

MAP Open Intrusion Interface

Application Notes



BOSCH

en Technical Description

Table Of Contents

1	Introduction.....	3
2	Requirements for OII Clients.....	3
3	Setup of OII	3
3.1	Configuration of OII	3
3.1.1	Activating OII	3
3.1.2	OII Users	4
3.2	Network Connection	4
4	First Connection with OII.....	5
4.1	Connection	5
4.2	Feature discovery	7
4.3	Configuration Information	7
4.4	Lists and individual resources.....	7
4.5	Issuing Commands.....	8
4.6	Case sensitivity.....	8
5	Event notification	9
5.1	Subscriptions.....	9
5.2	Fetching Events.....	11
5.3	Renew Subscription.....	11
5.4	Unsubscribe	11
6	Incidents (Alarms and Troubles)	11
7	Security Concept	12
7.1	Transport Layer Security	12
7.2	Certificate	13
7.3	Application Layer Security/User authentication.....	13
7.3.1	Rights Management	14
8	Connection Handling	15
9	Supervision.....	15
9.1	Configuring Supervision.....	15
9.2	Usage of OII Supervision.....	16
10	History Retrieval.....	16
11	Persistent Data	17
12	Panel load indicators.....	17
13	Automatic Arming of Area	17
14	References.....	17
15	Change History	18

1 Introduction

The open intrusion interface provides an IP based interface for integration of the MAP5000 to third party systems or applications like management systems or smart phone applications. The OII enables information exchange with and control of the MAP5000 over an IP network. It will allow retrieval of information from MAP5000 internal states, such as the status of an area, as well as peripheral and device information, such as a status of a detector. Further, it allows interaction with the MAP5000 like arming an area or bypassing a device. In particular, all information relevant to replicate the visualisation of the MAP5000 states on user interfaces like keypads is provided over the OII.

The intention of this document is to guide a user on setting up and first use of the OII.

2 Requirements for OII Clients

OII is based on state of the art web technologies that are available in different programming languages and operating systems.

In particular, the client needs to support the following features:

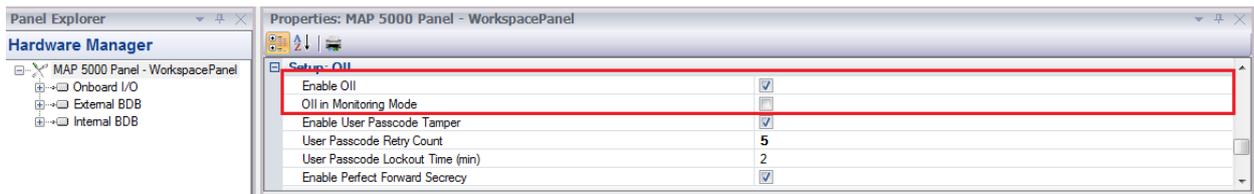
- **HTTP/1.1:** The commonly used version of the HTTP protocol
- **HTTPS:** All communication over OII is encrypted, thus HTTPS is required
- **TLS.1.2:** The OII enforces the use of the latest TLS version which currently is 1.2. Additionally, only a dedicated set of ciphers is supported. The client needs to support both TLS1.2 and the dedicated ciphers in order to communicate over the OII. Refer [Transport Layer Security](#) for certificates supported.
- **HTTP Digest Access Authentication:** Each request is authenticated using HTTP digest access authentication.
- **JSON:** The messages exchanged over the OII are formatted in JSON, as it provides a suitable trade-off between data size and a well-structured, human readable data format. A client needs to be able to create and parse JSON data structures to interact over the OII.

3 Setup of OII

3.1 Configuration of OII

3.1.1 Activating OII

Configuration of OII is done using the Remote Programming Software – RPS for MAP. OII can be activated in RPS under “Hardware Manager -> Setup: OII” section.

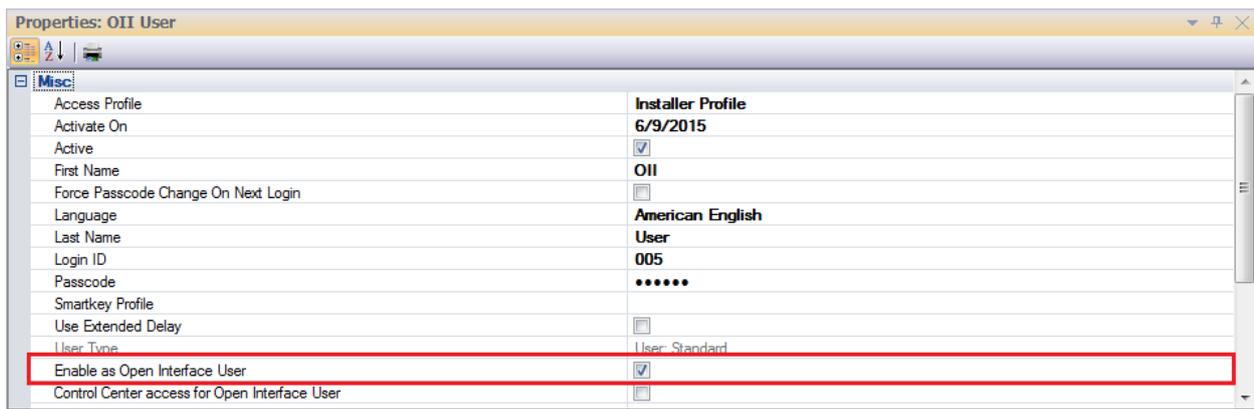


OII can be configured to operate only in Monitoring Mode. In monitoring mode, the client connected to the panel over OII can only retrieve statuses of entities in the panel. The client will be able to setup subscriptions to retrieve statuses of entities through eventing. The client will not be able to control the panel and issue commands to the panel (like e.g. arming an area).

3.1.2 OII Users

OII uses the existing user rights and access management framework of MAP 5000. A client attempting to connect to MAP over the OII will need to provide credentials of a user configured as an OII user on MAP 5000.

A user has to be configured as an OII user in the properties for the user. The user also has to have an access profile which has all permissions enabled.



Refer Section [Configuring Supervision](#) for configuration of OII Users for supervision.

3.2 Network Connection

The panel will either obtain an IP address via DHCP or use a statically configured IP address. Clients need to know the IP address of the panel to use the OII.

In case DHCP is activated and DNS is available, the serial number of the panel can be used as a domain name where the “.” is replaced by “-” (e.g. 94047.24269997507 results in 94047-24269997507 as a domain name).

4 First Connection with OII

4.1 Connection

This section describes establishing the first connection with OII using a Mozilla Firefox browser. (Note: All screen shots shown below are taken while using Mozilla Firefox browser and may look different on other browsers)

OII can be accessed over a secure connection by providing the IP address of the Panel + the url of a resource. For the first connection, the description resource can be used since this is always accessible at /desc (As explained in [Feature discovery](#) section) e.g. <https://192.168.1.33/desc>.

If DHCP is activated on the panel, OII can be accessed using the panels host name. e.g. <https://73990-36420840724.kor.apac.bosch.com/desc>

The OII is accessible on port 443. (Since it is the standard SSL port, it is not necessary to specify it explicitly)

On connection, the browser will indicate that the connection is untrusted. The panel provides a unique self-signed certificate. Since the certificate is self-signed, the browser would indicate that connection is untrusted as shown in the image below.

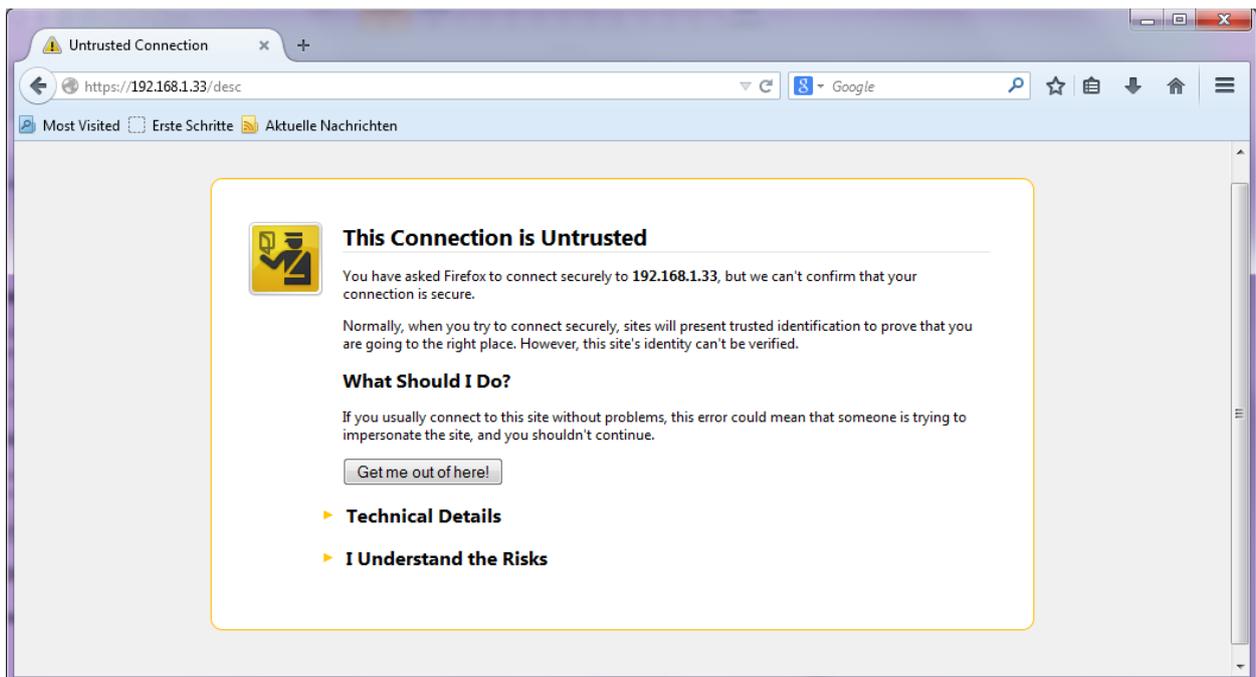


Figure 1 Untrusted Connection

To proceed with the connection, the user needs to accept the certificate by clicking the “I understand the Risks” link , adding it as an exception by clicking the “Add Exception ..” button and confirming the security exceptions as shown in the images below.

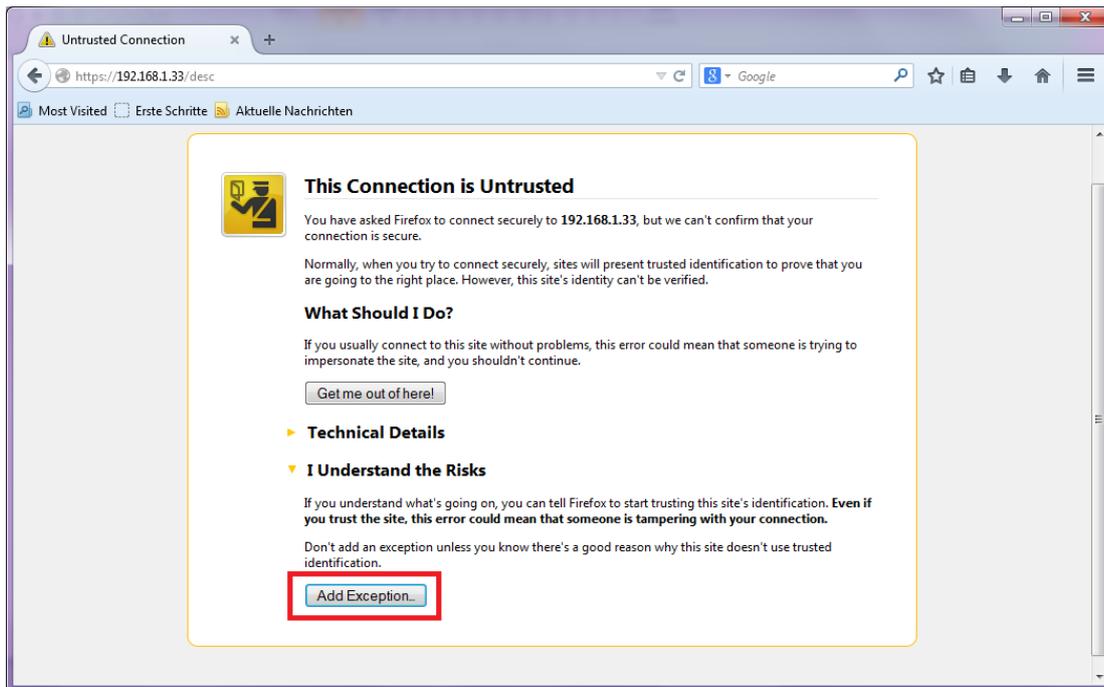


Figure 2 Add Exception

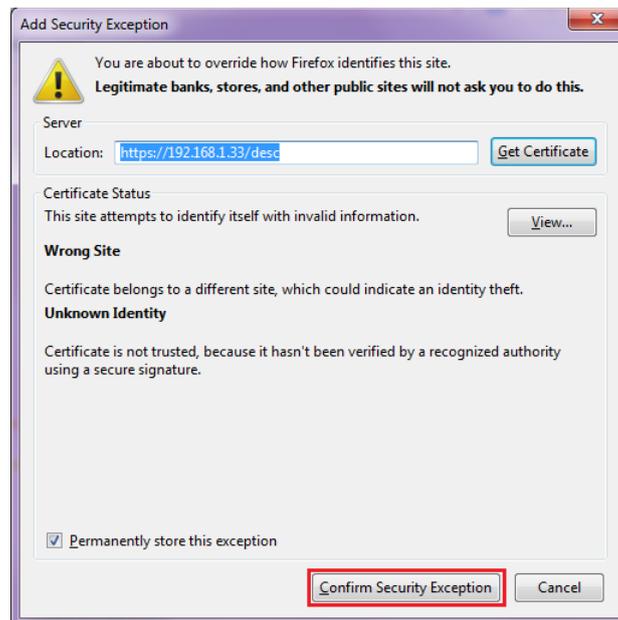


Figure 3 Confirm Security Exception

The browser will then prompt for user credentials. Credentials of a configured OII user needs to be provided. (Refer [OII Users](#) for configuration of OII Users from RPS)

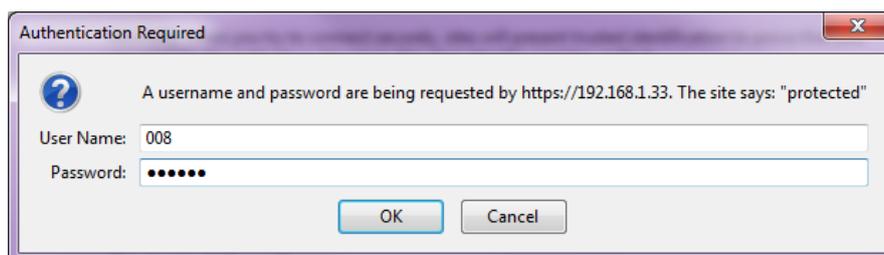


Figure 4 User Authentication

4.2 Feature discovery

In each version of OII, the location of resources will be fixed. But across future versions, the location of resources may change. In order for the client to discover the URLs of resources, the description resource is provided. The description resource will always be hosted by the MAP5000 at the URL “/desc” at its root. This resource will provide links to all major relevant resources. The client wanting to communicate to the panel over OII can always use this resource as the starting point.

4.3 Configuration Information

The configuration resource provides information about all the entities/elements that are configured in the MAP5000 system. The client can download the configuration on startup to understand the configured items on the MAP5000 and the relationship between entities like areas and devices. The config resource is available at the URL “/config”.

The configuration will provide information about the relationship of areas, areas and devices, internal programs and devices as well as between devices (e.g. in case a device is embedded in another device).

4.4 Lists and individual resources

All relevant MAP elements or entities are modelled as resources. For e.g. An area configured in the panel is modelled as a resource over the OII, a point configured in the panel is modelled as a resource over the OII. Each resource can be inspected or controlled at its individual URL e.g. “/1.1.area.2.2”, “/1.1.point.1.1”. A GET request on the resource can be used to get the status about the element. POST, DELETE can be used to modify the resource and thereby execute commands. Commands supported by the resource depends on the type of resource.

Resources of the same type are grouped together under a list resource thus allowing the client to inspect or control multiple resources using a single request to the list resource.

For e.g. “/areas” list resource can be used to retrieve status of all areas, “/points” list resource can be used to retrieve status of all points configured in the system. List resources can also be used to execute a command on all the resources under that list resource.

Specific resources under a list resource can be accessed by specifying a filter in the request e.g. /areas?url=/1.1.area.2.*, /areas/?url=/1.1.area.2.(81-85).

(Refer *MAP Open Intrusion Interface Base Specification* document for more details on usage of filters)

4.5 Issuing Commands

Status of resources can be inspected or retrieved by sending a GET request to the resource. A successful GET request will result in a response with a 200 “OK” response code and the response body containing the status of the resource in JSON format.

In addition to inspecting statuses of resources over the OII, the status of resources can also be modified over the OII. This is achieved by issuing commands to the system over the OII. Commands are issued in general by sending POST request to the resource with the request body containing details about the command.

Commands supported by resources depends on the type of resource. For e.g. an area resource would support commands such as ARM, DISARM, ARMINGINFO whereas a point resource would support commands such as ENABLE, DISABLE etc. (Refer *MAP Open Intrusion Interface Resource Model* document for details on resources and the commands they support)

Commands that are sent to the system over OII are forwarded for processing and a direct response provided to the client with a 202 “Accepted” return code. This implies that the command has been accepted for processing by the system, but is not guaranteed that the command would be executed as requested. Therefore a client would need to check the status of the resource to ensure that the command was in fact executed.

If application logic prevents execution of a command, an error response with a return code 409 “Conflict” with returned. The body of the response will contain text explaining why the command failed to execute.

In addition to the above mentioned return codes, standard return codes such as 400 “Bad Request”, 403 “Forbidden”, 404 “File Not Found”, 503 “Service Unavailable” could be returned. Refer *MAP Open Intrusion Interface Base Specification* and *MAP Open Intrusion Interface Resource Model* documents for further details on error codes.

4.6 Case sensitivity

All urls used to access resources over the OII are case insensitive. Hence a resource with id 1.1.Area.2.2 can be accessed at <https://<PanelIPorHostName>/1.1.area.2.2> or <https://<PanelIPorHostName>/1.1.aReA.2.2>.

The JSON content in the body of a request is case sensitive. Hence it should be ensured that the definition of the request body described in the MAP Open Intrusion Interface Resource Model document is strictly adhered to.

5 Event notification

The OII includes an event notification mechanism that allows spontaneous transmission of state changes (e.g. closing of a door contact) and occurrence of incidents (e.g. intrusion alarms, troubles). The notification mechanism uses a publish-subscribe pattern, where a client has to register with the MAP to specify which events it is interested in. This allows to custom tailor the events collected for each individual client. For example, a client can subscribe only for incidents in the system. In this case, it will only be notified about alarms and troubles but will not be notified about state changes of peripherals, significantly reducing the amount of events to be communicated. On a successful subscription creation, MAP will internally store events for the client.

The delivery of the events is done using a so called long polling mechanism. The request to fetch events will not return an answer until either a sufficient amount of events is available or a timer expires. A client can specify for each request how many events should be included in the response as a minimum (`minEvents`), as a maximum (`maxEvents`) and how long the call is allowed to block at most (`maxTime`).

5.1 Subscriptions

A client can create a subscription by issuing a POST request to the subscription list resource `/sub`. The body of the POST request contains all the relevant information for the subscription – buffer size, lease time, event type and resource urls.

```
{
  "bufferSize": 100,
  "leaseTime": 300,
  "subscriptions": [
    { "eventType": ["CHANGED", "CREATED", "DELETED"],
      "urls": ["/1.1.Area.2.5", "/1.1.Area.2.2"]
    }
  ]
}
```

Resource urls can also be specified as filters rather than having to specify individual resource urls.

```

{
  "bufferSize ": 100,
  "leaseTime": 600,
  "subscriptions": [
    { "eventType": ["CHANGED", "CREATED", "DELETED"],
      "urls": ["/inc/*"]
    }
  ]
}
    
```

If MAP accepts the request, a dedicated subscription is created. The response body will contain the unique subscription url, the lease time and the buffer size allocated for the subscription. The maximum buffer size for a subscription can be 640. Lease time is the maximum time up to which OII will keep a subscription on which there is no activity (fetching of events). The client is expected to renew the subscription within this period if it wants to keep the subscription alive. Lease time for a subscription can vary between 10 secs and 600 secs. The OII supports a maximum of 15 subscriptions at any point of time.

A client can also create multiple subscriptions so that the queue sizes and fetching behaviour fits to the expected occurrence of events. For example, it can be assumed that incidents will only occur rarely while device state changes will happen often. Thus, a client may create two subscriptions; one subscription for incidents only and one subscription for devices only. Thus, incidents can be prioritized and fetched quicker as if they would be part of the same subscription.

A client that wants to subscribe to all resources of the panel may specify a filter with all elements in the following list. Please note that the URL of the resources may change over time. Thus, a client shall look up the URL for the given resource type from the description resource to be forward compatible.

Description	Current URL
All devices	/devices
All areas	/areas
All incidents	/inc/*
OII interface	/1.1.system.15.1
Supervised connections	/supervisedConns
Internal Programs	/internalPrograms
All users	/user/*
Panel resource	/panel
All walktests	/walktest/*

5.2 Fetching Events

A client can fetch events from the buffer of the subscription by using a POST request with the defined, optional parameters in the body of the request. The following parameters are defined to allow a fine grained control of the event delivery to the client:

- **maxEvents:** The maximum number of events contained in the answer to the request. This is limited by the max buffer size allocated for the subscription.
- **minEvents:** The minimum number of events contained in the answer to the request. If not specified, will be considered as the same value as maxEvents.
- **maxTime:** The maximum time in seconds the request will block. The maximum value possible for max time is 100 secs.

The panel will respond to a POST request as soon as the first of the above specified conditions is fulfilled. Once events are fetched by the client, they will be deleted from the internal event buffer. Thus, events can only be fetched once.

(Refer *MAP Open Intrusion Interface Base Specification* document for further details on event retrieval)

5.3 Renew Subscription

OII will delete a subscription if there are no requests on the subscription for the lease time specified for the subscription. If the client doesn't need to fetch events from a subscription, but wants to keep the subscription alive, it can issue a POST request to fetch events with maxEvents specified as 0. The OII will consider this as a renewal request and renew the subscription.

5.4 Unsubscribe

Client can unsubscribe or delete a subscription by issuing a DELETE command on the subscription resource. MAP will free the event buffer associated with the subscription and also delete the subscription. Any events not fetched from the event buffer associated with the subscription will be lost.

6 Incidents (Alarms and Troubles)

Alarms and troubles of the MAP5000 are modelled as incidents. An incident resource will not always exist, but will be created when an alarm/trouble appears in the system. The incident resource will also be removed from the OII when the alarm/trouble is resolved and disappears from the system.

The incident resource itself contains all relevant information about the alarm. In particular, it conveys who has acknowledged or handled the alarm. Creation, deletion and state changes of an Incident are communicated via events.

Incidents are located under the root resource “/inc”. This resource acts as an incident list, where all pending incidents in the system can be accessed. The individual incidents are located under /inc with the following structure /inc/[Area ID]/[incident ID] where

Area ID: Complete Area SIID

Incident ID: Unique ID for an incident.

The status of the incidents can be inspected and the incidents can be handled over the OII. This can be done either at individual incident resources or at the incident list resource.

(Refer to *MAP Open Intrusion Interface Resource Model* document for further details on categorization of incidents and handling of incidents.)

7 Security Concept

The OII provides two levels of communication security. On transport level, TLS is used. In addition, each request to the OII is authorized using username and password information.

7.1 Transport Layer Security

The OII supports TLS v1.2 with support for the following ciphers with perfect forward secrecy

```
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256  
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256  
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384  
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384  
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256  
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256  
TLS_DHE_RSA_WITH_AES_256_CBC_SHA256  
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
```

These above mentioned ciphers are supported by default when OII is activated on the panel.

In addition to the above ciphers, the following ciphers are supported if support for non-perfect forward secrecy is enabled in the configuration through RPS.

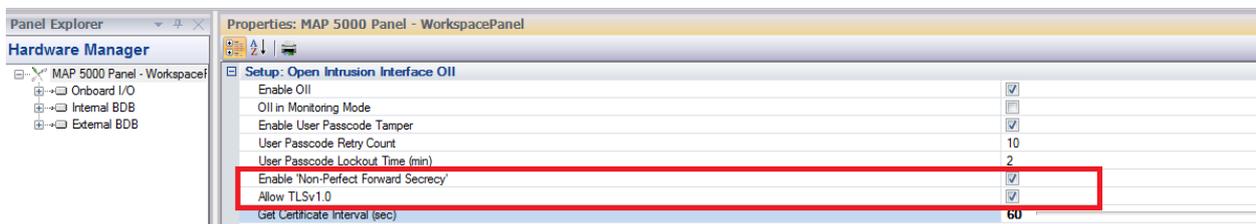
```
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256  
TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256  
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384  
TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384
```

Non-Perfect Forward Secrecy can be enabled through RPS under “Hardware Manager -> Setup: OII” section.

For clients that cannot support TLSv1.2, the OII shall allow downgrade to TLSv1.0. With TLSv1.0 enabled, the following cipher suites will be supported in addition to the above mentioned ciphers:

TLS_RSA_WITH_RC4_128_MD5
TLS_RSA_WITH_RC4_128_SHA
TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA
TLS_DHE_RSA_WITH_AES_128_CBC_SHA

The downgrade to TLSv1.0 can be enabled through RPS under “Hardware Manager -> Setup: OII” section. Downgrade to TLSv1.0 is possible only if Non-Perfect Forward Secrecy has been enabled.



7.2 Certificate

Each MAP5000 Panel has a unique, self-signed RSA certificate. The public part of the certificate can be downloaded as part of the TLS handshake. A client that wants to do sever authentication has to gain access to the certificate in a secure manner (e.g. directly connecting to the MAP5000 via the built in Ethernet plug or via a secure network) to ensure that the certificate belongs to the desired MAP.

The certificate is created in the panel itself during the first power up of the panel (Due to this, the first power up of the panel will take longer). Upload of certificates to the MAP is currently not possible.

The MAP5000 will not validate client certificates. Thus any client is allowed to connect to the MAP via TLS, but any activity on the OII will involve application layer security measures (username, password).

7.3 Application Layer Security/User authentication

OII requires HTTP authentication. HTTP Digest is used to provide username and password information. OII uses the existing user management framework of MAP and so Username and password are matched with the user database as configured via RPS (defined in [OII Users](#)). An OII request will only be executed if the password is valid and the user has the appropriate access rights to conduct the operation.

7.3.1 Rights Management

As defined in [OII Users](#), a user needs to be configured as an OII user in order to access the panel over the OII. Furthermore, an OII user can be configured to be allowed to login on the system keypad. If configured to do so, a user can use the credentials of the OII user to physically login on system keypad.

7.3.1.1 Access Permissions

In the current version of OII, only users with all permissions (over all the areas at all times) can be configured as an OII User. A user not matching these access permissions will not be allowed to be configured as an OII User by RPS.

7.3.1.1.1 *Installer User*

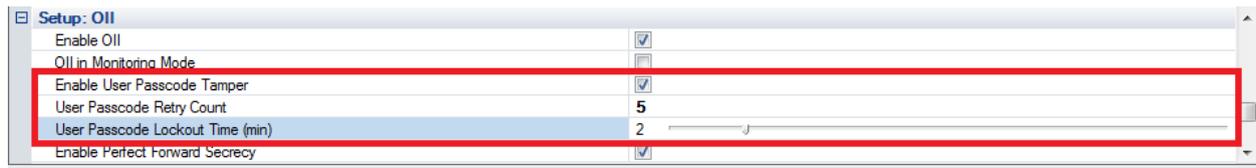
Installer users in general can access the MAP5000 only when the panel is in installer mode. A client monitoring the panel over OII would require uninterrupted access to the panel irrespective of the mode of the panel. Hence, an OII user irrespective of the type (End User or Installer User) can access the panel at all times (irrespective of the mode the Panel is in)

7.3.1.1.2 *Standard, Temporary and One Time users*

One time and temporary users have time constrained access to the MAP5000. One time users allowed only one login on the system keypad, temporary users are active only between a certain date range. Hence RPS would prevent configuring temporary and one time users as OII users. Only Standard (End Users and Installer type) users are allowed to be configured as OII users.

7.3.1.2 User Code Tamper

The OII provides the capability of detecting a possible brute force attack over the OII. This feature can be enabled by enabling “Enable User Passcode Tamper” in the OII section in RPS.



The number of incorrect login attempts can be configured (User Passcode Retry Count) after which the user will get locked out over the OII for a configured period of time (User Passcode Lockout Time). In addition to this, a User Code Tamper incident is detected by the MAP5000. If the incorrect logins are attempted with invalid user ids, only a user code tamper incident will be detected (since there is no user to lock out).

The resource `/1.1.system.15.1` can be used to evaluate the state of the User Code Tamper over the OII. (Since the resource URLs could vary across versions, it is recommended that this information be retrieved using the `/desc` resource)

8 Connection Handling

A client may use one or multiple TCP connections to communicate with the MAP. The MAP will follow the HTTP header information from the client, whether a TCP connection shall be closed or kept open after a HTTP response is sent. MAP will keep an idle connection alive for 5 seconds before which it is garbage collected.

Thus, a client may reuse the TCP connection for multiple HTTP requests and thereby save time and bandwidth as TCP and TLS handshake are only required once.

Currently, the OII allows a maximum of 20 connections in parallel.

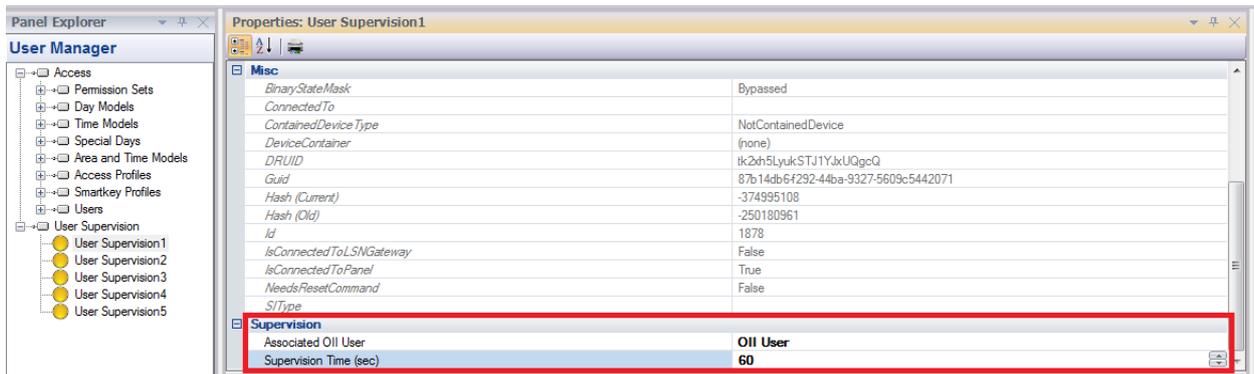
An http session on the OII will be kept open by the OII until it has responded to the request (unless session is closed earlier by the client).

9 Supervision

9.1 Configuring Supervision

MAP supports supervision on a user basis. A connection to a user will be considered “available” if the MAP receives any OII message (get status, post command, event fetch etc.) in a defined, configurable timespan (configurable using RPS). If no OII message is received in this interval, the connection is considered to be broken and a trouble will be detected. Please note that the existence of a TCP connection is not sufficient for supervision purposes an exchange of application level messages is required.

Users that need to be supervised are configured through RPS from the “User Manager -> User Supervision” section.



“Supervision Time” defines the timespan within which a OII message is expected from the client for the client connection to be considered as available. Supervision Time can range between 1 and 60 seconds.

At any point of time, only 5 OII users can be configured to be supervised.

9.2 Usage of OII Supervision

Since OII supervision is done based on user, a client will have to use a specific OII supervision user configured in RPS. The connection to the client will be considered broken if no message is received with the OII Supervision user credentials for the configured period of time.

If supervision of redundant paths is required, each redundant client will need to use a different user for supervision, so that failure of a connection to each client can be detected.

If it is required that a trouble is generated only if both the clients are unavailable, then both the clients can use the same OII Supervision user.

10 History Retrieval

MAP logs all events that have occurred in the Panel to a history database (Logging of events in history database is defined by the configuration). This history can be retrieved over the OII by using the history resources. OII provides 2 history related resources /history and /ipchistory. /history resource will retrieve events from the main history on the panel. Main History consists of all events in the panel configured to be logged.

/ipchistory resource will retrieve events from the history which are related to IP communicator. IPCHistory is applicable only if the panel has been configured to report over IP. Events in IPCHistory will not be logged in panel main history. A client will have to fetch events from main history and IPC history separately.

(Refer *MAP Open Intrusion Interface Resource Model* document for details on the resource structure for history resources and the event structure.)

11 Persistent Data

Certain statuses of elements/entities on the MAP are persisted across power cycles of the panel e.g. armed status of an area, disabled status of a device. Persistent data can be cleared in the panel by issuing a reset command on the /Panel resource with the persistData flag set as false.

Refer MAP Open Intrusion Interface Resource Model document for more details on the /Panel resource.

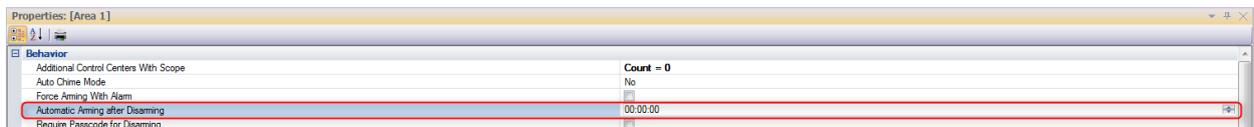
12 Panel load indicators

The attributes isPanelLoaded and restartImminent available as a part of the /Panel resource provides an indication of the load (processing load) currently in the panel. The isPanelLoaded flag indicates that the panel is temporarily busy.

The restartImminent flag indicates that the panel is overloaded and will restart due to the overload. A client can subscribe to the panel resource resulting in the client receiving an event notification when the restartImminent flag changes to true.

13 Automatic Arming of Area

On the MAP5000, an area can be configured to be automatically armed after being disarmed. This is configured by specifying a time out after which the area should be automatically armed when disarmed.



If configured for automatic arming, the area will arm automatically even if disarmed by a client over the OII after the specified time out.

14 References

MAP Open Intrusion Interface Base Specification	Document describing the communication technology and concept for the access to and control of the MAP5000 panel over OII
MAP Open Intrusion Interface Resource Model	Document describing the resources and operations provided by the MAP5000 via the OII

JSON Specification	Specification http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf , Official Website http://www.json.org/
--------------------	---

15 Change History

Date	Description	Author
05.10.2015	Initial Version	-
02.12.2015	Updated section 7.3.2.1 with details of 1.1.system.15.1 resource Added section 12 Panel Load Indicators	Arun Ram Moorthi
11.03.2016	Added Section 13 Automatic Arming of Area	Arun Ram Moorthi
10.04.2017	Corrected syntax in examples of Chapter 5.1	Steffen Neumann

Bosch Sicherheitssysteme GmbH
Robert-Bosch-Ring 5
85630 Grasbrunn
Germany
www.boschsecurity.com
© Bosch Sicherheitssysteme GmbH, 2017