

How to integrate OPTEX RLS3060SH Lasers

in Bosch DIVAR IP all-in-one 5000 recorders and Bosch VMS

An all round laser guided BVMS Management solution

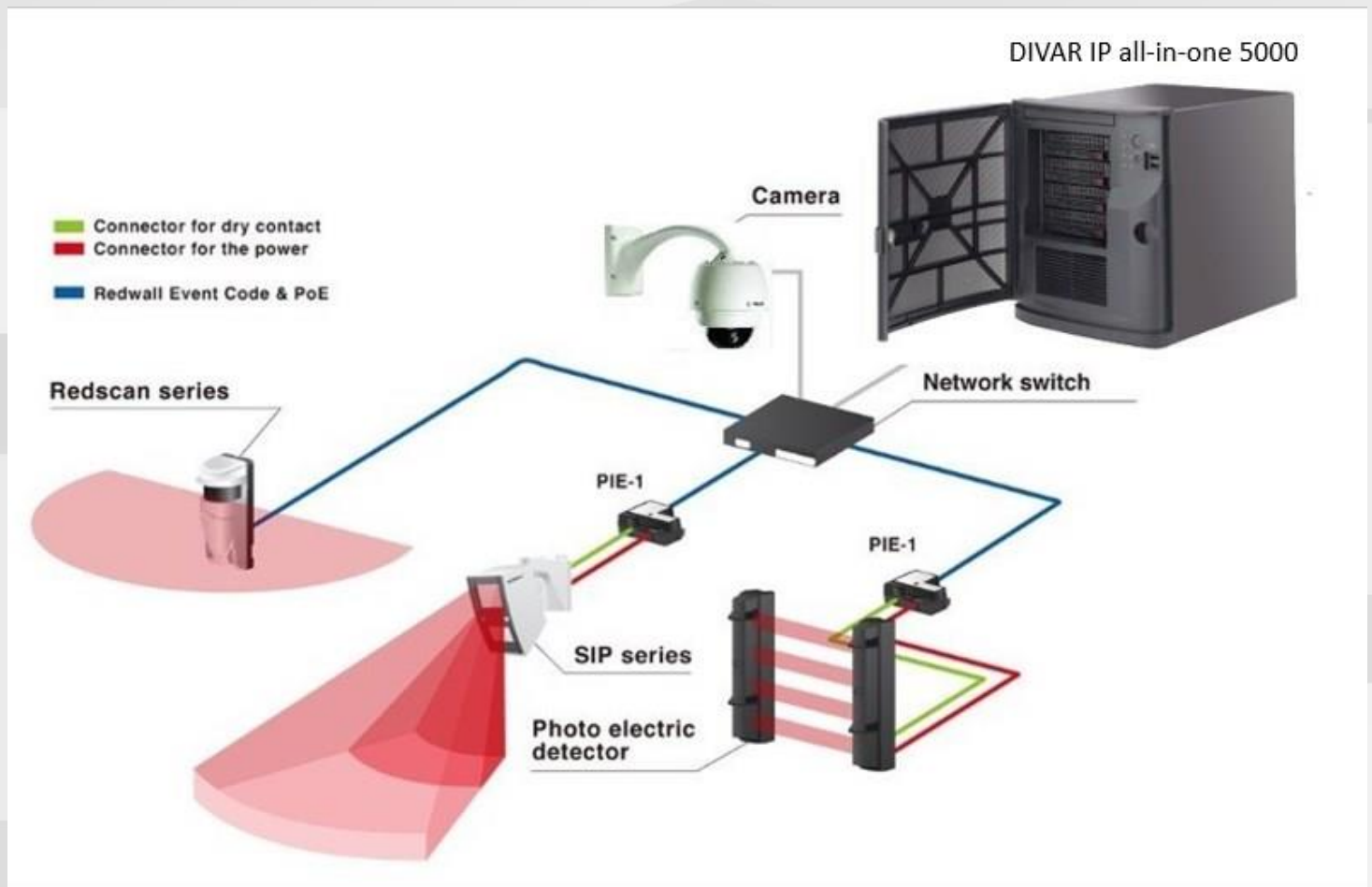


Table of Contents

| | |
|---|-----------|
| 1. Overview of the Integration | 4 |
| 1.1 Benefits of the integration..... | 4 |
| 1.2 Principle of the integration..... | 4 |
| 1.3 OPTEX products tested for this integration | 5 |
| 1.4 Bosch products used for this integration | 6 |
| 2. Technical details and Setup of the integration | 8 |
| 2.1 OPTEX RLS3060SH device preparation..... | 8 |
| 2.2 Bosch DIVAR IP all-in-one 5000 preparation..... | 9 |
| 2.2.1 Define the ATM/POS interface service | 9 |
| 2.2.2 How to configure the ATM/POS Service for OPTEX devices | 9 |
| 2.3 DIVAR IP all-in-one 5000 BVMS Configuration Client preparation..... | 10 |
| 3. Test and see the integration at work ! | 13 |
| 3.1 Test Setup | 13 |
| 3.2 OPTEX detection of specific Alarms as shown in OPTEX GUIs | 14 |
| 3.3 Event Code and alarm reporting as shown in BVMS Operator Client..... | 15 |
| 3.4 Search for video of an OPTEXT Event in the DIVAR IP Recorded Data Base..... | 15 |
| 3.5 OPTEX events found with Event Code as shown in the Operator Client GUI | 16 |
| 4. How to control BVMS via OPTEX Laser detection events | 18 |
| 4.1 Where to start ?..... | 18 |
| 4.2 The principle of controlling BVMS via external events | 18 |
| 4.3 A Server script to process incoming Event Data..... | 19 |
| 4.4 Creating Scriptlets | 19 |
| 4.5 Bringing your Server Script into action..... | 20 |
| 5. Appendix A Practical examples | 22 |
| 5.1 REDSCAN RLS-3080SH Laser alarm fires dedicated Bosch PTZ camera Presets | 22 |
| 5.2 REDSCAN RLS-3060SH Laser alarm populates Operator screen with 4 cameras | 23 |
| 5.3 Debug hints for Integrators testing the examples | 26 |
| 6. Appendix B Reference Sites for further info | 27 |
| 7. Disclaimer | 27 |

History

| Version | Date | Author | Description |
|---------|--------------|-----------------------|--------------------------------|
| V1.1 | January 2020 | Jan Noten (BT-SC/MKI) | First version for distribution |

1. Overview of the Integration

This integration Application Note outlines the feature set, connectivity, interfacing and settings to be done in order to use the OPTEX product line in BVMS or BVMS managed products like DIVAR IP all-in-one 5000 Series.

This Application Note also assumes that the reader is knowledgeable on both Bosch BVMS as well as OPTEX products.

1.1 Benefits of the integration

This integration solution with the OPTEX sensors and Bosch BVMS products provides security solutions for a wide range of application sectors like Remote Video Monitoring, Petrol/Gas, Government & Military, Transportation, Retail & Logistics, Industrial Sites, Banking, Museums & Art Galleries, Leisure, Residential & VIP.

The OPTEX detectors provide accurate detection information that is passed on to BVMS allowing the security manager to set up any desirable video follow up scenario like automated Goto Preset of IP cameras and/or start recording scenarios. Furthermore the OPTEX detector information will be stored in BVMS and can be used as forensic search data to easily find corresponding incidents as recorded video evidence. This OPTEX-BOSCH integration provides a tailor made solution for almost any intrusion scenario imaginable.



1.2 Principle of the integration

OPTEX lasers detect, analyze and track objects and can be programmed to scan areas, horizontally and vertically. At any alarm, the OPTEX device will create a ASCII string or so called REDWALL Basic Event Code that is passed on to the Bosch interface embedded within the DIVAR IP all-in-one 5000. This interface is called ATM/POS_Service. This interface will capture the Basic Event Code, filters and processes the data and passes this on to the internal BVMS standard ATM/POS socket. The ATM/POS service is capable to process up to 16 OPTEX devices at the time allowing 16 OPTEX devices to send their detector information to BVMS at any time. The OPTEX device will send the OPTEX Basic Event Code format that describes the detected infringement. This Basic Event Code format and syntax looks like an ASCII string of the following format and content (Fig1):

List of basic event codes (Redscan codes shown)

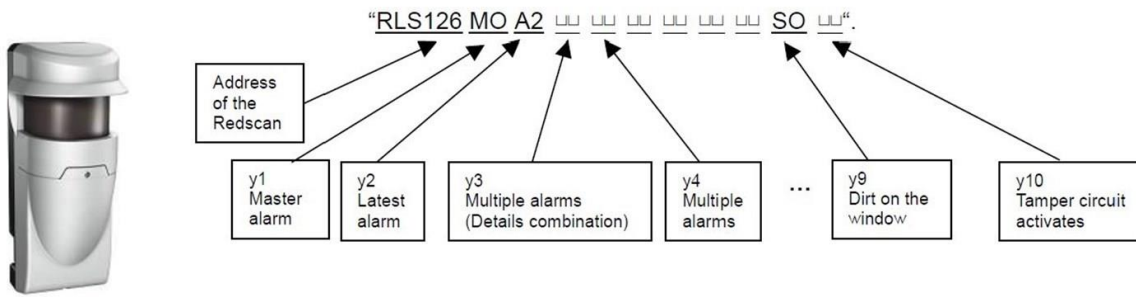


Fig. 1 Basic Event Codes

OPTEX Event Codes are product type dependent. Please consult OPTEX support for the latest updated list of codes (*R.E.C. Event codes with combi codes.xlsx*)

Inside the BVMS configuration, the Basic Event Code data is regarded as Text Data and can internally be stored and processed as Event Data. This allows the user to do dedicated BVMS event search to quickly find all events of certain types but also allows the installer to pre-program dedicated tailored scripts in the BVMS Server to activate a live (distributed) alarm and enhanced follow-up scenario for Operator Workstations and/or cameras on the site.

1.3 OPTEX products tested for this integration

The OPTEX products tested are:

- a. REDSCAN RLS-3060SH Intrusion Detector Laser Unit

Find details here: <https://www.optex-europe.com/products/intrusion-detection/rls-3060sh>



Fig. 2 REDSCAN RLS-3060SH

- b. REDWALL PIE-1 Encoder 5 input detector

Find details here: <https://www.optex-europe.com/products/intrusion-detection/pie-1>



Fig. 3 PIE-1 Encoder

- c. Redscan Manager Version: 8.3.0.3 Software
 This software can be downloaded or obtained via OPTEX.
tech-support@optex-europe.com

Note:

This AppNote will concentrate on the integration set up of the RLS-3060SH Laser in BVMS only. In principle, all other OPTEX products that supply the same interfacing via the OPTEX Basic Event Code format, can be used in the same way as this RLS-3060SH and are equally to be programmed in BVMS as shown below.

1.4 Bosch products used for this integration

The Bosch DIVAR IP all-in-one 5000 products are all standardly licensed in order to be used for this integration.

- a. All DIVAR IP all-in-one 5000 versions as described in the Datasheet here can be used:
https://resource.boschsecurity.com/documents/DIP_ao_5000_Data_sheet_enUS_71312221451.pdf



Fig. 4 DIVAR IP all-in-one 5000

- b. ATMPOS_Service_1.00.00.09_Package.
 This can be retrieved via IPP or via the following link:
<https://partner.ipp.boschsecurity.com/cloud/index.php/s/FNsWoWrj2xdTX06>

The ZIP Package includes the following files. See Fig.5

| Name | Size |
|---|-----------|
| VC++ 2010 SP1 Redistributable | 4 995 416 |
| ATM POS Service Installation Manual.pdf | 284 796 |
| Setup.msi | 2 097 664 |

Fig. 5 ATMPOS_Service_1.00.00.09_Package

The Setup.msi should be run in Administrator mode (right mouse.. Run as Administrator)
The installer will also copy a document “ATM POS Service User Guide.pdf” to you disk.

2. Technical details and Setup of the integration

2.1 OPEX RLS3060SH device preparation

To setup the REDSCAN Laser detection areas, masks, I/O settings and Network specs you can use the OPEX REDSCAN Manager software as supplied via OPEX.

Make sure that the Laser Network Settings are programmed correctly (The Laser default IP number is 192.168.0.126) . In the illustrations following, the Laser IP has been reconfigured to IP number 192.168.10.126

In order to connect the Laser to the DIVAR IP all-in-one 5000 you must also program the DIVAR IP socket destination here where the Laser Basic Code Events have to be reported to. This is done via the I/O button in the top menu line of the REDSCAN Manager Software.

See figure 6.

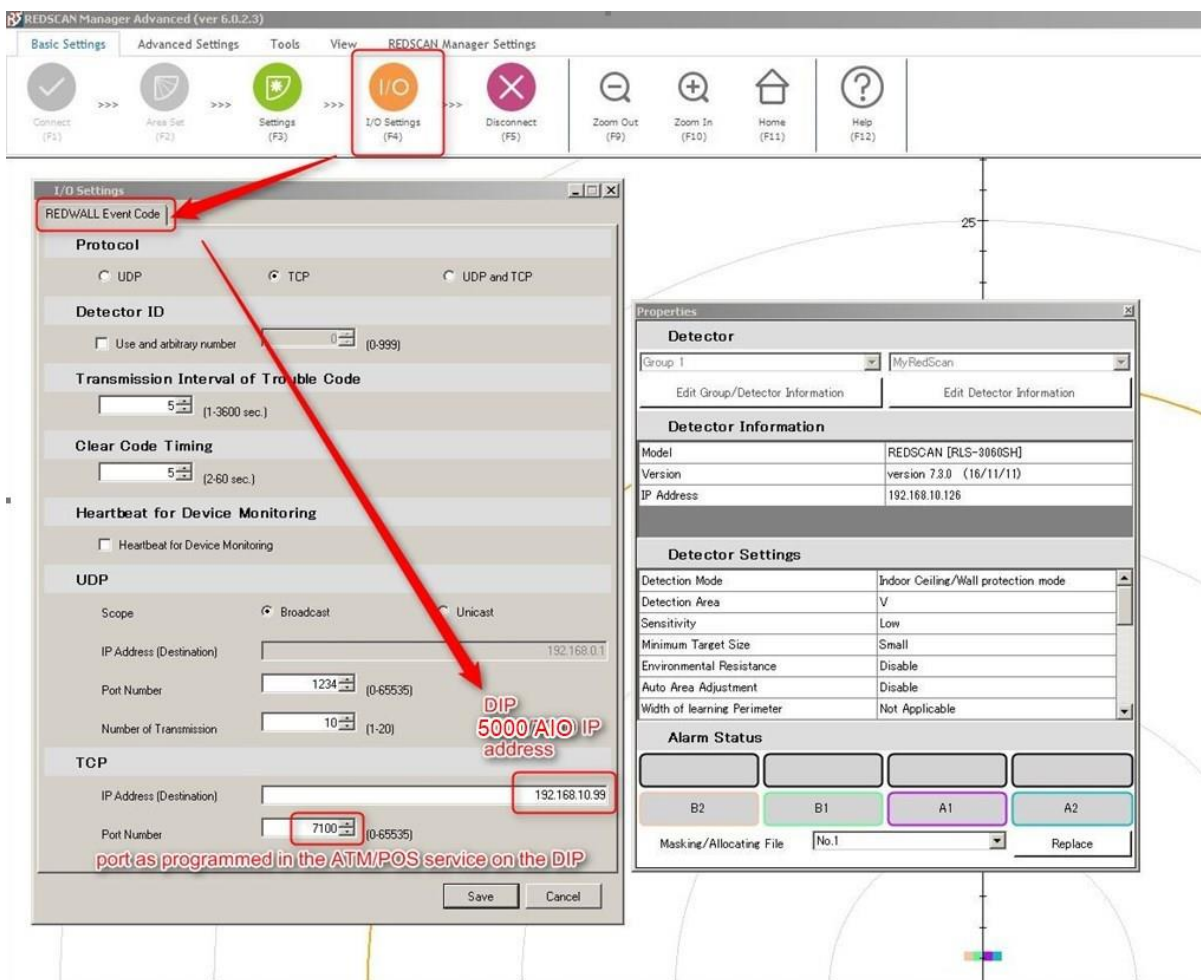


Fig. 6 REDSCAN Manager Software

Since the ATM/POS service, which has to receive the Event code, is running on the DIVAR IP all-in-one 5000 unit, you have to insert the IP number and the ATM/POS service port (in this case 7100) that corresponds to the Terminal-ID in the ATM/POS service configuration receiving this event.1

2.2 Bosch DIVAR IP all-in-one 5000 preparation

2.2.1 Define the ATM/POS interface service

The ATM/POS Service is a separate Windows service that runs in the background on a DIVAR IP all-in-one 5000. It provides a general data collection interface between an external application and BVMS.

It therefore takes external ASCII Text data received on its IP socket and passes this on to the ATM/POS internal IP input of DIVAR IP all-in-one 5000 (BVMS) local host IP (127.0.0.1).

Please install the ATM/POS Service in the **C:/Program Files(x86)/Bosch/ATM POS Service** directory of a DIVAR IP all-in-one 5000, you can use this service now on the DIVAR IP all-in-one 5000 right away.

For BVMS Systems, the service can be installed separately on the BVMS Server or any other Windows PC in the Network.

Please consult the **ATM POS Service Installation Manual** of the **ATM/POS Service** to install the ATM/POS service step by step in case you use a BVMS system and not a DIVAR IP all-in-one 5000.

2.2.2 How to configure the ATM/POS Service for OPTEx devices

The next step is to set up the OPTEx channels as an Text Data supplier device.

A new Text Data source like OPTEx devices can be added using the configurator that can be found here:

C:/Program Files(x86)/Bosch/ATM POS Service BoschATMPOSConfigurator.exe See Fig.7.

This configuration tool can be used to add, to edit, to delete, and to save these configuration settings.

This will create the Service.xml file content for the runtime environment at start of the service and also for the BoschATMPOSConfigurator tool as own settings .

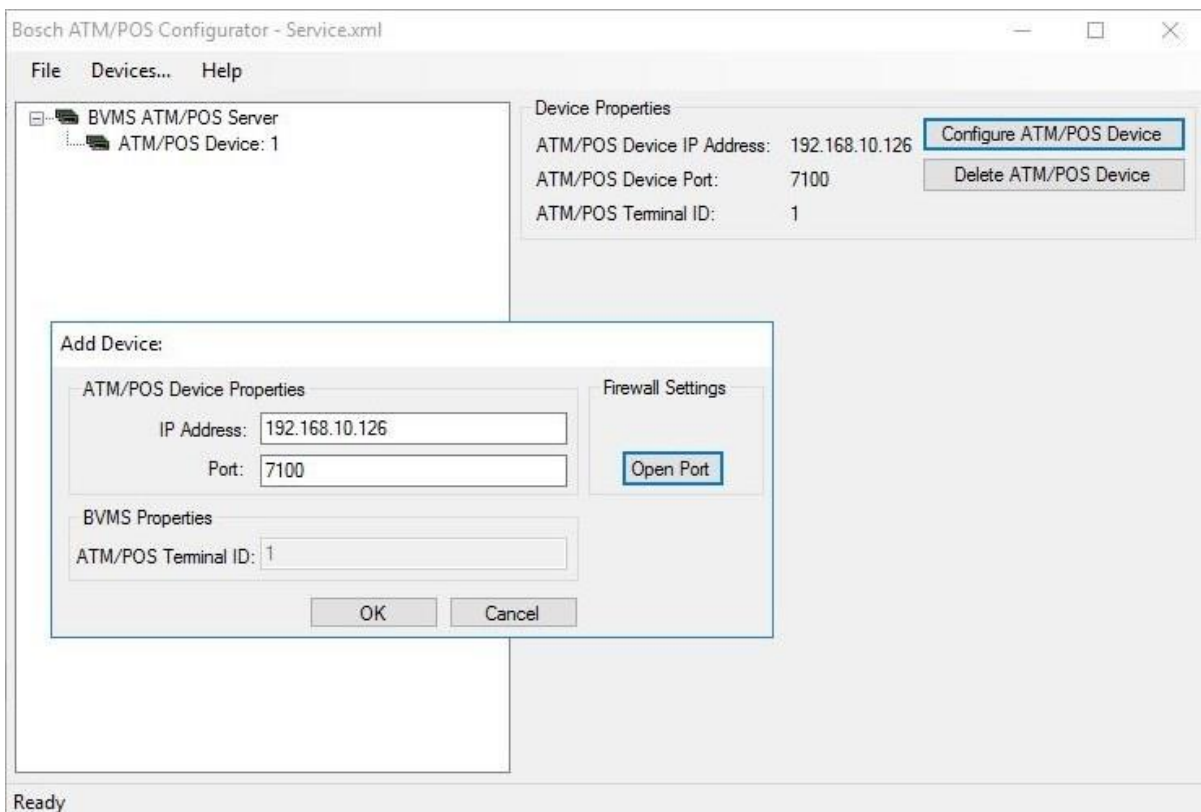


Fig. 7 The Bosch ATM/POS Configurator Tool

Note:

The button “Open Port” automatically opens the defined port in your firewall to receive the Text Data.

Each ATM port must be unique for each Text Data device configured.

The default configuration file **Service.xml** also should be located in the same folder. At launch, the BoschATMPOSConfigurator tool should load the configuration settings from the **Service.xml** file. The user can manual modify and save things.

Note: the Codepage language is fixed to English.

The **Service.xml** shows then like Fig8:

```
<Service>
  <DefaultAtm TerminalId="1" CodePage="US" DeviceId="0" />
  <Atm IpAddress="192.168.10.126" TerminalId="1" CodePage="US" DeviceId="0" />
  <Brd IpAddress="127.0.0.1" ConnectionPort="4201" TxToBrdPort="4200" RecvTimeout="250" TempFolder="C:\AtmPosDump" RecordingLanguage="GERMAN">
    <Terminal Id="1" Name="POS Input 1" />
  </Brd>
  <AtmListener TcpPort="7100" RecvTimeout="1000" MaxSizeData="7000" AcceptsAny="1" />
</Service>
```

Fig.8 The file **Service.xml** with all initial settings

A finished setup looks like Fig. 7 and 8

This is all it takes to setup the data capture service for the REDSCAN Laser.

In the same directory mentioned above you can find this service named as **AtmPosService.exe** . You can check if this service is running as a Windows background (consult services.msc). After a configuration change, either in Service.xml or via the tool, you must manually restart this ATM/POS service.

To do this, please run the file **RestartATMPOSService.bat** in the install directory.

(BVMS restarts this service automatically at every restart of the BVMS server).

Note: If the service does not start then there could be an error in your configuration like the same port used twice..

2.3 DIVAR IP all-in-one 5000 BVMS Configuration Client preparation

In order to receive the OPTEX Event data and to store this data along with video footage also BVMS must be setup to receive the data that is routed from the OPTEX device via the ATM/POS service towards the BVMS server in the DIVAR IP (or on a PC)

To configure the DIVAR IP or BVMS for this integration, you can use the BVMS Config Client as installed in the Start menu of Windows.

The steps to do are:

- a. Define an ATM/POS device (representing the ATM/POS service)
- b. Define the Event reaction in BVMS resulting from this event and when.
- c. Define which camera should record the event data along its video footage
- d. Restart the BVMS server and Operator Clients.

The various figures below show these steps.

There are 16 inputs in the following picture on the right. Make sure that the input number corresponds with the TerminalID of the defined ATM/POS device in Fig 7.

Configuration Client (127.0.0.1, User: Admin)

System Hardware Tools Reports Settings Help

Devices Maps and Structure Schedules Cameras and Recording Events

Change device passwords

Device Tree [11]

- Enterprise System [1]
- Bosch Recording Station/DiBos
- DVR (Digital Video Recorder)
- Matrix Switches
- Workstations [1]
- Monitors [3]
- Other Devices [12]
 - Communication Devices
 - ATM/POS [1]
 - OPTEX Events**
 - Foyer Card Readers
 - Virtual Inputs
- SNMP [2]
- CCTV Keyboards
- I/O Modules
- Allegiant CCL Emulation
- Mobile Video Services
- Intrusion Panels
- Video Analytics

Bosch ATM/POS-Bridge **Inputs**

| | | | | |
|---|-------------------------------------|------------|--------------------------|--|
| 1 | <input checked="" type="checkbox"/> | RLS-3080SH | <input type="checkbox"/> | |
| 2 | <input type="checkbox"/> | | <input type="checkbox"/> | |
| 3 | <input type="checkbox"/> | | <input type="checkbox"/> | |
| 4 | <input type="checkbox"/> | | <input type="checkbox"/> | |
| | <input type="checkbox"/> | | <input type="checkbox"/> | |
| | <input type="checkbox"/> | | <input type="checkbox"/> | |
| | <input type="checkbox"/> | | <input type="checkbox"/> | |
| | <input type="checkbox"/> | | <input type="checkbox"/> | |

use here the same input number corresponding to the Terminal ID as defined in figure 7

Fig. 9 Define an ATM/POS device (representing the ATM/POS service)

To define the OPTEX device event follow up in the Event TAB, see Fig 10 below.

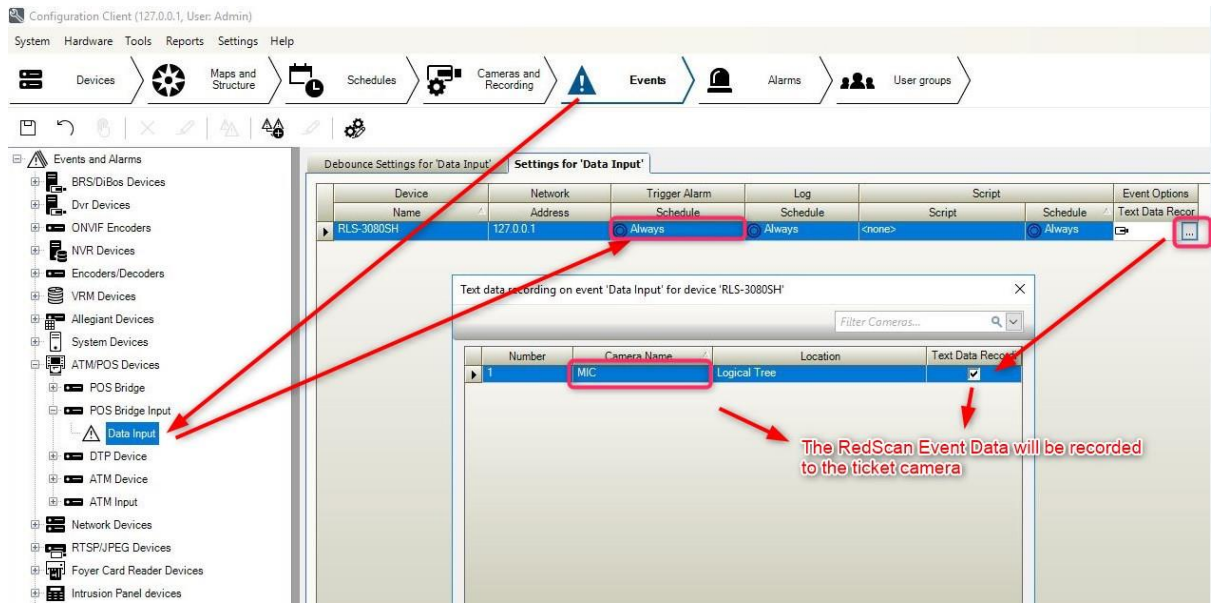


Fig. 10 Define the Event reaction in BVMS resulting from this event and when.

In the above setup you can see follow-up programmed that initiated an alarm (scheduled as “Always”) for that REDSCAN Data Input Event. On the far right you find the button to program to which camera(s) the Event Text data should be stored.

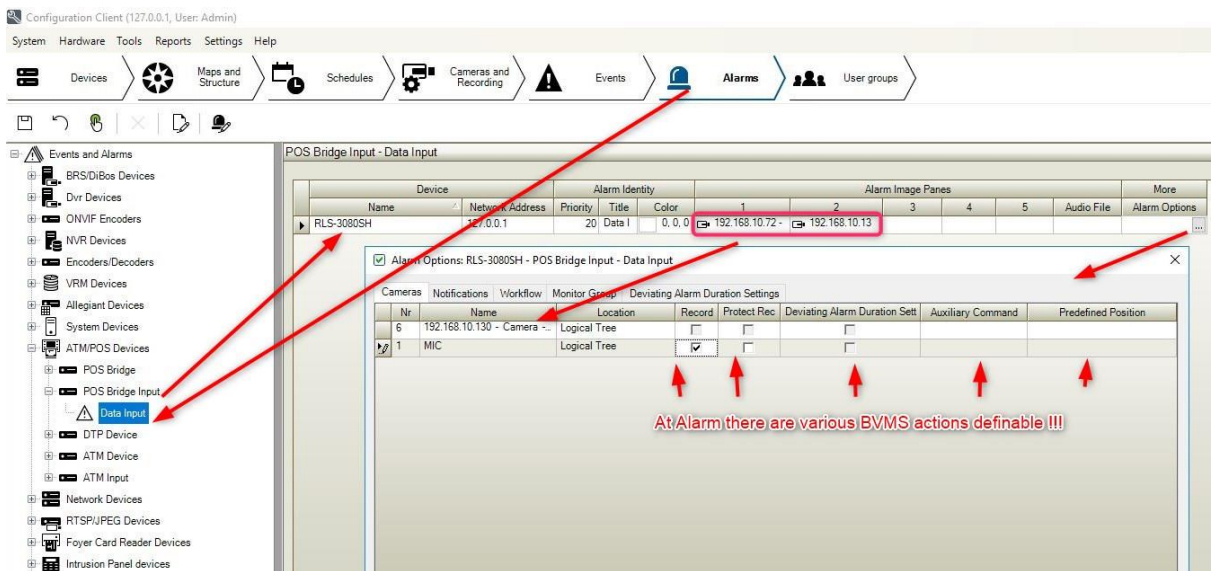


Fig. 11 Define the ALARM reaction in BVMS resulting from this event and when.

The above alarm settings and features addressed are activated as a result of an alarm acknowledge. Also notice in the red circle that the REDSCAN Laser detector will automatically call-up the 2 event involved Cameras.

When all BVMS settings are inserted you can re-activate the BVMS Server by pressing the Activation button in the left top screen of the Config Client. See Fig. 12

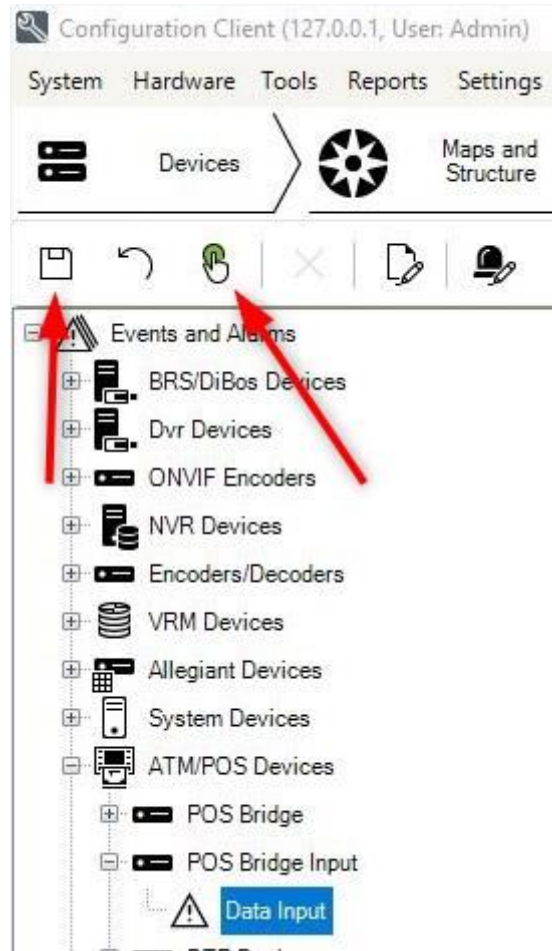


Fig. 12 Restart the BVMS server and Operator Clients.

Also at this point, the Operator Clients must restart with the new database content and will then show the altered settings in their GUI.

3. Test and see the integration at work !

Provided you have followed the above guideline and programmed the various elements correctly, you can now put the OPTEX – Bosch integration to the test.

3.1 Test Setup

For the DIVAR IP , the ATM/POS service Version 1.00.00.09 was used on a DIVAR IP all-in-one 5000 with Version 10.0 BVMS and some cameras were added to the system.

To test the OPTEX device Event Code generation, as listed in Fig. 1 , the REDSCAN Laser was forced into Alarm Mode via an intrusion simulation.

This was done on the REDSCAN laser by setting up a scan environment with 4 Alarm zones (B2,B1,A1,A2) and penetrating the zones.

3.2 OPEX detection of specific Alarms as shown in OPEX GUIs

See the device alarm in below Fig13

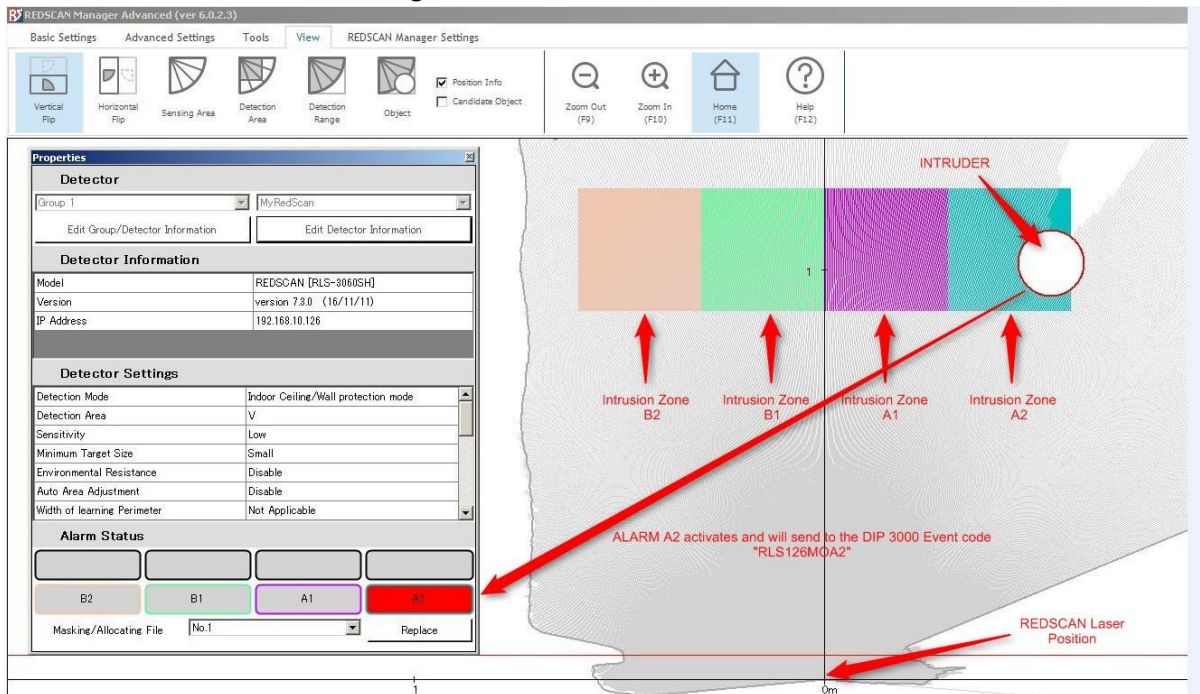


Fig. 13 A REDSCAN laser event triggers an alarm in Zone 2 in the REDSCAN Manager GUI

The device's alarm will be send along with their dedicated Event Codes to the DIVAR IP all-in-one 5000. The default Event Code start characters , identifying the device in the below screen shot are:

- REDSCAN Laser (192.168.10.126) = **RLS126.....**

At an OPEX alarm, the Event Code transmission to the DIVAR IP will trigger the DIVAR IP event settings. As a result and according to the DIVAR IP settings, this event will be processed.

3.3 Event Code and alarm reporting as shown in BVMS Operator Client

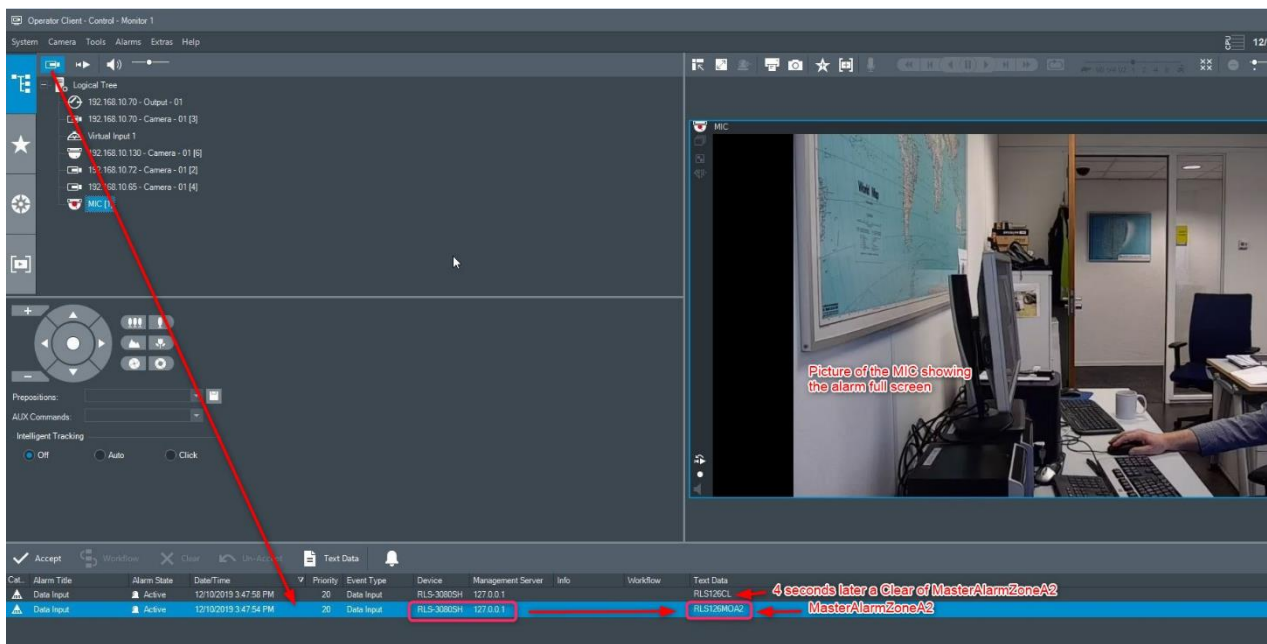


Fig.14 The Operator Client receives the REDSCAN Laser Master Alarm for Zone A2

The Event Code received along with the Alarm will also be stored to the recordings of the camera(s) that you nominated in Fig.10 in the tickboxes for the column called Text Data Recording. We will use this for further tests if needed.

3.4 Search for video of an OPTTEXT Event in the DIVAR IP Recorded Data Base

For finding video Evidence you can either use the “Find Video By Event” button in BVMS or use the “Text Data Search” button (as shown in Fig 15).

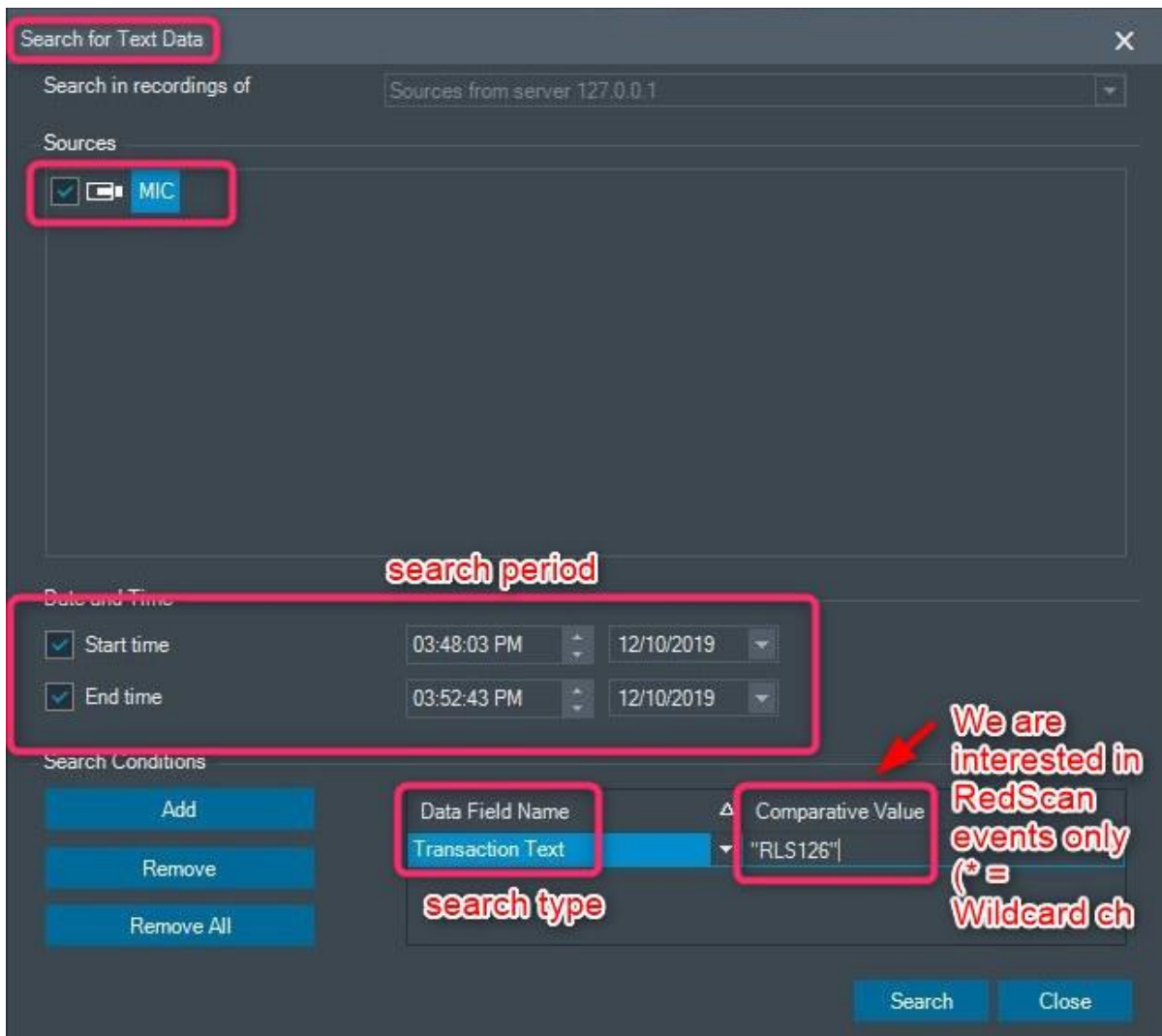


Fig. 15 Find any OPEX event over a certain time period for a particular device.

3.5 OPEX events found with Event Code as shown in the Operator Client GUI

After the search has finished, the result shows in the Operator Client with all evidence on one screen.

The recorded event data that can be consulted here is:

1. all Alarm details like time / date , source device etc (also logged in log files).
2. The involved camera details closest to the event location.
3. The reported OPEX Event Code synchronously reproduced from the live occurrence

The Operator can instantly use all playback options and also click any other search result lines for further consultation. See Fig. 16

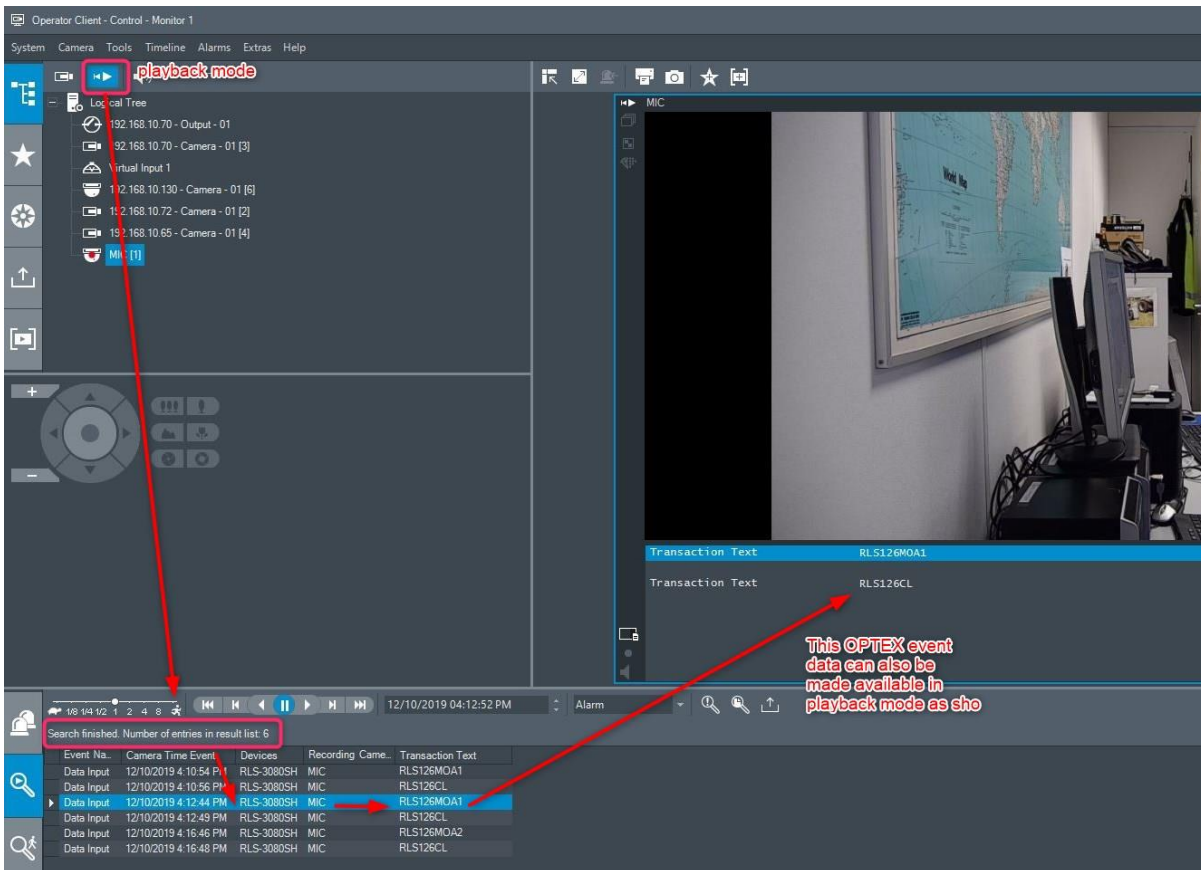


Fig. 16 List and show any OPEX REDSCAN event over a certain time period for a particular device.

If this is all working fine, you can start to fine tune the integration application.

The options are that you activate functions and features via the Event and Alarm Screens in the Configuration Client as a result of OPEX messages.

It is easy for instance to filter on specific OPEX Event Code in order to invoke specific action in BVMS as a follow-up to the incident for operators.

To design such specific action, the Integrator can extend the standard BVMS feature set for his end user or tailor the integration solution by programming scriptlets in the BVMS on-board Script Editor/Compiler in the Config Client of BVMS. This extra functionality allows Integrators to invoke adequate BVMS response on a very high level of automation and optional integration communication to on-site subsystems.

You can find some examples in the Appendix of this document.

The details about how to compose, edit and compile C# or VB .NET scriptlets are part of the BVMS Advanced Integration Workshop in Eindhoven. This workshop teaches you how to design integration features which are only limited by your own imagination. Scripting is outside the scope of the main BVMS Software feature set.

For further details, please contact the EMEA Integrator Partner Program (IPP) department in Eindhoven.

4. How to control BVMS via OPTEX Laser detection events

4.1 Where to start ?

In order to build tailored event follow-up for BVMS and BVMS controlled devices such as presets for PTZ cameras or populating Operator Client screens with involved cameras automatically, the above described Standard Integration should work OK first. It has no sense to continue building new features at this point when the Standard Integration lacks incoming event code generation or lacks communication between BVMS and OPTEX.

Following the guidelines above, a working Standard Integration will show the R.E.C event codes in the BVMS Operator Client as alarm overview line each time the OPTEX device detects an incident. See fig 16 above.

With the Standard Integration, any OPTEX event code sent to BVMS generates an alarm and the event code can be observed by the Operator. So far, there is no distinct or dedicated follow-up inside BVMS for each alarm individually. If dedicated action is desired to happen you must create a so called Server Script in BVMS's Server Script code editor to follow up this alarm tailor made. Also, as we will learn later, via a Client Script we can follow up this alarm event on the local Operators Workstation like populating incident cameras to the operators screen.

4.2 The principle of controlling BVMS via external events

In principle, events are "triggers" which can occur inside the BVMS Server or activated by external IP network devices such as OPTEX. With this type of integration, the OPTEX events propagate from the IP network via the background ATM/POS service into the core of the BVMS server. Such an event carries detailed event data about the Laser detected event.

In the Configuration Client of BVMS (in the **EVENT TAB**) you can link each ATM/POS incoming Data Input, containing the OPTEX event (ASCII) code, to a Server Script.

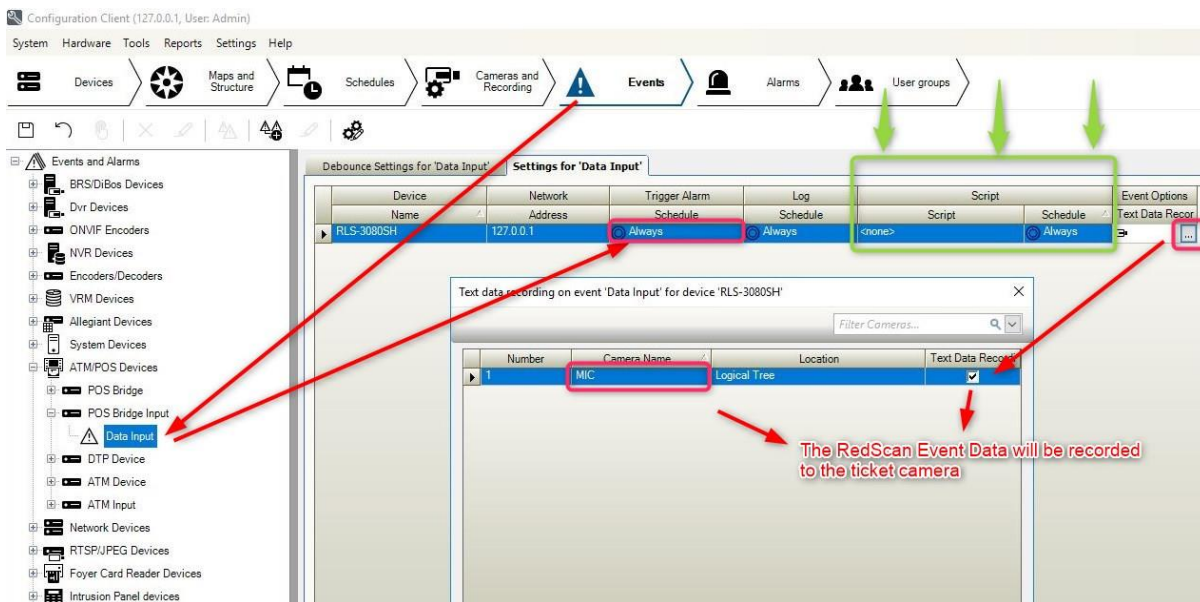


Fig. 17. The Green box shows where Server Scripts show up and can be selected and scheduled for execution once being added and compiled (see chapter 4.3).

Compiled Script code can interpret this incoming Event data (OPTEX Event codes) and by that activate internal BVMS actions as well as external calls to devices and subsystems. This makes BVMS suitable for almost any integration project and required feature set.

4.3 A Server script to process incoming Event Data

Since BVMS contains all tools itself to design dedicated Scripts to follow-up reported incident events, no external development software is needed to edit, nor to compile your script C# source code. Scripts can be developed with the on board BVMS – SDK components and their methods.

You can add Scripts via the Config Client software of BVMS by selecting the Tools menu and then “Command Script Editor”

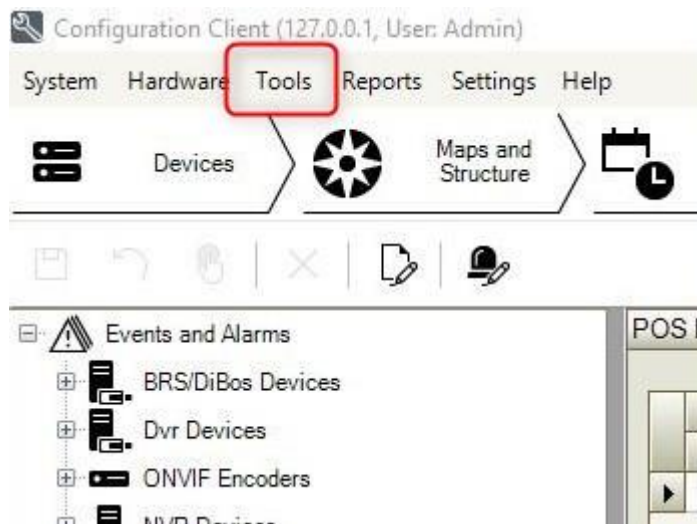


Fig. 18. Invoking the Command Script editor in BVMS Config Client Software

The Script tools window opens with 2 panels; on the left the Client and Server Script survey of existing scripts and on the right the Client and Server Editing Tabs.

For events, to invoke Script actions as a follow up to incoming BVMS events to the server, you need to add your script code using the SERVER script Tab. On the contrary, Client Scripts are invoked by Operators and need to be added as Client Script via the Client Script Tab.

4.4 Creating Scriptlets

To start editing, select in the left panel, the desired script type (ClientScript or ServerScript) and then select “New Scriptlet” via a right mouse click. This action will automatically open in the right panel the associated Script Tab and a new entry for your script code.

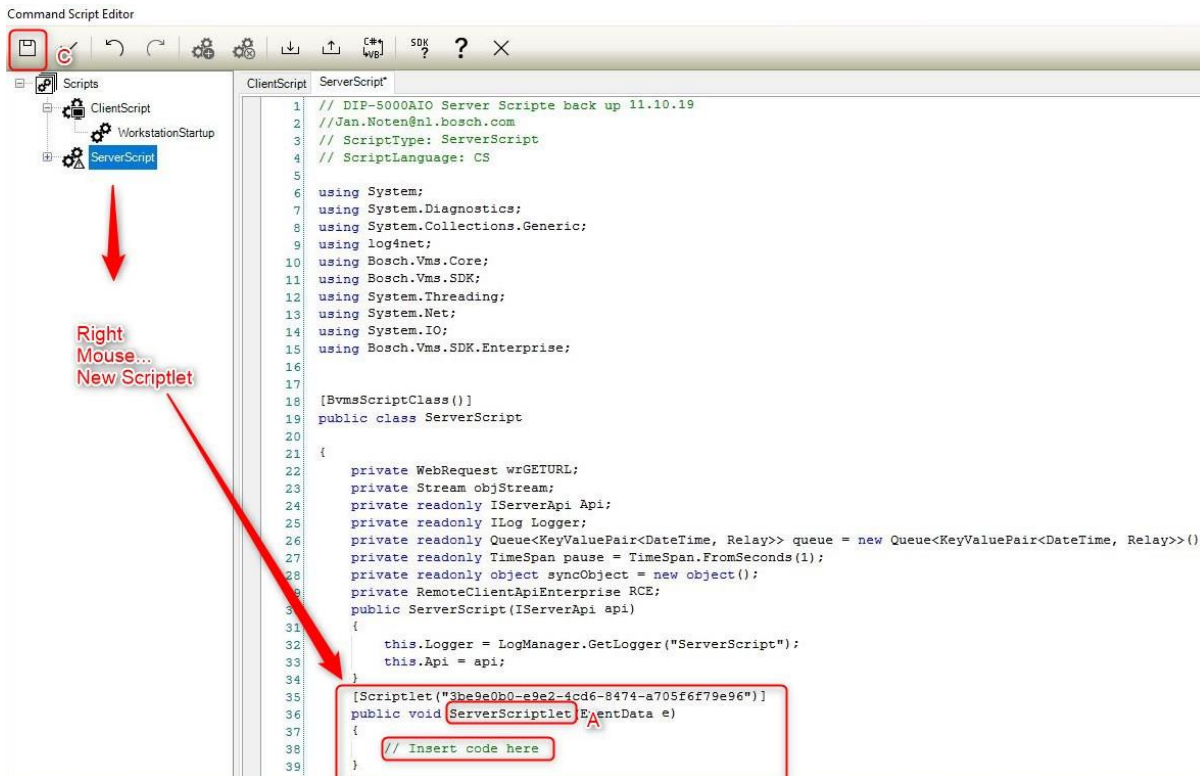


Fig. 19. Logical steps to create a Server Scriptlet.

The next step is to change the Scriptlet Name to a dedicated name corresponding the content action of the scriptlet (inset A).

Note:

- For Server Scripts, this new name will later also automatically appear in the Event Tab, at your device event row where you can select the “Scripts” for that event to follow-up tht event.
- For Client Scripts, this new name will be used for adding Scriptlets to the Operators Logical Tree.

Next step is to add your dedicated source code (in this document C#) to the scriptlet where is says “Insert code here” by overriding the green comment line with your code. (inset B)

Your script can be written in the editor area on the right using the onboard BVMS SDK (API). To do so and to get help please open the API Help button on the top indicating “SDK/?”.

It is outside the scope of this Application Note to go into SDK details. (Please contact MKI/IPP Eindhoven for further BVMS SDK details if required)

When your code has been added, you can compile by pressing the green tick in the left top corner or the Floppy Icon which will compile and save you code (provided no errors are detected) (inset C) . Please check and correct your code until there are no errors reported from the compiler at the bottom of the screen.

Leave the code screens when done.(your code must be error free and saved, before you can leave this page !!).

4.5 Bringing your Server Script into action.

A server scriptlet can be associated to any incoming BVMS event, once a new server scriptlet has been successfully added. To do this, go to the Event TAB and select your invoking event on the left. On the right, all event invocation actions can be programmed (to show alarms, to log and to activate a scriptlet) Important to know is that Server Scriptlets also follow the BVMS programmed Time schedules. (so do not forget to select a time schedule , See Fig 17 Green box).

To test your script, save your settings with the Save Changes (floppy icon - left top of the screen) or the adjacent ticky button. (If you get compilation errors in the bottom pane, correct them first).

Now fire the event and check the outcome.

Note:

Diagnosing Scriptlets and C#/VB6 Code is expected to be known and is not part of this document.
If you require further help then please contact the EMEA IPP department in Eindhoven the Netherlands.

5. Appendix A Practical examples

5.1 REDSCAN RLS-3080SH Laser alarm fires dedicated Bosch PTZ camera Presets

This integration example creates the following Bosch and OPTEx automated security feature.

1. Whenever the Laser detects an Area 1 violation, the Bosch PTZ camera will automatically go to Preset 1.
 2. Whenever the Laser detects an Area 2 violation, the Bosch PTZ camera will automatically go to Preset 2.
- Etc..

To make this work create a Server script as follows:

(Replace the “//insert code here” section in an empty Server script by the red code below)

```
public void FilterOPTExEventCode(EventData e)
{
    if(e.Type=="DataInputEvent" && e.DeviceName== "..see fig 9. used input name.. ")
    {
        //Send PTZ camera with logical number 1 to the scene of interest according to the reported
        laser Event Code
        DomeCamera ThisDome=Api.DomeCameraManager.GetDomeCameraByLogicalNumber(1);
        string LaserData= e["PosData"];

        if (LaserData.Contains("RLS126MOA1"))
        {
            //Sent PTZ Camera to Preset 1
            Api.DomeCameraManager.MoveToPredefinedPosition(ThisDome,1);
        }
        else if (LaserData.Contains("RLS126MOA2"))
        {
            //Sent PTZ Camera to Preset 2
            Api.DomeCameraManager.MoveToPredefinedPosition(ThisDome,2);
        }
    }
}
```

Fig. 20. Server Script to automatically show a crime scene fired from an OPTEx Laser Device.

How does this all work?:

1. All event data sent from the Laser device will adhere to the OPTEx Event code syntax, in this case **RLS** followed by the **lowest IP Octet of your Laser's IP number + specific event code**. Example **RLS126CL** meaning alarm cleared. (CL)The PTZ Presets must be programmed in the PTZ camera in advance and correspond to the desired Laser area.
2. The script can contain further code to execute actions not available as standard functionality in BVMS. Example: activating lights, opening barriers, interfacing to other systems etc.
3. The GUID is the uniquely given script GUID by the BVMS server when you added a “New Scriptlet”. DO NOT ALTER this given GUID !
4. The Script name “FilterOPTExEventCode” is a given name. You can take any name.
5. The 2 **If(.....)** lines, filter on only ATM/POS data AND only data from the Device as named in the ATM/POS data input field representing the Laser. In this case “RLS-3080SH” . See fig 9.
6. The Camera Logical Number must correspond with the given Logical Number in the Camera TAB of the BVMS Config Client of that PTZ camera. (Column called “Number”)

7. Make sure there is no compilation error before leaving this script. If there are errors reported then find the error line indication in the error message. Consult the SDK/? Button if needed.
8. This script is an example showing the operational basics of scripts. The integrator is however responsible to implement a smooth and save functionality taking operational and security aspects in account.
9. Also be informed that the OPTEX Event code can be a combination of multiple events and also there are codes that indicate clearance of Alarms. (See Fig 1.) You can anticipate as programmer to these possible events in your C# code and make you script rugged and sabotage/fool proof. See Appendix B and their Event Code publications for details.

5.2 REDSCAN RLS-3060SH Laser alarm populates Operator screen with 4 cameras

This integration example showcases the following Bosch and OPTEX automated security feature. Assume a Laser is continuously scanning the outside carpark on activities.

1. At a laser alarm, the Operator Client Monitor 1 must go into Quad mode automatically.
2. The 4 cameras located on the corners of the carpark should be shown automatically on this quad as Camera 1 in Cameo1, Camera 2 in Cameo2, Camera 3 in Cameo3, etc
3. When the Laser reports no activity anymore via a Clear (CL event), Monitor 1 returns to full screen mode and shows Camera 1.

To make this work, we must:

- a) Subscribe the workstation at login to receive the correct event from the BVMS Server
- b) Instantiate an EventReceiver in the Workstation to capture that Laser event and follow up.
- c) Derive and filter details of the received Eventdata, and as a result populate the workstation with the 4 cameras or full screen OR Camera 1 in full screen at rest.

To use reported Laser events from the BVMS Server in the Operator GUI on his Workstation, we need to make a Client Script that loads at Operator login and runs in the Workstation later.

This script contains all actions at runtime mentioned above, we will name it i.e. "WorkstationStartup".

Create the Client script in the Scripting code editor as follows:


```

1 // ScriptType: ClientScript
2 // ScriptLanguage: CS
3 using System;
4 using System.Diagnostics;
5 using System.Collections.Generic;
6 using log4net;
7 using Bosch.Vms.Core;
8 using Bosch.Vms.SDK;
9
10 [BvmsScriptClass()]
11 public class ClientScript
12 {
13     private readonly IClientApi Api;
14     private readonly ILog Logger;
15     private EventReceiver eventReceiverWS;
16     RemoteServerApi RSApi;
17     public ClientScript(IClientApi api)
18     {
19         this.Logger = LogManager.GetLogger("ClientScript");
20         this.Api = api;
21     }
22     public void Dispose()
23     {
24         if (eventReceiverWS != null)
25         {
26             RSApi.EventManager.Unregister(eventReceiverWS);
27             eventReceiverWS=null;
28             RSApi=null;
29         }
30     }
31
32     [Scriptlet("50a3717f-0162-40d1-8d09-3a5cea2b9c6f")]
33     public void WorkstationStartup()
34     {
35         System.Threading.Thread.Sleep(2000);
36         try
37         {
38             RSApi = new RemoteServerApi("192.168.10.249:5390","admin","");
39             eventReceiverWS = new EventReceiverWS(Api, Logger);
40             RSApi.EventManager.Register(eventReceiverWS,"DataInputEvent");//InputState
41             //Api.ApplicationManager.ShowMessage("Registered for OPTEX Events");
42         }
43         catch(Exception ex)
44         {
45             Api.ApplicationManager.ShowMessage("Error executing Workstation logon script");
46         }
47     }
48
49

```

Fig. 21 The “WorkstationStart” script, event registration at the BVMS Server

How does this all work?:

1. The code will define the connection to the Server (Please also add the 3 red arrow associated code lines)
2. A local Eventreceiver is defined and instantiated.
3. Next this local EventReceiver is subscribed to the Server for a specific event namely “DataInputEvent” as exposed by the Laser. The effect is that the Workstation will ONLY receive the subscribed events.

Next add the specific code to the “WorkstationStartup” script in order to receive the eventdata ,filter this data and do the video switching etc
 Add the following code here :


```

31
32 [Scriptlet("50a3717f-0162-40d1-8d09-3a5cea2b9c6f")]
33 public void WorkstationStartup()
34 {
35     System.Threading.Thread.Sleep(2000);
36     try
37     {
38         RSApi = new RemoteServerApi("192.168.10.249:6390", "admin", "");
39         eventReceiverWS = new EventReceiverWS(Api, Logger);
40         RSApi.EventManager.Register(eventReceiverWS, "DataInputEvent");//InputState
41         //Api.ApplicationManager.ShowMessage("Registered for OPEX Events");
42     }
43     catch(Exception ex)
44     {
45         Api.ApplicationManager.ShowMessage("Error executing Workstation logon script");
46     }
47 }
48
49
50 public class EventReceiverWS : EventReceiver
51 {
52     private IClientApi mApi;
53     private ILog mLogger;
54     public EventReceiverWS(IClientApi api, ILog logger)
55     {
56         mApi = api;
57         mLogger = logger;
58     }
59     public override void OnEvent(EventData e)
60     {
61         if (e.Type=="DataInputEvent" && e.DeviceName=="RLS-3080SH") //intercept when Laser sends data
62         {
63             string LaserData=e["PosData"];
64
65             if(LaserData.Contains("RLS126MOA2"))
66             {
67                 mApi.ContentManager.SetGranularity(1,2); //set monitor 1 on quad
68                 for(int i=1;i<5;i++)
69                 {
70                     mApi.ContentManager.DisplayCamera(new ImagePane(1,i),mApi.CameraManager.GetCameraByLogicalNumber(i));
71                 }
72             }
73             else if(LaserData.Contains("RLS126CL"))
74             {
75                 mApi.ContentManager.SetGranularity(1,1); //set monitor 1 on full and camera 1
76                 mApi.ContentManager.DisplayCamera(new ImagePane(1,1),mApi.CameraManager.GetCameraByLogicalNumber(1));
77             }
78         }
79     }
80 }
81
82
83 }

```

Fig.22 The EventReceiver populating cameras on a Quad Screen

Next please compile , fix errors if needed, and save the configuration.

The last step is to add the “WorkstationStartup” script to the Workstation definition in the Devices table. (The script name is available and selectable only when the script compilation was successful)

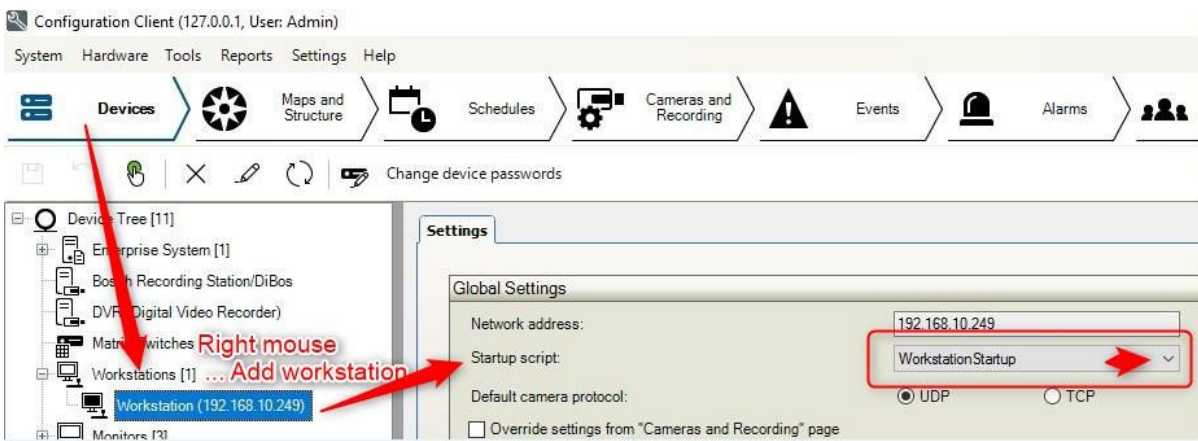


Fig. 23 Definition and initializing the Operator Workstation as a Startup Script

5.3 Debug hints for Integrators testing the examples

Here are some hints to debug your scripts when scripts are not executed.

1. First make sure the laser events are fired. You can set an alarm for the same event and check via the ALARM panel in the Operator Client if the event is shown.
2. If events are correctly shown, check the reception of the EventData in you script by temporarily showing that data on the Operator GUI.

To do so please add temporarily the following code in the red box:

```
public override void OnEvent(EventData e)
{
    if (e.Type=="DataInputEvent" && e.DeviceName== "RLS-3080SH") //intercept when Laser sends data
    {
        string LaserData=e["PosData"];
    }
}
```

mApi.ApplicationManager.ShowMessage(LaserData);

You should now see the EventData on top of the Operator Gui if events are executed in your script:

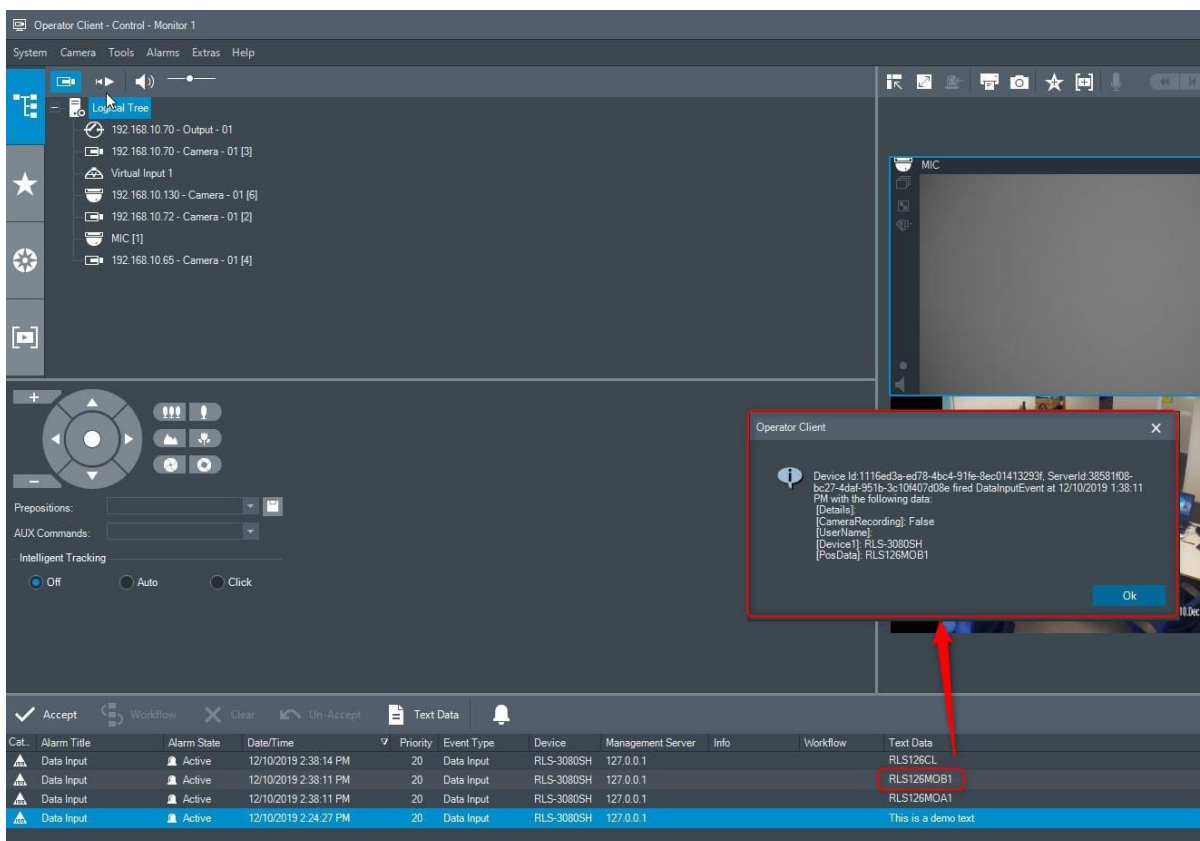


Fig.23 Laser EventData shown in the Operator GUI as debug information

If at this point events are executed but not switching video then diagnose your code within the if statements. Also make sure the cameras are selectable (authorized to see for this Operator).

6. Appendix B Reference Sites for further info

For OPTEX Product information: www.OPTEX-europe.com

For Bosch Integration Partner Program (IPP) :<https://www.boschsecurity.com/xc/en/partners/integration-tools/>

For Bosch DIVAR IP all-in-one 5000:

https://resource.boschsecurity.com/documents/DIP_aio_5000_Data_sheet_enUS_71312221451.pdf

If you require further help then please contact the EMEA IPP department in Eindhoven the Netherlands.

7. Disclaimer

Bosch cannot not accept any liability on implementing nor using scripts mentioned in this document. Your activity in developing products that interface with Bosch products is at your own risk and responsibility regarding fitness for use, completeness, faultlessness, or any claims of third parties which may arise based on such further development.



Robert Bosch GmbH
Bosch Security Systems B.V
P.O. Box 80002
5600 JB Eindhoven
The Netherlands