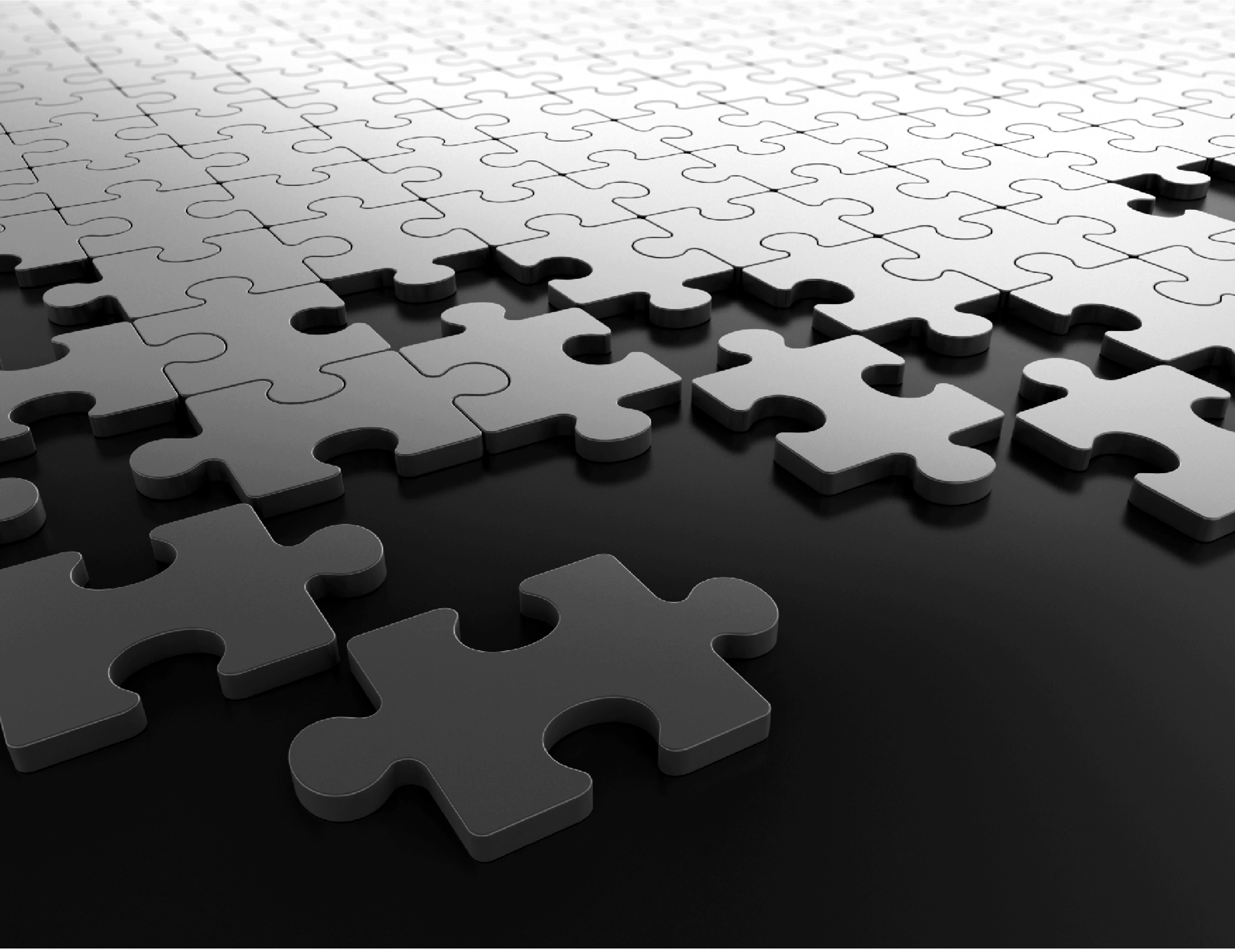
COSEC PANEL200 API Guide



MATRIX
SECURITY SOLUTIONS

COSEC Panel200 API

User Guide

Documentation Disclaimer

Matrix Comsec reserves the right to make changes in the design or components of the product as engineering and manufacturing may warrant. Specifications are subject to change without notice.

This is a general documentation for all variants of the product. The product may not support all the features and facilities described in the documentation.

Information in this documentation may change from time to time. Matrix Comsec reserves the right to revise information in this publication for any reason without prior notice. Matrix Comsec makes no warranties with respect to this documentation and disclaims any implied warranties. While every precaution has been taken in the preparation of this system manual, Matrix Comsec assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained herein.

Neither Matrix Comsec nor its affiliates shall be liable to the buyer of this product or third parties for damages, losses, costs or expenses incurred by the buyer or third parties as a result of: accident, misuse or abuse of this product or unauthorized modifications, repairs or alterations to this product or failure to strictly comply with Matrix Comsec operating and maintenance instructions.

Copyright

Version 10.0
Release date: January 5, 2023

Contents

List of Tables

About the Document

Welcome to the *COSEC Panel200 API User Guide*. Using this document you can learn more about COSEC APIs, browse through detailed descriptions of individual APIs and test them using sample scenarios.

Document Conventions

This API User Guide will follow a set of document conventions to make it consistent and easier for you to read. These are as follows:

1. Text within angle brackets (e.g. “<request-type>”) denotes content in URL syntax and should be replaced with either a value or a string. The angle brackets should be ommitted in all instances except those used to denote “tags” within XML responses (e.g. “<name></name>”).

2. The term *device* used in this document, will refer only to **Panel200** in standalone mode or otherwise mentioned explicitly.

3. Any expression resembling ***<x~y>***, indicates that the field should be repeated for index values ***x*** to index values ***y***. This is to avoid duplicating the same parameter for multiple index numbers.

4. Additional information about any section appears in the form of notices. The following symbols have been used for notices to draw your attention to important items.

   Important: *to indicate something that requires your special attention or to remind you of something you might need to do when you are using the system.*

   Caution: *to indicate an action or condition that is likely to result in malfunction or damage to the system or your property.*

   Warning: *to indicate a hazard or an action that will cause damage to the system and or cause bodily harm to the user.*

   Tip: *to indicate a helpful hint giving you an alternative way to operate the system or carry out a procedure, or use a feature more efficiently.*

Document Organization

This document has been organized into the following topics:

1. About the Document
2. API Overview
3. Supported APIs
4. Details of APIs
5. Error Responses
6. API Response Codes
7. Appendix

Topics 1 and 2 will provide a general understanding of COSEC Panel200 APIs and the basic interface communication. Topic 3 provides a list of all supported APIs. Topics 4 provides detailed explanation of individual APIs. The following information has been provided on each request type:

- Description of the functionality.
- Action requested.
- Generic query syntax.
- Mandatory and optional parameters (argument-value table).
- Examples (*Sample Request* and *Sample Response*).

Topic 5 provides illustrations of error messages. Topic 6 provides a list of API Response Codes and their meaning. The *Appendix* will provide additional material for the user's reference.

For a list of all tables provided in the document, refer to List of Tables. Click on the links to view the respective tables for the required data.

Who Can Use This Document

The COSEC Devices API User Guide is meant for *third-party software developers* who wish to operate COSEC Devices via another remote application. This guide will provide information to users on how to request and receive services from COSEC Devices using a COSEC API.

API Overview

COSEC Devices APIs provide an interface for communication with COSEC Devices via HTTP methods. These APIs will enable specific functions to be performed on your remote devices such as setting basic and advanced device configurations, configuring users on device, performing enrollment of credentials, monitoring events and sending commands to device. For a complete list of COSEC Device APIs, refer to *Supported APIs*.

How It Works

Following is an illustration of how the COSEC system typically communicates in a client-server based architecture.

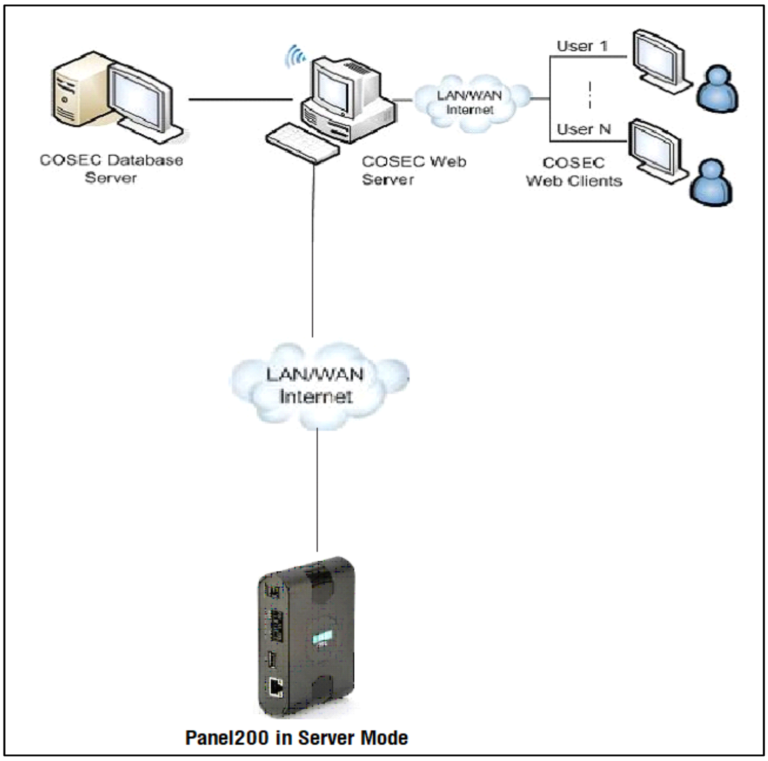


Fig. Communication through COSEC Web Server

However, here the communication with COSEC devices occurs via the COSEC Web server. On the other hand, Devices APIs enable a client application to access and monitor a remotely installed COSEC device directly, without installing the COSEC server/Monitor.

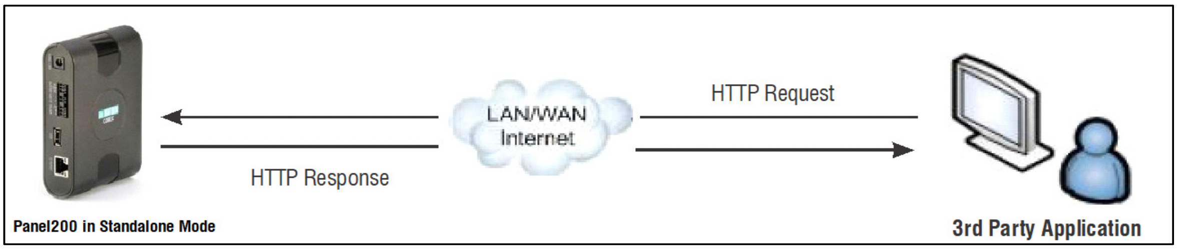


Fig. Communication through COSEC API

Using APIs, the third party can send a simple HTTP request to configure, control or command a device. The device then processes and executes this request to return an appropriate response.

General Features

All COSEC APIs -

- Are Web-based *HTTP* APIs.
- Use basic *HTTP Request-Response* for interface communication.
- Generate response in either *text* or *XML* (Extensible Markup Language) format.
- Use simple *HTTP commands* such as *GET*, *SET*, *DELETE* etc.
- Use a generic syntax for all queries.
- Support some predefined parameters and their corresponding values for each action. Each parameter will either be mandatory or bear a system-defined default value (when no value is specified).
- Use a mandatory parameter ***action*** universally, which takes action values (such as ***get, set, delete*** etc.) and specifies the action to be requested.

What the User Should Know

It is assumed that developers using this document have prior knowledge of:

- Basic functioning of the COSEC system
- Basic HTTP request-response communication
- XML

Prerequisite

In order to use a Panel200 API, the user will require:

- A Panel200 (pre-installed)
- A network enabled for accessing the COSEC Device.
- The credentials for API Authentication

For information on installing a Panel200 and assigning an IP address to it, please refer to the respective device documentation.

Authentication

The device shall request basic authentication for granting access. Default username and password for HTTP session authentication are:

Username: admin
Password: password set on device

HTTP Request-Response

Basic HTTP communication is based on a request-response paradigm. The message structure for both request and response has a generic format.

```
HTTP-message = Request | Response; HTTP/1.1 messages
```

`Generic-message = start-line`	*The start line*
`*(message-header CRLF)`	*Zero or more header fields or ‘headers’*
`CRLF`	*An empty line*
`[Message-body]`	*A message-body (chunk or payload)*

```
Start-line = Request-Line | Status-Line
```

Communication Flow

The communication takes place in the following manner:

1. The client checks availability of the device.

2. If available, the client issues a request for the device.

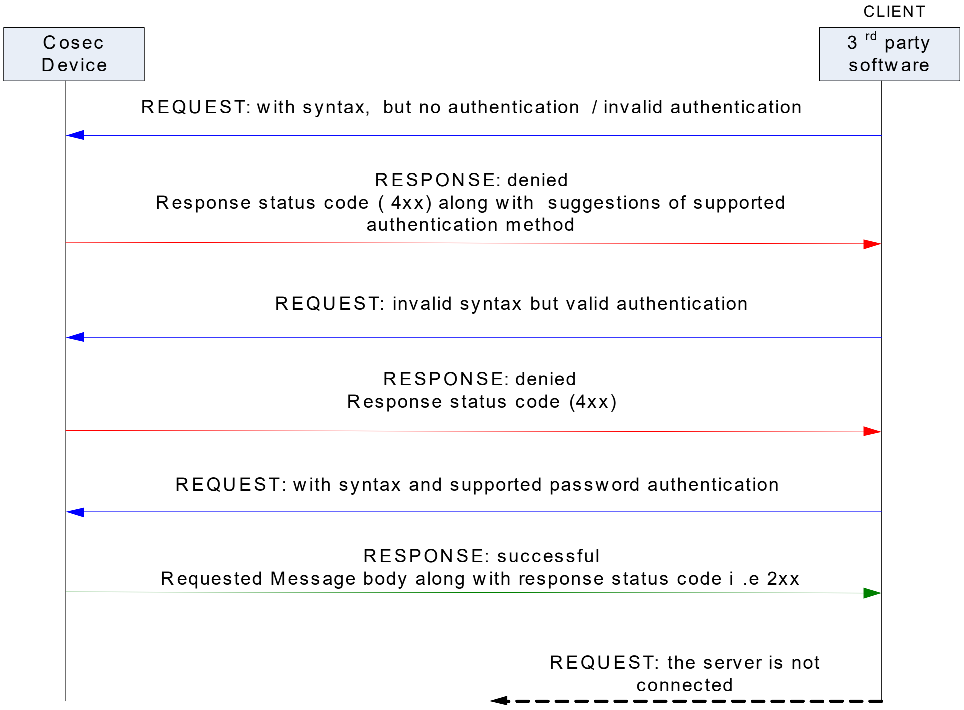

CLIENT
Cosec Device
3 rd party software
REQUEST: with syntax, but no authentication / invalid authentication
RESPONSE: denied
Response status code (4xx) along with suggestions of supported authentication method
REQUEST: invalid syntax but valid authentication
RESPONSE: denied
Response status code (4xx)
REQUEST: with syntax and supported password authentication
RESPONSE: successful
Requested Message body along with response status code i .e 2xx
REQUEST: the server is not connected

Fig: communication flow

3. The device parses the request for the action to be taken.

4. In case of an error (*invalid syntax*, *invalid authentication* etc.), the request is denied and an error response is returned. Else, the requested data is returned with the appropriate response code.

Request Format

All HTTP Requests follow a generic message format. It consists of the following components:

1.	Request Line	This line is constituted by the following three elements which must be separated by a space: • The method type (GET, HEAD, POST, PUT etc.) • The requested URL • The HTTP version to use For e.g.: `GET http://192.168.1.2/device.cgi/command?action=geteventcount HTTP/1.0`

2.	Header Fields	Add information about the request using these header fields: • A General Header (<Header-name>:<value>). • A Request Header (<Header-name>:<value>). • An Entity Header (<Header-name>:<value>).
3	Empty Line	This is an empty line separating headers from the message body.
4	Message Body	This is the chunk or payload.

Example:

```
GET http://matrix.com/ HTTP/1.0
Accept: text/html
If-Modified-Since: Saturday, 15-January-2000 14:37:11 GMT
User-Agent: Mozilla/4.0 (compatible; MSIE 5.0; Windows 95)
```

Response Format

An HTTP response is a collection of lines sent by the server to the client. A generic HTTP response format will resemble the following:

```
VERSION-HTTP CODE EXPLANATION<crlf>
HEADER: Value<crlf>
.
.
.
HEADER: Value<crlf>
Empty line<crlf>
BODY OF THE RESPONSE
```

It consists of the following components:

1.	A status line	This line is constituted by the following three elements which must be separated by a space: • The version of the protocol used (e.g. *HTTP/1.0*). • The status code (indicates the status of the request being processed). • The explanation of the code.
2.	The response header fields	These optional lines allow additional information to be added to the response header. This information appears in the form of a name indicating the header type followed by a value for the header type. The name and value are separated by a colon (:).
3.	The body of the response	Contains the requested data.

Example

When the server gets a request, it will respond with a standard HTTP status code as illustrated in the following sample response:

```
HTTP/1.0 200 OK
Date: Sat, 15 Jan 2000 14:37:12 GMT
Server: Microsoft-IIS/2.0
Content-Type: text/HTML
Content-Length: 1245
Last-Modified: Fri, 14 Jan 2000 08:25:13
GMT
```

HTTP Status Codes: *Status codes are 3-digit numeric codes returned in HTTP responses that enable recipients to understand the successful or failed status of the request issued. In general, codes in the 1xx range indicate an informational message only, 2xx codes indicate a successful request, 3xx codes indicate an incomplete request that requires further action, 4xx codes point at client-side errors while 5xx codes point at server-side errors.*

URL Syntax

All Panel200 APIs follow a common HTTP query syntax for the third party to generate a request. The generic URL is stated below.

Syntax

```
http://<deviceIP:deviceport>/device.cgi/<request-type>?<argument>=<value>&<argument>=<value>......
```

Take a close look at the URL and its basic elements:

URL element	Description
http://	This is the protocol used to communicate with the client. **Note:** All HTTP commands are in plain text, and almost all HTTP requests are sent using TCP port 80, though any port can be used.
<deviceIP:deviceport>	This identifies the device with which communication is to be performed. It consists of two components: deviceIP: Device IP address deviceport: Device Port Number
device.cgi	This is a mandatory entity required to specify the CGI directory for all the device-related commands.
<request-type>	This specifies the type of API request. For the mandatory request types, please refer to the individual API descriptions.

URL element	Description
<argument>	This defines a specific action or command depending on the function to be performed. **A mandatory argument for all COSEC API functions is *action*. This argument always takes an action as its value (For eg. *action=get*).** For more information on the common HTTP actions used in COSEC APIs, please refer to section *Common Actions*.
<value>	These are argument values that determine the output.

Example

Let us assume that the target device has the IP address 192.168.x.y and the device port number is *80*. The user wants to fetch basic configured parameters for the device. In this case, a sample request would resemble the following:

```
http://192.168.x.y:80/device.cgi/device-basic-config?action=get&format=xml
```

In this case, the query uses an ***action=get*** parameter which is commonly used to retrieve information from the device-side. The URL takes another argument called ***format*** which specifies that the response returned should be in the XML format.

- Special characters (&, ‘, “, **<**, **>**, #, % and **;**) will not be allowed in arguments or their values. Special character “**&**” will be allowed as a separator between consecutive arguments and “**?**” will be allowed as a separator between the request-type and an argument.
- The request line and headers must all end with <CR><LF> that is carriage return character followed by a line feed character.
- The status line and header must all end with <CR><LF>.
- The empty line must consist of only <CR><LF> and no other white space.

Common Actions

The following actions are commonly used in COSEC APIs as values for the ‘***action***’ argument:

Action	Use
GET	To fetch required data from device.
SET	To set required parameters for a given function.
GETDEFAULT	This is used to get default the parameters of all/ specified argument. If any argument is specified then default value of that particular argument is returned else default value of complete group is returned.
SETDEFAULT	This is used to default the parameters. If any argument is specified then default that particular value else default complete group

Action	Use
DELETE	To delete data from device.
ENROLL	To enroll an entity to a device.

Additional Information

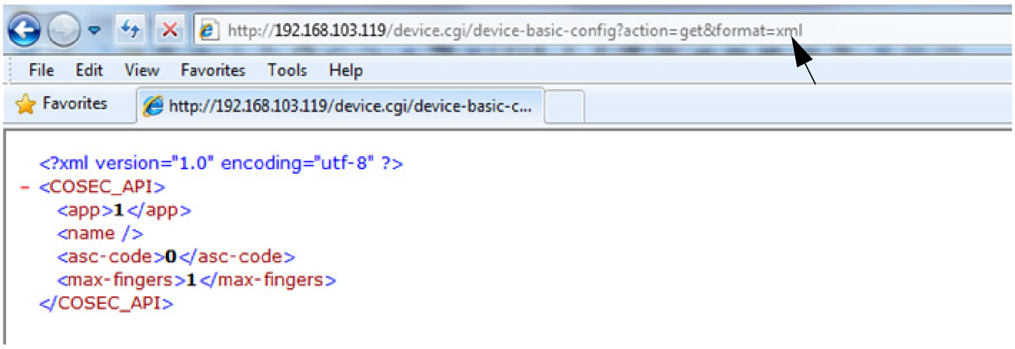
- Generally, all the commands will be supported in the GET Method and hence the arguments and valid values will be expected in the URL. Wherever applicable POST method will be specified explicitly. For the POST method, the parameters must be included in the body of the HTTP request.

- To set blank values in a particular field, a blank can follow the “=”. E.g. “argument=&”

- If the format is not specified then by default, the values should be returned in text format.

- For all arguments other than ‘action’, the position in the URL may be changed.

COSEC APIs use basic authentication and can be tested on any standard Web browser. Enter the request URL in the address field of your browser and press the ‘Enter’ key to send query to the device. Enter the authentication credentials when prompted. The response will be displayed on your browser in the specified format.



http://192.168.103.119/device.cgi/device-basic-config?action=get&format=xml

http://192.168.103.119/device.cgi/device-basic-c...

```
  <?xml version="1.0" encoding="utf-8" ?>
- <COSEC_API>
    <app>1</app>
    <name />
    <asc-code>0</asc-code>
    <max-fingers>1</max-fingers>
  </COSEC_API>
```

Supported APIs

COSEC Devices support the following groups of APIs categorized on the basis of functions to be performed:

- Panel Configuration
- Panel Door Configuration
- User Configuration
- Enrollment
- Events
- Access Zone
- Sending Commands to Panel
- IMEI Registration Request
- Request for Security Code
- Access Request on QR Scanning
- To get Random Key on Code

Panel Door Configuration

This group of APIs enables users to perform the following types of Panel Door Configuration:

- Panel Door List
- Panel Door Configuration
- Reader Configuration
- Commands
- Function Key Configuration
- To Download/Upload Multi-Language File
- Alarm Configuration

Panel Door List

Description: To get the List of all connected Panel Doors along with door’s pdid, door-name, door-type in the response.

Actions: get

Syntax: http://deviceIP:deviceport/device.cgi/panel-door-list?action=get

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Request - Panel Door List

Argument	Valid Values	Mandatory	Description
format	Text, xml	No	Specify the format in which response is expected

Table: Response - Panel Door List

Parameters	Description
pdid	Defines the Door Id
door-name	Defines the Door Name
door-type	Defines the Door Type

Example

Sample Request

http://deviceIP:deviceport/device.cgi/panel-door-list?action=get&format=xml

Sample Response

```
HTTP Code: 200 OK

Content-Type: <type>

Content-Length: <length>

Body:

<COSEC_API>
<COSEC DOOR>
 <pdid>1</pdid>

<door-name>test1-v3</door-name>

<door-type>12</door-type>

</COSEC DOOR>


<COSEC DOOR>
 <pdid>2</pdid>

<door-name>test2-vega</door-name>

<door-type>9</door-type>

</COSEC DOOR>



<COSEC DOOR>
 <pdid>3</pdid>

<door-name>test3-io</door-name>

<door-type>14</door-type>

</COSEC DOOR>
</COSEC_API>
```

Panel Door Configuration

Description: To enable, disable, define or retrieve configuration parameters related to various door features such as Aux input, Aux Output, Door sense, Alarm etc.

Actions: get, set, delete

Syntax: http://<deviceIP:deviceport>/device.cgi/panel-door-config?<argument>=<value>[&<argument>=<value>....]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Set Panel Door Configuration Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255(for Panel200), 1 to 75 (for Panel/Panellite v1)	Yes	To define the door id
door-name	Max 30 ASCII characters	Yes	To define the door name

Table: Set Panel Door Configuration Parameters

Argument	Valid Values	Mandatory	Description
door-type	1 - V1 Door 3 - V2 Door 6 - Compact Door (PATH V1) 7 - PVR Door 9 - VEGA Door 11 - ARC DC100 12 - V3 Door 14 - ARC IO800 16 - Path V2 17 - ARC DC200 19 - V4 DOOR 20 - ARGO 21 - ARGO FACE	Yes	To specify the door type ARGO FACE applicable for Server Mode Panel only.
active	0 = Inactive 1 = Active	No	To activate/deactivate the door
communication-type	0 = Ethernet 1 = RS-485	Yes	To define the communication type of the door with the Panel
ip-address	15 characters (0-9 digits and dot (.) only)	Yes	To define the IP address
rs-485-address	00-31	Yes	To define the RS-485 address of the door. Either IP or RS-485 address is mandatory.
mac-address	17 characters ("A-F, a-f, 0 -9, : ")	Yes Not Mandatory for RS485	To specify the MAC address of the door.
mute-buzzer	0 = Un mute 1 = Mute	No	To mute/unmute the door buzzer
access-zone	1 to 99	No	To define the access zone
door-sense	0- Inactive 1- Active	No	To enable/disable the door sense
door-sense-supervised	0- Un-supervised 1- Supervised	No	To enable/disable the supervised sense of the door
door-sense-type	0- NO 1- NC	No	To define the sense type of the door
door-relay-unlock-output-group-num	1 to 99	No	To define the output group number of the door for door relay for unlocking the door
door-relay-lock-output-group-num	1 to 99	No	To define the output group number of the door for door relay for unlocking the door
aux-input	0- Inactive 1- Active	No	To enable the aux input of the door.

Table: Set Panel Door Configuration Parameters

Argument	Valid Values	Mandatory	Description
aux-input-2	0- Inactive 1- Active	No	To enable the aux input of the door. Parameter applicable for IO controller device type and ARC DC200 Single Door only.
aux-input-3	0- Inactive 1- Active	No	To enable the aux input of the door. Parameter applicable for IO controller device type only.
aux-input-4	0- Inactive 1- Active	No	To enable the aux input of the door. Parameter applicable for IO controller device type only.
aux-input-5	0- Inactive 1- Active	No	To enable the aux input of the door. Parameter applicable for IO controller device type only.
aux-input-6	0- Inactive 1- Active	No	To enable the aux input of the door. Parameter applicable for IO controller device type only.
aux-input-7	0- Inactive 1- Active	No	To enable the aux input of the door. Parameter applicable for IO controller device type only.
aux-input-8	0- Inactive 1- Active	No	To enable the aux input of the door. Parameter applicable for IO controller device type only.
aux-input-supervised	0- Un-supervised 1- Supervised	No	To enable/disable the supervised sense of the aux input
aux-input-supervised-2	0- Un-supervised 1- Supervised	No	To enable/disable the supervised sense of the aux input. Parameter applicable for IO controller device type only.
aux-input-supervised-3	0- Un-supervised 1- Supervised	No	To enable/disable the supervised sense of the aux input. Parameter applicable for IO controller device type only.
aux-input-supervised-4	0- Un-supervised 1- Supervised	No	To enable/disable the supervised sense of the aux input. Parameter applicable for IO controller device type only.
aux-input-supervised-5	0- Un-supervised 1- Supervised	No	To enable/disable the supervised sense of the aux input. Parameter applicable for IO controller device type only.
aux-input-supervised-6	0- Un-supervised 1- Supervised	No	To enable/disable the supervised sense of the aux input. Parameter applicable for IO controller device type only.

Table: Set Panel Door Configuration Parameters

Argument	Valid Values	Mandatory	Description
aux-input-supervised-7	0- Un-supervised 1- Supervised	No	To enable/disable the supervised sense of the aux input. Parameter applicable for IO controller device type only.
aux-input-supervised-8	0- Un-supervised 1- Supervised	No	To enable/disable the supervised sense of the aux input. Parameter applicable for IO controller device type only.
aux-input-sense-type	0- NO 1- NC	No	To define the sense type of the aux input
aux-input-sense-type-2	0- NO 1- NC	No	To define the sense type of the aux input. Parameter applicable for IO controller device type and ARC DC200 Single Door only.
aux-input-sense-type-3	0- NO 1- NC	No	To define the sense type of the aux input. Parameter applicable for IO controller device type only.
aux-input-sense-type-4	0- NO 1- NC	No	To define the sense type of the aux input. Parameter applicable for IO controller device type only.
aux-input-sense-type-5	0- NO 1- NC	No	To define the sense type of the aux input. Parameter applicable for IO controller device type only.
aux-input-sense-type-6	0- NO 1- NC	No	To define the sense type of the aux input. Parameter applicable for IO controller device type only.
aux-input-sense-type-7	0- NO 1- NC	No	To define the sense type of the aux input. Parameter applicable for IO controller device type only.
aux-input-sense-type-8	0- NO 1- NC	No	To define the sense type of the aux input. Parameter applicable for IO controller device type only.
debounce-time(sec)	0 to 99	No	To define the debounce time
debounce-time-2 (sec)	0 to 99	No	To define the debounce time. Parameter applicable for IO controller device type and ARC DC200 Single Door only
debounce-time-3 (sec)	0 to 99	No	To define the debounce time. Parameter applicable for IO controller device type only.

Table: Set Panel Door Configuration Parameters

Argument	Valid Values	Mandatory	Description
debounce-time-4 (sec)	0 to 99	No	To define the debounce time. Parameter applicable for IO controller device type only.
debounce-time-5 (sec)	0 to 99	No	To define the debounce time. Parameter applicable for IO controller device type only.
debounce-time-6(sec)	0 to 99	No	To define the debounce time. Parameter applicable for IO controller device type only.
debounce-time-7 (sec)	0 to 99	No	To define the debounce time. Parameter applicable for IO controller device type only.
debounce-time-8 (sec)	0 to 99	No	To define the debounce time. Parameter applicable for IO controller device type only.
aux-output	0- Inactive 1- Active	No	To enable/disable the aux output of the door
aux-output-2	0- Inactive 1- Active	No	To enable/disable the aux output of the door
aux-output-3	0- Inactive 1- Active	No	To enable/disable the aux output of the door
aux-output-4	0- Inactive 1- Active	No	To enable/disable the aux output of the door
aux-output-5	0- Inactive 1- Active	No	To enable/disable the aux output of the door
aux-output-6	0- Inactive 1- Active	No	To enable/disable the aux output of the door
aux-output-7	0- Inactive 1- Active	No	To enable/disable the aux output of the door
aux-output-8	0- Inactive 1- Active	No	To enable/disable the aux output of the door
aux-output-group-num	1 to 99	No	To define the Aux output group number of the door
aux-output-group-num-2	1 to 99	No	To define the Aux output group number of the door
aux-output-group-num-3	1 to 99	No	To define the Aux output group number of the door
aux-output-group-num-4	1 to 99	No	To define the Aux output group number of the door
aux-output-group-num-5	1 to 99	No	To define the Aux output group number of the door
aux-output-group-num-6	1 to 99	No	To define the Aux output group number of the door

Table: Set Panel Door Configuration Parameters

Argument	Valid Values	Mandatory	Description
aux-output-group-num-7	1 to 99	No	To define the Aux output group number of the door
aux-output-group-num-8	1 to 99	No	To define the Aux output group number of the door
pulse-time	1 to 65535 sec	No	To define pulse time
auto-relock	0- Inactive 1- Active	No	To activate and deactivate the auto relock.
auto-relock-timer	1 to 65535 sec	No	To define relock time
tamper-readergrp1	0- NO 1-NC	No	To select whether it is NO/NC
tamper-readergrp2	0-NO 1-NC	No	To select whether it is NO/NC
format	text,xml	No	Specifies the format in which the response is expected.

Table: Get Panel Door Configuration Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 TO 255	Yes	To define the door id
format	Text, xml	No	Specify the format in which response is expected

Table: Response Panel Door Configuration

Parameters	Description
pdid	Defines the door id
door-name	Defines the door name
door-type	Defines the door type
active	Defines whether the door is active or not
communication-type	Defines the communication type of the door with the panel
ip-address	Defines the ip address of the door
rs-485-address	Defines the RS-485 address of the door
mac-address	Defines the MAC address of the door. Not mandatory if the communication type is defined as RS-485.
mute-buzzer	Defines whether the door buzzer is Mute/unmute
access-zone	Defines the access zone
door-sense	Defines whether the door is enabled/disabled.
door-sense-supervised	Defines whether the supervised sense of the door is enabled/disabled.
door-sense-type	Defines the sense type of the door.

Parameters	Description
door-relay-unlock-output-group-num	Defines the output group number of the door for door relay for unlocking the door.
door-relay-lock-output-group-num	Defines the output group number of the door for door relay locking the door.
aux-input	Defines whether the aux input of the door is enabled/disabled.
aux-input-2	Defines whether the aux input of the door is enabled/disabled. Parameter applicable for IO controller device type and ARC DC200 Single Door only.
aux-input-3	Defines whether the aux input of the door is enabled/disabled. Parameter applicable for IO controller device type only
aux-input-4	Defines whether the aux input of the door is enabled/disabled. Parameter applicable for IO controller device type only
aux-input-5	Defines whether the aux input of the door is enabled/disabled. Parameter applicable for IO controller device type only
aux-input-6	Defines whether the aux input of the door is enabled/disabled. Parameter applicable for IO controller device type only
aux-input-7	Defines whether the aux input of the door is enabled/disabled. Parameter applicable for IO controller device type only
aux-input-8	Defines whether the aux input of the door is enabled/disabled. Parameter applicable for IO controller device type only
aux-input-supervised	Defines whether the supervised sense of the door is enabled/disabled
aux-input-supervised-2	Defines whether the supervised sense of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-input-supervised-3	Defines whether the supervised sense of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-input-supervised-4	Defines whether the supervised sense of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-input-supervised-5	Defines whether the supervised sense of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-input-supervised-6	Defines whether the supervised sense of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-input-supervised-7	Defines whether the supervised sense of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-input-supervised-8	Defines whether the supervised sense of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-input-sense-type	defines the sense type of the aux input
aux-input-sense-type-2	defines the sense type of the aux input Parameter applicable for IO controller device type and ARC DC200 Single Door only
aux-input-sense-type-3	defines the sense type of the aux input Parameter applicable for IO controller device type only
aux-input-sense-type-4	defines the sense type of the aux input Parameter applicable for IO controller device type only
aux-input-sense-type-5	defines the sense type of the aux input Parameter applicable for IO controller device type only

Parameters	Description
aux-input-sense-type-6	defines the sense type of the aux input Parameter applicable for IO controller device type only
aux-input-sense-type-7	defines the sense type of the aux input Parameter applicable for IO controller device type only
aux-input-sense-type-8	defines the sense type of the aux input Parameter applicable for IO controller device type only
debounce-time (sec)	defines the debounce time
debounce-time-2 (sec)	defines the debounce time Parameter applicable for IO controller device type and ARC DC200 Single Door only
debounce-time-3 (sec)	defines the debounce time Parameter applicable for IO controller device type only
debounce-time-4 (sec)	defines the debounce time Parameter applicable for IO controller device type only
debounce-time-5 (sec)	defines the debounce time Parameter applicable for IO controller device type only
debounce-time-6 (sec)	defines the debounce time Parameter applicable for IO controller device type only
debounce-time-7 (sec)	defines the debounce time Parameter applicable for IO controller device type only
debounce-time-8 (sec)	defines the debounce time Parameter applicable for IO controller device type only
aux-output	defines whether the aux output of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-output-2	defines whether the aux output of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-output-3	defines whether the aux output of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-output-4	defines whether the aux output of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-output-5	defines whether the aux output of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-output-6	defines whether the aux output of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-output-7	defines whether the aux output of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-output-8	defines whether the aux output of the door is enabled/disabled Parameter applicable for IO controller device type only
aux-output-group-num	defines the aux output group number of the door.
aux-output-group-num-2	defines the aux output group number of the door. Parameter applicable for IO controller device type only
aux-output-group-num-3	defines the aux output group number of the door. Parameter applicable for IO controller device type only

Parameters	Description
aux-output-group-num-4	defines the aux output group number of the door. Parameter applicable for IO controller device type only
aux-output-group-num-5	defines the aux output group number of the door. Parameter applicable for IO controller device type only
aux-output-group-num-6	defines the aux output group number of the door. Parameter applicable for IO controller device type only
aux-output-group-num-7	defines the aux output group number of the door. Parameter applicable for IO controller device type only
aux-output-group-num-8	defines the aux output group number of the door. Parameter applicable for IO controller device type only
pulse-time	Defines pulse timer of the door
auto-relock	Defines whether auto-relock is enabled/ disabled in the door
auto-relock-timer	Defines relock timer of the door
tamper-readergrp1	To select whether it is NO/NC
tamper-readergrp2	To select whether it is NO/NC

Table: Delete Panel Door Configuration Parameters

Argument	Valid Values	Mandatory	Description
pdid	Refer configuration table for valid value	Yes	To select the ID on which the specified operation is to be done.
format	Text, xml	No	Specify the format in which response is expected

For IO Controller, Valid Parameters are: action, pdid, door-name, door-type, active, communication-type, ip-address, rs-485-address, mac-address, aux-input, aux-input-(2 to 8), aux-input-supervised, aux-input-supervised-(2 to 8), aux-input-sense, aux-input-sense-(2 to 8), debounce-time, debounce-time-(2 to 8), aux-output, aux-output-(2 to 8), aux-output-group-num, aux-output-group-num-(2 to 8), format

Reader Configuration

Description: To set or retrieve configuration parameters for internal and external readers such as reader type, access mode, entry-exit mode and the tag re-detection delay time.

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi/reader-config?action=<value>&<argument>=<value>….

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Reader Configuration Parameters (All doors except COSEC ARC Controller)

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door
internal-card-reader	0 = None 1 = EM Prox Reader 2 = HID Prox Reader 3 = MiFare Reader 4 = HID iCLASS-U Reader 5 = HID iCLASS-W Reader	No	To define the internal card reader.
internal-biometric-reader	0 = None 1 = Finger Reader 2 = Palm Vein Reader	No	To define the internal biometric reader.
external-reader	0 = None 1 = EM Prox Reader 2 = HID Prox Reader 3 = MiFare U Reader 4 = HID iCLASS-U Reader 5 = Finger Reader 6 = HID iCLASS-W Reader 7 = UHF Reader 8 = Combo Exit Reader 9 = MiFare-W Reader 10=PIN-W Reader 11=CB - U Reader 12=CB - W Reader 13=ATOM RD300 14=ATOM RD200 15=ATOM RD100	No	To define the external reader.
internal-reader-mode	0 = Entry 1 = Exit	No	To define the door mode for internal reader.
external-reader-mode	0 = Entry 1 = Exit	No	To define the door mode for external reader.
exit-switch	0 = Inactive 1 = Active	No	To activate exit switch.
format	text,xml	No	Specifies the format in which the response is expected.
tag-re-detect-delay	0-60 mins	No	To define the tag re-detection delay time.

Table: Reader Configuration Parameters (COSEC ARC Controller only)

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes if action is desired on Panel door	To select the Panel door
rs-485-readergrp1	0 = None 1 = EM Prox Reader 2 = HID Prox Reader 3 = MiFare Reader 4 = HID iCLASS-U Reader 5 = Combo Reader 6 = CB - U Reader 7 = ATOM RD300 8 = ATOM RD200 9 = ATOM RD100	No	To define the RS-485 reader in reader1 group.
wiegand-readergrp1	0 = None 1 = Short-Range Reader 2 = Long-Range Reader 3 = PIN Reader 4 = Card+ PIN W Reader 5 = CB - W Reader	No	To define the Wiegand reader in reader1 group.
readergrp1-entry-exit-mode	0 = Entry 1 = Exit	No	To define the mode (entry/exit) for reader1 group
rs-485-readergrp2	0 = None 1 = EM Prox Reader 2 = HID Prox Reader 3 = MiFare Reader 4 = HID iCLASS-U Reader 5 = Combo Reader 6 = CB - U Reader 7 = ATOM RD300 8 = ATOM RD200 9 = ATOM RD100	No	To define the RS-485 reader in reader2 group.
wiegand-readergrp2	0 = None 1 = Short-Range Reader 2 = Long-Range Reader 3 = PIN Reader 4 = Card+ PIN W Reader 5 = CB - W Reader	No	To define the Wiegand reader in reader2 group.
readergrp2-entry-exit-mode	0 = Entry 1 = Exit	No	To define the mode (entry/exit) for reader2 group
exit switch	0 = Entry 1 = Exit	No	To activate exit switch
tag-re-detect-delay	0-60 mins	No	To define the tag re-detection delay time.

Example

1. **To configure internal card reader as an HID Prox reader and internal reader mode as entry**.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/reader-config?action=set&pdid=1&reader1=2&door-access-mode=0
```

Sample Response

```
HTTP Code: 200 OK
Content-Type: <type>
Content-Length: <length>
Body: Response-Code=0
```

Commands

Description: To send commands to Panel Door.

Syntax: http://<deviceIP:deviceport>/device.cgi/door-commands?action=<value>&<argument>=<value>….

1. To normalize the Door

Action= normalizedoor

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Normalize Door Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door

2. To lock the door

Action=lockdoor

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: To Lock the Door Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door

3. To unlock the door

Action=unlockdoor

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: To Unlock the Door Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door

4. To disable aux input of the door

Action= disableauxinput

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Disable Aux Input Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door
port-no	1 to 8	No	To select the port Only applicable for IO controller, Port 1 and Port 2 will be applicable for ARC DC200 Single Door Dual Reader.

5. To normalize aux input of the door

Action= normalizeauxinput

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Normalize Aux Input Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door
port-no	1 to 8	No	To select the aux port Only applicable for IO controller, Port 1 and Port 2 will be applicable for ARC DC200 Single Door Dual Reader.

6. To disable aux output of the door

Action= disableauxoutput

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Disable Aux Output Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door
port-no	1 to 8	No	To select the aux port Only applicable for IO controller

7. To normalize aux output of the door

Action= normalizeauxoutput

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Normalize Aux Output Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door
port-no	1 to 8	No	To select the auxport Only applicable for IO controller

8. To disable door sense

Action= disabledoorsense

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: To Disable the Door Sense Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door

9. To normalize door sense

Action= normalizedoorsense

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: To Normalize the Door Sense Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door

10. To reset aux output latch of the door

Action= resetauxoutputlatch

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Reset Aux Output Latch Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door
port-no	1 to 8	No	To select the aux port Only applicable for IO controller

11. To clear biometric credentials of the door

Action=clearbiometriccredentials
Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Clear Biometric Credentials Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door

12. To synchronize credentials

Action= synccredentials

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Synchronize Credentials Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door

13. To calibrate FP sensor

Action= calibratefpsensor

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Calibrate FP Sensor Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select Panel Door

14. To Activate Aux Relay

Action= activateauxrelay

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Activate Aux Relay Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select the ID on which the specified operation is to be done.
port-no	1 to 8	No	To select the aux port Port 2 to 8 are valid for IO Controller only

15. To Deactivate Aux Relay

Action= deactivateauxrelay

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Activate Aux Relay Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes (if action is desired on Panel door.)	To select the ID on which the specified operation is to be done.
port-no	1 to 8	No	To select the aux port Port 2 to 8 are valid for IO Controller only

Function Key Configuration

Description: To set or retrieve configuration of Function Keys on the Panel door keypad. COSEC enables its users to map up to 4 special functions to the arrow keys on a Door keypad. These functions can then be performed at the door by using the keypad shortcuts. Use this API to specify which special functions are to be assigned shortcuts on COSEC devices.

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi/function-key?action=<value>&<argument>=<value>….

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Function Key Configuration Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes	To define the Panel door ID
F1	0 = None 1 = Official IN 2 = Official OUT 3 = Short Leave IN 4 = Short Leave OUT 5 = Regular IN 6 = Regular OUT 7 = Post Break IN 8 = Pre - Break OUT 9 = Overtime IN 10 = Overtime OUT	No	Assigning special functions to respective function keys.
F2			
F3			
F4			
format	text,xml	No	Specifies the format in which the response is expected.

Example

1. **To configure function key F1 as official work – IN.**

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/function-key?action=set&f1=1
```

Sample Response

```
HTTP Code: 200 OK
Content-Type: <type>
Content-Length: <length>
Body: Response-Code=0
```

To Download/Upload Multi-Language File

Description: To download/upload multi-language file for custom message display on supported COSEC panel doors for which multi-language support has been enabled.

File uploaded can be in XLS or CSV format only.

Actions: get, set, getdefault

Syntax: http://<deviceIP:deviceport>/device.cgi/multi-language-file-door?<argument>=<value>[&<argument>=<value>]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Download/Upload Multi-Language File - Parameters

Argument	Valid Values	Mandatory	Description
action	get	Yes	To download the multi-language file for the panel door.
action	getdefault	Yes	To download the sample multi-language file for the panel door
action	set	Yes	To upload a custom message file for the panel door

Alarm Configuration

Description: To set or retrieve configuration of alarm on the Panel door keypad.

Actions: get, set, getdefault, setdefault

Syntax: http://**<deviceIP:deviceport>**/**device.cgi/**door-alarm?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Alarm Configuration- Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255(for Panel200), 1 to 75 (for Panel/Panellite v1)	Yes	To define the door id
alarm	0- Inactive 1- Active	No	To enable/disable Alarm
tamper-alarm	0- Inactive 1- Active	No	To enable/disable the “Tamper” Alarm
duress-alarm	0- Inactive 1- Active	No	To enable/disable "Duress" alarm
door-held-open-alarm	0- Inactive 1- Active	No	To enable/disable “Door Held Open” alarm
door-abnormal-alarm	0- Inactive 1- Active	No	To enable/disable “Door Abnormal” alarm
door-force-open-alarm	0- Inactive 1- Active	No	To enable/disable “Door Force Open” alarm
door-fault-alarm	0- Inactive 1- Active	No	To enable/disable “Door Fault” alarm
panic alarm	0- Inactive 1- Active	No	To enable/disable “Panic” alarm
deadman-alarm	0- Inactive 1- Active	No	To enable/disable “Deadman” alarm
occupancy-violated alarm	0- Inactive 1- Active	No	To enable/disable “Occupancy Violated” alarm
tail-gating-alarm	0- Inactive 1- Active	No	To enable/disable “Tail Gating” alarm
man-trap-violation-alarm	0- Inactive 1- Active	No	To enable/disable “Man Trap Violation” alarm
anti-pass-back-alarm	0- Inactive 1- Active	No	To enable/disable “Access Denied-Anti Pass Back” alarm
access-denied-othr-reason-alarm	0- Inactive 1- Active	No	To enable/disable “ Access Denied – Other Reason” alarm
user-unidentified-alarm	0- Inactive 1- Active	No	To enable/disable “User Unidentified” alarm

Table: Alarm Configuration- Parameters

Argument	Valid Values	Mandatory	Description
mltpl-unauth-atmpt-alarm	0- Inactive 1- Active	No	To enable/disable “Multiple Unauthorized Attempts” alarm
door-lock-held-open-alarm	0- Inactive 1- Active	No	To enable/disable “Door Lock Held Open” alarm
door-lock-abnormal-alarm	0- Inactive 1- Active	No	To enable/disable “Door Lock Abnormal” alarm
door-lock-manual-open-alarm	0- Inactive 1- Active	No	To enable/disable “Door Lock Manual Open” alarm
thresh-temp-excded	0- Inactive 1- Active	No	To enable/ disable the “User Denied – Threshold Temperature Exceeded” alarm
access-route-alarm	0- Inactive 1- Active	No	To enable/ disable the “Access Route Violation” alarm
access-route-timer-alarm	0- Inactive 1- Active	No	To enable/ disable the “Access Route Timer Violation” alarm
other-alarm	0- Inactive 1- Active	No	To enable/disable “Other” alarms
format	Text, XML	No	Specifies the format in which the response is expected

Panel Configuration

This group of APIs enables users to perform the following types of Panel Configuration:

- *Panel Basic Configuration*
- *Access Settings Configuration*
- *Panel Advance Configuration*
- *Enrollment Configuration*
- *Alarm Configuration*
- *Date and Time Configuration*
- *System Timers Configuration*
- *Special Function Configuration*
- *Multi-Language Support*
- *To Download/Upload Multi-Language File*
- *Access Features*

Panel Basic Configuration

Description: To set or retrieve basic configuration parameters for a Panel such as name and maximum number of finger templates on device.

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi/panel-basic-config?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Panel Basic Configuration Parameters

Argument	Valid Values	Mandatory	Description
name	Alphanumeric, Max. 30 characters	No	To identify/configure the device name.
Template-per-finger	0- Single Template/Finger 1- Dual Template/Finger	No	To specify the number of finger template to be enrolled per user.
max-fingers	Single Template/Finger: 0-9 where, 0 - 1 Finger 1 - 2 Fingers 2 - 3 Fingers 3 - 4 Fingers 4 - 5 Fingers 5 - 6 Fingers 6 - 7 Fingers 7 - 8 Fingers 8 - 9 Fingers 9 - 10 Fingers Dual Template/Finger: 0-4 where, 0 - 1 Finger 1 - 2 Fingers 2 - 3 Fingers 3 - 4 Fingers 4 - 5 Fingers	No	Maximum no. of finger templates that can be stored per user on this device.

Table: Panel Basic Configuration Parameters

Argument	Valid Values	Mandatory	Description
max-palms	0 - 1 Palm 1 - 2 Palms 2 - 3 Palms 3 - 4 Palms 4 - 5 Palms 5 - 6 Palms 6 - 7 Palms 7 - 8 Palms 8 - 9 Palms 9 - 10 Palms	No	Maximum no. of palm templates that can be stored per user on this device.
palm-operation-mode	0 - Non-guide mode 1 - Guide mode	No	To specify the mode as Guided or Un-guided mode. Default mode is Un-guided mode.
palm-mode-adaptive	0-Basic Template 1-Compressed Template	No	to define whether PVR will run in Adaptive mode or not
mode	0 - Server mode 1 - Standalone mode	No	To set the panel mode as Server mode or Standalone mode. Default mode is server mode.
format	text, xml	No	specifies the format in which the response is expected.

To get the default values for any parameter, use the ***action=getdefault*** *method. To restore configuration parameters on device to default values, use the* ***action=setdefault*** *method.*

Access Settings Configuration

Description: To set or retrieve configuration parameters for enabling basic access control on a device for users.

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi /access-setting?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Access Settings Configuration Parameters

Argument	Valid Values	Mandatory	Description
week-day<0~6>	sun (0) to sat (6) 0 = Inactive 1 = Active	No	To define the active working days. This parameter is repeated for each day of the week.
work-start-hh	00-23	No	Define the work start time
work-start-mm	00-59	No	Define the work start time
work-end-hh	00-23	No	Define the work stop time
work-end-mm	00-59	No	Define the work stop time
format	text, xml	No	Specifies the format in which the response is expected

Panel Advance Configuration

Description: To set or retrieve advance configuration parameters for a device such as application type and Additional Security Code on device.

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi/panel-advanced-config?action=<value>&<argument>=<value>….

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Device Configuration Parameters

Argument	Valid Values	Mandatory	Description
facility-code	16 bits, 1-65535 range, Should be non-zero	No	
asc-active	0- Inactive 1- Active	No	To enable/disable the additional security code.
asc-code	Numeric, 16 bits, 1-65535 range	No	To configure an Additional Security Code (ASC). Should be non-zero.
generate-invalid-user-events	0 - No 1 - Yes	No	To generate invalid user events when invalid user is punched in.
generate-exit-switch-events	0 - No 1 - Yes	No	To generate exit switch events.
degrade-access	0- Inactive 1- Active	No	To enable the access in degrade mode.
degrade-access-wait-timer	1 to 99 sec	No	To specify the degrade wait timer.
format	text, xml	No	specifies the format in which the response is expected.

Enrollment Configuration

Description: To set or retrieve configuration parameters for enrollment of credentials on a device such as number of credentials allowed, number of templates allowed per finger, enrollment mode etc.

Actions: get, set, getdefault, setdefault

Syntax:http://<deviceIP:deviceport>/device.cgi /enroll-options?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Enrollment Configuration Parameters

Argument	Valid Values	Mandatory	Description
enroll-on-device	0 = Inactive 1 = Active	No	To enable/disable the feature to enroll through special function
enroll-using	0 = User ID 1 = Reference No.	No	To define the option to enroll the credential using the user's Reference No. or User ID, for enrollment through special function. Note: This parameter will not be valid for NGT Direct Door and Vega Controller where enrollment must be performed by User ID.
enroll-finger-count	Single Template/Finger: 0-9 where, 0 = 1 Finger 1 = 2 Fingers 2 = 3 Fingers 3 = 4 Fingers 4 = 5 Fingers 5 = 6 Fingers 6 = 7 Fingers 7= 8 Fingers 8 = 9 Fingers 9 = 10 Fingers Dual Template/Finger: 0-4 where, 0 = 1 Finger 1 = 2 Fingers 2 = 3 Fingers 3 = 4 Fingers 4 = 5 Fingers	No	No. of fingers allowed to be enrolled in one enrollment cycle. Note: For the **action=set** method, this value should not be greater than the **max-finger** value set in Panel Device Configuration API.

Table: Enrollment Configuration Parameters

Argument	Valid Values	Mandatory	Description
enroll-palm-count	0 = 1 Palm 1 = 2 Palms 2 = 3 Palms 3 = 4 Palms 4 = 5 Palms 5 = 6 Palms 6 = 7 Palms 7 = 8 Palms 8 = 9 Palms 9 = 10 Palms	No	No. of palms allowed to be enrolled in one enrollment cycle.
enroll-card-count	0 = 1 Card 1 = 2 Cards 2 = 3 Cards 3 = 4 Cards	No	No. of special function cards allowed to be enrolled in one enrollment cycle.
enroll-mode	0 = Read Only Card 1 = Smart Card 2 = Biometric 3 = Biometric then Card 4 = Face 5 = Duress Finger	No	To define the enrollment mode for enrollment through device.
format	text,xml	No	Specifies the format in which the response is expected.

- *If the **temp-per-finger** mode is changed, then the templates have to be restored to the device explicitly by the third party software, else mismatch will occur in the module.*

- *If **Single Template/Finger** mode is selected on the device and some users are already enrolled according to it and if abruptly the mode is changed to **Dual Template/Finger** then:*

  i. *If the maximum finger count was greater than 5 fingers in Single Template/Finger mode, then after changing the mode to the Dual Template/Finger, the finger count will set to 5.*

  ii. *If the maximum finger count was less than 5 fingers in Single Template/Finger mode, then after changing the mode to the Dual Template/Finger, the finger count will remain same.*

- *If the mode is changed back to Single Template/Finger, then finger count should not be changed. If users want to increase the finger count they should mention it explicitly.*

Alarm Configuration

Description: To set or retrieve configuration parameters for enabling/disabling alarms and related functions on a COSEC device such as Auto Alarm Acknowledgement.

Actions: get, set, getdefault, setdefault

Syntax:http://<deviceIP:deviceport>/device.cgi /alarm?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Alarm Configuration Parameters

Argument	Valid Values	Mandatory	Description
duress-alarm	0 = Inactive 1 = Active	No	To enable/disable the Duress alarm.
deadman-alarm	0 = Inactive 1 = Active	No	To enable/disable the Deadman alarm.
panic-alarm	0 = Inactive 1 = Active	No	To enable/disable the panic alarm.
doorcontroller-offline-alarm	0 = Inactive 1 = Active	No	To enable or disable the alarm.
doorcontroller-fault-alarm	0 = Inactive 1 = Active	No	To enable or disable the alarm.
auto-clear-alarm	0 = Inactive 1 = Active	No	To enable or disable the auto clear alarm.
auto-alarm-ack	0 = Inactive 1 = Active	No	To enable or disable the auto alarm ACK.
format	text,xml	No	Specifies the format in which the response is expected.
thresh-temp-excded	0 = Inactive 1 = Active	No	To enable/ disable the “User Denied – Threshold Temperature Exceeded” alarm
access-route-alarm	0 = Inactive 1 = Active	No	To enable/ disable the “Access Route Violation” alarm
access-route-timer-alarm	0 = Inactive 1 = Active	No	To enable/ disable the “Access Route Timer Violation” alarm

Date and Time Configuration

Description: To set or retrieve date and time configurations on a COSEC device. The user can configure the date and time to be displayed on the device, the display format, the time update mode, the NTP server settings as well as the Daylight Savings Time (DST) settings on the selected device.

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi /date-time?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Date and Time Configuration Parameters

Argument	Valid Values	Mandatory	Description
year	2009 to 2037	No	To set year value
month	01 to 12	No	To set month value
date	01 to 31	No	To set date
hour	00 to 23	No	To set hour
minute	00 to 59	No	To set minutes
second	00 to 59	No	To set seconds
time-format	0 = 24 hours 1 = 12 hours	No	Defines the time format to be displayed on the device display. Note: This is applicable only for the time shown on the device display and not for general date-time which will always be in 24 hours format.
time-zone	00-74 (Tool supported by Windows), default: GMT (+05:30) Chennai, Kolkata, Mumbai, New Delhi. Refer to *"Table: Universal Time Zone Reference" on page 95*	No	To define the universal time zone.
update-mode	0 = Auto 1 = Manual	No	Defines whether the update mode is manual or through NTP Server.
ntp-server-type	0 = Predefined 1 = User Defined	No	Defines whether the NTP server is a predefined server or user-defined server address.
ntp-server	0 = ntp1.cs.wisc.edu 1 = time.windows.com 2 = time.nist.gov	No	To define the NTP Address.
user-defined-ntp	Alphanumeric, Max. 40 characters.	No	To define the user-defined NTP.
dst-enable	0 = Disable 1 = Enable	No	To enable/disable DST.

Table: Date and Time Configuration Parameters

Argument	Valid Values	Mandatory	Description
fwd-month	0 = January 1 = February 2 = March 3 = April 4 = May 5 = June 6 = July 7 = August 8 = September 9 = October 10 = November 11 = December	No	Forward clock day
fwd-week	0 = 1st 1 = 2nd 2 = 3rd 3 = 4th 4 = Last		
fwd-day	0 = Sunday 1 = Monday 2 = Tuesday 3 = Wednesday 4 = Thursday 5 = Friday 6 = Saturday		
fwd-time-hh	00 - 23 (24 hours format only)	No	Forward clock time instance
fwd-time-mm	00 - 59		
rev-month	0 = January 1 = February 2 = March 3 = April 4 = May 5 = June 6 = July 7 = August 8 = September 9 = October 10 = November 11 = December	No	Reverse clock day
rev-week	0 = 1st 1 = 2nd 2 = 3rd 3 = 4th 4 = Last	No	

Table: Date and Time Configuration Parameters

Argument	Valid Values	Mandatory	Description
rev-day	0 = Sunday 1 = Monday 2 = Tuesday 3 = Wednesday 4 = Thursday 5 = Friday 6 = Saturday	No	Reverse clock day
rev-time-hh	00 - 23 (24 hours format only)	No	Reverse clock time instance
rev-time-mm	00 - 59		
duration-hh	00 - 23 (24 hours format only)	No	Time by which clock should be forwarded or reversed.
duration-mm	00 - 59		
format	text,xml	No	Specifies the format in which the response is expected.

- *When user sets the time locally it should be GMT time. And in GET command also the time value to be returned will be GMT time irrespective of the time displaying on the device.*
- *While configuring Daylight Saving Parameters, users are responsible to define the forward and reverse time properly.*

System Timers Configuration

Description: To set or retrieve configurations for the following system timers:

Auto Alarm Acknowledgement Timer	Specifies the time period in seconds after which an unacknowledged alarm will acknowledge itself automatically.
Inter Digit Wait Timer	Specifies time period in seconds between two key inputs on the device keypad. On the expiry of this timer, the system considers the user input to be complete and is ready for the next input.
Multi Access Wait Timer	Defines the time in seconds for which the system needs to wait for the second credential input from a user when more than one credential is required to grant access.
Palm Enrollment Time Out Timer	Defines the time period in seconds within which a palm must be enrolled after generating the enrollment command.
Door Abnormal Wait Timer	Defines the time in seconds required for a door to be energized for a valid credential. If the opened door does not return to its closed state before the expiry of this timer, the door will generate a “Door Abnormal Alarm”.
Special Function Timer	Defines the time in minutes for which the Late-IN and Early-OUT special functions will remain active after being enabled at the door controller.

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi /system-timer?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: System Timers Configuration Parameters

Argument	Valid Values	Mandatory	Description
alarm-ack-timer	10 to 65535 (sec)	No	To define the timer for Auto Alarm Acknowledgement.
alarm-reissue-wait-timer	3 to 99 min	No	To specify the time for an acknowledged alarm to wait before alarming again.
door-abnormal-wait-timer	1 to 255 sec	No	To specify the time for the system to wait before generating door abnormal alarm.
degrade-wait-timer	1 to 99 sec	No	To specify the time before the door switches from network fault to degrade mode.
idwt	1-99 (sec)	No	To define the Inter Digit Wait Timer.

Table: System Timers Configuration Parameters

Argument	Valid Values	Mandatory	Description
multi-access-wait-timer	3-99 (sec)	No	To define the Multi Access Wait Timer. We recommend you to set the timer value as greater than or equal to 10 seconds to avoid access denial issues to users.This is applicable when the system reads the credentials (biometric) from the user's Smart Cards.
palm-enroll-time-out	3-99 (sec)	No	To define the Palm Enrollment Time out Timer.
sp-function-timer	1-99 (mins)	No	To define the Special Function Timer.
format	text,xml	No	Specifies the format in which the response is expected.

Special Function Configuration

Description: COSEC enables its users to perform certain pre-defined operations directly from the COSEC device. These are known as special functions. An RFID card can be encoded for a special function and the card-holder can perform this function at the device just by showing this special card.

Use this API to enable, disable, define or retrieve Special Functions configuration on a device.

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi/special-function?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Special Function Configuration Parameters

Argument	Valid Values	Mandatory	Description
Sp-fn-Index	1 = Offical Work - IN 2 = Official Work - OUT 3 = Short Leave - IN 4 = Short Leave - OUT 5 = Regular - IN 6 = Regular - OUT 7 = Break End 8 = Break Start 9 = Over Time - IN 10 = Over Time - OUT 12 = Set Panic Alarm 13 = Enroll User 14 =Enroll Special Card 15 = Delete Credentials 16 = Late IN - Start 17 = Late IN - Stop 18 = Early OUT - Start 19 = Early OUT- Stop 20 = View User Profile 21 = Activate DND 22 = Deactivate DND 23 = Activate Dead Man 24 = Deactivate Dead Man 25 = Door Lock 26 = Door Unlock 27 = Door Normal 28 = Zone Local 29 = Zone Unlock 30 = Zone Normal 31 = Mute Door Buzzer 32 = Mute Panel Buzzer 33 = Clear Door Aux Output 34 = Clear Panel Aux Output 35 = Door Arm 36 = Door Disarm 37 = Zone Arm 38 = Zone Disarm 39 = Clear Alarm	Yes	The index number of a special function.

Table: Special Function Configuration Parameters

Argument	Valid Values	Mandatory	Description
enable	0 = Disable 1 = Enable	No	To enable/disable special functions on the device.
card1	64 Bits (20 Numeric Digits approx.)	No	For Panel- Standalone Mode:- If value entered by the user contains comma, then maximum characters to be allowed to enter should be 21 characters. If no comma is detected in the card value then maximum characters supported should be 20 characters only. To define the special function card 3. To define the special function card 4.
card2	64 Bits (20 Numeric Digits approx.)	No	
card3	64 Bits (20 Numeric Digits approx.)	No	
card4	64 Bits (20 Numeric Digits approx.)	No	
format	text,xml	No	Specifies the format in which the response is expected.

Multi-Language Support

Description: To enable/disable multiple language support for custom message display on supported COSEC devices.

Languages supported are: English, Spanish, Albanian, Thai, Vietnamese

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi/multi-language?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Multi-Language Support Parameters

Argument	Valid Values	Mandatory	Description
multi-language-support	0 - Inactive 1 - Active	No	To enable/disable multi-language support.
Multi-language-input-data	0 - Inactive 1 - Active	No	To enable/disable multi-language support.

To Download/Upload Multi-Language File

Description: To download/upload multi-language file for custom message display on supported COSEC devices for which multi-language support has been enabled.

File uploaded can be in XLS or CSV format only.

Actions: get, set, getdefault

Syntax: http://<deviceIP:deviceport>/device.cgi /multi-language-file?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Download/Upload Multi-Language File - Parameters

Argument	Valid Values	Mandatory	Description
action	get	Yes	To download the multi-language file.
action	getdefault	Yes	To download the sample multi-language file.
action	set	Yes	To upload a custom message file.

Access Features

Description: To set or retrieve configuration parameters for enabling/disabling access features

Actions: get, set,

Syntax: http://**<deviceIP:deviceport>**/**device.cgi/**access-feature?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Access Feature - Parameters

Argument	Valid Values	Mandatory	Description
access-route-enable	0 to 1 0= Inactive 1= Active	No	To enable/disable access route policy on panel
allow-access-not-in route	0 to 1 0= Inactive 1= Active	No	To enable/disable "Allow access while not in Route" flag on panel
format	Text, XML	No	To specify the format.

User Configuration

The various COSEC devices have capacity to support the following number of users:

Door V2:	2000
Door V3:	50,000
Door V4:	50,000
NGT Direct Door:	10,000
Wireless Door:	50,000
Path V1 Controller:	10,000
PVR Door:	50,000
Vega Controller:	50,000
ARC DC100:	10,000
ARGO:	50,000
Path V2 Door:	50,000
ARC DC200:	50,000

This group of APIs enables users to add or delete users, set user photographs, add or fetch various configurations related to users on or from a device as well as synchronize credentials with device. The following functions can be called:

- *Setting/Retrieving User Configuration*
- *Deleting a User*
- *Setting User Credentials*
- *Retrieving User Credentials*
- *Deleting User Credentials*
- *Setting/Retrieving/Deleting User Photo*

Setting/Retrieving User Configuration

Description: To set basic user configuration parameters on a device using the ***action=set*** parameter and retrieve configuration details using ***action=get***.

Actions: get, set

Syntax: `http://<deviceIP:deviceport>/device.cgi/users?<argument>=<value>[&<argument>=<value>….]`

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: User Configuration Parameters

Argument	Valid Values	Mandatory	Description
user-id	Maximum 15 characters	Yes	To set or retrieve the alphanumeric user ID for the selected user. Note: If a ***set*** request is sent against an existing user ID, then configuration for this user will be updated with the new values.
user-index	Direct Door V2= 1 - 2,000 Path V1 Controller = 1 - 2,000 Wireless Door = 1 - 50,000 PVR = 1 - 50,000 NGT = 1 - 10,000 Vega Controller = 1 - 50,000 Path V2 Door = 1 - 50,000 ARC DC200 = 1-50,000 Door V4 = 1- 50,000 ARGO = 1- 50,000	No	To identify the index number for the selected user ID (only ***get*** parameter)
ref-user-id	Maximum 8 digits	Yes (Not mandatory for the ***get*** action)	To select the numeric user ID on which the specified operation is to be done.
name	Alphanumeric. Max. 15 characters	No	To define the user name
user-active	0 = Inactive 1 = Active	No	to activate or deactivate a user.
vip	0 = Inactive 1 = Active	No	To define a user as VIP. Note: A VIP user is a user with the special privilege to access a particular door.
validity-enable	0 = Inactive 1 = Active	No	To enable/disable the user validity.

Table: User Configuration Parameters

Argument	Valid Values	Mandatory	Description
validity-date-dd	1-31	No	To define the end date for user validity.
validity-date-mm	1-12	No	
validity-date-yyyy	2000-2037	No	
user-pin	1 to 15 Digits	No	To set the user PIN or get the event from user PIN. Note: The user-pin can be set to a blank value.
by-pass-finger	0 = Inactive 1 = Active	No	To enable/disable the bypass finger option.
by-pass-palm	0 = Inactive 1 = Active	No	To enable/disable the bypass palm option.
card1	64 Bits (8 bytes) (max value - 18446744073709551615)	No	For Panel- Standalone Mode:- If value entered by the user contains comma, then maximum characters to be allowed to enter should be 21 characters. If no comma is detected in the card value then maximum characters supported should be 20 characters only.
card2	64 Bits (8 bytes) (max value - 18446744073709551615)	No	
user-group	0-999	No	To set the user group number. **Note:** A user can be assigned to any user group ranging from 1 to 999. User group number can be set/update via “Set” action. To remove a user from an assigned user group, user group should be set to 0.
restrict-access	0 = Inactive 1 = Active	No	To enable/disable restrict user option.
route-id	1 to 255	No	To define the route id assigned to the user
format	text, xml	No	Specifies the format in which the response is expected.

- *For **set** requests only one user’s complete data should be sent at a time. Attempting to set data for multiple users at a time will return an error response. For more examples of error responses, see Error Responses.*
- *To create a new user on device, both **user-id** and **ref-user-id** are mandatory parameters to be provided, and these should be unique for each user.*
- *If a user is already configured in the system and admin wants to update the user with new information/data, only Alphanumeric User ID is sufficient but if the reference user ID is also mentioned then it would be verified whether this belongs to the same user or not.*
- *Whenever an event is generated related to a user, the required user ID field upon calling the event will always show user’s reference user ID. Whereas if “Get” action is sent to call user configuration then it will show alphanumeric user ID.*

Example

1. **To get user names for user-id = 1**

Sample Request

```
http://deviceIP:deviceport/device.cgi/users?action=get&user-id=1&format=xml
```

Sample Response

```
HTTP Code: 200 OK
Content-Type: <xml>
Content-Length: <length>
Body:
<COSEC_API>
<user-id>1</user-id>
<user-index>0</user-index>
<ref-user-id></ref-user-id>
<name></name>
<user-active>0</user-active>
<vip>0</vip>
<validity-enable>0</validity-enable>
<validity-date-dd>1</validity-date-dd>
<validity-date-mm>1</validity-date-mm>
<validity-date-yyyy>2009</validity-date-yyyy>
<user-pin></user-pin>
<by-pass-finger>0</by-pass-finger>
<card1>0</card1>
<card2>0</card2>
</COSEC_API>
```

Deleting a User

Description: To delete a user from a device. Deleting a user will result in deletion of the credentials of that user along with all the other configurations set on the device.

Actions: delete

Syntax: http://<deviceIP:deviceport>/device.cgi/users?action=delete&<argument>=<value>….

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Delete User - Parameters

Argument	Valid Values	Mandatory	Description
user-id	Maximum 15 characters	Yes	To specify the alphanumeric user ID for the user to be deleted.
format	text,xml	No	Specifies the format in which the response is expected.

Setting User Credentials

Description: To set a user's biometric or card credentials on a device.

Actions: set

Syntax: http://<deviceIP:deviceport>/device.cgi/credential?action=set&<argument>=<value>….

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Setting User Credentials - Parameters

Argument	Valid Values	Mandatory	Description
type	1 = Finger 2 = Card 3 = Palm 4 = Palm template with guide mode	Yes	To define the user credentials type.
user-id	Alphanumeric (Max 15 characters)	Yes	To select the user-id for which the credential is to be fetched.
card1	64 Bits (8 bytes) (max value - 18446744073709551615)	No	For Panel- Standalone Mode:- If value entered by the user contains comma, then maximum characters to be allowed to enter should be 21 characters. If no comma is detected in the card value then maximum characters supported should be 20 characters only.
card2	64 Bits (8 bytes) (max value - 18446744073709551615)	No	
format	text,xml	No	Specifies the format in which the response is expected.
data	-	No	This is the data of respective credential type, which is to be stored at given index number for the respective user id.

Retrieving User Credentials

Description: To retrieve a user’s credential information from a device.

Actions: get

Syntax: http://<deviceIP:deviceport>/device.cgi/credential?action=get&<argument>=<value>….

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Retrieving User Credentials - Parameters

Argument	Valid Values	Mandatory	Description
type	1 = Finger 2 = Card 3 = Palm 4 = Palm template with guide mode	Yes	To define the user credentials type.
user-id	Alphanumeric (Max. 15 characters)	Yes	To select the user-id for which the credential is to be fetched.
card1	64 Bits (8 bytes) (max value - 18446744073709551615)		For Panel- Standalone Mode:- If value entered by the user contains comma, then maximum characters to be allowed to enter should be 21 characters. If no comma is detected in the card value then maximum characters supported should be 20 characters only.
card2	64 Bits (8 bytes) (max value - 18446744073709551615)		

Table: Retrieving User Credentials - Parameters

Argument	Valid Values	Mandatory	Description
finger-index	1 = 1 Finger 2 = 2 Finger 3 = 3 Finger 4 = 4 Finger 5 = 5 Finger 6 = 6 Finger 7 = 7 Finger 8 = 8 Finger 9 = 9 Finger 10 = 10 Finger	No	Identifies the finger template/ palm template is to be set or retrieved from the device. The template will be set and retrieved from the data portion of the request and response.
palm-index	1 = 1 Palm 2 = 2 Palm 3 = 3 Palm 4 = 4 Palm 5 = 5 Palm 6 = 6 Palm 7 = 7 Palm 8 = 8 Palm 9 = 9 Palm 10 = 10 Palm 11 = 11 Palm (Compressed)	No	
format	text,xml	No	Specifies the format in which the response is expected.
data	-	No	This is the data of respective credential type, which is to be stored at given index number for the respective user id.

- *Credential parameters to be applied will depend on the credential type selected.*
- *At a time only finger print or palm can be get/set. Both cannot be set at the same time.*
- *The set command is basically similar to adding and duplication of finger template will not be verified by the device. It is expected to be handled by the 3rd party software.*
- *The method used in this case should be POST method as it consists of raw/ hex data in the data portion of the request and the response.*
- *Finger/palm index fields are not mentioned as mandatory fields because if user selects credential type card then there is no need to specify the finger or palm index, similarly if credential type is finger then palm index in not a mandatory field and vice versa.*

Deleting User Credentials

Description: To delete selected credentials of a user from a device.

Actions: delete

Syntax: http://<deviceIP:deviceport>/device.cgi/credential?action=delete&<argument>=<value>….

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Deleting User Credentials - Parameters

Argument	Valid Values	Mandatory	Description
user-id	Alphanumeric (Max. 15 characters)	Yes	To delete the credential of a particular user.
type	0 = All 1 = Finger 2 = Card 3 = Palm 4 = Palm template with guide mode	Yes	Defines the credential type to be deleted. Note: For the selected type, all credentials will be deleted.
format	text,xml	No	Specifies the format in which the response is expected.

For delete if type is all then both card and biometric credentials should be deleted.

Example

1. **To delete finger templates of user id 1.**

Sample Request

```
http://deviceIP:deviceport/device.cgi/credential?action=delete&user-id=1&type=1
```

Sample Response

```
HTTP Code: 200 OK
Content-Type: <type>
Content-Length: <length>
Body: Response-Code=0
```

Setting/Retrieving/Deleting User Photo

Description: To set/upload the user photo on the Panel using the ***action=set*** parameter, retrieve the user photo using ***action=get*** and delete the user photo using ***action=delete***

Actions: get, set, delete

Syntax: http://<deviceIP:deviceport>/device.cgi/photo?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: User Photo Parameters

Argument	Valid Values	Mandatory	Description
user-id	Maximum 15 characters	Yes	To set or retrieve the alphanumeric user ID for the selected user. Note: If a ***set*** request is sent against an existing user ID, then configuration for this user will be updated with the new values.
photo-format	0 = jpeg 1 = jpg 2 = png 3 = bmp	Yes (No for Get and Delete action)	To define the format of photograph.
format	text, xml	No	Specifies the format in which the response is expected.

Enrollment

The Enrollment APIs can be used to generate an enrollment request for a device. Once the enrollment request is successfully sent on the device, the device will initiate the enrollment process and request credentials to be provided physically, as per the credential type and sequence specified.

Perform the enrollment function on a remote door controller using these enrollment APIs:

- Enrolling a User
- Enrolling Special Cards

Enrolling a User

Description: To command a device to initiate enrollment for a user based on parameters specified.

Actions: enroll

Syntax: http://<deviceIP:deviceport>/device.cgi/enrolluser?action=enroll&<argument>=<value>….

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Enrolling User - Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes	To select the panel door
type	0 = Read Only Card 1 = Smart Card 2 = Biometric 3 = Biometric Then Card 4 = Mobile Device 7 = Face 8 = Duress Finger	Yes	Defines the credential to be enrolled.
user-id	Maximum 15 characters	Yes	Defines the alphanumeric User ID of the user whose credential is to be enrolled.
enroll-using	0 = Reader Group1 1 = Reder Group2	No	To specify on which reader the enrollment is to be done. It is only applicable for ARC controllers.
finger-count	Single Template/Finger: 0-9 where, 0 = 1 Finger 1 = 2 Fingers 2 = 3 Fingers 3 = 4 Fingers 4 = 5 Fingers 5 = 6 Fingers 6 = 7 Fingers 7= 8 Fingers 8 = 9 Fingers 9 = 10 Fingers Dual Template/Finger: 0-4 where, 0 = 1 Finger 1 = 2 Fingers 2 = 3 Fingers 3 = 4 Fingers 4 = 5 Fingers	Yes	To specify the number of fingers to be enrolled.

Table: Enrolling User - Parameters

Argument	Valid Values	Mandatory	Description
card-count	0 = 1 Card 1 = 2 Cards 2 = 3 Cards 3 = 4 Cards	Yes	To specify the number of cards to be enrolled.
palm-count	0 = 1 Palm 1 = 2 Palms 2 = 3 Palms 3 = 4 Palms 4 = 5 Palms 5 = 6 Palms 6 = 7 Palms 7 = 8 Palms 8 = 9 Palms 9 = 10 Palms	Yes	To specify the number of palms to be enrolled.
w-asc	0 = Inactive 1 = Active	No	To enable/disable the Additional Security Code (ASC) to be written on the Smart Card.
w-fc	0 = Inactive 1 = Active	No	To enable/disable the Facility Code (FC) to be written on the Smart Card.
w-ref-user-id	0 = Inactive 1 = Active	No	To enable/disable the User ID to be written on the Smart Card.
w-name	0 = Inactive 1 = Active	No	To enable/disable the User Name to be written on the Smart Card.
w-designation	0 = Inactive 1 = Active	No	To enable/disable the designation to be written on the Smart Card.
w-branch	0 = Inactive 1 = Active	No	To enable/disable the branch name to be written on the Smart Card.
w-department	0 = Inactive 1 = Active	No	To enable/disable the department name to be written on the Smart Card.
w-bg	0 = Inactive 1 = Active	No	To enable/disable the blood group to be written on the Smart Card.
w-contact	0 = Inactive 1 = Active	No	To enable/disable Emergency Contact information to be written on the Smart Card.
w-medical-history	0 = Inactive 1 = Active	No	To enable/disable the medical history to be written on the Smart Card.
w-fp-template	0 = No Templates 1 = 1 Finger Template 2 = 2 Finger Templates	No	To enable/disable the finger templates to be written on the Smart Card.

Table: Enrolling User - Parameters

Argument	Valid Values	Mandatory	Description
w-palm-template	0 = No 1 = Yes	No	To enable/disable the Palm templates to be written on the Smart Card.
name	Alphanumeric, 15 Chars, ASCII Code	No	Defines the values for the respective fields to be written on the Smart Card.
designation			
branch			
department			
bg	Maximum 4 characters. Valid Values: A+ A- B+ B- AB+ AB- O+ O- A1- A1+ A1B- A1B+ A2- A2+ A2B- A2B+ B1+	No	Defines the values for the respective fields to be written on the Smart Card. Note: '**bg**' stands for blood group of the user.
contact	Alphanumeric, 15 Chars, ASCII Code	No	
medical-history	Alphanumeric, 15 Chars, ASCII Code	No	
format	text,xml	No	Specifies the format in which the response is expected.

- *This is only to send enrollment command, if the credential is to be retrieved then it has to be retrieved explicitly using the get and set credential command.*

- *By default, if count is not specified for enroll command then consider it as one and perform the enroll operation.*

- *This enrollment has no links to the parameter configured on the device for "enroll through special function".*

Example

1. **To start enrollment of two fingers for user id 45.**

Sample Request

```
http://deviceIP:deviceport/device.cgi/enrolluser?action=enroll&pdid=1&user-id=45&type=2&finger-count=2
```

Sample Response

```
HTTP Code: 200 OK
Content-Type: <type>
Content-Length: <length>
Body: Response-Code=0
```

Enrolling Special Cards

Description: A Special Card is an RFID card which can be encoded for a special function. This API enables the user to perform enrollment of special cards on the selected device based on specified parameters such as special function ID and number of cards to be enrolled as special cards.

Actions: enroll

Syntax: http://<deviceIP:deviceport>/device.cgi/enrollspcard?action=enroll&<argument>=<value>….

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Enroll Special Cards - Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255	Yes	To select the panel door
sp-fn-id	All configured Special Functions (special function ID)	Yes	Defines the special function identification number.
type	0- Read Only Card 1- Smart Card	Yes	Defines the credential type to be enrolled
enroll-using	0 = Reader Group1 1 = Reder Group2	No	To specify on which reader the enrollment is to be done. It is only applicable for ARC controllers.
card-count	0 = 1 Card 1 = 2 Cards 2 = 3 Cards 3 = 4 Cards	No	To specify the number of cards to be enrolled.
format	text,xml	No	Specifies the format in which the response is expected.

Events

Any action that occurs or is performed using a live COSEC device is referred on the COSEC system as an Event. A client application can directly request event logs to be fetched from a specific device or be fed with live events data via the device listening port. The functions available in this API group are as follows:

- Retrieving Events
- Retrieving Events in the TCP Socket

Retrieving Events

Description: To request all or specified events from a device.

Actions: getevent

Syntax: http://<deviceIP:deviceport>/device.cgi/events?action=getevent&<argument>=<value>….

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Retrieving Events - Parameters

Argument	Valid Values	Mandatory	Description
roll-over-count	0 to 65535	Yes	This identifies the first event that is to be sent to the 3rd party from a set of events sent in this response. If the "no-of-events" field value is 1, then this will be the only event sent to the server.
seq-number	1 to 5,00,000	Yes	
no-of-events	1 to 5 (for Direct Door V2 and Path Controller) 1 to 100 (for all other Direct Doors)	No	Specifies the number of events to be fetched.
format	text,xml	No	Specifies the format in which the response is expected.

- *For different kind of events, different fields are required, to understand the functionality of an event, which are denoted as **Detail** fields.*
- *The **Detail** field in the response depends on the type of device. For further information, refer to relevant tables in the Event Configuration Reference (Appendix).*

Retrieving Events in the TCP Socket

Description: To receive all or specific events through the TCP listening port of the device.

Actions: getevent

Syntax: http://<deviceIP:deviceport>/device.cgi/tcp-events?action=getevent&<argument>=<value>….

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Retrieving Events in the TCP Socket - Parameters

Argument	Valid Values	Mandatory	Description
trigger	1 = Start 0 = Stop	No	To start/ stop the process.
keep-live-events	0 = inactive 1 = active	No	To specify whether the events should be sent continuously or till the Maximum limit.
ipaddress	0 to 15 char ASCII 0-9,'.'	Yes	Defines the IP Address on which the events are to be sent.
port	1024 – 65535	Yes	Defines the listening port on which the events are to be sent.
roll-over-count	0 to 65535	Yes	It is used to specify the exact sequence number of an event stored at any port.
seq-number	1 to 5,00,000	Yes	It is used to specify the sequence number of any event. The maximum value for this can be from 1 to the event log capacity of that device.
response-time	3 - 15 seconds	No	To specify the response time to wait for a confirmation of established network.
interface	0 = Ethernet 1 = Wi-Fi 2 = Mobile Broadband	No	Specifies the interface. Note: If no interface is defined, **Ethernet** will be tried by default.
format	text,xml	No	Specifies the format in which the response is expected.

Due to memory constraints, this API is not supported on Door V2.

Example

1. **To request to send the events continuously on the TCP port from event seq 1 and roll over count 0 on IP address 192.168.102.42 and tcp listening port 80.**

Sample Request

```
http://deviceIP:deviceport/device.cgi/tcp-events?action=getevent&ipaddress=192.168.102.42&port=80&roll-over-
count=0&seq-number=1
```

Sample Response

```
HTTP Code: 200 OK
Content-Type: <type>
Content-Length: <length>
Body: Response-Code=0
```

- *The default TCP protocol acknowledgement should be used to send the next event. If in case any event is missed in between, then it is the responsibility of the 3rd party to re-request for that event. This shouldn't be done via TCP port but missed events can be re-requested through HTTP API.*

- *If during the event transferring if reboot occurs then the prior command (to send events) will no longer be valid and client must re-request events. In such a case, the events which have already been sent, will be overwritten by the same.*

- *The user ID against which an event is stored must be the Reference ID for a user. This being numeric (max. 8 digits), will enable efficient utilization of storage space on devices, especially those having high event logging capacity (upto 5,00,000 events).*

Access Zone

Access Zones are areas with well defined boundaries, which are defined to effectively implement an Access Security System with Access Policies. A site can have multiple Access Zones, each Zone having multiple door controllers. User needs to define the Access Zones before defining the door controllers and assigning the Access Zones.

Syntax: http://<deviceIP:deviceport>/device.cgi/access-zone?<argument>=<value>[&<argument>=<value>….]

For Action= Get

Argument	Valid Values	Mandatory	Description
zone-id	1 to 99	Yes	To specify the zone-id whose details are to be fetched.
format	text,xml	No	Specifies the format in which the response is expected.

Response contains following parameters and their values:

Argument	Description
zone-name	Defines the zone name
access-level	Defines the access level
degrade-mode	Defines if degrade mode functionality is enabled or not
degrade-mode-type	Defines the level of degrade mode access ie. Basic or Advance
access-control-on-exit	Defines if basic access policies are to be checked on exit or not
internal-access-mode	Defines access mode of internal reader or reader group 1 reader
external-access-mode	Defines access mode of external reader or reader group 2

For Action= Set

Argument	Valid Values	Mandatory	Description
zone-id	1 to 99	Yes	To specify the zone-id whose details are to be fetched.
zone-name	15 char, ASCII	No	To specify the zone name.
access-level	1 to 15	No	To specify the access level of the zone.
degrade-mode	0 - Inactive 1 - Active	No	To enable degrade mode functionality.
degrade-mode-type	0 - Basic 1 - Advance	No	To define the type of degrade mode: Basic or Advance. By default basic will be set.
access-control-on-exit		No	To specify whether to check basic access policies on exit or not

Argument	Valid Values	Mandatory	Description
internal-access-mode		No	To define access mode of internal reader or reader group 1 reader
external-access-mode		No	To define access mode of external reader or reader group 2

For Action= delete

Argument	Valid Values	Mandatory	Description
zone-id	1 to 99	Yes	To specify the zone-id whose details are to be fetched.

Sending Commands to Panel

It is possible to send CGI commands to a panel in order to perform certain functions.

The generic URL for these commands:

```
http://<deviceIP:deviceport>/device.cgi/command?action=<value>&<argument>=<value>….
```

Table: List of Commands to Device

S.No.	Command to Device	Action	Description
1	Clear Alarm	clearalarm	To command the device to clear an alarm. For parameters, refer Table: clear alarm - Parameters below.
2	Get Credential Count for Enrolled Credentials	getcount	To get the count of already enrolled templates and credentials for a user on the selected device. For parameters, refer *Table: Get Credential Count Command - Parameters* below.
3	Acknowledge Alarm	acknowledgealarm	To command the device to acknowledge an alarm without clearing it. For parameters, refer *Table: acknowledge alarm - Parameters* below.
4	Get User Count on Device	getusercount	To obtain the total number of users added on a device.
5	Get Current Event Sequence Number	geteventcount	To get the current event sequence number and roll over count in a device.
6	Default the System Configuration	systemdefault	To set all the configurations on the device to default status.
7	Delete Credentials for All Users	deletecredential	To delete all biometric credentials of users from device. For parameters, refer *Table: Deleting Credentials for All Users - Parameters* below.
8	Reset IO Link	reset-io-link	To reset the active latch type of IO link

For action =clearalarm

For valid values of this action, refer to the following argument-value table.

Table: clear alarm - Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255,0	Yes	To select the panel door or panel. 0= to perform the action on panel, 1-255= to perform the action on the desired panel door
format	Text, xml	No	Specifies the format in which the response is expected

For action=getcount

For valid values of this action, refer to the following argument-value table.

Table: Get Credential Count Command - Parameters

Argument	Valid Values	Mandatory	Description
user-id	1 to max. User ID in the door (2 bytes)	Yes	Defines the numeric ID of the user whose data is to be fetched.
card-count		No	To get the number of cards enrolled.
finger-count		No	To get the count pass the parameter as argument
palm-count		No	To get the count pass the parameter as argument
format	text,xml	No	To specify the format in which the response is expected.

- *If no parameter is requested then all the count values will be returned by default (of supported credential types e.g. for PVR door, only card and palm template count will be returned).*
- *Palm template count and finger template counts depend on the device type i.e. Palm template count is only applicable for PVR doors and FP template counts are applicable for other devices. The specified credential should be applicable for the device on which the command is sent.*

For action=acknowledgealarm

For valid values of this action, refer to the following argument-value table.

Table: acknowledge alarm - Parameters

Argument	Valid Values	Mandatory	Description
pdid	1 to 255,0	Yes	To select the panel door or panel. 0= to perform the action on panel, 1-255= to perform the action on the desired panel door

For action=deletecredential

For valid values of this action, refer to the following argument-value table.

Table: Deleting Credentials for All Users - Parameters

Argument	Valid Values	Mandatory	Description
pdid	1-255	Yes	To define the panel door ID.

Table: Deleting Credentials for All Users - Parameters

Argument	Valid Values	Mandatory	Description
type	0 = All 1 = Finger 2 = Palm	Yes	To specify the type of credential to be deleted.

For action= resetaccesspolicy

Syntax= http://<deviceIP:deviceport>/device.cgi/command?action=resetaccesspolicy

For valid values of this action, refer to the following argument-value table.

Table: Reset access policy - Parameters

Argument	Valid Values	Mandatory	Description
user-id	15 Character ASCII	Yes	To select the user for which access route access policy needs reset
access-route	0= unchecked 1= checked	Yes	To enable/disable access route access policy which is to be Reset

Example

Following are some sample cases for your reference:

1. **To get the current rollover count and sequence number of events in the device.**

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/command?action=geteventcount&format=xml
```

Sample Response

```
HTTP Code: 200 OK
Content-Type: <xml>
Body:
<COSEC_API>
<Roll-over-count>1</roll-over-count>
<seq-number>1</seq-number>
</COSEC_API >
```

IMEI Registration Request

To register the IMEI number of the user on COSEC Standalone Panel200, the IMEI Registration request is sent from the COSEC APTA.

Description: To send IMEI Registration Request from COSEC APTA installed in mobile device.

This API will be applicable with Third Party Software.

Actions: set

Syntax: http://<deviceIP:deviceport>/device.cgi/imei-register?<argument>=<value>[&<argument>=<value>…...]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: IMEI Request

Argument	Valid Values	Mandatory	Description
user-id	15 characters ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz 1234567890 and white space allowed	Yes	The user ID on which the specified operation is to be done.The user id & password appended in the API for access of the Url will be: User id: admin Password: password set on device
imei	40 characters ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz 1234567890 allowed	Yes	The imei through which the panel will authenticate the user's mobile device.
device-id	Upto 8 digits	No	Random number generated by Panel for auto sign-in feature
format	xml	-	-

Table: IMEI Request API Response

Response	Response Code	Rights
User identified & IMEI received matches with IMEI stored against the identified user in the panel	1	1,2 *value 1 or 2 will sent depend upon the rights assigned against the user in the panel database. 1= authorized to gain access through API 2=not authorized to gain access through API
User identified & registered the received IMEI against the identified user. This is case will be created when: 1.no IMEI was present against the user and Auto register flag=1 2.IMEI was already present against the user and auto register flag=1. (here the new IMEI received will be replaced with old IMEI) 3. The Device ID stored in IMEI field against the user matches with the Device ID received in Request API (here the Device ID will be replaced by IMEI umber once it matches)	1	
When the received IMEI is stored against any other user. API should fail	2	0 * Whenever code =2,3,4,5 mobile device should ignore rights parameter.
If this request is received but the IMEI wait timer [1 minute timer] is not active for the user.	3	
When Panel Door is offline	4	
When the received user id does not exists in panel database. API should fail	5	
When access through mobile feature is not enabled API should fail.	6	
When the Device ID/IMEI both received	7	

*Note:
when code =1 ; it indicates the IMEI registration is successful
when code =2,3,4,5,6,7; it indicates the IMEI registration is not successful

Request for Security Code

To establish secured connection with the Panel200 through COSEC APTA after registering IMEI number of the user.

Description: To get the security code for the user.

This API will be applicable with Third Party Software.

Actions: get

Syntax: http://<deviceIP:deviceport>/device.cgi/security-code?<argument>=<value>[&<argument>=<value>…]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Security Code Request

Argument	Valid Values	Mandatory	Description
user-id	15 characters ABCDEFGHIJKLMNOPQ RSTUVWXYZabcdefghijkl mnopqrstuvwxyz 1234567890 and white space allowed	Yes	The user ID on which the specified operation is to be done.The user id & password appended in the API for access of the Url will be: User id: admin Password: password set on device
format	xml	-	-

The response should have following parameters:

Argument	Valid Values	Description
security-code	2 bytes(2 char) ABCDEFGHIJKLMNOPQ RSTUVWXYZabcdefghijkl mnopqrstuvwxyz 1234567890	As per configured in the panel

Table:Security Code Request API Response

Response	Response Code	Reason
When received user id doesn't exist in the panel	13	user id not exists
When mobile access feature is not enabled in the panel	24	Feature not enabled in config
When the door is offline	30	Door Offline

To get Random Key on Code

Description: To get a random key for particular user for secured access.

Actions: get

Syntax: http://<deviceIP:deviceport>/device.cgi /random-key?<argument>=<value>[&<argument>=<value>….]

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Access Request-qr

Argument	Valid Values	Mandatory	Description
user-id	15 characters ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz 1234567890 and white space allowed	Yes	The user ID (alphanumeric user ID) on which the specified operation is to be done.
pdid	Numeric (1-255)	Yes	The door ID on which the specified operation is to be done.
format	xml	-	-

Table: Access Request-qr Response

Field	Valid Values
random-key	random number upto 8 digit

Note:

- Whenever the received user id doesn't exist in the panel, response code13(user id not exists) will be sent.
- Whenever the access through API is not enabled in the panel, response code24 (Feature not enabled in config) will be sent.
- Whenever the feature mobile access is not enabled for the received user id in the panel, response 24 (Feature not enabled in config) will be sent
- Whenever the door is offline, response code30 (Door Offline) will be sent.

Access Request on QR Scanning

To access COSEC device without biometric credentials is a new feature. You can access COSEC Panel doors using COSEC APTA by QR scanning.

Description: To access a device from COSEC APTA installed in mobile device.

This API will be applicable with COSEC Server and Third Party Software also.

Actions: set

Syntax: `http://<deviceIP:deviceport>/device.cgi/access-request-qr?<argument>=<value>[&<argument>=<value>….]`

Parameters: All arguments for this query and their corresponding valid values are listed below:

Table: Group:access-request-qr

Argument	Valid Values	Mandatory	Description
security-key	Encrypted random number	Yes	A security Key encrypted by mobile is sent to authenticate the API.
imei	40 characters ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz 1234567890 allowed	Yes	The imei through which the panel will authenticate the user's mobile device.
user-id	15 characters ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz 1234567890 and white space allowed	Yes	The user ID on which the specified operation is to be done.The user id & password appended in the API for access of the Url will be: User id: admin Password: password set on device
mode	0,1	Yes	To define the mode of the request 0=ENTRY 1=EXIT
pdid	1 to 255	Yes	To specify the panel door id
format	xml	-	-

Table: QR Request API Response

Response	Response Code	Reason	Applicable Doors
User verified-Access Allowed or When visitor is allowed	1	0	All doors
APB-soft(when exit was not done by user)	1	0	All doors

Table: QR Request API Response

Response	Response Code	Reason	Applicable Doors
APB-soft(when entry was not done by user)	1	0	All doors
User Allowed-Mode =Entry(When Schedule door mode is Active)	1	0	Linux based direct doors
APB-soft(when exit was not done by user)	1	0	Linux based direct doors
User Allowed-Mode =Exit (When Schedule door mode is Active)	1	0	Linux based direct doors
APB-soft(when entry was not done by user)	1	0	Linux based direct doors
User Access Denied	2	0	All doors
2-person rule violated/ Invalid input for 2-person rule	2	1	All doors
Access deniedAPB(HARD) denied because exit was not made by user	2	2	All doors
Access deniedAPB(HARD) denied because entry was not made by user	2	3	All doors
User blocked	2	4	All doors
When no door is present in the panel against the pdid received in the API	2	5	Panel doors
When restrict access flag is enabled to the user on the device.	2	6	Panel doors
Door Locked	2	7	All doors
Access Denied-First-in user rule violation	2	8	All doors
Occupancy rule violation-when maximum occupancy rule is violated	2	9	All doors
Occupancy rule violation- when occupancy control for zone has been violated	2	10	Panel doors
When minimum occupancy required rule is violated and exit has been denied to the user.	2	11	Direct doors
User validity expired	2	13	All doors
System error-Access denied Generate this when user is allowed but an error occurred while writing the events in the flash.	2	14	All doors
Denied-visitor escort rule violation(2nd person not identified)/ Visitor escort violation cases/ Waiting for visitor escort state	2	15	Panel doors
when requested user id is not assigned or not found on device	2	16	All doors

Table: QR Request API Response

Response	Response Code	Reason	Applicable Doors
Entry Restricted cases	2	17	All doors
When the User is not active from the server	2	18	All doors
1) If the "Access through API" has not been set & API from Mobile has been received 2) If API access mode is API + biometric and biometric reader is not configured with API's mode i.e entry/ exit	2	19	All doors
1) When the security key received from the api don't match with the API Security key stored in the device. 2) Whenever access request through API feature is not enabled for the received User. 3) When the user is not in the route assigned to the door 4) Whenever the IMEI received in the API doesn’t match with the IMEI stored against the identified user.	2	20	All doors
Waiting for second user -when 2-person is active and first person has been verified	3	1	All doors
Waiting for second user -Occupancy 1st user identified	3	0	Panel doors & Linux based direct doors
If device is serving a command like enrollment, firmware upgrade or deletion of credentials and API is received	4	0	All doors
When the access mode= API+Biometric & user has identified and devices waits for Biometric credentials	5	0	All doors
When passed pdid, i.e. panel door is offline	6	0	Panel Doors

In above table:

All doors include:

- "All direct doors and all panel doors

Linux based direct doors include:

- "NGT, Vega, Wireless/V3, PVR

Direct doors include:

- "NGT, Vega, Arc, Wireless/V3, PVR

Panel doors include:

- "Vega, Arc, Wireless/V3, PVR, V1, V2, PATH

Access Policies

Action: set, get, delete

Syntax: http:// <deviceIP:deviceport>/device.cgi/access-route?<argument>=<value>[&<argument>=<value>….]

Table: Access Policies- Action= set

Arguments	Valid Values	Mandatory	Description
route-id	1 to 255	Yes	To set or retrieve alphanumeric route ID.
active	0= Inactive 1= active	No	To enable or disable Access Route Policy. By default it is disabled/deactivated
route name	15 Char ASCII Code	Yes	To define the access route name
restriction	0= soft 1= hard	No	To set or retrieve the restriction for un-sequenced route. By default it is 'Soft'
sequenced	0= unsequenced 1= sequenced	No	To enable or disable sequencing in access route. By default it is disabled
reset-lowest-level	0= Inactive 1= active	No	To reset access route at lowest level. By default it is disabled
check-last-exit-level	0= Inactive 1= active	No	To Enable/Disable checking on last Exit Level.by default it is disabled. This is applicable only when route is sequenced
last-exit-level-restrct	0= soft 1= hard	No	To set or retrieve the last exit level restriction. By default it is 'Soft'. This is applicable only when route is sequenced
door-info	Id (1-255),level(1-32)	Yes	To define the door id and level as an array of max. 255 doors
format	Text, XML	No	To specify the format.

For action = get
For valid values of this action, refer to the following argument-value table.

Table: Access Policies- Action= get

Argument	Valid Values	Mandatory	Description
route-id	1 to 255	Yes	To define the Access Route ID
format	Text, XML	No	To specify the format.

For action= delete
For valid values of this action, refer to the following argument-value table.

Table: Access Policies- Action= delete

Argument	Valid Values	Mandatory	Description
route-id	1 to 255	Yes	To select the route ID on which the specified operation is to be done.
format	Text, XML	No	Specifies the format in which the response is expected

Advanced Parameter for Access Route

Advances parameter for access route will be applicable only when the route is sequenced.

Action: set, get

Syntax: http:// <deviceIP:deviceport>/device.cgi/ad-access-route?<argument>=<value>[&<argument>=<value>….]

For valid values of this action, refer to the following argument-value table.

Table: Advance Parameter for Access Route - Action: set

Argument	Valid Values	Mandatory	Description
route-id	1 to 255	Yes	To set or retrieve alphanumeric route ID.
door-id	1 to 255(for Panel200), 1 to 75 (for Panel/Panellite v1)	Yes	To define the door id of the device
entry-enable	0= inactive 1= active	No	To enable/disable sequenced access route at entry. By default it is disabled
entry-route-timer	1-65535 seconds	No	To set or retrieve entry route timer. By default it is 600 seconds
entry-level-restrct	0= soft 1= hard	No	To set or retrieve the entry level restriction. By default it is 'Soft'
entry-perform-action	0= on user punch 1= on timer elapse	No	To set or retrieve the Perform Action on Entry. By default it is "On User Punch"
exit-enable	0= inactive 1= active	No	To enable/disable sequenced access route at exit. By default it is disabled
exit-route-timer	1-65535 seconds	No	To set or retrieve exit route timer. By default it is 600 seconds
exit-level-restrct	0= soft 1= hard	No	To set or retrieve the exit level restriction. By default it is 'Soft'
exit-perform-action	0= on user punch 1= on timer elapse	No	To set or retrieve the Perform Action on Exit. By default it is "On User Punch"
format	Text, XML	No	To specify the format

Action: get

For valid values of this action, refer to the following argument-value table.

Table: Advance Parameter for Access Route- Action: get

Argument	Valid Values	Mandatory	Description
route-id	1 to 255	Yes	To define the Access Route ID.
format	Text, XML	No	To specify the format

API Response Codes

These numerical codes will be returned with an API response. These response codes shall indicate the result of a particular request made by the client. For e.g. the response code ‘0’ will indicate that the requested action was performed successfully. Refer to the given table for a list of response codes and their meanings.

Table: API Response Codes

Response Code	Description	Test Condition
0	Successful	-
1	Failed - Invalid Login Credentials	On every Authentication/Verification while logging In
2	Date and time – manual set failed	If unable to set the RTC for date and time API
3	Invalid Date/Time	In User API, if validity-date or date of birth is set wrong. If the starting time and end time of a shift is configured as same.
4	Maximum users are already configured.	On every set command for user API
7	Card 1 and card 2 are identical	On every set command for user API and set credential API
8	Card ID exists	On every set command for user API and set credential API, Set Special Function API
9	Finger print template/ Palm template already exists	Set credential API
10	No Record Found	Event sequence number and roll over count not found, image not found, file not found. No panel doors are added in panel
11	Template size/ format mismatch	If the expected template size is not as per the required size, format or any checksum error etc. in Set credential API
12	FP Memory full	In Set credential API, if the max FP template is set in the module.
13	User id/reference id not found	In enroll user command if user id is not available in the device and in User Configuration API, to update a user if provided reference user ID doesn’t belong to that user verified with alphanumeric user ID. Also, to be used for get security code API
14	Credential limit reached	In enroll user command, if max no. of credentials is already enrolled.
15	Reader mismatch/ Reader not configured	The enroll request is for smart card and the device has proximity reader or if enroll request has palm template but door has finger reader and similar cases.
16	Device Busy	All cases of enrollment when the device is unable to process a request as it is in a different menu state
17	Internal process error	Internal error like configuration, firmware or event or calibration failure occur
18	PIN already exists	Set User API: PIN is already assigned to another user
19	Biometric credential not found	In enroll user smart card, write FP is enabled, but FP is not enrolled, Get FP/Palm template command is sent but template is not present.
20	Memory Card Not Found	In case memory card is not connected, and a command related to getting an image (user photo) is sent.
21	Reference User ID exists	When an already existing User ID is entered against a user having unique User ID.

Table: API Response Codes

Response Code	Description	Test Condition
22	Wrong Selection	For enrolling user, if writing FP template on smart card is enabled, but no fingerprint is enrolled. When palm/finger/card count exceeds the maximum number of available places.
23	Palm template mode mismatch	In Set Credentials API, when palm template with particular mode does not match with the selected mode.
24	Feature not enabled in the configuration	In configuration, if a particular parameter is not enabled and is required for the process.
25	Message already exists for same user for same date	In Device Display API, if message is configured for same date and same user that is already configured.
26	Error in Import Data	Import API if any of the import rules are violated
27	Maximum doors are already configured	On every set command for Panel door API
28	Panel door already exists	Whenever the IP/RS 485 address/MAC address conflicts with an existing panel door This will be also applicable whenever the user tries to change the door type of an already existing door.
29	Invalid value	Whenever the value set for any parameter is not as per its field type defined. Whenever IP/MAC entered is not valid. Whenever username/door name is set as blank. Whenever the Pdid received in the API doesnot exists in the panel.
30	Door Offline	When Panel door is offline
31	Photo not uploaded	Whenever user's Photo is not present in the panel and Get User Photo has been fired

Error Responses

These are some possible error response types obtained from incorrect API requests.

- **Argument is mentioned in request but valid value is not assigned.**

Sample Response

```
HTTP code: <code>
Content-type: <type>
Body:
Request failed: Incomplete command “<argument>=”
```

- **Invalid value is assigned to argument in request.**

Sample Response

```
HTTP code: <code>
Content-type: <type>
Body:
Request failed: Invalid command “<argument>=<invalid value>”
```

- **Syntax of request is incorrect or any unexpected arguments are received.**

Sample Response

```
HTTP code: <code>
Content-type: <type>
Body:
Request failed: Invalid syntax “<entire request>”
```

- **Mandatory fields are not mentioned in request.**

Sample Response

```
HTTP code: <code>
Content-type: <type>
Body:
Request failed: Incomplete command “<entire request>”
```

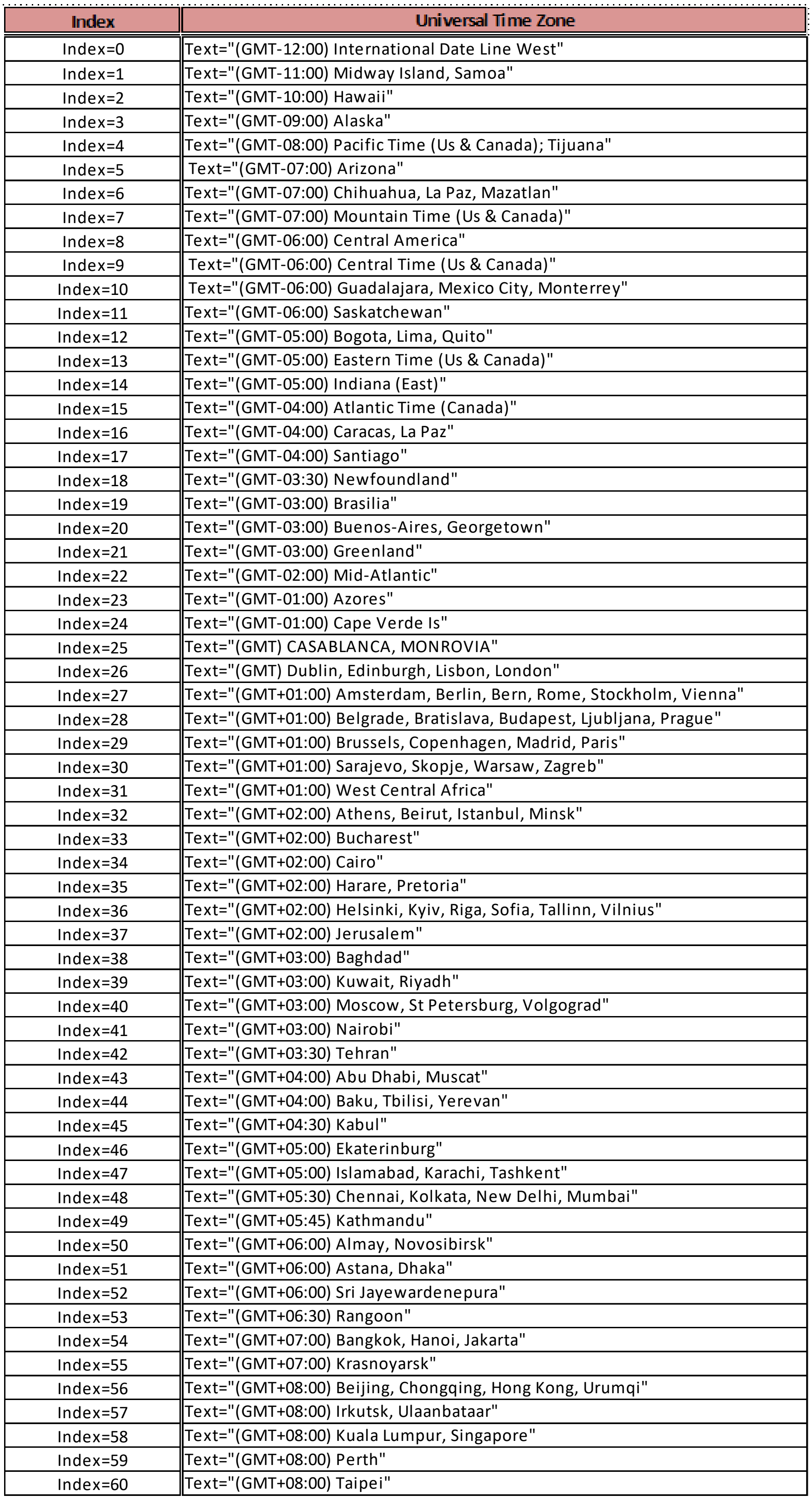
- **Syntax of request is valid but no data found.**

Sample Response

```
HTTP code: <code>
Content-type: <type>
Body:
Request failed: No record found “<argument>=<value>”
```

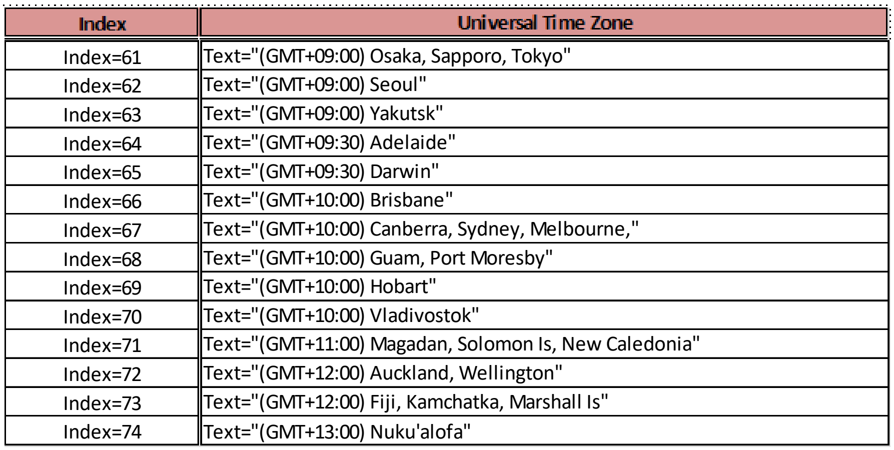
Appendix

Table: Universal Time Zone Reference



Index	Universal Time Zone
Index=0	Text="(GMT-12:00) International Date Line West"
Index=1	Text="(GMT-11:00) Midway Island, Samoa"
Index=2	Text="(GMT-10:00) Hawaii"
Index=3	Text="(GMT-09:00) Alaska"
Index=4	Text="(GMT-08:00) Pacific Time (Us & Canada); Tijuana"
Index=5	Text="(GMT-07:00) Arizona"
Index=6	Text="(GMT-07:00) Chihuahua, La Paz, Mazatlan"
Index=7	Text="(GMT-07:00) Mountain Time (Us & Canada)"
Index=8	Text="(GMT-06:00) Central America"
Index=9	Text="(GMT-06:00) Central Time (Us & Canada)"
Index=10	Text="(GMT-06:00) Guadalajara, Mexico City, Monterrey"
Index=11	Text="(GMT-06:00) Saskatchewan"
Index=12	Text="(GMT-05:00) Bogota, Lima, Quito"
Index=13	Text="(GMT-05:00) Eastern Time (Us & Canada)"
Index=14	Text="(GMT-05:00) Indiana (East)"
Index=15	Text="(GMT-04:00) Atlantic Time (Canada)"
Index=16	Text="(GMT-04:00) Caracas, La Paz"
Index=17	Text="(GMT-04:00) Santiago"
Index=18	Text="(GMT-03:30) Newfoundland"
Index=19	Text="(GMT-03:00) Brasilia"
Index=20	Text="(GMT-03:00) Buenos-Aires, Georgetown"
Index=21	Text="(GMT-03:00) Greenland"
Index=22	Text="(GMT-02:00) Mid-Atlantic"
Index=23	Text="(GMT-01:00) Azores"
Index=24	Text="(GMT-01:00) Cape Verde Is"
Index=25	Text="(GMT) CASABLANCA, MONROVIA"
Index=26	Text="(GMT) Dublin, Edinburgh, Lisbon, London"
Index=27	Text="(GMT+01:00) Amsterdam, Berlin, Bern, Rome, Stockholm, Vienna"
Index=28	Text="(GMT+01:00) Belgrade, Bratislava, Budapest, Ljubljana, Prague"
Index=29	Text="(GMT+01:00) Brussels, Copenhagen, Madrid, Paris"
Index=30	Text="(GMT+01:00) Sarajevo, Skopje, Warsaw, Zagreb"
Index=31	Text="(GMT+01:00) West Central Africa"
Index=32	Text="(GMT+02:00) Athens, Beirut, Istanbul, Minsk"
Index=33	Text="(GMT+02:00) Bucharest"
Index=34	Text="(GMT+02:00) Cairo"
Index=35	Text="(GMT+02:00) Harare, Pretoria"
Index=36	Text="(GMT+02:00) Helsinki, Kyiv, Riga, Sofia, Tallinn, Vilnius"
Index=37	Text="(GMT+02:00) Jerusalem"
Index=38	Text="(GMT+03:00) Baghdad"
Index=39	Text="(GMT+03:00) Kuwait, Riyadh"
Index=40	Text="(GMT+03:00) Moscow, St Petersburg, Volgograd"
Index=41	Text="(GMT+03:00) Nairobi"
Index=42	Text="(GMT+03:30) Tehran"
Index=43	Text="(GMT+04:00) Abu Dhabi, Muscat"
Index=44	Text="(GMT+04:00) Baku, Tbilisi, Yerevan"
Index=45	Text="(GMT+04:30) Kabul"
Index=46	Text="(GMT+05:00) Ekaterinburg"
Index=47	Text="(GMT+05:00) Islamabad, Karachi, Tashkent"
Index=48	Text="(GMT+05:30) Chennai, Kolkata, New Delhi, Mumbai"
Index=49	Text="(GMT+05:45) Kathmandu"
Index=50	Text="(GMT+06:00) Almay, Novosibirsk"
Index=51	Text="(GMT+06:00) Astana, Dhaka"
Index=52	Text="(GMT+06:00) Sri Jayewardenepura"
Index=53	Text="(GMT+06:30) Rangoon"
Index=54	Text="(GMT+07:00) Bangkok, Hanoi, Jakarta"
Index=55	Text="(GMT+07:00) Krasnoyarsk"
Index=56	Text="(GMT+08:00) Beijing, Chongqing, Hong Kong, Urumqi"
Index=57	Text="(GMT+08:00) Irkutsk, Ulaanbataar"
Index=58	Text="(GMT+08:00) Kuala Lumpur, Singapore"
Index=59	Text="(GMT+08:00) Perth"
Index=60	Text="(GMT+08:00) Taipei"

Table: Universal Time Zone Reference



Index	Universal Time Zone
Index=61	Text="(GMT+09:00) Osaka, Sapporo, Tokyo"
Index=62	Text="(GMT+09:00) Seoul"
Index=63	Text="(GMT+09:00) Yakutsk"
Index=64	Text="(GMT+09:30) Adelaide"
Index=65	Text="(GMT+09:30) Darwin"
Index=66	Text="(GMT+10:00) Brisbane"
Index=67	Text="(GMT+10:00) Canberra, Sydney, Melbourne,"
Index=68	Text="(GMT+10:00) Guam, Port Moresby"
Index=69	Text="(GMT+10:00) Hobart"
Index=70	Text="(GMT+10:00) Vladivostok"
Index=71	Text="(GMT+11:00) Magadan, Solomon Is, New Caledonia"
Index=72	Text="(GMT+12:00) Auckland, Wellington"
Index=73	Text="(GMT+12:00) Fiji, Kamchatka, Marshall Is"
Index=74	Text="(GMT+13:00) Nuku'alofa"

Event Configuration Reference

Table: List of Events

Event ID	Event Description
101	User Allowed
102	User Allowed – with Duress
104	User Allowed - Dead-man Zone
105	User Allowed – Door Not open
151	User Denied – User Invalid
154	User Denied – Time Out
157	User Denied – Disabled User
158	User Denied – Blocked User
160	User Denied – DND Enabled
161	User denied – Control zone
162	User Denied – Door Lock
163	User Denied – Invalid Access Group
164	User Denied – Validity date expired
201	Door Status changed
202	Dead-man timer changed
203	DND status changed
204	Aux input status changed
205	Aux output status changed
206	Door sense input status
207	Door Controller Communication status
208	Door Open/ Close
209	Lock relay status changed

Table: List of Events

Event ID	Event Description
301	Dead-man timer expired Alarm– User IN
302	Duress detection
303	Panic Alarm
304	FP Memory Full – Alarm
305	Door Held open too long
306	Door Abnormal
307	Door force open
308	Door Controller Offline
309	Door Controller -Fault
310	Tamper Alarm
311	Master Controller Mains fail Alarm
312	Master Controller Battery fail
313	Master Alarm – MC Alarm input
314	RTC
315	Event Buffer Full
317	Intercom - panic
321	Access Denied Aalrm
322	Multiple Unauthorized Access Alarm
323	Custom Alarm 1
324	Custom Alarm 2
325	Custom Alarm 3
326	User Unidentified
329	Raise Alarm
351	Alarm acknowledged
352	Alarm cleared
353	Alarm Re-issued
401	User Block/Restore
402	Login to ACS
403	Message transaction confirmation to ACMS
405	Enrolment
406	Master Alarm sense input status
407	Master Aux Output status
409	Credentials Deleted
451	Configuration Change
452	Roll over of events
453	Master Controller Power ON

Table: List of Events

Event ID	Event Description
454	Configuration Defaulted
456	Backup and Update
457	Default System
458	Sensor Calibration
459	User Denied – invalid card
460	User PIN Change

Table: Size of Event Fields

Door	Field 1	Field 2	Field 3	Field 4	Field 5	Event Log Capacity
Panel200	4 bytes	2 bytes	2 bytes	4 bytes	4 bytes	5,00,000 events

Table: Event Structure

<table>
<tr><th colspan="9">Reply Event</th></tr>
<tr><td>RPL_EVT</td><td>TYPE</td><td>MID</td><td>ROLL_OVER_COUNT</td><td>EVT_SEQ_NUM</td><td>SOURCE_ID</td><td>DATE</td><td>TIME</td><td>EVT_ID</td></tr>
</table>

<table>
<tr><th colspan="5">Reply Event</th></tr>
<tr><td>FIELD1</td><td>FIELD2</td><td>FIELD3</td><td>FIELD4</td><td>FIELD5</td></tr>
</table>

The above structure will be used to send events to COSEC Server.

<table>
<tr><th colspan="9">Send Event</th></tr>
<tr><td>SND_EVT</td><td>EVT_SEQ_NUM</td><td>SOURCE_ID</td><td>DATE</td><td>TIME</td><td>EVT_ID</td><td>FIELD1</td><td>FIELD2</td><td>FIELD3</td></tr>
</table>

This structure will be used to send events to CCC Server or when TCP Events API is fired.

<table>
<tr><th colspan="3">Send Event</th></tr>
<tr><td>FIELD4</td><td>FIELD5</td><td>MAC ADDRESS</td></tr>
</table>

Table: User Events

<table>
<tr><th colspan="5">Event Details</th><th colspan="3">Device</th></tr>
<tr><th>Event ID</th><th>(Field 1)
Reference ID</th><th>(Field 2)
Special Code</th><th>(Field 3)
Entry/Exit</th><th>(Field 4) and (Field 5)</th><th colspan="3">Panel200/Panel/Panel-Lite/Standalone Panellite</th></tr>
<tr><td colspan="6">User Allowed Events</td><td></td><td></td></tr>
</table>

Table: User Events

Event Details					Device
Event ID	**(Field 1)** **Reference ID**	**(Field 2)** **Special Code**	**(Field 3)** **Entry/Exit**	**(Field 4) and (Field 5)**	**Panel200/Panel/Panel-Lite/Standalone Panellite**
101	Xxxx (Reference ID=0 for REX input)	Special Function code (Reference ID=0 for REX input)	Detail (Reference ID=0 for REX input)	(Reference ID=0 for REX input)	✓
102	Xxxx	Special Function code	Detail		✗
103	Xxxx	Special Function code	Detail		✓
104	Xxxx	Special Function code	Detail		✓
105	Xxxx	Special Function code	Detail		✓
106	Xxxx	Special Function code	Detail		✓
107	Xxxx	Special Function code	Detail		✓
108	Xxxx	Special Function code	Detail		✓
109	Xxxx	Special Function code	Detail		✓
110	Xxxx	Special Function code	Detail		✓
111	First four bytes of extension number	last two bytes of extension number	Detail		✗
112	Xxxx	0 = Door unlock 1 = Door lock	Detail		✓
User Denied Events					
151	Xxxx	Special Function code	Detail		✓
152	Xxxx		Detail		✓
153	Xxxx		Detail		✓
154	Xxxx		Detail		✓
155	Xxxx		Detail		✓
156	Xxxx		Detail		✓
157	Xxxx		Detail		✓
158	Xxxx		Detail		✓

Table: User Events

Event Details					Device
Event ID	(Field 1) Reference ID	(Field 2) Special Code	(Field 3) Entry/Exit	(Field 4) and (Field 5)	Panel200/Panel/Panel-Lite/Standalone Panellite
159	Xxxx		Detail		✓
160	Xxxx		Detail		✓
161	Xxxx		Detail		✓
162	Xxxx		Detail		✓
163	Xxxx		Detail		✓
164	Xxxx		Detail		✓
165	Xxxx	0=Door Not in Sequence 1=Door Not in Route	Detail		✓
		2=Door Not in Sequence for Smart card based Route 3=Door Not in Smart card based Route 4=Credential Invalid for Smart card based Route Access			✓
166	Xxxx	0=Outside working hours 1=Holiday	Detail		✓
		2=Week off 3=Field Break 4=Rest Day			✓
167	Xxxx		Detail		✓
171	First four bytes of extension number	last two bytes of extension number	Detail		✕
172	Xxxx		Detail		✕

Table: Special Function Codes Reference

S.No.	Special Function Name	Special Function Code	Applicable for Allowed Events	Applicable for Denied Events
1	Official Work-IN Marking in T&A	1	✓	✕

Table: Special Function Codes Reference

S.No.	Special Function Name	Special Function Code	Applicable for Allowed Events	Applicable for Denied Events
2	Official Work-OUT Marking in T&A	2	✓	✗
3	Short Leave-IN Marking in T&A	3	✓	✗
4	Short Leave-OUT Marking in T&A	4	✓	✗
5	Clock - IN Marking in T&A	5	✓	✗
6	Clock - OUT Marking in T&A	6	✓	✗
7	Post Lunch-IN Marking in T&A	7	✓	✗
8	Pre Lunch -OUT Marking in T&A	8	✓	✗
9	Over time – IN Marking in T&A	9	✓	✗
10	Over time – OUT Marking in T&A	10	✓	✗
11	Late –IN Allowed Marking in T&A	11	✓	✗
12	Early - OUT Allowed Marking in T&A	12	✓	✗
13	Access in Degrade Mode Marking	99	✓	✓
14	Smart Identification	98	✗	✓
15	e-Canteen	97	✗	✓

Table: Field 3 Detail (User Events) Reference

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RFU						API	Group	Palm	Finger	Card	PIN	RFU		RFU	Entry/ Exit

Table: Information of Bit 0 and Bit 1

Credential	Bit 1	Bit 0	Value	
Entry	0	0	0	✓
Exit	0	1	1	✓

Table: Information of Bit 4 and Bit 9

Credential	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Value
PIN	0	0	0	0	0	1	1
Card	0	0	0	0	1	0	2
Card + PIN	0	0	0	0	1	1	3
Finger	0	0	0	1	0	0	4
Finger + PIN	0	0	0	1	0	1	5
Finger + Card	0	0	0	1	1	0	6
Finger + Card + PIN	0	0	0	1	1	1	7
Palm	0	0	1	0	0	0	8
PIN + Palm	0	0	1	0	0	1	9
Card + Palm	0	0	1	0	1	0	10
PIN + Card + Palm	0	0	1	0	1	1	11
Group + Palm	0	1	1	0	0	0	24
API	1	0	0	0	0	0	32
API + Finger	1	0	0	1	0	0	36
API + Palm	1	0	1	0	0	0	40
API + PIN	1	0	0	0	0	1	33
Group + Finger	0	1	0	1	0	0	20

Table: Alarm Events

Event Details					Applicable Devices							
Event ID	(Field 1)	(Field 2)	(Field 3)	(Field 4) and (Field 5)	Direct Door V2	Path Controller	Wireless Door	NGT Door	PVR Door	Vega Controller	Door FMX	Panel200/ Panel/ Panel Lite
301	Reference ID Xxxx	1 = Critical	Alarm Sequence Number		✓	✓	✓	✓	✓	✓	✗	✓
302	Reference ID Xxxx	1 = Critical	Same as above		✗	✗	✗	✗	✗	✗	✗	✓

Table: Alarm Events

Event Details					Applicable Devices							
Event ID	**(Field 1)**	**(Field 2)**	**(Field 3)**	**(Field 4) and (Field 5)**	**Direct Door V2**	**Path Controller**	**Wireless Door**	**NGT Door**	**PVR Door**	**Vega Controller**	**Door FMX**	**Panel200/ Panel/ Panel Lite**
303	Reference ID Xxxx	1 = Critical	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
304	1= Internal 2= External	3 = Minor	Same as above		✓	✓	✓	✓	✓	✓	✗	✓
305		3 = Minor	Same as above		✓	✓	✓	✓	✓	✓	✗	✓
306		2 = Major	Same as above		✓	✓	✓	✓	✓	✓	✓	✓
307		1 = Critical	Same as above		✗	✗	✗	✗	✗	✗	✓	✓
308		2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
309		2 = Major	Same as above		✓	✓	✓	✓	✓	✓	✓	✓
310		1 = Critical	Same as above		✓	✓	✓	✓	✓	✓	✓	✓
311		2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
312		1 = Critical	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
313		1 = Critical	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
314	1= Power ON/OFF Detected (time not in sync) 2= low battery detected 3= RTC Not Detected	2 = Major 1 = Critical	Same as above		✓	✓	✓	✓	✓	✓	✓	✗
315		2 = Major 1 = Critical	Same as above		✗	✗	✗	✗	✗	✗	✗	✗
317	First four bytes of Extension number	Last two bytes of extension number	Same as above		✓	✓	✓	✓	✓	✓	✓	✗
318	Reference ID Xxxx	2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
319	Reference ID Xxxx	2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
320	Reference ID Xxxx	2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
321	Reference ID Xxxx	2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
322	Reference ID Xxxx	2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
323		2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
324		2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
325		2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
326		2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓

Table: Alarm Events

Event Details					Applicable Devices							
Event ID	**(Field 1)**	**(Field 2)**	**(Field 3)**	**(Field 4) and (Field 5)**	**Direct Door V2**	**Path Controller**	**Wireless Door**	**NGT Door**	**PVR Door**	**Vega Controller**	**Door FMX**	**Panel200/ Panel/ Panel Lite**
327	Reference ID Xxxx	2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
328	Reference ID Xxxx	2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
329	Reference ID Xxxx	2 = Major	Same as above		✗	✗	✗	✗	✗	✗	✗	✓
351		4 = SysInterlock 5 = User_Jeeves 6 = User_ACMS 9 = Auto	Same as above		✓	✓	✓	✓	✓	✓	✓	✓
352		4 = SysInterlock 5 = User_Jeeves 6 = User_ACMS 7= Special Function	Same as above		✓	✓	✓	✓	✓	✓	✓	✓
353			Same as above		✗	✗	✗	✗	✗	✗	✗	✓

Table: System Events

Event Details					Applicable Devices								
Event ID	**(Field 1)**	**(Field 2)**	**(Field 3)**	**(Field 4) and (Field 5)**	**Direct Door V2**	**Path Controller**	**Wireless Door**	**NGT Door**	**PVR Door**	**Vega Controller**	**Door FMX**	**Panel200/Panel/Panel Lite/Standalone Panel lite**	**IO Controller**
401	User ID: xxxx	0= Unused (Restore User) 2=Unauthorized access 4=Invalid PIN	1= Blocked 0= Restored		✓	✓	✓	✓	✓	✓	✓	✓	✗
402		5= SA 6= SE 7= Operator	1=Success 0=Fail		✗	✗	✓	✓	✓	✓	✓	✓	✓
403	Transaction ID: Xxxx		1=Success 0=Fail		✗	✗	✓	✓	✓	✓	✓	✓	✗
404	Guard Tour no. Xxxx + cycle no.		1=Success 0=Fail		✗	✗	✗	✗	✗	✗	✗	✗	✗
405	ID: Xxxx	8 = User Card 9 = User Finger 10 = Special Cards 14 = Palm 16 = Palm template with guide mode 17 = User Finger-Suprema ISO format	0= FP-1/Palm-1 1= Card1/FP-2/Palm-2 2= Card-2/ F-3/Palm-3 3 = Card-3/FP-4/Palm- 4 = Card-4/FP-5/Palm-5 5= FP-6/Palm-6 6= FP-7/Palm-7 7= FP-8/Palm-8 8= FP-9/Palm-9 9= FP-10/Palm-10		✓	✓	✓	✓	✓	✓	✓	✓	✗

Table: System Events

Event Details					Applicable Devices								
Event ID	**(Field 1)**	**(Field 2)**	**(Field 3)**	**(Field 4) and (Field 5)**	**Direct Door V2**	**Path Controller**	**Wireless Door**	**NGT Door**	**PVR Door**	**Vega Controller**	**Door FMX**	**Panel200 /Panel/ Panel Lite/ Standalone Panel lite**	**IO Controller**
406			1=Normal 2=Fault (Open) 3= Fault(Short) 4= Activated		✗	✗	✗	✗	✗	✗	✗	✓	✗
407			1=Normal 4=Activated		✗	✗	✗	✗	✗	✗	✗	✓	✗
408	I/O Link ID	11 = Pulse 12 = Interlock 13 = Latch 15 = Toggle (only with activated event)	1=Normal 4=Activated		✓	✓	✓	✓	✓	✓	✓	✓	✓
409	ID: Xxxx	8 = User Cards 9 = User Fingers 14 = Palm	5= Web Jeeves 6= ACMS 7= Special Function		✓	✓	✓	✓	✓	✓	✓	✓	✗
410	Time Triggered Function Id		1=Normal/ Deactivated 4=Activated		✓	✓	✓	✓	✓	✓	✓	✓	✓
411	Time Stamping Function ID		1=Normal/ Deactivated 4=Activated		✗	✗	✗	✓	✗	✗	✗	✗	✗
412	Guard tour no. +cycle no.	Door Controller sequence no.	1=Success 0=Fail		✗	✗	✗	✗	✗	✗	✗	✗	✗
413	event sequence number	roll over count	1=Success 0=Fail		✗	✗	✗	✓	✗	✗	✗	✗	✗
451	Configuration Table ID xxx	Index start	Index end		✗	✗	✓	✓	✓	✓	✓	✓	✓
452	Roll over number 00 to 99				✓	✓	✓	✓	✓	✓	✓	✓	✓
453					✓	✓	✓	✓	✓	✓	✓	✓	✓
454	Configuration Table ID xxx	Index start	Index end		✗	✗	✓	✓	✓	✓	✓	✓	✗

Table: System Events

Event Details					Applicable Devices								
Event ID	**(Field 1)**	**(Field 2)**	**(Field 3)**	**(Field 4) and (Field 5)**	**Direct Door V2**	**Path Controller**	**Wireless Door**	**NGT Door**	**PVR Door**	**Vega Controller**	**Door FMX**	**Panel200 /Panel/ Panel Lite/ Standalone Panel lite**	**IO Controller**
455	Time Period = xxx (configured value) (this field is used only with Overridden events) Resume events will have blank	1= 2-person Rule 2= Access Policies 3= Alarms 4= Anti-pass back 5= First In User 6= Mantrap 7= Occupancy control 8= Visitor Escort Rule	1= Overridden 0= Resumed		✗	✗	✗	✗	✗	✗	✗	✓	✗
456	1=Backup 2=Update	1=Configuration 2=Event 3=Firmware	0 = Fail 1 = Success 2 = CRC Check Fail		✗	✗	✓	✓	✓	✓	✓	✓	✗
457			6 = from ACMS 8 = from Hardware		✓	✓	✓	✓	✓	✓	✓	✓	✓
458		0 = Internal Finger Reader 1 = External Finger Reader	0 = Fail 1 = Success 2 = Not Supported		✓	✓	✓	✓	✗	✓	✓	✓	✗
459	Card ID	Card ID	Card ID		✓	✓	✓	✓	✓	✓	✓	✓	✗
460	Reference ID: xxxx				✗	✗	✓	✓	✓	✓	✓	✓	✗
461	Reference ID: xxxx	0 = Authorized 1 = Rejected	System User Index 1 to 10		✗	✗	✗	✗	✗	✗	✗	✓	✗

MATRIX
SECURITY SOLUTIONS

MATRIX COMSEC PVT. LTD.

Head Office:
394-GIDC, Makarpura, Vadodara - 390010, India.
Ph.:+91 265 2630555, +918511173344
E-mail: Tech.Support@MatrixComSec.com

Website: www.MatrixComSec.com