPort="9002" />
</Controller>
```

- **Camera:** The camera name or ID as programmed in PlateSmart. This will be displayed as the **<VMSCameraName>** in the incoming XML data.
- **ControllerIP:** The IP address of the controller that the license plate data from that camera will be sent to. The integration service can send data to multiple controllers.
- **EthernetPort:** The TCP/IP port that the integration service will use to send data from this camera to the controller. This must be equal to the controller's **Ethernet Port** (see page 5) + the **Configured Address** of the smart reader (see page 6).

If the integration service maps two cameras to the same port, the controller will consider them a single smart reader.

5. Save the file.

## Running the Service

The PlateSmart LPR Integration Service can be run as:

- A console application that is designed for testing purposes only.
- A Windows service for live deployment of the integration.

When the service is running, it will listen for incoming plate data over the port defined in the config file. It will then send this data on to the controller, using a different port for each individual camera.

Note that additional configuration may be required in the LPR application to configure cameras to send XML data to Protege GX.

## Running as a Console Application

The service can be run as a console application to allow live viewing of events. This is designed for testing purposes only as once the console is shut down the service will cease to operate.

1. Navigate to the installation directory.

If you accepted the defaults during installation this will be C:\Program Files (x86)\Integrated Control Technology\Protege GX PlateSmart LPR Integration Service.

2. Right click on the **LPRSVC.exe** file and select Run as Administrator. This will open a console logging the incoming and outgoing data from the service.
3. If the Windows Defender Firewall is on, you will see a Windows Security Alert popup indicating that some features of the integration service have been blocked.  
To allow the service to function, check the boxes to allow the service to communicate on **Domain networks** and **Private networks** (or as appropriate for your installation). Then click **Allow Access**.
4. If the application is configured correctly, it should display the following messages:

```
Entered OnStart()...
Service started successfully.
Starting RunProcess{}
```

Listening on [192.168.1.7:399]

The IP address of your workstation should be displayed in the final line.

5. When the integration service receives data from PlateSmart, it will be displayed here in XML format.
  - The **VMSCameraName** field in the XML data contains the camera name or ID. This is matched to a **Camera** in the LPR section of the config file, and the data is downloaded to the corresponding controller over the defined **ControllerIP** and **EthernetPort**.
  - The **PlateTag** field in the XML data holds the license plate that is sent to the Controller.

The console will display the result of each communication and whether the controller accepts the license plate credential.

## Running as a Windows Service

The service can be started from the Windows Service Manager.

1. Open the **Services** snap-in by:
  - Pressing the **Windows + R** keys
  - Typing **services.msc** into the search bar and pressing **Enter**
2. Navigate through the list of services and locate the **Protege GX PlateSmart LPR Integration Service**.
3. Right click on the service and select **Start**.

To view events logged by the service, navigate to the installation directory to locate the debug.log file, which can be opened with any text editor.

## Disabling Service Logging

For live deployment of the integration it is recommended that logging is disabled. To disable logging to the debug.log file:

1. Open the Windows Service Manager (see above) and stop the **Protege GX PlateSmart LPR Integration Service**.
2. Open the LPRSVC.exe.config file with a text editor. Locate the section in bold below and change the **<threshold>** value from **All** (debugging enabled) to **Off** (debugging disabled).

```
<appender name="RollingLogFileAppender"  
type="log4net.Appender.RollingFileAppender">  
  <threshold value="Off"/>  
  ...  
</appender>
```

3. Return to the Windows Service Manager, right click on the **Protege GX PlateSmart LPR Integration Service** and click **Start**.

# Testing the Integration

---

Once both Protege GX and the integration service have been configured, we can test the entire PlateSmart integration.

This test assumes that you have access to the PlateSmart system. If this is not available, the integration can be tested by sending the relevant XML data to the integration service via another application, such as [TCP Client Server](#). Any connections of this sort used for testing should be disabled in the firewall before deployment.

In this testing example, building manager Brett Lamb (license plate: ABC123) will enter the carpark via the Carpark Gate Entry.

1. Ensure that the PlateSmart system is ready to send LPR data to the integration service.
2. Run the PlateSmart LPR Integration Service as a console application so that you can view the live service event log. Navigate to the installation directory, right click on **LPRSVC.exe** and click **Run as Administrator**.

Alternatively, you can run the application as a windows service with logging enabled (see previous page), and view events in the log file.

3. The service should log the following events when it starts and receives a connection from PlateSmart:

```
Entered OnStart() ...
Service started successfully.
Starting RunProcess{}...
Listening on [IntegrationServiceIP:Port]
Client connected [PlateSmartIP:Port]
```

4. Brett Lamb drives towards the boom gate. The Carpark Camera Entry captures his license plate and sends it to PlateSmart, which sends it to the integration service. XML data similar to the following is displayed in the console or log:

```
<PlateSmartMetaData-V3>
  <Plate>
    <EventDateTime>Thursday, August 06, 2020 3:59:28.541
  PM</EventDateTime>
    <PSCameraID></PSCameraID>
    <PSCameraName></PSCameraName>
    <VMSCameraID>1</VMSCameraID>
    <VMSCameraName>Carpark Camera Entry</VMSCameraName>
    <PlateID>0</PlateID>
    <PlateTag>ABC123</PlateTag>
    <PlateStateName>Ohio</PlateStateName>
    <PlateCountryName>United States Of America</PlateCountryName>
    <VehicleImagePath></VehicleImagePath>
    <PlateImagePath></PlateImagePath>
    <Longitude>1</Longitude>
    <Latitude>1</Latitude>
    <MapLocationURL></MapLocationURL>
    <AccountName></AccountName>
  </Plate>
</PlateSmartMetaData-V3>
```

5. The integration service sends the data on to the controller over the port defined for that camera. If it is successful, you will see the following messages:

```
Number of plates found = 1
1 valid plates found.
Found camera 'Carpark Camera Entry' OK.
```

```
Attempting to send License Plate 'ABC123' to Controller  
[ControllerIP:9001]  
Successfully downloaded license plate to controller.  
Buffer contents (post-processing):
```

6. The controller receives the credential and evaluates whether the user is allowed access. You should see the following events in the Protege GX event log:  
    User Brett Lamb (UN1) Granted Entry To Carpark Gate Entry (DR22) Access  
    Level Manager (AL1) Using Credentials : ABC123  
    Door Carpark Gate Entry (DR22) Unlocked By Access
7. The carpark entry boom gate will open.
8. When Brett Lamb leaves work, the above steps are repeated for the Carpark Gate Exit.

When testing is complete, it is recommended that you disable logging for the integration service (see page 11).

# Troubleshooting

---

1. **Problem:** The license plate data is not being received by the controller, and the integration service displays the following event:

```
Exception raised in SendDataToController(): Failed to send License Plate 'ABC123' to Controller
```

**Cause:** This error will be raised in situations where:

- The controller is offline.
- An incorrect controller IP address has been provided in the **<Controller>** section of the config file (see page 9).
- The port that is configured in the config file does not match a port that the controller is listening on for smart reader data, so the controller refuses the connection.

The exception message should provide more information about the issue.

2. **Problem:** The license plate data is not being received by the controller, and the integration service displays the following event:

```
Provided <VMSCameraName> 'Carpark Camera 1' in XML data could not be matched to a 'Camera' in the <LPR> section in the config file.
```

**Cause:** The PlateSmart data uses a camera that does not match the cameras configured in the integration service. Ensure that the **<VMSCameraName>** sent by PlateSmart exactly matches an **<add Camera="" />** element in the **<Controller>** section of the config file (see page 9).

3. **Problem:** The data has successfully downloaded to the controller, but the door does not open. The Protege GX event log displays a 'Read raw credential data' event.

**Cause:**

- There may be no user with this credential assigned. If this is the case, right click on the event to assign the credential to a new or existing user.
- If the credential has been assigned already, that user may not be downloaded to this controller. Ensure that the user has a valid access level.

Designers & manufacturers of integrated electronic access control, security and automation products.  
Designed & manufactured by Integrated Control Technology Ltd.  
Copyright © Integrated Control Technology Limited 2003-2022. All rights reserved.

**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.