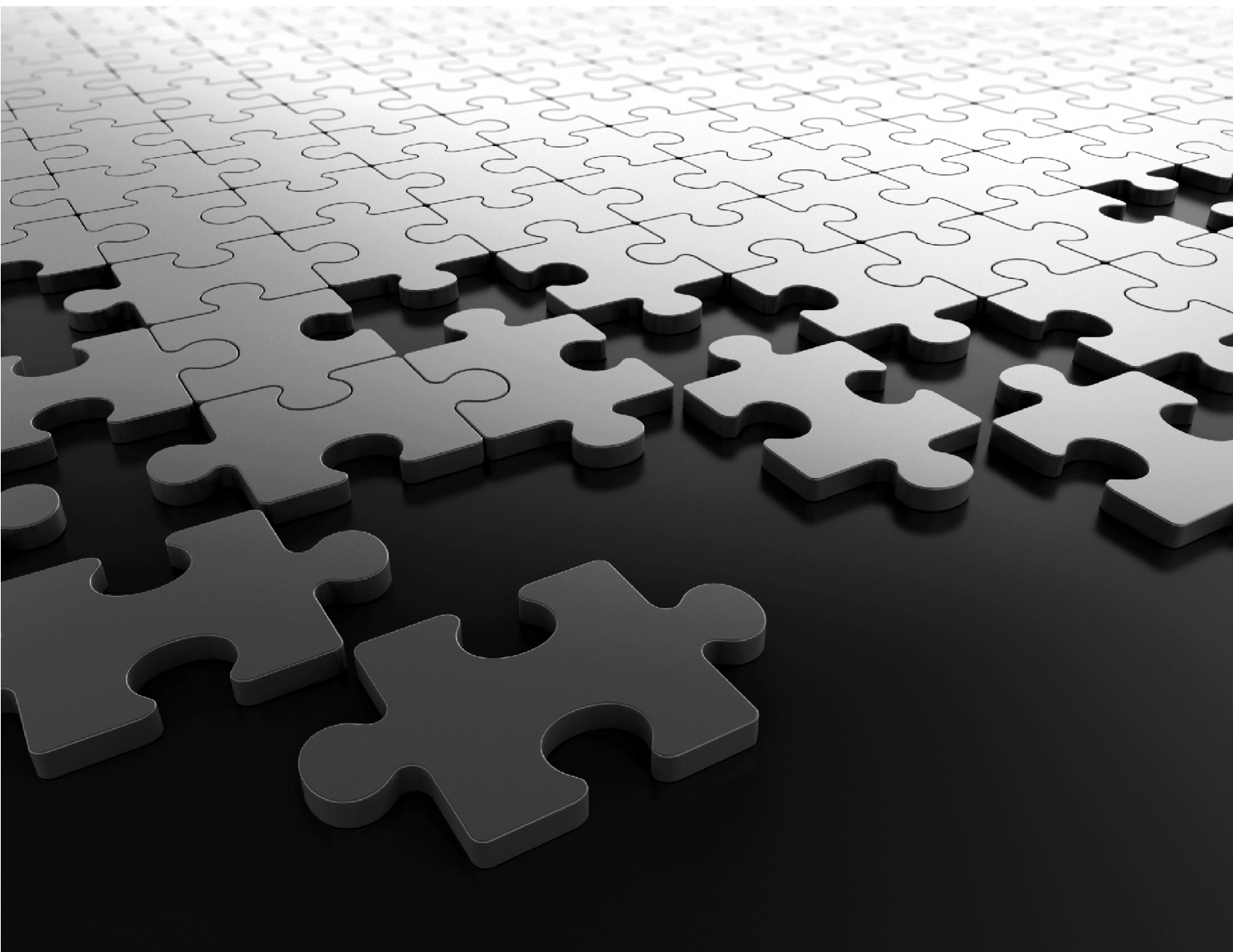


COSEC Devices API User Guide



COSEC Devices 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

All rights reserved. No part of this system manual may be copied or reproduced in any form or by any means without the prior written consent of Matrix Comsec.

Version 25

Release date: November 14, 2024

Contents

List of Tables	iii
About the Document	1
Document Conventions	1
Document Organization	2
Who Can Use This Document	2
API Overview	3
How It Works	3
Supported Devices	4
General Features	4
What the User Should Know	5
Prerequisite	5
Authentication	5
HTTP Request-Response	5
Communication Flow	7
Request Format	7
Response Format	8
URL Syntax	9
Common Actions	10
Additional Information	11
Supported APIs	12
Device Configuration	13
Basic Device Configuration	14
Function Key Configuration	18
Reader Configuration	19
Finger Reader Parameter Configuration	23
Palm Sensor Parameter Configuration	25
Enrollment Configuration	26
Access Settings Configuration	28
Alarm Configuration	29
Date and Time Configuration	30
Door Features Configuration	33
System Timers Configuration	36
Special Function Configuration	37
Internal/External Reader Configuration	39
Device Display Configuration	41
Device-LED & Buzzer Cadence Configuration	44
Multi-Language Support	46
To Download/Upload Multi-Language File	47
To Select the Mode of Wiegand Interface	49
Temperature Logging Configuration	50
Face Mask Compulsion	52
Smart Card Format	53
Smart Card Key Change	61
FR Settings	63
User Configuration	66
Setting/Retrieving User Configuration	67
Setting a User Photo	70
Deleting a User	72
Setting User Credentials	72
Retrieving User Credentials	74
Deleting User Credentials	78
Enrollment	79

<i>Enrolling a User</i>	80
<i>Enrolling Special Cards</i>	84
<i>Read/Write Card</i>	85
<i>Test Enrollment</i>	91
Events	92
<i>Retrieving Events</i>	93
<i>Retrieving Events in the TCP Socket</i>	96
Sending Commands to Device	98
Cafeteria Reset and Recharge	103
Verify Biometric and Open Door	105
Temperature Reading	106
Error Responses	107
API Response Codes	109
Appendix	112
<i>Event Configuration Reference</i>	113

List of Tables

Table: Device Configuration Parameters.....	14
Table: Function Key Configuration Parameters.....	18
Table: Reader Configuration Parameters (All doors except COSEC ARC Controller).....	19
Table: Reader Configuration Parameters (COSEC ARC Controllers only).....	21
Table: Finger Reader Parameter Configuration - Parameters.....	23
Table: Finger Reader Parameter Configuration - Parameters.....	25
Table: Enrollment Configuration Parameters.....	26
Table: Access Settings Configuration Parameters.....	28
Table: Alarm Configuration Parameters.....	29
Table: Date and Time Configuration Parameters.....	30
Table: Door Features Configuration Parameters.....	33
Table: System Timers Configuration Parameters.....	36
Table: Special Function Configuration Parameters.....	37
Table: Internal/External Reader Configuration Parameters.....	39
Table: Device Display Settings Parameters.....	41
Table: LED & Buzzer Cadence Configuration Parameters.....	44
Table: Multi-Language Support Parameters.....	46
Table: Download/Upload Multi-Language File - Parameters.....	47
Table: Wiegand Interface Mode Parameters.....	49
Table: Temperature Logging Configuration Parameters.....	50
Table: Smart Card Format Parameters.....	52
Table: Smart Card Format Parameters.....	53
Table: Smart Card Configuration Parameters - iClass 2K2.....	54
Table: Smart Card Configuration Parameters - iClass 16K2.....	55
Table: Smart Card Configuration Parameters - iClass 16K16.....	56
Table: Smart Card Configuration Parameters - Mifare 4K.....	57
Table: Smart Card Configuration Parameters - Mifare 1K.....	58
Table: Smart Card Key Change Parameters.....	61
Table: Smart Card Key Change Response.....	61
Table: FR Settings.....	63
Table: User Configuration Parameters.....	67
Table: Setting a User Photo - Parameters.....	70
Table: Delete User - Parameters.....	72
Table: Setting User Credentials - Parameters.....	72
Table: Retrieving User Credentials - Parameters.....	74
Table: Deleting User Credentials - Parameters.....	78
Table: Enrolling User - Parameters.....	80
Table: Enroll Special Cards - Parameters.....	84
Table: Read/Write Card Parameters.....	85
Table: Write Card Response.....	87
Table: Read Card Response.....	87
Table: Retrieving Events - Parameters.....	93
Table: Value Range for Event Sequence Numbers.....	93
Table: Retrieving Events in the TCP Socket - Parameters.....	96
Table: List of Commands to Device.....	98
Table: Get Credential Count Command - Parameters.....	99
Table: Open Door Command - Parameters.....	101
Table: Deleting Credentials for All Users - Parameters.....	101
Table: Activate Aux Relay command - Parameters.....	101
Table: Deactivate Aux Relay command - Parameters.....	102
Table: Deny User Command- Parameters.....	102
Table: Door Panic Command - Parameters.....	102
Table: Cafeteria Reset-Recharge - Parameters.....	103
Table: Smart Card Key Change Response.....	103
Table: Verify biometric and Open door- Parameters.....	105

Table: Arguments and Valid Values	106
Table: API Response Codes	109
Table: Universal Time Zone Reference	112
Table: List of Events	113
Table: Size of Event Fields	116
Table: User Events	116
Table: Special Function Codes Reference	119
Table: Field 3 Detail (User Events) Reference*	120
Table: Information of Bit 0 and 1*	120
Table: Information of Bit 4 to 13*	120
Table: Field 3 Detail (User Events) Reference*	122
Table: Information of Bit 0 and Bit 1*	122
Table: Alarm Events	124
Table: System Events	127
Table: Cafeteria Events	131

About the Document

Welcome to the *COSEC Devices API User Guide*. This document will provide you a comprehensive overview and complete user-guidance for all *COSEC Devices APIs*. 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 omitted in all instances except those used to denote “tags” within XML responses (e.g. “<name></name>”).
2. Cross-references and other links appear as follows: [Document Conventions](#)

For e.g. To learn more about APIs, please refer to section [Who Can Use This Document](#)

3. The term *device* used in this document, will refer only to direct doors.
4. 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.
5. 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 Devices APIs and the basic interface communication. Topic 3 provides a list of all supported APIs. Topics 4 provide an overview of API categories with 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

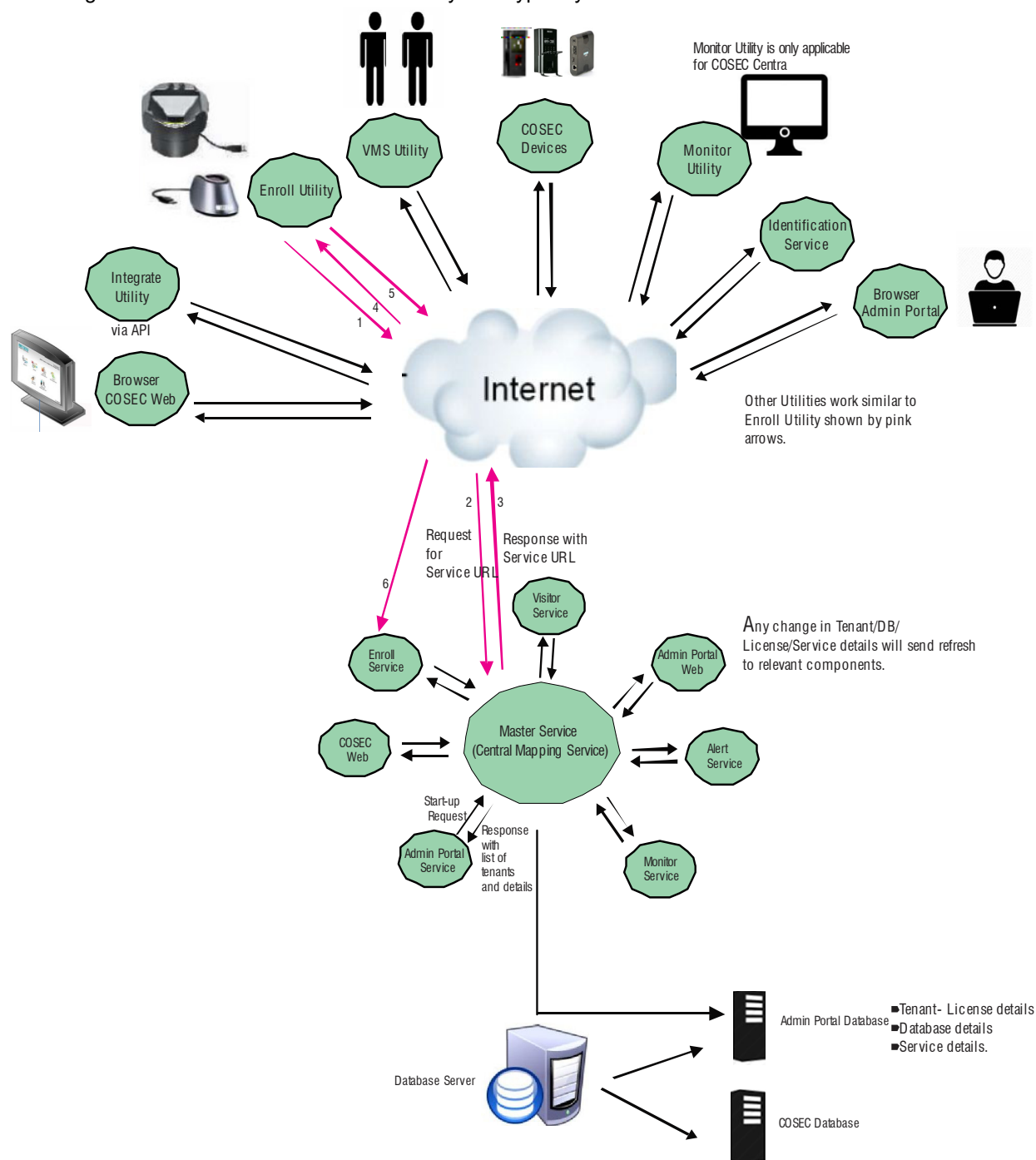
The COSEC Devices API User Guide is meant for *third-party software developers* who wish to operate COSEC Devices via another remote application. This guide will provide information to users on how to request and receive services from COSEC Devices using a COSEC API.

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.



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.

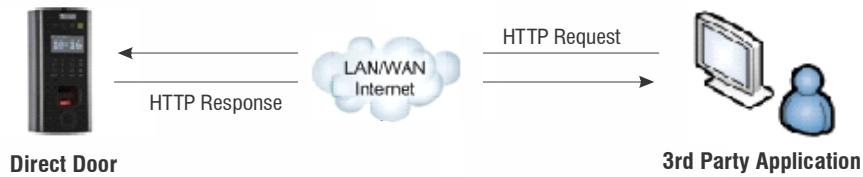


Fig. Communication through COSEC API

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.

Supported Devices

COSEC Devices APIs are dependant on the device type. Currently, Device APIs are supported on the following COSEC Door Controllers and their variants:

- COSEC Direct Door V2
- COSEC Direct Door V3
- COSEC Direct Door V4
- COSEC Path Controller
- COSEC Wireless Door
- COSEC NGT Door
- COSEC PVR Door
- COSEC Vega Controller
- COSEC Argo Controller
- COSEC ARC Controller
- COSEC Door FMX
- COSEC ARGON FACE Door

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 COSEC API, the user will require:

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



For information on installing a COSEC device 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	<i>The start line</i>
*(message-header CRLF)	<i>Zero or more header fields or 'headers'</i>
CRLF	<i>An empty line</i>
[Message-body]	<i>A message-body (chunk or payload)</i>

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.

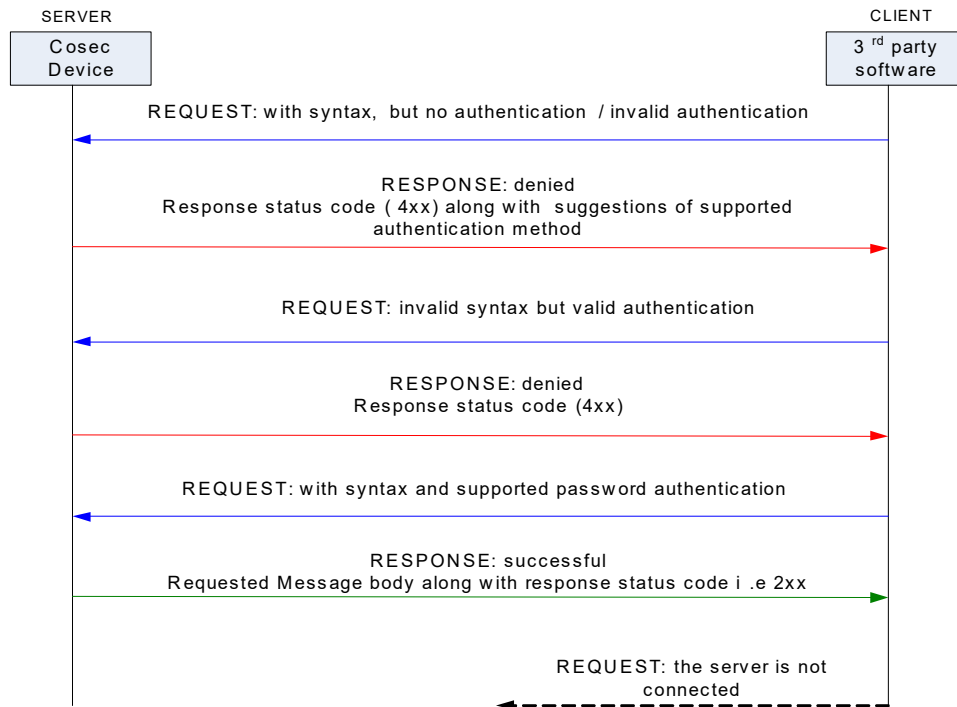


Fig: communication flow

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	<p>This line is constituted by the following three elements which must be separated by a space:</p> <ul style="list-style-type: none"> • The method type (GET, HEAD, POST, PUT etc.) • The requested URL • The HTTP version to use <p>For e.g.:</p> <p>GET http://192.168.1.2/device.cgi/command?action=geteventcount HTTP/1.0</p>
----	--------------	---

2.	Header Fields	<p>Add information about the request using these header fields:</p> <ul style="list-style-type: none"> • 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	<p>This line is constituted by the following three elements which must be separated by a space:</p> <ul style="list-style-type: none"> • The version of the protocol used (e.g. <i>HTTP/1.0</i>). • 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 COSEC 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
<i>http://</i>	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.
<i><deviceIP:deviceport></i>	This identifies the device with which communication is to be performed. It consists of two components: deviceIP: Device IP address deviceport: Device Port Number (Default- 80)
<i>device.cgi</i>	This is a mandatory entity required to specify the CGI directory for all the device-related commands.
<i><request-type></i>	This specifies the type of API request. For the mandatory request types, please refer to the individual API descriptions.

URL element	Description
<argument>	<p>This defines a specific action or command depending on the function to be performed.</p> <p>A mandatory argument for all COSEC API functions is <i>action</i>. This argument always takes an action as its value (For eg. <i>action=get</i>).</p> <p>For more information on the common HTTP actions used in COSEC APIs, please refer to section Common Actions.</p>
<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
<i>GET</i>	To fetch required data from device.
<i>SET</i>	To set required parameters for a given function.
<i>GETDEFAULT</i>	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.
<i>SETDEFAULT</i>	This is used to default the parameters. If any argument is specified then default that particular value else default complete group

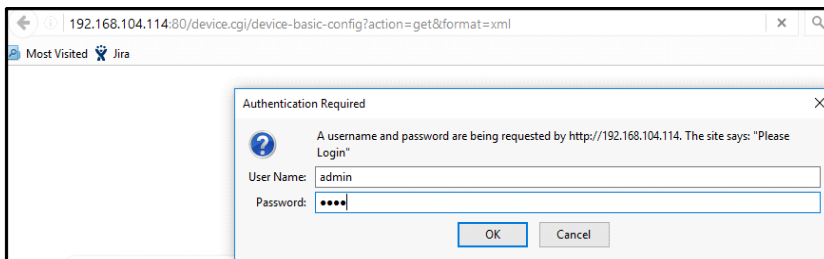
Action	Use
DELETE	To delete data from device.
ENROLL	To enroll an entity to a device.

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.



```
<COSEC_API>
<app>1</app>
<name>Door V3-Device-18</name>
<asc-code>0</asc-code>
<generate-invalid-user-events>0</generate-invalid-user-events>
<generate-exit-switch-events>0</generate-exit-switch-events>
<manual-door-mode-selection>0</manual-door-mode-selection>
<mifare-custom-key-enable>0</mifare-custom-key-enable>
<hid-iclass-custom-key-enable>0</hid-iclass-custom-key-enable>
<max-fingers>1</max-fingers>
<finger-format>0</finger-format>
</COSEC_API>
```

Supported APIs

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

- *Device Configuration*
- *User Configuration*
- *Enrollment*
- *Events*
- *Sending Commands to Device*
- *Access Request on QR Scanning*
- *Cafeteria Reset and Recharge*
- *Get Random Key*
- *Verify Biometric and Open Door*

Device Configuration

This group of APIs enables users to perform the following types of device configuration:

- *Basic Device Configuration*
- *Function Key Configuration*
- *Reader Configuration*
- *Finger Reader Parameter Configuration*
- *Palm Sensor Parameter Configuration*
- *Enrollment Configuration*
- *Access Settings Configuration*
- *Alarm Configuration*
- *Date and Time Configuration*
- *Door Features Configuration*
- *System Timers Configuration*
- *Special Function Configuration*
- *Internal/External Reader Configuration*
- *Device Display Configuration*
- *Device-LED & Buzzer Cadence Configuration*
- *Multi-Language Support*
- *To Download/Upload Multi-Language File*
- *To Select the Mode of Wiegand Interface*
- *Temperature Logging Configuration*
- *Face Mask Compulsion*
- *Smart Card Format*
- *Smart Card Key Change*
- *FR Settings*

Basic Device Configuration

Description: To set or retrieve basic configuration parameters for a device such as application type, name, Additional Security Code and maximum number of finger templates on device.

Actions: get, set, getdefault, setdefault

Syntax: `http://<deviceIP:deviceport>/device.cgi/device-basic-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
app	1, 2 1 = Advanced Access Control 2 = Basic Access Control	No	To define the application.
name	Alphanumeric, Max. 30 characters	No	To identify/configure the device name.
asc-code	Numeric, 16 bits, 1-65535 range	No	To configure an Additional Security Code (ASC). Should be non-zero. Not applicable for ARGO FACE200T.
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: Device 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.
max-faces	0 - 1 Face 1 - 2 Face 2 - 3 Face 3 - 4 Face 4 - 5 Face 5 - 6 Face 6 - 7 Face 7 - 8 Face 8 - 9 Face 9 - 10 Face 10 - 11 Face 11 - 12 Face 12 - 13 Face 13 - 14 Face 14 - 15 Face 15 - 16 Face 16 - 17 Face 17 - 18 Face 18 - 19 Face 19 - 20 Face 20 - 21 Face 21 - 22 Face 22 - 23 Face 23 - 24 Face 24 - 25 Face 25 - 26 Face 26 - 27 Face 27 - 28 Face 28 - 29 Face 29 - 30 Face	No	Maximum no. of templates that can be stored per user on this device.
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. Not applicable for ARGO FACE200T.

Table: Device Configuration Parameters

Argument	Valid Values	Mandatory	Description
mifare-custom-key-enable	0 - Disable 1 - Enable	No	To enable the custom key for mifare card variants. Not applicable for ARGO FACE200T.
mifare-custom-key	Numeric 6 Bytes (12 Hex Digits)	No	To define custom keys for Mifare card variants. Not applicable for ARGO FACE200T.
hid-iclass-custom-key-enable	0 - Disable 1 - Enable	No	To enable the custom key for hid-iclass variants. Not applicable for ARGO FACE200T.
hid-iclass-custom-key	8 Bytes (16 Hex Digits)	No	To define custom keys for HID i-class card variants. Not applicable for ARGO FACE200T.
card-custom-key-auto-update	0 - Disable 1 - Enable	No	To enable the auto update of custom key. Not applicable for ARGO FACE200T.
finger-format	0 - Proprietary 1 - ISO	No	To set the mode of the finger template. By default, proprietary format will be selected.
manual-door-mode-selection	0 - Disable 1 - Enable	No	To enable/disable the Door Mode Selection feature
format	text, xml	No	specifies the format in which the response is expected.



For parameters “mifare-custom-key” and “hid-iclass-custom-key”; “Get & get default action are not applicable.



The **Additional Security Code** is a code that can be written on a smart card for adding an additional layer of security check during door access.



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.



For ARC IO800, Valid Parameters: action, name, format.

Example

Following are some test cases for your reference:

1. To get all parameters.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/device-basic-config?action=get
```


Sample Response

HTTP Code: 200 OK
Content-Type: <code>
Content-Length: <type>
Body:
app=1 name=NGT Direct Door-Device-11 asc-code=0 generate-invalid-user-events=0 generate-exit-switch-events=0 max-fingers=1 finger-format=0

2. To get device name, when expected value is blank and the response format is in text.

Sample Request

http://<deviceIP:deviceport>/device.cgi/device-basic-config?action=get&name&app

Sample Response

HTTP Code: 200 OK
Content-Type: <code>
Content-Length: <type>
Body:
app=1 name=NGT Direct Door-Device-11

3. To get device name, when the expected value is blank and the response format is XML.

Sample Request

http://<deviceIP:deviceport>/device.cgi/device-basic-config?action=get&name&app&format=xml

Sample Response

HTTP Code: 200 OK
Content-Type: <code>
Content-Length: <type>
Body:
<COSEC_API>
<app>1</app>
<name>NGT Direct Door-Device-11</name>
</COSEC_API>

4. To set device name as blank – Valid argument.

Sample Request

http://<deviceIP:deviceport>/device.cgi/device-basic-config?action=set&name=

Sample Response

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

Function Key Configuration

Description: To set or retrieve configuration of Function Keys on the Device keypad. COSEC enables its users to map up to 4 special functions to the arrow keys on a Direct 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
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

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
reader1	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. Not applicable for ARGO FACE200T.
reader2	0 = None 1 = Finger Reader 2 = Palm Vein Reader	No	To define the internal biometric reader. Not applicable for ARGO FACE200T.
reader3	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 8 = UHF Reader 9 = Combo Exit Reader 10 = MiFare-W Reader 11= PIN-W-Reader 12=Card+Pin-W-Reader 13=CB-U-Reader 14=CB-W-Reader 15=ATOM-1 16=ATOM-2 17=ATOM-3	No	To define the external reader. Not applicable for ARGO FACE200T.

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

Argument	Valid Values	Mandatory	Description
door-access-mode	0 = Card 1 = Finger 2 = Card + PIN 3 = PIN + Finger 4 = Card + Finger 5 = Card + PIN + Finger 6 = Any 7 = Palm 8 = Palm + PIN 9 = Card + Palm 10 = Card + PIN + Palm 11 = Palm + Group (Optional) 12 = Finger then Card 13 = Palm then Card 14= None 15= Face 16= Card & Face 17= PIN & Face 18= finger & Face 20= Card then Biometrics	No	To define the access mode applicable for door access.
door-entry-exit-mode	0 = Entry 1 = Exit	No	To define the whether the internal reader is to be set on an entry or exit mode.
reader-access-mode	0 = Card 1 = Biometric 2 = Card + PIN 3 = Biometric + PIN 4 = Biometric + Card 5 = Biometric + Card + PIN 6 = Any One 12 = Biometric then Card 19 = BLE 20 = Card then Biometrics	No	To define the access mode applicable for the external reader. Not applicable for ARGO FACE200T. reader-access-mode= 2, 3 and 5 are only applicable for Vega, Argo, Argo Face and PathV2.
reader-entry-exit-mode	0 = Entry 1 = Exit	No	To define the whether the external reader is to be set on an entry or exit mode. Not applicable for ARGO FACE200T.
tag-re-detect-delay	00 - 3600 seconds	No	To define the tag re-detection delay time. Not applicable for ARGO FACE200T.
format	text, XML	No	Specifies the format in which the response is expected.

Table: Reader Configuration Parameters (COSEC ARC Controllers only)

Argument	Valid Values	Mandatory	Description
rs-485-readergrp1	0 = None 1 = EM Prox Reader 2 = HID Prox Reader 3 = MiFare Reader 4 = HID iCLASS-U Reader 5 = Combo Reader 13= CB-U Reader 15= ATOM-1 16= ATOM-2 17= ATOM-3	No	To define the RS-485 reader in reader1 group.
wiegand-readergrp1	0 = None 1 = Short-Range Reader 2 = Long-Range Reader 11= PIN- W-Reader 12= Card +PIN-W Reader 14= CB -W Reader	No	To define the Wiegand reader in reader1 group. PIN-W-Reader and Card+ PIN-W Reader, Cb-W Reader is applicable for ARC DC 200 only.
readergrp1-entry-exit-mode	0 = Entry 1 = Exit	No	To define the mode (entry/exit) for reader1 group
readergrp1-access-mode	0 = Card 1 = Finger 4 = Card + Finger 6 = Any 12 = Finger then Card 20= Card then Biometrics	No	To define the access mode applicable 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 13= CB-U Reader 15= ATOM-1 16= ATOM-2 17= ATOM-3	No	To define the RS-485 reader in reader2 group.
wiegand-readergrp2	0 = None 1 = Short-Range Reader 2 = Long-Range Reader 11= PIN- W-Reader 12= Card +PIN-W Reader 14= CB -W Reader	No	To define the Wiegand reader in reader2 group. PIN-W-Reader and Card+ PIN-W Reader, Cb-W Reader is applicable for ARC DC 200 only.
readergrp2-entry-exit-mode	0 = Entry 1 = Exit	No	To define the mode (entry/exit) for reader2 group
readergrp2-access-mode	0 = Card 1 = Finger 4 = Card + Finger 6 = Any 12 = Finger then Card 20= Card then Biometrics	No	To define the access mode applicable for reader2 group.

Table: Reader Configuration Parameters (COSEC ARC Controllers only)

Argument	Valid Values	Mandatory	Description
tag-re-detect-delay	00 - 3600 seconds	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&reader1=2&door-access-mode=0`

Sample Response

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

Finger Reader Parameter Configuration

Description: To set the finger reader calibration for fingerprint enrollment.

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi /finger-parameter?<argument>=<value>&<argument>=<value>...

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

Table: Finger Reader Parameter Configuration - Parameters

Argument	Valid Values	Mandatory	Description
security	0 = Level 1 (Low) 1 = Level 2 2 = Level 3 3 = Level 4 4 = Level 5 (High) Default = 0 (For Suprema device) Default = 2 (For Lumidigm device)	Yes	To define the security type while enrollment. Note: 0-4 is only applicable for FMX. 0-2 is applicable for suprema based devices.
lighting-condition	0 = Out door 1 = In door Default = 1	No	To define the lighting condition.
sensitivity	0 = Level 1 (Low) 1 = Level 2 2 = Level 3 3 = Level 4 4 = Level 5 5 = Level 6 6 = Level 7 7 = Level 8 (High) Default = 7 (For Suprema device) Default = 1 (For Lumidigm device)	No	To define the sensitivity levels from low to high. Note: 0-2 is only applicable for FMX. 0-7 is applicable for suprema based devices.
fast-mode	0 = Mode 1 (Normal) 1 = Mode 2 2 = Mode 3 3 = Mode 4 4 = Mode 5 5 = Mode 6 (Fastest) 6 = Auto Default = 6	No	To define the mode to be used during enrollment.

Table: Finger Reader Parameter Configuration - Parameters

Argument	Valid Values	Mandatory	Description
image-quality	0 = Weak 1 = Moderate 2 = Strong 3 = Strongest Default = 1	No	To define the acceptable image quality for enrollment.
format	text,xml	No	Specifies the format in which the response is expected

Palm Sensor Parameter Configuration

Description: To set the palm sensor calibration for palm enrollment.

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi /palm-parameter?<argument>=<value>&<argument>=<value>...

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

Table: Finger Reader Parameter Configuration - Parameters

Argument	Valid Values	Mandatory	Description
security	0 = Normal 1 = Highest 2 = High 3 = Low 4 = Lowest Default = 2	Yes	To define the security type while enrollment.
palm-matching-timeout	0 to 9999 sec Default = 15 sec	No	To define the palm matching timeout.
palm-temp-quality	0 = Good 1 = Moderate 2 = Poor Default = 1	No	To define the acceptable image quality for enrollment.
format	text, XML	No	Specifies the format in which the response is expected.
custom-palm-app-key-enable	0 = Inactive 1 = Active Default = 0	No	To enable the custom key.
custom-palm-app-key	Alphanumeric: 16 characters	No	It specifies the custom key used for identification and verification.

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?action=<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.
temp-per-finger	0 = Single Template/ Finger 1 = Dual Template/Finger	No	To define the number of templates to be saved per finger.
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 Basic 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. Not applicable for ARGO FACE200T.
enroll-mode	0 = Read Only Card 1 = Smart Card 2 = Biometric 3 = Biometric then Card 7 = Face 8 = Duress Finger	No	To define the enrollment mode for enrollment through device. enroll-mode=0 and 1, that is, Read Only Card and Smart Card is not applicable for ARGO FACE200T.
palm-operation-mode	0 = Non-guide mode 1 = Guide mode	No	To specify whether the enrollment is in Guide mode or Non-guide mode for the COSEC PVR Door.
palm mode adaptive	0=Basic Template 1=Compressed Template	No	To define whether PVR is running in Adaptive mode or not
self-enrollment-enable	0 = Disable 1 = Enable	No	To enable/disable self-enrollment on device.
self-enrollment-retry-count	0 - 255	No	To specify the self-enrollment retry count for users.
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.
- The palm adaptive mode is by default disabled, if enabled, device will generate one additional compressed palm template and store it as 11th palm template

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?action=<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

Example

1. To get data for all parameters in the text format.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/access-setting?action=get&format=xml
```

Sample Response

```
HTTP Code: 200 OK
Content-Type: <code>
Content-Length: <type>
Body:
week-day0=1 week-day1=1 week-day2=1 week-day3=1 week-day4=1 week-day5=1 week-day6=1 work-start-hh=0
work-start-mm=0 work-end-hh=23 work-end-mm=59
```

Alarm Configuration

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

Actions: get, set, getdefault, setdefault

Syntax: `http://<deviceIP:deviceport>/device.cgi/alarm?action=<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
alarm	0 = Inactive 1 = Active	No	To enable/disable alarm.
tamper-alarm	0 = Inactive 1 = Active	No	To enable or disable the feature. Not applicable for ARGO FACE200T.
tamper-readergrp1	0 = NO 1 = NC	No	For ARC Controllers reader group 1. To specify normal state as NO/NC to generate tamper alarm.
tamper-readergrp2	0 = NO 1 = NC	No	For ARC Controllers reader group 2. To specify normal state as NO/NC to generate tamper alarm.
auto-alarm-ack	0 = Inactive 1 = Active	No	To enable or disable the Auto Alarm Acknowledgement feature.
format	text, XML	No	Specifies the format in which the response is expected.
thresh-temp-exceeded	0 = Inactive 1 = Active	No	To enable/ disable the “User Denied – Threshold Temperature Exceeded” alarm
mask-not-detected	0 = Inactive 1 = Active	No	To enable/ disable the “User Denied – Face Mask Not detected” 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?action=<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.
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.
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 112	No	To define the universal time zone.
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.



For IO800: Valid Parameters are: action, year, month, date, hour, minute, second, time-format, time-zone, update-mode, ntp-server-type, ntp-server, user-defined-ntp, format.

Door Features Configuration

Description: To enable, disable, define or retrieve configuration parameters related to various door features such as auto-relock, ASC, door sense, exit switch, greeting message display, voice guidance etc.

Actions: get, set, getdefault, setdefault

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

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

Table: Door Features Configuration Parameters

Argument	Valid Values	Mandatory	Description
allow-exit-when-locked	0 = Inactive 1 = Active	No	To allow exit when door is locked. Not applicable for ARGO FACE200T.
auto-relock	0 = Inactive 1 = Active	No	To enable/disable the Auto-relock feature. Not applicable for ARGO FACE200T.
asc-active	0 = Inactive 1 = Active	No	To enable/disable the Additional Security Code (ASC). Not applicable for ARGO FACE200T.
buzzer-mute	0 = Unmute 1 = Mute	No	To mute/un-mute the buzzer.
door-sense-active	0 = Inactive 1 = Active	No	To enable/disable sensing of door states. Not applicable for ARGO FACE200T.
door-sense	0 = NO 1 = NC	No	To define the normal door state as as normally open (NO) or normally closed (NC). Not applicable for ARGO FACE200T.
supervised	0 = Unsupervised 1 = Supervised	No	To enable/disable supervised sensing of door states (four-state monitoring of door controllers). Not applicable for ARGO FACE200T.
exit-switch	0 = Inactive 1 = Active	No	To enable/disable the exit switch. Not applicable for ARGO FACE200T.
aux-output-enable	0 = Inactive 1 = Active	No	To enable/disable aux output. Not applicable for ARGO FACE/ ARGO FACE200T.
aux-output-enable-2	0 = Inactive 1 = Active	No	To enable/disable aux output. Not applicable for ARGO FACE/ ARGO FACE200T.
aux-output-enable-3	0 = Inactive 1 = Active	No	To enable/disable aux output. Not applicable for ARGO FACE/ ARGO FACE200T.

Table: Door Features Configuration Parameters

Argument	Valid Values	Mandatory	Description
aux-output-enable-4	0 = Inactive 1 = Active	No	To enable/disable aux output. Not applicable for ARGO FACE/ ARGO FACE200T.
aux-output-enable-5	0 = Inactive 1 = Active	No	To enable/disable aux output. Not applicable for ARGO FACE/ ARGO FACE200T.
aux-output-enable-6	0 = Inactive 1 = Active	No	To enable/disable aux output. Not applicable for ARGO FACE/ ARGO FACE200T.
aux-output-enable-7	0 = Inactive 1 = Active	No	To enable/disable aux output. Not applicable for ARGO FACE/ ARGO FACE200T.
aux-output-enable-8	0 = Inactive 1 = Active	No	To enable/disable aux output. Not applicable for ARGO FACE/ ARGO FACE200T.
greeting-msg-enable	0 = Inactive 1 = Active	No	To enable/disable the display greeting message.
greeting-msg<1~4>	Alphanumeric, Max. 21 ASCII characters	No	To define upto 4 display greeting messages, the start time and the end time for displaying each message.
greeting-start-time-hh<1~4>	00-23	No	
greeting-start-time-mm<1~4>	00-59	No	
greeting-end-time-hh<1~4>	00-23	No	
greeting-end-time-mm<1~4>	00-59	No	
voice-guidance	0 = Inactive 1 = Active	No	To enable/disable Voice Guidance (Only for NGT doors).
duplicate-access-time-interval	0-9999	No	To define subsequent punch restriction time. Applicable for ARGO, VEGA and ARGO FACE only.
format	text,xml	No	Specifies the format in which the response is expected.



- *When greeting messages are defined in an order then first message will always have precedence over second and second over third and so on. Hence, if two messages defined with overlapped timing range, the first defined message between two will have the priority.*
- *Third party should always take care of setting the time range for different messages.*



For IO800; Valid Parameters are: action, aux-output-enable-2,3,4,5,6,7,8, format.

System Timers Configuration

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

Auto Alarm Acknowledgment 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 Open Pulse 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?action=<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	10-65535 (sec)	No	To define the timer for Auto Alarm Acknowledgment.
idwt	1-99 (sec)	No	To define the Inter Digit Wait Timer.
multi-access-wait-timer	3-99 (sec)	No	To define the Multi Access Wait Timer.
palm-enroll-time-out	3-99 (sec)	No	To define the Palm Enrollment Time out Timer.
pulse-time	1-65535 (sec) 0.1-65535.0 (sec) for ARGO, VEGA, ARGO FACE, Path V2 and ARC DC200	No	To define the Door Pulse time. Not applicable for ARGO FACE200T.
sp-function-timer	1-99 (mins)	No	To define the Special Function Timer.
auto-relock-timer	1-65535 (sec)	No	To define the auto relock timer. Not applicable for ARGO FACE200T.
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?action=<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 = Official Work - IN 2 = Official Work - OUT 3 = Short Leave - IN 4 = Short Leave - OUT 5 = Regular - IN 6 = Regular - OUT 7 = Post Break - IN 8 = Pre Break - OUT 9 = Over Time - IN 10 = Over Time - OUT 11 = Enroll User 12 = Enroll Special Card 13 = Delete Credentials 14 = Late IN - Start 15 = Late IN - Stop 16 = Early OUT - Start 17 = Early OUT- Stop 18 = Door Lock 19 = Door Unlock 20 = Door Normal 21 = Clear Alarm 22 = Sold Out 23 = Available	Yes	The index number of a special function. Sp-fn-Index=12,18,19 and 20 is not applicable for ARGO FACE200T.
enable	0 = Disable 1 = Enable	No	To enable/disable special functions on the device.
card1	64 Bits (20 Numeric Digits approx.)	No	To define the special function card 1.
card2	64 Bits (20 Numeric Digits approx.)	No	To define the special function card 2.
card3	64 Bits (20 Numeric Digits approx.)	No	To define the special function card 3.
card4	64 Bits (20 Numeric Digits approx.)	No	To define the special function card 4.

Table: Special Function Configuration Parameters

Argument	Valid Values	Mandatory	Description
format	text, XML	No	Specifies the format in which the response is expected.

Internal/External Reader Configuration



Internal/External Reader Configuration is not applicable for ARGO FACE200T.

Description: To define custom card format for internal/external reader.

FC = Facility Code
O = Odd Parity
CSN = Card Serial No.
E = Even Parity

Actions: get, set, getdefault, setdefault

Syntax:

Internal Reader Card format:

`http://<deviceIP:deviceport>/device.cgi/internal-card-format?action=<value>&<argument>=<value>...`

External Reader Card format:

`http://<deviceIP:deviceport>/device.cgi/external-card-format?action=<value>&<argument>=<value>...`

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

Table: Internal/External Reader Configuration Parameters

Argument	Valid Values	Mandatory	Description
Format-id	1- 5	No	To specify which format index is to be used
max-no-of-bits	1 - 128 bits	No	To specify no. of bits to be read by card reader.
card-structure	1 - max value 0 = Blank 1 = Facility Code 2 = Odd Parity 3 = Card Serial No. 4 = Even Parity	No	To specify structure of the card format by the selection of even parity, odd parity, card serial no., facility code etc. e.g. 26 bits card format 411111111333333333333332 where, 1 = FC 3 = Card ID 4 = Even Parity 2 = Odd Parity Blank is not a part of configuring card structure. It is to store the default card structure.
card-read-order	0 = Forward 1 = Reverse bitwise 2 = Reverse byte-wise	No	To specify the order in which the card is to be read by the reader. By default, card is read in forward order.

Table: Internal/External Reader Configuration Parameters

Argument	Valid Values	Mandatory	Description
Include-facility-code	0 = Disable 1 = Enable	No	To specify whether facility code is a part of card ID or not. By default it is enabled.
Seq-of-operation	0,1	No	To specify which seq to use 0 = Reading order then bit configuration 1 = Bit Configuration then reading order

Device Display Configuration

Description: To customize the device display on COSEC PVR Doors. The COSEC PVR carries a DOT Matrix LCD display. This API can be used to configure the various display elements at the time of accessing the device or when the device is in the Idle mode.

Actions: get, set, getdefault, setdefault

Syntax: `http://<deviceIP:deviceport>/device.cgi/device-display?action=<value>&<argument>=<value>...`

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

Table: Device Display Settings Parameters

Argument	Valid Values	Mandatory	Description
message-id	Numeric, 1 to 100	Yes	To give message ID. Note: Two message IDs having same value of user field & user-id for same date & day will not be allowed. e.g. if some message is configured for 1st nov to 1st dec for 3 days Monday, Tuesday, Wednesday for all users, then another message cannot be configured for same date range for say, Thursday, Friday and Saturday.
mode	0 = Idle 1 = Access Allowed/ Denied	Yes	To specify the display mode.
enable	0 = Disable 1 = Enable	No	To enable/disable the message.
user	0 = All 1 = User ID	No	This field shall be used to select whether message is to be shown to all users or random users. NA to idle.
user-id	Alphanumeric, 15 characters.	No	This will be used only if a random user is selected in "user" field.
week-day<0~6>	Sunday(0) to Saturday(6) 0 = Inactive 1 = Active	No	To define the active working days. This parameter is repeated for each day of the week.
from-date-dd	Date:1-31; Month: 1-12; Year:2009-2037 Default Value: 1/1/2009	No	This field shall be used to store date from which this message is to be displayed.
from-date-mm			
from-date-yyyy			
to-date-dd	Date:1-31; Month: 1-12; Year:2009-2037 Default Value: 1/1/2009	No	This field shall be used to store date up to which this message is to be displayed.
to-date-mm			
to-date-yyyy			

Table: Device Display Settings Parameters

Argument	Valid Values	Mandatory	Description
duration	Alphanumeric, Sec:00-59 Default Value: 3	No	This field shall be used to store the duration for which message is to be displayed at the time of access allowed/denied. NA to idle
message<0~5>	126 chars(21*6) ASCII Set: A-Z a-z 0-9 - _ . () : @ ! \$ % ' + \ /	No	This field shall be used to configure the message that is to be displayed. One for each line. message0=line 2 message1=line 3 message2=line4 message3=line5 message4=line6 message5=line7 Default value: Blank

Example

1. To set message that is to be shown in the Idle mode.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/device-display?action=set&mode=0&message-id=1&enable=1&week-day6=1&from-date-dd=22&from-date-mm=7&from-date-yyyy=2014&to-date-dd=22&to-date-mm=7&to-date-yyyy=2014&message0=Merry Christmas
```

Sample Response

Response-code=0

Output on Screen:

line 1	<Reserved>															
line 2	M	e	r	r	y		C	h	r	i	s	t	m	a	s	
line 3																
line 4																
line 5																
line 6																
line 7																
line 8	<Reserved>															

2. To set message that is to be shown in the Idle mode.

Sample Request

http://<deviceIP:deviceport>/device.cgi/device-display?action=set&mode=0&message-id=1&enable=1&week-day6=1&from-date-dd=22&from-date-mm=7&from-date-yyyy=2014&to-date-dd=25&to-date-mm=7&to-date-yyyy=2014&message0=_____Merry Christmas

Sample Response

Response-code=0

Output on Screen:

line 1	<Reserved>																					
line 2																		M	e	r	r	y
line 3																						
line 4																						
line 5																						
line 6																						
line 7																						
line 8	<Reserved>																					

3. To delete a specific message.

Sample Request

http://<deviceIP:deviceport>/device.cgi/device-display?action=setdefault&message-id=1

Sample Response

Response-code=0

The message on the message-ID =1 will be defaulted that is equivalent to delete, as message is blank by default.

4. To delete all messages.

Sample Request

http://<deviceIP:deviceport>/device.cgi/device-display?action=setdefault

Sample Response

Response-code=0

All the messages will be defaulted to blank.

Device-LED & Buzzer Cadence Configuration

Description: To configure LED and Buzzer cadence for COSEC PVR Doors.

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi/device-led-buzzer?action=<value>&<argument>=<value>....

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

Table: LED & Buzzer Cadence Configuration Parameters

Argument	Valid Values	Mandatory	Description
state	0 = Idle online 1 = Idle offline 2 = Degraded mode 3 = Wait 4 = Success 5 = Deny 6 = Time out 7 = Alarm minor 8 = Alarm major 9 = Alarm critical	Yes	This specifies the state for which LED and buzzer cadence is to be shown.
led	0 = Off 1 = On	No	This is to specify "on-off" status for LED. By default it is set to "on".
buzzer	0 = Disable 1 = Enable	No	This is to enable or disable the door buzzer. By default, it is set to "enable". (This is not applicable for states: idle online, idle offline and degraded mode, because only LED cadence is shown for these states) For other states, buzzer should be active till LED cadence is shown.
on-off	Numeric. 200-3000 ms (only in multiple of 200 and both together should be in summation equal to 3000 ms) Note: Mention the value of "on-off" as "x-y" where x= on time in milliseconds and y= off time in milliseconds.	No	This is to specify the on-off duration for each state. If not configured, default values should be considered for that state. Default values for states: Idle online - 200 ON/2200 OFF Idle offline - 200 ON/2200 OFF Degraded - 200 ON/2200 OFF Wait - 200 ON/1000 OFF Success - 1200 ON/0000 OFF Deny - 200 ON/200 OFF (3 cycles) Time out - 200 ON/200 OFF Alarm minor - 200 ON/1000 OFF Alarm major - 400 ON/800 OFF Alarm critical - ON till Reset

Table: LED & Buzzer Cadence Configuration Parameters

Argument	Valid Values	Mandatory	Description
cycles	0 - 254	No	<p>This specifies the no. of cycles for which cadence is to be shown for each state.'0' indicates the infinite cycle.</p> <p>Default values for states:</p> <p>Success - 1 Deny - 3 Time out - 3 Others - 0</p>



In case of online state for device, if cycle is changed from infinite to limited no. of cycle, the response will be effective as soon as values are saved but vice-versa is not possible, if user changes the cycle from limited no. of cycle to infinite, it will be effective only after event for the same state will be generated.

1. To show buzzer cadence only for a state.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/device-led-buzzer?action=set&state=4&led=0&buzzer=1&on-off=400-400&cycles=3
```

Sample Response

Response-code=0

Output

The buzzer cadence will be played on the device for 400 ms on and 400 ms off for 3 cycles for wait state and Led cadence will not be shown.

2. To show led and buzzer cadence for a state.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/device-led-buzzer?action=set&state=4&led=1&buzzer=1&on-off=200-1000&cycle=0
```

Sample Response

Response-code=0

Output

The Led and buzzer cadence will be played on the device for 200 ms on and 1000 ms off for infinite cycle till user allowed state.

Multi-Language Support

Description: To enable/disable multiple language support for custom message display on supported COSEC devices (*Door V3, NGT Controller, PVR Door, Vega Controller*).

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

Actions: get, set, getdefault, setdefault

Syntax: `http://<deviceIP:deviceport>/device.cgi/multi-language?action=<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 the multi-language for data input

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?action=<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.

Example

1. To download file.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/multi-language-file?action=get
```

Sample Response

Body:
Response-Code=0

Custom Message File



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.

2. To get sample file.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/multi-language-file?action=getdefault
```

Sample Response

Body:
Response-Code=0

Custom Message File

3. To upload a file.

Sample Request

`http://<deviceIP:deviceport>/device.cgi/multi-language-file?action=set`

Sample Response

Body: Response-Code=0

To Select the Mode of Wiegand Interface



Wiegand Interface Mode is not applicable for ARGO FACE200T.

Description: To set the Wiegand interface mode for supported COSEC doors (*PVR Door, PATH Controller*) for reader input or output to third party panel.

Actions: get, set, getdefault, setdefault

Syntax: `http://<deviceIP:deviceport>/device.cgi/wiegand-interface?action=<value>&<argument>=<value>...`

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

Table: Wiegand Interface Mode Parameters

Argument	Valid Values	Mandatory	Description
enable	0 = Reader Input 1 = Output to 3rd party panel	No	This specifies whether Wiegand interface is configured as a reader input or as an output to 3rd party panel. By default, it is in reader input mode.
enable-signal-wait	0 = Inactive 1 = Active	No	This specifies whether the door should wait for response from the 3rd party panel before unlocking the door after user verification. By default it is disabled.
signal-wait-timer	0 - 255 sec	No	This specifies the time duration for which the door should wait for the 3rd party panel signal
output-format	0 = 26 Bit 1 = Actual 2 = 37 Bit	No	This specifies the format in which the output is to be sent to 3rd party panel.
send-from	0 = MSB Bit 1 = LSB Bit	No	Send MSB or LSB first

Temperature Logging Configuration

Description: To set the temperature parameters like temperature unit, temperature sensors, threshold temperature value etc in devices.

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi/temp-log?<argument>=<value>[&<argument>=<value>....]

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

Table: Temperature Logging Configuration Parameters

Argument	Valid Values	Mandatory	Description
enable	0= Inactive 1= Active	No	To enable or disable Temperature Logging feature. By default it is disabled
temp-unit	0= Fahrenheit 1=Celsius	No	To set the temperature unit in which the user temperature will be measured. By default it is Fahrenheit
sensor-type	0=AST 1= Web-based 2=Reserved 3=FEVOBOT	No	To allow user to select the thermal sensor type. By default it is FEVOBOT.
interface	0= RS-232 1=USB 2=HTTP/S	No	To allow user to select the interface. HTTP/S is applicable only when sensor type is Web-Based. By default it is "USB". For non USB devices default value is "RS – 232". For sensor type FEVOBOT only USB is applicable.
restriction	0=Soft 1=Hard	No	To allow user to switch the restriction between Soft and Hard. By default it is Soft
bypass	0=Disable 1=Enable	No	To give provision of bypass if sensor is disconnected. By default it is disable
thresh-temp	97.0 – 110.0 °F 36.0 – 43.0 °C	No	To allow user to enter the Threshold temperature value. By default it is 99.5°F or 37.5°C
calibration-para	-5.0 to 5.0	No	To calibrate the temperature with increment of 1. By default it is 0. Not applicable for FEVOBOT
emissivity	0.95-0.98	No	To set the emissivity parameter for AST sensor. Default is 0.95. Not Applicable for FEVOBOT
format	text, XML	No	Specifies the format in which the response is expected



- *temp-unit and thresh-temp are interrelated fields.*

For setdefault/getdefault: temp-unit is Fahrenheit so the thresh-temp will be 99.5.

For set:

If both the fields are passed and the thresh-temp value is not as per temp-unit then API will fail.

If temp-unit is only passed then if the temp-thresh value in device is not according to temp-unit passed then temp-thresh value will get defaulted as per temp-unit.

If thresh-temp is only passed, temp-unit value already set in device will be checked and if mismatch is found then API will fail.

- *Sensor-type and interface are interrelated fields:*

For setdefault/getdefault: sensor-type is 0 and the interface is 1 for devices that support USB and the interface is 0 for devices that don't support USB.

- *For set:*

If both the fields are passed and sensor-type is web-based then the interface value should be HTTP/S. If any other interface value is sent then API will fail.

If both the fields are passed and sensor-type is FEVOBOT then the interface value will be USB. If any other interface value is sent then API will fail

If both the fields are passed and sensor-type is not web-based then the interface value should not be HTTP/S. If HTTP/S interface value is sent then API will fail.

For devices which don't support USB if interface value is passed as USB, the API will fail

If only sensor-type is passed then the interface value will be checked.

- *Sensor-type is passed as Web-based and interface value is not HTTP/S in device then the interface value will be set as HTTP/S.*
- *If sensor-type is passed as AST and interface value is HTTP/S in device then the interface value will be set as USB if device supports USB else it will be set as RS-232.*
- *If sensor-type is passed as FEVOBOT and interface value is other than USB in device then the interface value will be set as USB.*
- *If only interface is passed, then sensor-type value will be checked.*
- *Interface is passed as HTTP/S and sensor-type value in device is not Web-based then API will fail.*
- *Interface is passed as USB and device doesn't support USB then API will fail.*
- *Interface is passed as USB and sensor-type is Web-based in device then API will fail.*
- *Interface is passed as RS-232 and sensor-type is Web-Based in device then API will fail.*
- *Interface is passed as RS-232 and sensor-type is Web-Based/FEVOBOT in device then API will fail*

Face Mask Compulsion

Description: Face Mask Compulsion feature is used to enforce users to wear masks while they are within the premises

Actions: get, set, getdefault, setdefault

Syntax: http://<deviceIP:deviceport>/device.cgi/face-mask?<argument>=<value>[&<argument>=<value>....]

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

Table: Smart Card Format Parameters

Argument	Valid Values	Mandatory	Description
enable	0= Inactive 1= Active	No	To enable or disable Face Mask Compulsion Feature. By default it is disabled
approach-cam-timer	0.0 to 15.0s	No	To set the approach to camera wait-timer. By default it is 3.0s
detection-timeout	1 to 99 s	No	To set time-out for face mask detection. By default it is 4s
restriction	0=Soft 1=Hard	No	To allow user to switch the restriction between Soft and Hard. By default it is Soft
detection-threshold-identification	1.00 to 99.99	No	To specify the threshold value below which the mask status will be returned as 0 ie. Not detected for identification. By default the value is set as 98.00
detection-threshold-enrollment	1.00 to 99.00	No	To specify the threshold value below which the mask status will be returned as 0 ie. Not detected for enrollment. By default the value is set as 10.00
format	text, XML	No	Specifies the format in which the response is expected

Smart Card Format



Smart Card Format is not applicable for ARGO FACE200T.

Description: To personalize the smart card format by configuring user-defined fields and defining their location on the available card memory as per card type selected. This API will not be applicable for Path Controllers and Arc Controllers due to memory constraints.

Actions: get, set, setdefault

Syntax: `http://<deviceIP:deviceport>/device.cgi/smart-card-format?action=<value>&<argument>=<value>...`

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

Table: Smart Card Format Parameters

Argument	Valid Values	Mandatory	Description
card-type	Indexed. 1 = iClass 2K2 2 = iClass 16K2 3 = iClass 16K16 4 = Mifare 1K 5 = Mifare 4K	Yes	To specify the card type.
mode	Boolean. 0 = Default 1 = Custom	No	To specify the mode of reading/writing a card.
card-no	Numeric 0 = CSN 1 = UID 2 = Custom		
read-csn	Boolean. 0 = Inactive 1 = Active	No	To decide whether to read CSN when card id gets failed to read. Only applicable when card id is custom.

Table: Smart Card Format Parameters

Argument	Valid Values	Mandatory	Description
index	Refer the respective card type tables for values as provided below.	Yes	To specify the index number of the field to be configured. This field is not mandatory if we want to just change the mode of a particular card type.
field-type		No	To specify the field type of the field to be configured.
date-format		No	To specify the date format.
separator		No	To specify the separator
date-field-type		No	To specify the date field format i.e. BCD/ASCII/Hex
start-sector		No	To specify the start sector for mifare 1K and 4K.
start-block		No	To specify the start block in iclass 2K2, 16K2, 16K16 and mifare 1K , 4K
start-page		No	To specify the start page for iclass 16K16.
start-byte		No	To specify the start byte for iclass 2K2, 16K2, 16K16 and mifare 1K , 4K
start-bit		No	To specify the start bit for iclass 2K2, 16K2, 16K16 and mifare 1K , 4K
read-from		No	To specify the order in which the field has to be read
length		No	To specify the Length of the field.
format	Text, XML	No	Specifies the format in which the response is expected

Table: Smart Card Configuration Parameters - iClass 2K2

Argument	Valid Values	Mandatory	Description
card-type	1 = iClass 2K2	Yes	To specify the card type.
index	Numeric. 1-99	Yes	To specify the index number of the field to be configured. This field is not mandatory if just the mode of a particular card type is to be changed.
field-type	0 = Text 1 = Numeric 2 = Date 3 = Raw	No	To specify the field type of the field to be configured.

Table: Smart Card Configuration Parameters - iClass 2K2

Argument	Valid Values	Mandatory	Description
date-format	0 = ddmmyy 1 = ddmmyyyy 2 = ddmmmyy 3 = ddmmmyyyy 4 = mmddyy 5 = mmddyyyy 6 = yymmdd 7 = yyyymmdd	No	To specify the date format.
separator	0 = none 1 = - 2 = /	No	To specify the separator
date-field-type	0 = Hex 1 = BCD 2 = ASCII	No	To specify the date field format i.e. BCD/ASCII/Hex
start-sector	0	No	-
start-block	19 - 31	No	To specify the start block in iClass 2K2.
start-page	0	No	-
start-byte	0 - 7	No	To specify the start byte for iclass 2K2.
start-bit	0 - 7	No	-
length	0 - 104	No	To specify the Length of the field.

Table: Smart Card Configuration Parameters - iClass 16K2

Argument	Valid Values	Mandatory	Description
card-type	2 = iClass 16K2	Yes	To specify the card type.
index	Numeric. 1-99	Yes	To specify the index number of the field to be configured. This field is not mandatory if just the mode of a particular card type is to be changed.
field-type	0 = Text 1 = Numeric 2 = Date 3 = Raw	No	To specify the field type of the field to be configured.
date-format	0 = ddmmyy 1 = ddmmyyyy 2 = ddmmmyy 3 = ddmmmyyyy 4 = mmddyy 5 = mmddyyyy 6 = yymmdd 7 = yyyymmdd	No	To specify the date format.
separator	0 = none 1 = - 2 = /	No	To specify the separator

Table: Smart Card Configuration Parameters - iClass 16K2

Argument	Valid Values	Mandatory	Description
date-field-type	0 = Hex 1 = BCD 2 = ASCII	No	To specify the date field format i.e. BCD/ASCII/Hex
start-sector	0	No	-
start-block	19 - 255	No	To specify the start block in iClass 16K2.
start-page	0	No	-
start-byte	0 - 7	No	To specify the start byte for iClass 16K2.
start-bit	0 - 7	No	-
length	0 - 1896	No	To specify the Length of the field.

Table: Smart Card Configuration Parameters - iClass 16K16

Argument	Valid Values	Mandatory	Description
card-type	3 = iClass 16K16	Yes	To specify the card type.
index	Numeric. 1-99	Yes	To specify the index number of the field to be configured. This field is not mandatory if just the mode of a particular card type is to be changed.
field-type	0 = Text 1 = Numeric 2 = Date 3 = Raw	No	To specify the field type of the field to be configured.
date-format	0 = ddmmyy 1 = ddmmyyyy 2 = ddmmmyy 3 = ddmmmyyyy 4 = mmddyy 5 = mmddyyyy 6 = yymmdd 7 = yyymmdd	No	To specify the date format.
separator	0 = none 1 = - 2 = /	No	To specify the separator
date-field-type	0 = Hex 1 = BCD 2 = ASCII	No	To specify the date field format i.e. BCD/ASCII/Hex
start-sector	0	No	-
start-block	For Page 0, applicable blocks = 19-31 For Pages 1-6, applicable blocks = 6-31	No	To specify the start block in iclass 16K16.

Table: Smart Card Configuration Parameters - iClass 16K16

Argument	Valid Values	Mandatory	Description
start-page	0-6	No	To specify the start page for iClass 16K16.
start-byte	0 - 7	No	To specify the start byte for iClass 16K16.
start-bit	0 - 7	No	-
length	0 - 1352 (bytes)	No	To specify the Length of the field.

Table: Smart Card Configuration Parameters - Mifare 4K

Argument	Valid Values	Mandatory	Description
card-type	5 = Mifare 4K	Yes	To specify the card type.
index	Numeric. 1-99	Yes	To specify the index number of the field to be configured. This field is not mandatory if we want to just change the mode of a particular card type.
field-type	0 = Text 1 = Numeric 2 = Date 3 = Raw	No	To specify the field type of the field to be configured.
date-format	0 = ddmmyy 1 = ddmmyyyy 2 = ddmmmyy 3 = ddmmmyyyy 4 = mmddyy 5 = mmddyyyy 6 = yyymmdd 7 = yyyyymmdd	No	To specify the date format.
separator	0 = none 1 = - 2 = /	No	To specify the separator
date-field-type	0 = Hex 1 = BCD 2 = ASCII	No	To specify the date field format i.e. BCD/ASCII/Hex
start-sector	0-39	No	To specify the start sector for Mifare 4K.
start-block	For sector 0, applicable blocks = 1,2 For sector 1-31, applicable blocks = 0-2 For sector 32-39, applicable blocks = 0-14	No	To specify the start block in Mifare 4K.
start-page	0	No	-
start-byte	0 - 15	No	To specify the start byte for Mifare 4K.

Table: Smart Card Configuration Parameters - Mifare 4K

Argument	Valid Values	Mandatory	Description
start-bit	0 - 7	No	-
length	0 - 3440	No	To specify the Length of the field.

Table: Smart Card Configuration Parameters - Mifare 1K

Argument	Valid Values	Mandatory	Description
card-type	4 = Mifare 4K	Yes	To specify the card type.
index	Numeric. 1-99	Yes	To specify the index number of the field to be configured. This field is not mandatory if just mode of a particular card type is to be changed.
field-type	0 = Text 1 = Numeric 2 = Date 3 = Raw	No	To specify the field type of the field to be configured.
date-format	0 = ddmmyy 1 = ddmmyyyy 2 = ddmmmyy 3 = ddmmmyyyy 4 = mmddy 5 = mmddyyyy 6 = yymmdd 7 = yyyyymmdd	No	To specify the date format.
separator	0 = none 1 = - 2 = /	No	To specify the separator
date-field-type	0 = Hex 1 = BCD 2 = ASCII	No	To specify the date field format i.e. BCD/ASCII/Hex
start-sector	0-15	No	To specify the start sector for Mifare 1K.
start-block	For sector 0, applicable blocks = 1,2 For sector 1-15, applicable blocks = 0-2	No	To specify the start block in iclass 2K2, 16K2, 16K16 and mifare 1K , 4K
start-page	0	No	-
start-byte	0 - 15	No	To specify the start byte for Mifare 1K.
start-bit	0 - 7	No	-
length	0 - 752 bytes	No	To specify the Length of the field.



- For mode set as “default”, no other parameters are not be configured other than “card-type”.
- For action= setdefault, card-type & index are only the required parameters. If “card-type” parameter is given in setdefault API then all the parameters of a particular card-type will be defaulted. If “index” is also provided with “card-type” as parameter, then all the properties of that specific index field will be set to default.
- For action=get, if the mode of the device is default mode, then the parameters shown in the response will contain parameters applicable according to the default mode. i.e. all the Matrix predefined fields and their corresponding parameters.
- For action = set, all the parameters will be configurable.

Example

1. To set the fields for Mifare 1K in custom mode.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/smart-card-format1?action=set&card-type=3&mode=1&index=22&field-type=1&start-sector=8&start-block=1&start-byte=1&length=4
```

Sample Response

```
HTTP Code: 200 OK
Content-Type: <type>
Content-Length: <length>
Body: <response code>
```

2. To set the mode for Mifare 1K.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/smart-card-format1?action=set&card-type=3&mode=1
```

Sample Response

```
HTTP Code: 200 OK
Content-Type: <type>
Content-Length: <length>
Body: <response code>
```

In the above example, this API will set the mode of the Mifare 1K card type as custom globally, irrespective of the field type. Setting the mode of a card is only dependent upon the card type and not the index.

3. To default Mifare 1K.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/smart-card-format1?action=setdefault&card-type=3
```

Sample Response

HTTP Code: 200 OK
Content-Type: <type>
Content-Length: <length>
Body: <response code>

Smart Card Key Change



Smart Card Key Change is not applicable for ARGO FACE200T.

Description: This API will enable to change the key of the Smart card. The key-type parameter will decide which key i.e. matrix predefined key or the custom key is to be written on the shown smart card.

Actions: set

Syntax: `http://<deviceIP:deviceport>/device-cgi/key-change?action=set[&<argument>=<value>...]`

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

Table: Smart Card Key Change Parameters

Argument	Valid Values	Mandatory	Description
new-key-type	0 = matrix key 1 = custom key stored in the device	Yes	To specify the type of key to be written on the card shown.
old-key-type	0 = matrix key 1 = custom key stored in the device	Yes	To specify the type of key currently written on the card shown.
card-type	0 = Mifare Classic 1 = Mifare DESFire	Yes	Applicable only if Internal Reader= Mifare.
old-key	8 Bytes (16 Hex digits)	Yes	This key will be verified with the old key written in the card.
format	Text, XML	No	Specifies the format in which the response is expected



- This API will only change the key for a single card shown.
- Using this API only key can be changed. Key verification is not supported by this API.
- Server shall send the old key type written in the card along with key value. The old key value should be sent as blank if the old key type is matrix key.

Table: Smart Card Key Change Response

Field	Valid Values	Response
Response-code	0 = success 16 = Device Busy 24 = Feature not configured 28 = Read/Write Failed 29 = Wrong Card Type 30 = Key mismatch	-

Table: Smart Card Key Change Response

Field	Valid Values	Response
card-no	0-18446744073709551615; 8 bytes	<p>1.In CP if card no= CSN; device will read the csn of the card and send the same as response in this parameter.</p> <p>2.In cp if card no= custom; device will write the value received in the card-no parameter in the field22 and send the same as response in this parameter</p> <p>3.This value will be sent has zero if card no is failed to be read.</p> <p>*Note: card no will represent CSN / field22 as per card no parameter defined in CP</p>
Card-type	<p>1 = iclass2K2</p> <p>2 = iclass16K2</p> <p>3 = iclass16K16</p> <p>4 = Mifare 1K</p> <p>5 = Mifare 4K</p> <p>6 = Mifare DESFire 2K</p> <p>7 = Mifare DESFire 4K</p> <p>8 = Mifare DESFire 8K</p> <p>9 = Read Only Card</p>	<p>Read only card will be send if the read card is a HID prox card or EM prox card</p>

FR Settings

Actions: set, get, setdefault, getdefault

Syntax: http://<deviceIP:deviceport>/device.cgi/fr-settings?<argument>=<value>[&<argument>=<value>....]

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

Table: FR Settings

Argument	Valid Values	Mandatory	Description
enable	0 = Disable 1 = Enable Default = 0	No	To enable or disable Face Recognition.
mode	0 = Local	No	Only Local mode is applicable for FR Settings. For Server Assisted Mode, the API will be handled by the Server.
capturing-mode	0 = Tap & Go 1 = Free Scan Default = 0	No	To allow user to switch the mode between tap & go and free scan mode.
enable-freescan-timeout	0 = Disable 1 = Enable Default = 0	No	This will allow user can set the free scan timeout
freescan-timeout	5 to 999 seconds Default: 30 seconds	No	If free scan timeout is enabled, this will set the timer value.
identification-timeout	1-99 s Default: 4seconds	No	This is the timeout value for tap and go mode.
group-fr	0 = Disable 1 = Enable Default = 0	No	To enable or disable Group Face Recognition.
unidentified-face	0 = Disable 1 = Enable Default = 0	No	To enable or disable Capture Face of Unidentified User.
format	Text,xml	No	Specifies the format in which the response is expected
conflict-check	0 = Disable 1 = Enable Default = 1	No	To enable or disable Conflict Checking with other users during Face Enrollment

Table: FR Settings

Argument	Valid Values	Mandatory	Description
conflict-matching-threshold	1.0 - 99.99	No	This is the Matching Threshold used to check conflict with other users during Face Enrollment.
face-anti-spoofing	0 = Disable 1 = Enable Default = 0	No	This flag defines whether Face Anti-Spoofing feature is enabled or not.
face-anti-spoofing-threshold	1.00 - 99.99 Default: 62.00 for ARGO and VEGA Default: 90.00 for ARGO FACE	No	This parameter defines the minimum required value of identified user's face liveness for considering him/her as real person.
adaptive-face-enrollment	0 = Disable 1 = Enable Default = 0	No	To enable or disable Adaptive Face Enrollment
threshold-deviation	0.0 - 10.0 Default: 2.0	No	This parameter defines the threshold value from which the value can be deviated.
multi-user-matching-score-deviation	0.0 - 10.0 Default: 2.0	No	This parameter defines the deviation expected in Matching Score from Multi - User.
confirm-before-adaptive-face-enrollment	0 = Disable 1 = Enable Default = 0	No	To enable or disable Asking for user confirmation before Adaptive Face Enrollment.
face-matching-score	0.0 - 100.0 Default = 94.0	No	This parameter defines the Matching of Face.
threshold-face-detection	0 - 100 Default = 50	No	This parameter defines the threshold that should be maintained for face detection.
adaptive-face-templates-per-user	1-10 Default = 60	No	This parameters defines the number of adaptive face templates that can be stored against a user.
Unidentified-face-feedback	0 = Disable 1 = Enable Default = 0	No	To enable or disable Show Feedback for Unidentified Face.
Unidentified-face-event	0 = Disable 1 = Enable Default = 0	No	To enable or disable Generate Unidentified Face Event.

Table: FR Settings

Argument	Valid Values	Mandatory	Description
Access-via-qr	0 = Disable 1 = Enable Default = 0	No	To enable or disable Allow Access via QR.

User Configuration

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

S. No.	COSEC Device	No. of Users Supported
1	Direct Door V1	500
2	Direct Door V2	2000
3	Direct Door V3	50,000
4	Direct Door V4	50,000
5	NGT Direct Door	10,000
6	Wireless Door	50,000
7	PVR Door	10,000
8	PATH Door	10,000
9	ARC Door	10,000
10	ARC IO 800	NA
11	Panel	10,000
12	Panel-Lite	25,000
13	Panel200	25,000
14	Vega Controller	50,000
15	Door FMX	50,000
16	MODE Device	50,000
17	ARGO	50,000
18	ARGO FACE	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](#)
- [Setting a User Photo](#)
- [Deleting a User](#)
- [Setting User Credentials](#)
- [Retrieving User Credentials](#)
- [Deleting User Credentials](#)

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?action=<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 Controller = 1 - 2,000 Path V2= 1 - 50,000 Wireless Door = 1 - 50,000 PVR = 1 - 10,000 NGT = 1 - 10,000 Vega Controller = 1 - 50,000 Door FMX = 1 - 50,000 Panel Lite V2= 1 - 25,000 Panel/ Panel Lite= 1 - 25,000 V4 DOOR= 1 - 50,000 ARGO= 1 - 50,000 ARGO FACE= 1 - 50,000 ARC DC 200= 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	2000-2099	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	It defines the value of access card 1 and 2.
card2	64 Bits (8 bytes) (max value - 18446744073709551615)	No	
dob-enable	0 = Enable 1 = Disable	No	To enable/disable the display of a birthday message.
dob-dd	1-31	No	To set or delete the date of birth for a user.
dob-mm	1-12		
dob-yyyy	1990-2037		
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.
self-enrollment-enable	0 = Disable 1 = Enable	No	To enable/disable self-enrollment for user.
enable-fr	0 = Disable 1 = Enable	No	To enable/disable face recognition for a 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 data for user-id = 1

Sample Request

```
http://192.168.104.114:80/device.cgi/users?action=get&user-id=1
```

Sample Response

```
user-id=1 user-index=2 ref-user-id=1 name=Chirag user-active=1 vip=0 validity-enable=0 validity-date-  
dd=1 validity-date-mm=1 validity-date-yyyy=2009 user-pin= card1=3280168263 card2=0 self-enrollment-  
enable=0 by-pass-finger=0
```

Setting a User Photo

Description: To set, fetch or delete a photograph against a user's profile on the device using a third party application.

Actions: get, set, delete

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

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

Table: Setting a User Photo - Parameters

Argument	Valid Values	Mandatory	Description
user-id	Maximum 15 characters	Yes	To specify the alphanumeric user ID for the user whose photo is to be set.
user-photo	N/A	Yes	To get, set or delete the user photo. This should be done in the data portion of the request / response.(applicable only for VEGA and NGT doors)
photo-format	0 = jpeg 1 = jpg 2 = png 3 = bmp	Yes (only for set parameter)	To define the format for the photograph.
format	text,xml	No	Specifies the format in which the response is expected.

Example

Following are some test cases for your reference:

1. To add an image file in .jpeg format for user-id 1.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/userphoto?action=set&user-id=1&photo-format=0
Data:
Image data
```

Sample Response

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

2. To fetch the user photo for the same user.

Sample Request

```
http://<deviceIP:deviceport>/device.cgi/userphoto?action=get&user-id=1
```

Sample Response

HTTP Code: 200 OK
Content-Type: image/jpeg
Content-Length: 12345
Body:

<JPEG Image Data>



This is an example only. The actual response will vary depending on product model and configuration.

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 5 = Face Template 6 = Face image	Yes	To define the user credentials type. Type= 5 and 6 are applicable only for ARGO FACE.
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	It defines the value of access card 1 and 2.
card2	64 Bits (8 bytes) (max value - 18446744073709551615)	No	

Table: Setting User Credentials - Parameters

Argument	Valid Values	Mandatory	Description
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.

Example:

The API for setting user credential on device is mentioned in URL field of 3rd party application as shown below:
`http://192.168.104.114:80/device.cgi/credential?action=set&type=1&user-id=101`

Form1

URL:

UserName: Password: content-type:

Response

☐ Is Success

The **User Name** and **Password** are the device login credentials.

Click **Browse** button to select the credential template file say finger template which is required to set on the device.

Now click **Send Request** to set the finger template on device. The success response of API will be shown as below.

Form1

URL:

UserName: Password: content-type:

Response

☒ Is Success

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 5 = Face Template 6 = Face Image	Yes	To define the user credentials type. Type= 5 and 6 are applicable only for ARGON FACE.
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)		It defines the value of access card 1 and 2.
card2	64 Bits (8 bytes) (max value - 18446744073709551615)		
finger-index	1 = 1 Finger 2 = 2 Fingers 3 = 3 Fingers 4 = 4 Fingers 5 = 5 Fingers 6 = 6 Fingers 7 = 7 Fingers 8 = 8 Fingers 9 = 9 Fingers 10 = 10 Fingers	No	Identifies the number of finger templates/palm/ face templates/ face image to be set or retrieved, on or from the device. The template/image will be set and retrieved from the data portion of the request and response. Face-index is applicable for both type= 5 and type=6.
palm-index	1 = 1 Palm 2 = 2 Palms 3 = 3 Palms 4 = 4 Palms 5 = 5 Palms 6 = 6 Palms 7 = 7 Palms 8 = 8 Palms 9 = 9 Palms 10 = 10 Palms 11 = 11 Palms	No	

Table: Retrieving User Credentials - Parameters

Argument	Valid Values	Mandatory	Description
face-index	1 - 1 Face 2 - 2 Face 3 - 3 Face 4 - 4 Face 5 - 5 Face 6 - 6 Face 7 - 7 Face 8 - 8 Face 9 - 9 Face 10 - 10 Face 11 - 11 Face 12 - 12 Face 13 - 13 Face 14 - 14 Face 15 - 15 Face 16 - 16 Face 17 - 17 Face 18 - 18 Face 19 - 19 Face 20 - 20 Face 21 - 21 Face 22 - 22 Face 23 - 23 Face 24 - 24 Face 25 - 25 Face 26 - 26 Face 27 - 27 Face 28 - 28 Face 29 - 29 Face 30 - 30 Face 31 - 31 Face 32 - 32 Face 33 - 33 Face 34 - 34 Face 35 - 35 Face 36 - 36 Face 37 - 37 Face 38 - 38 Face 39 - 39 Face 40 - 40 Face	No	Identifies the number of finger templates/palm/ face templates to be set or retrieved, on or from the device. The template will be set and retrieved from the data portion of the request and response.
format	text,xml	No	Specifies the format in which the response is expected. Applicable only for type=2
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. See example in Setting User Credential.*
- *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 is not a mandatory field and vice versa.*
- *Compressed palm template will be given with index 11.*
- *If the user passes the argument: Face-index and type=5/6 in the Credentials API with action=GET, then device should return the face template/ face image present at that specific face-index in response.*
- *At a time only one face image can be obtained via action=GET and passing the face-index.*
- *In data field, if data of 832 bytes is received then only the data will be treated as compressed palm template and stored.*
- *At a time only one face image can be set. Multiple face images cannot be set a time.*
- *For type=6, maximum size allowed= 200Kb and supported file format= jpg only.*

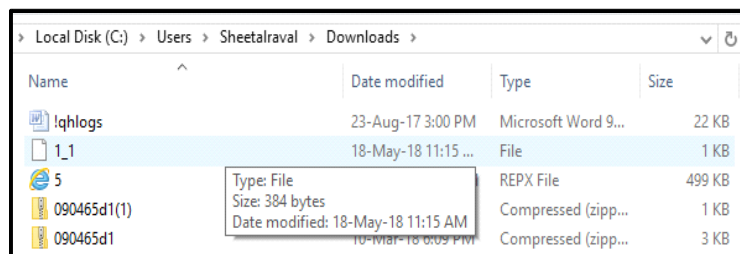
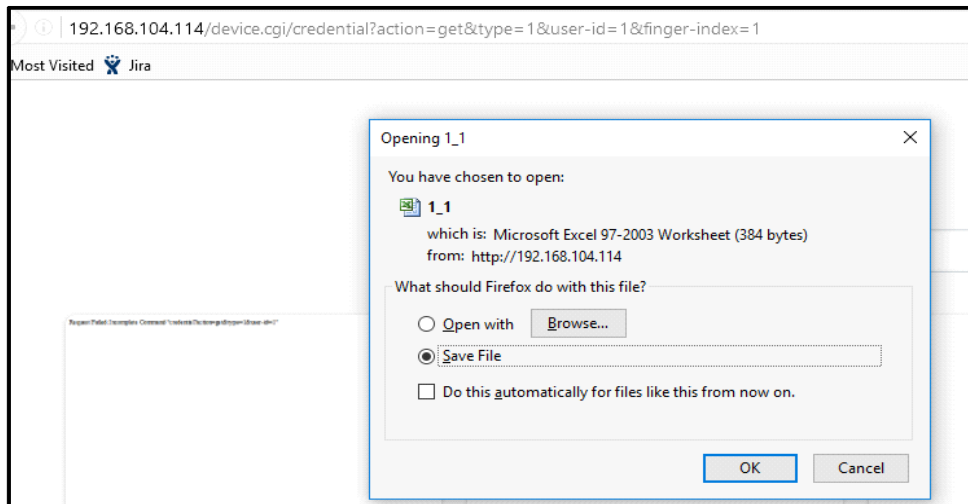
Example1

1. To get finger credential of user-id = 1

Sample Request

`http://192.168.104.114:80/device.cgi/credential?action=get&type=1&user-id=1&finger-index=1`

Sample Response



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 5 = Face Template 6 = Face Image	Yes	Defines the credential type to be deleted. Note: For the selected type, all credentials will be deleted. Type= 5 and 6 are applicable only for ARGON FACE. NOTE- For delete if type is all then all card, biometric and face credentials should be deleted. For type= 5/6, both face images and face templates will be deleted.
format	text,xml	No	Specifies the format in which the response is expected.

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

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*](#)
- [*Read/Write Card*](#)
- [*Test Enrollment*](#)

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
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. type=0 and 1, that is, Read Only Card and Smart Card is not applicable for ARGO FACE200T.
user-id	Maximum 15 characters	Yes	Defines the alphanumeric User ID of the user whose credential is to be enrolled.
enroll-using	0 = Reader Group 1 1 = Reader Group 2 2 = Device	No	To specify on which reader the enrollment is to be done. <i>Only applicable for ARC DC 100 and ARC DC 200.</i>
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	To specify the number of fingers to be enrolled.
card-count	0 = 1 Card 1 = 2 Cards 2 = 3 Cards 3 = 4 Cards	No	To specify the number of cards to be enrolled.

Table: Enrolling User - Parameters

Argument	Valid Values	Mandatory	Description
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	To specify the number of palms to be enrolled.
face-count	0 - 1 Face 1 - 2 Face 2 - 3 Face 3 - 4 Face 4 - 5 Face 5 - 6 Face 6 - 7 Face 7 - 8 Face 8 - 9 Face 9 - 10 Face 10 - 11 Face 11 - 12 Face 12 - 13 Face 13 - 14 Face 14 - 15 Face 15 - 16 Face 16 - 17 Face 17 - 18 Face 18 - 19 Face 19 - 20 Face 20 - 21 Face 21 - 22 Face 22 - 23 Face 23 - 24 Face 24 - 25 Face 25 - 26 Face 26 - 27 Face 27 - 28 Face 28 - 29 Face 29 - 30 Face	No	To specify the number of faces 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.

Table: Enrolling User - Parameters

Argument	Valid Values	Mandatory	Description
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.
w-palm-template	0= No 1=Yes	No	To enable/disable the finger 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			

Table: Enrolling User - Parameters

Argument	Valid Values	Mandatory	Description
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&user-id=45&type=2&finger-count=1`

Sample Response

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

Enrolling Special Cards



Enrolling Special Cards is not applicable for ARGO FACE200T.

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
sp-fn-id	All configured Special Functions (special function ID)	Yes	Defines the special function identification number.
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.

Read/Write Card



Read/Write Card is not applicable for ARGO FACE200T.

Description: This API will be used to read and write a card. The API will only read/write values from the card/on the card. No values will be permanently stored on the device.

Actions: read, write

Syntax: `http://<deviceIP:deviceport>/device.cgi/card-read-write?<argument>=<value>[&<argument>=<value>...]`

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

Table: Read/Write Card Parameters

Argument	Valid Values	Mandatory	Description
ref-user-id	0-9; 8 digits	No	To specify the numeric User-ID.
user-id	15 char, Alphanumeric	No	To specify the alpha-numeric User-ID.
name	15 char, Alphanumeric	No	To specify the user name.
asc	1-65535	No	To specify the additional security code.
fc	1-65535	No	To specify the facility code.
designation	15 char, ASCII Code	No	To specify the designation of the user.
branch		No	To specify the branch of the user.
department		No	To specify the department of the user.
bg	0- NA 1- A+ 2- A- 3- B+ 4- B- 5- AB+ 6- AB- 7- O+ 8- O- 9- A1- 10- A1+ 11- A1B- 12- A1B+ 13- A2- 14- A2+ 15- A2B- 16- A2B+ 17- B1+	No	To specify the blood-group of the user.
contact	15 char, ASCII Code	No	To specify the emergency contact of the user.
medical-history	15 char, ASCII Code	No	To specify the medical-history of the user.

Table: Read/Write Card Parameters

Argument	Valid Values	Mandatory	Description
pin	0-9; 6 digits	No	To specify the user pin.
VIP	0- Inactive 1- Active	No	To specify the user as VIP or not.
bypass-palm	0=Inactive 1= Active	No	To specify the user has authority to bypass palm verification
bypass-fp	0- Inactive 1- Active	No	To specify the user has authority to bypass finger print verification or not
validity-date-dd	1-31	No	To specify the validity date
validity-date-mm	1-12	No	To specify the validity month
validity-date-yyyy	2000-2037	No	To specify the validity year
access-level	1-15	No	To specify the access level of user for smart identification
card-no	0-18446744073709551615; 8 bytes	No	To specify CSN/Card ID of the card
fp1	0=not present 1=present	No	To specify if FP template1 is to be written on card or not.
fp2	0=not present 1=present	No	To specify if FP template2 is to be written on card or not.
palm	0=not present	No	To define whether the palm compressed template is to be written on the card [field-25] or not.
format	Text, XML	No	To specify the format in which response is needed
fp-index	1 or 2	Yes	To identify whether the template received is 1st template or second template
Smart-access-route-id	1-99	No	When Smart Access Route is enabled for SI user
Smart-access-user-level	1-75	No	When Smart Access Route is enabled for SI user

Action= Write

- For this action device will write the values received in the API into the respective fields in the card.
- Note that all the values that are to be written in the card should be received in the API.
- Device is not expected fetch the values from the device configuration and write them into the card.
- **Card no:**
If valid card no parameter is received and card no in CP configuration = default: device will ignore this parameter, write the card and return the CSN in the card no parameter of the API response.

- Server shall send the old key type written in the card along with key value. The old key value should be sent as blank if the old key type is matrix key.

Table: Write Card Response

Field	Valid Values	Response
Response-code	0 = success 16 = Device Busy 26 = Parameters not applicable as per card type defined. 27 = Time-Out 28 = Read/Write Failed 29 = Wrong Card Type 30 = Key mismatch	-
card-no	0-18446744073709551615; 8 bytes	4.In CP if card no= CSN; device will read the csn of the card and send the same as response in this parameter. 5.In cp if card no= custom; device will write the value received in the card-no parameter in the field22 and send the same as response in this parameter 6.This value will be sent has zero if card no is failed to be read. *Note: card no will represent CSN / field22 as per card no parameter defined in CP
Card-type	1 = iclass2K2 2 = iclass16K2 3 = iclass16K16 4 = Mifare 1K 5 = Mifare 4K 6 = Mifare DESFire 2K 7 = Mifare DESFire 4K 8 = Mifare DESFire 8K 9 = Read Only Card	Read only card will be send if the read card is a HID prox card or EM prox card

Action= Read

- If any parameters received in the API are not configured in the CP configuration then API will be failed with reason 26.
- After reading the values, device will send the read values in the API response and No further action will be taken at the device.
- Device will not store the read values in its memory.Test Enrollment

Table: Read Card Response

Field	Valid Values	Response
Response-code	0 = success 16 = Device Busy 27 = Time-Out 28 = Read/Write Failed 29 = Wrong Card Type 30 = Key mismatch	-

Table: Read Card Response

Field	Valid Values	Response
card-no	0-18446744073709551615; 8 bytes	1.In CP if card no= CSN; device will read the csn of the card and send the same as response in this parameter. 2.In cp if card no= custom; device will write the value received in the card-no parameter in the field22 and send the same as response in this parameter 7.This value will be sent has zero if card no is failed to be read. *Note: card no will represent CSN / field22 as per card no parameter defined in CP
Card-type	1 = iclass2K2 2 = iclass16K2 3 = iclass16K16 4 = Mifare 1K 5 = Mifare 4K 6 = Mifare DESFire 2K 7 = Mifare DESFire 4K 8 = Mifare DESFire 8K 9 = Read Only Card	Read only card will be send if the read card is a HID prox card or EM prox card

Table: Read Card Response

Field	Valid Values	Response
ref-user-id	Value read from the respective field from the card	If the corresponding fields are not configured in the CP [for CP card format = custom] or if the debug field of respective field = 0 [for CP card format = default] then that respective field should not be a part of the response.
user-id	Value read from the respective field from the card	
name	Value read from the respective field from the card	
asc	Value read from the respective field from the card	
fc	Value read from the respective field from the card	
designation	Value read from the respective field from the card	
branch	Value read from the respective field from the card	
department	Value read from the respective field from the card	
bg	Value read from the respective field from the card	
contact	Value read from the respective field from the card	
medical-history	Value read from the respective field from the card	
pin	Value read from the respective field from the card	
vip	Value read from the respective field from the card	
bypass-fp	Value read from the respective field from the card	
bypass-palm	Value read from the respective field from the card	
validity-date-dd	Value read from the respective field from the card	
validity-date-mm	Value read from the respective field from the card	
validity-date-yyyy	Value read from the respective field from the card	
access-level	Value read from the respective field from the card	

Table: Read Card Response

Field	Valid Values	Response
card-no	Value read from the respective field from the card	If the corresponding fields are not configured in the CP [for CP card format = custom] or if the debug field of respective field = 0 [for CP card format = default] then that respective field should not be a part of the response.
Smart-access-route-id	Value read from the respective field from the card	
Smart-access-user-level	Value read from the respective field from the card	
fp-count	0,1,2	<p>1.For card format in cp = default Return 0 if both FP debug fields = 0 Return 1 if any FP debug field = 1 Return 2 if both FP debug field = 1</p> <p>2.For card format in cp = custom Always return [device side limitation]</p>
palm-count	0,1	<p>For card format in cp = default Return 0 if palm debug fields = 0 Return 1 if palm debug field = 1 For card format in cp = custom Always return [device side limitation]</p>

Test Enrollment

Description: This API will be used to know whether the device is applicable to be used as a enrollment device or not. There is a requirement from the COSEC ENROLL that it should check whether the device is online or not. To confirm the same the following API must be fired from the COSEC Enroll.

Actions: get

Syntax: `http://<deviceIP:deviceport>/device.cgi /check-enrollment?<argument>=<value>[&<argument>=<value>...]`

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	Refer to "Table: Value Range for Event Sequence Numbers" for the valid values on different devices.	Yes	
no-of-events	1 to 5 (for Direct Door V2, Path Controller, IO 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.

Table: Value Range for Event Sequence Numbers

Door	Event Sequence Number
V2	1 to 50,000
CDC	1 to 50,000
Wireless	1 to 5,00,000
NGT	1 to 1,00,000
PVR	1 to 1,00,000
Vega Controller	1 to 5,00,000
ARC Controller	1 to 1,00,000
IO Controller	1 to 1,00,000
Door FMX	1 to 5,00,000
ARGO FACE	1 to 5,00,000
ARGO	1 to 5,00,000



- 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).

Example

1. To request specific events with roll over count = 0 and sequence number = 1. No. of events requested is 3, for an NGT door.

Sample Request

`http://deviceIP:deviceport/device.cgi/events?action=getevent&roll-over-count=0&seq-number=1&no-of-events=3`

Sample Response

```
HTTP Code: 200 OK
Content-Type: xml
Content-Length: 12345
Body:
<COSEC_API>
<Events>
<roll-over-count>0</roll-over-count>
<seq-No>1</seq-No>
<date>16/4/2014</date>
<time>14:56:20</time>
<event-id>457</event-id>
<detail-1>0</detail-1>
<detail-2>0</detail-2>
<detail-3>6</detail-3>
<detail-4>0</detail-4>
<detail-5>0</detail-5>
</Events>
<Events>
<roll-over-count>0</roll-over-count>
<seq-No>2</seq-No>
<date>16/4/2014</date>
<time>14:56:20</time>
<event-id>453</event-id>
<detail-1>0</detail-1>
<detail-2>0</detail-2>
<detail-3>0</detail-3>
<detail-4>0</detail-4>
<detail-5>0</detail-5>
</Events>
<Events>
<roll-over-count>0</roll-over-count>
<seq-No>3</seq-No>
<date>16/4/2014</date>
<time>14:57:28</time>
<event-id>453</event-id>
<detail-1>0</detail-1>
<detail-2>0</detail-2>
<detail-3>0</detail-3>
<detail-4>0</detail-4>
<detail-5>0</detail-5>
</Events>
</COSEC_API>
```

For example if an enrollment event is called in which three fingers have been enrolled with the dual template per finger then the detail fields will be as follows:

For first finger:

- Event-ID: 405 (code for enrollment event)
- Detail-1: user-id
- Detail-2: 9 (code for finger credential)
- Detail-3: **12**
- Detail-4: 0
- Detail-5: 0

For second finger:

- Event-ID: 405 (code for enrollment event)
- Detail-1: user-id
- Detail-2: 9 (code for finger credential)
- Detail-3: **24**
- Detail-4: 0

- Detail-5: 0

For third finger:

- Event-ID: 405 (code for enrollment event)
- Detail-1: user-id
- Detail-2: 9 (code for finger credential)
- Detail-3: 36
- Detail-4: 0
- Detail-5: 0

If the template per finger mode was selected as single template per finger then the respective values for detail 3 will be 11, 22 and 33, where LSB denotes the template index.

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	It is used to start and stop the process
keep-live-events	0: inactive 1: active	No	It is used to send the events continuously or till the maximum limit.
ipaddress	IP address and port number validations are same as for network configuration settings.	Yes	Defines the IP Address and the listening port on which the events are to be sent.
port			
roll-over-count	0 to 65535	Yes No; if keep-live-events=1	It is used to specify the exact sequence number of an event stored at any port.
seq-number	Refer to "Table: Value Range for Event Sequence Numbers" for the valid values on different devices.	Yes No; if keep-live-events=1	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 = WiFi 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 Direct 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 acknowledgment 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 (up to 5,00,000 events).*

Sending Commands to Device

It is possible to send CGI commands to a device 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.
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.
4	Lock Door	lockdoor	To command the door to return to a locked state. Not applicable for ARGO FACE200T.
5	Unlock Door	unlockdoor	To command the door to return to an unlocked state. Not applicable for ARGO FACE200T.
6	Normalize Door	normalizedoor	To command the door to return to a normal state. Not applicable for ARGO FACE200T.
7	Open Door	opendoor	To open the door. The door should be opened only for the defined pulse time. Not applicable for ARGO FACE200T. For parameters, refer Table: Open Door Command - Parameters below.
8	Get User Count on Device	getusercount	To obtain the total number of users added on a device.
9	Get Current Event Sequence Number	geteventcount	To get the current event sequence number and roll over count in a device.
10	Default the System Configuration	systemdefault	To set all the configurations on the device to default status.
11	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.
12	Activate the Aux Relay	activateauxrelay	To activate the Aux Relay.
13	De-activate the Aux Relay	deactivateauxrelay	To de-activate the Aux Relay.
14	Deny User	denyuser	To deny a user. For parameters, refer Table: Deny User Command- Parameters below.

Table: List of Commands to Device

S.No.	Command to Device	Action	Description
15	Door Panic	door panic	To create door panic situation when some number is entered by user from his extension. For parameters, refer Table: Door Panic Command - Parameters below.

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	0 = 1 Card 1 = 2 Cards 2 = 3 Cards 3 = 4 Cards	No	To get the number of cards enrolled.
finger-count	Single Template/Finger: 0-9 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 0 = 1 Finger 1 = 2 Fingers 2 = 3 Fingers 3 = 4 Fingers 4 = 5 Fingers	No	To get the number of fingers 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	No	To get the number of palms enrolled.

Table: Get Credential Count Command - Parameters

Argument	Valid Values	Mandatory	Description
face-count	0 - 1 Face 1 - 2 Face 2 - 3 Face 3 - 4 Face 4 - 5 Face 5 - 6 Face 6 - 7 Face 7 - 8 Face 8 - 9 Face 9 - 10 Face 10 - 11 Face 11 - 12 Face 12 - 13 Face 13 - 14 Face 14 - 15 Face 15 - 16 Face 16 - 17 Face 17 - 18 Face 18 - 19 Face 19 - 20 Face 20 - 21 Face 21 - 22 Face 22 - 23 Face 23 - 24 Face 24 - 25 Face 25 - 26 Face 26 - 27 Face 27 - 28 Face 28 - 29 Face 29 - 30 Face	No	To get the number of faces enrolled. Currently applicable for ARGO FACE only
format	text,xml	No	Specifies 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.

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 >
```

For action=opendoor

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

Table: Open Door Command - Parameters

Argument	Valid Values	Mandatory	Description
extra-info1	4 ASCII bytes	Yes	This field is kept mandatory to identify the source of the command.
extra-info2	2 ASCII bytes	No	-
extra-info3	0 to 65535	No	-
show-a/v-response	0 = Do not display audio/video (default value) 1 = Display audio/video	No	To enable/disable display of audio/video.



The valid values in the “extra-info” fields are numeric values which can be replaced by valid enumerated codes and can be included in the event by the third party requesting the command.

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
type	0 = All 1 = Finger 2 = Palm 3 = Face (Currently applicable for ARGO FACE only)	Yes	To specify the type of credential to be deleted.

For action=activateauxrelay

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

Table: Activate Aux Relay command - Parameters

Argument	Valid Values	Mandatory	Description
port-no	1 to 8	No	To select the Aux port. This parameter is applicable only for IO controller.

For action=deactivateauxrelay

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

Table: Deactivate Aux Relay command - Parameters

Argument	Valid Values	Mandatory	Description
port-no	1 to 8	No	To select the Aux port. This parameter is applicable only for IO controller.

For action=denyuser

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

Table: Deny User Command- Parameters

Argument	Valid Values	Mandatory	Description
extra-info1	4 ASCII bytes	Yes	This field is kept mandatory to identify the source of the command.
extra-info2	2 ASCII bytes	No	-
extra-info3	0 to 65535	No	-
show-a/v-response	0 = Do not display audio/video (default value) 1 = Display audio/video	No	To enable/disable display of audio/video.

For action=doorpanic

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

Table: Door Panic Command - Parameters

Argument	Valid Values	Mandatory	Description
extra-info1	4 ASCII bytes	Yes	This field is kept mandatory to identify the source of the command.
extra-info2	2 ASCII bytes	No	-
extra-info3	0 to 65535	No	-

Cafeteria Reset and Recharge



Cafeteria Reset and Recharge is not applicable for ARGO FACE200T.

Description: This API will be used to recharge/reset the cafeteria balance on card.

Actions: read, write

Syntax: `http://<deviceIP:deviceport>/device.cgi/cafeteria-reset-recharge?<argument>=<value>[&<argument>=<value>...]`

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

Table: Cafeteria Reset-Recharge - Parameters

Argument	Valid Values	Mandatory	Description
card-no1	0-18446744073709551615; 8 bytes	Yes	To define the CSN/Card ID of the user.
card-no2	0-18446744073709551615; 8 bytes	Yes	To define the CSN/Card ID of the user.
balance	4 bytes	No	To specify the available cafeteria balance of the user.
format	text,xml	No	Specifies the format in which the response is expected.

Table: Smart Card Key Change Response

Field	Valid Values	Response
Response-code	0 = success 16 = Device Busy 26 = Parameters not applicable as per card type defined. 28 = Read/Write Failed 29 = Wrong Card Type 30 = Key mismatch 31 = Invalid card	-
card-no	0-18446744073709551615; 8 bytes	1.In CP if card no= CSN; device will read the csn of the card and send the same as response in this parameter. 2.In cp if card no= custom; device will write the value received in the card-no parameter in the field22 and send the same as response in this parameter 3.This value will be sent has zero if card no is failed to be read. *Note: card no will represent CSN / field22 as per card no parameter defined in CP

Table: Smart Card Key Change Response

Field	Valid Values	Response
Card-type	1 = iclass2K2 2 = iclass16K2 3 = iclass16K16 4 = Mifare 1K 5 = Mifare 4K 6 = Mifare DESFire 2K 7 = Mifare DESFire 4K 8 = Mifare DESFire 8K 9 = Read Only Card	Read only card will be send if the read card is a HID prox card or EM prox card

Verify Biometric and Open Door

Description: When this API is fired; device will prompt for User finger and it will be compared with the finger templates received from APIs.

Actions: set

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

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

Table: Verify biometric and Open door- Parameters

Argument	Valid Values	Mandatory	Description
user-id	15 Alphanumeric characters	Yes	The user ID on which the specified operation is to be done.
name	15 Alphanumeric characters with space	Yes	The name of the user
operation	0-2	Yes	0-Verify and open 1-Only Verify 2-Only open
Format	Text, XML	No	Specifies the format in which the response is expected.



Finger data must be sent in JSON format only.
JSON object names must be fp1, fp2,..., fp10 only.

If operation = 0; device will verify scanned finger with received fingers in the API and if verified then device will open the door.

If operation =1 then device will verify the scanned finger with received fingers in the API and if verified device will send success response but door will not be opened.

If operation = 2 then device will directly open the door.

Temperature Reading

Description: This API will be used by 3rd Party Thermal Sensors to send temperature readings to the device.

Actions: set

Syntax: http://<deviceIP:deviceport>/device.cgi/temp-reading?<argument>=<value>[&<argument>=<value>....]

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

Table: Arguments and Valid Values

Argument	Valid Values	Mandatory	Description
temp-value	6 characters (0-9 and dot (.) only)	Yes	The temperature value recorded by sensor
temp-unit	0=Fahrenheit 1= Celsius 2=Kelvin	Yes	The unit in which the temperature value is recorded by sensor
Format	Text, XML	No	Specifies the format in which the response is expected.

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>"

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
5	Image – size is too big	On every set command for user API
6	Image – format not supported	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/ Face template already exists/ Face Image already exists	Set credential API
10	No Record Found	Event sequence number and roll over count not found, user id not found in Set User API
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 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.
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	Credential not found	Get FP/Palm/ Face template or Face Image command is sent but template/ image is not present.

Table: API Response Codes

Response Code	Description	Test Condition
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.
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. On Set Credential API- for Face Image, if the device's FR is disabled Or On Set Credential API- for Face Image/ Face Template, if the device's FR Mode is server-assisted.
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	Invalid smart card format/ Parameters not applicable as per card type defined.	If any of the entered values are incorrect as per specified range./ If any parameter received in the API is configured in the CP configuration of the device.
27	Time Out	If the card is not shown on the device.
28	Read/Write failed	If the card has been failed to read/write due to any reason like: key mismatch, wrongly placed card etc
29	Wrong Card Type	If the card shown on the device is not as per the reader set on the device for example, if write operation is to be done & read only card is shown or reader is set as HID i-class & mifare card has been shown etc
30	key mismatch	If the old key written in the card mismatches with the read old key.
31	invalid card	This should be sent whenever the cafeteria card shown is a wrong card for recharge/reset API.
32	Scan failed	When the finger/palm scanning fails dues to reason like improper finger/palm placement or bad quality template in scan template API
33	Invalid value	If the value of the parameters passed in the API is not applicable. Example if template format is passed as 0/1 for FMX door
34	Credential does not match	If finger templates received in the API and scanned templates does not match.
35	Failure	Fail to find any FP template/image for FP Template API Applicable for Face Image Upload via Web API
36	Face Not Detected	On Set Credential API- for Face Image, if the received face image fails face detection
37	User Conflict	On Set Credential API- for Face Image, if the received face image is already enrolled against any other user
38	Enroll Conflict	On Set Credential API- for Face Image, if for the user any previously enrolled images exist and the received face image doesn't match with the previously enrolled face images
39	Face Mask Detected	On Set Credential API- for Face Image, if the received face image has face with mask worn.

Table: API Response Codes

Response Code	Description	Test Condition
40	Full Face Not Visible	On Set Credential API- for Face Image, if the received face image fails Full Face Visible conditions
41	Face Not Straight	On Set Credential API- for Face Image, if the extracted face from the received face image is not straight

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
103	User Allowed – Anti-Pass Back-soft
104	User Allowed - Dead-man Zone
105	User Allowed – Door Not open
106	User Allowed – Smart Secure Access
107	User Allowed – Smart card based route access - soft
108	User Allowed – Panel route access - soft
109	User Allowed – two person rule - primary user
110	User Allowed – two person rule - secondary user
111	Visitor Allowed
112	User Allowed – Multi-Level Access
151	User Denied – User Invalid
152	User Denied – Occupancy Control
153	User Denied – 2-Person Rule
154	User Denied – Time Out
155	User Denied – Visitor Escort Rule
156	User Denied – Anti-Pass Back
157	User Denied – Disabled User
158	User Denied – Blocked User
159	User Denied – First IN User
160	User Denied – DND Enabled
161	User denied – Control zone

Table: List of Events

Event ID	Event Description
162	User Denied – Door Lock
163	User Denied – Invalid Access Group
164	User Denied – Validity date expired
165	User Denied – Invalid Route Access
166	User Denied – Invalid Shift Access
167	User Denied – Invalid Access Cluster
171	Visitor Denied
172	User Denied – FP sensor busy
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
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
318	Occupancy Violated Alarm
319	Tail- Gating Alarm
320	Man Trap Timer Violated Alarm

Table: List of Events

Event ID	Event Description
321	Access Denied Aalrm
322	Multiple Unauthorized Access Alarm
323	Custom Alarm 1
324	Custom Alarm 2
325	Custom Alarm 3
326	User Unidentified
327	Anti-Pass Back Violated Alarm
328	Access Route Violated Alarm
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
404	Guard Tour-status
405	Enrolment
406	Master Alarm sense input status
407	Master Aux Output status
408	Input Output Group Link status
409	Credentials Deleted
410	Time Triggered Function
411	Time Stamping Function
412	Guard tag
413	Camera Event for time stamp
451	Configuration Change
452	Roll over of events
453	Master Controller Power ON
454	Configuration Defaulted
455	Soft Override
456	Backup and Update
457	Default System
458	Sensor Calibration
459	User Denied – invalid card
460	User PIN Change



Some of the events listed are applicable only on Panels/Panel Doors and not on Direct Doors. Refer the respective event tables to see the applicable doors for each event.

Table: Size of Event Fields

Door	Field 1	Field 2	Field 3	Field 4	Field 5	Event Log Capacity
Direct Door V2	4 bytes	2 bytes	2 bytes	N.A.	N.A.	50,000 events
Path Controller	4 bytes	2 bytes	2 bytes	N.A.	N.A.	50,000 events
Wireless Door	4 bytes	2 bytes	2 bytes	4 bytes	4 bytes	5,00,000 events
NGT Direct Door	4 bytes	2 bytes	2 bytes	4 bytes	4 bytes	1,00,000 events
PVR Door	4 bytes	2 bytes	2 bytes	4 bytes	4 bytes	1,00,000 events
Vega Controller	4 bytes	2 bytes	2 bytes	4 bytes	4 bytes	5,00,000 events
Panel200	4 bytes	2 bytes	2 bytes	4 bytes	4 bytes	5,00,000 events
ARC	4 bytes	2 bytes	2 bytes	N.A.	N.A.	1,00,000 events
IO Controller	4 bytes	2 bytes	2 bytes	N.A.	N.A.	1,00,000 events
Door FMX	4 bytes	2 bytes	2 bytes	4 bytes	4 bytes	5,00,000 events
Direct Door V3	4 bytes	2 bytes	2 bytes	4 bytes	4 bytes	5,00,000 events
Direct Door V4	4 bytes	2 bytes	2 bytes	4 bytes	4 bytes	5,00,000 events
ARGO	4 bytes	2 bytes	2 bytes	4 bytes	4 bytes	5,00,000 events
ARGO FACE	4 bytes	2 bytes	2 bytes	4 bytes	4 bytes	5,00,000 events

Table: User Events

Event Details					Applicable Devices											
Event ID	(Field 1) User ID	(Field 2) Special Code	(Field 3) Entry/Exit	(Field 4) and (Field 5)	Direct Door V2	Path Controller	Wireless Door / Door V3	NGT Door	PVR Door	Vega Controller	Panel/ Panel-Lite/ Standalone Panellite	ARC	Door FMX	ARGO	Direct Door V4	
User Allowed Events																
101	Xxxx (user ID=0 for REX input)	Special Function code	Detail			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
102	Xxxx	Special Function code	Detail			✗	✗	✓	✗	✓	✗	✗	✗	✗	✓	
103	Xxxx	Special Function code	Detail			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
104	Xxxx	Special Function code	Detail			✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	

Table: User Events

Event Details					Applicable Devices										
Event ID	(Field 1) User ID	(Field 2) Special Code	(Field 3) Entry/Exit	(Field 4) and (Field 5)	Direct Door V2	Path Controller	Wireless Door / Door V3	NGT Door	PVR Door	Vega Controller	Panel/ Panel-Lite/ Standalone Panellite	ARC	Door FMX	ARGO	Direct Door V4
105	Xxxx	Special Function code	Detail		X	X	X	X	X	X	✓	X	X	X	X
106	Xxxx	Special Function code	Detail		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
107	Xxxx	Special Function code	Detail		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
108	Xxxx	Special Function code	Detail		X	X	X	X	X	X	✓	X	X	X	X
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		✓	✓	✓	✓	✓	✓	X	✓	✓	✓	✓
112	Xxxx	0 = Door unlock 1 = Door lock	Detail		X	X	X	X	X	X	✓	X	X	X	X
User Denied Events															
151	(User ID = 0 if not identified)	Special Function code	Detail		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
152	Xxxx		Detail		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
153	Xxxx		Detail		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
154	Xxxx		Detail		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
155	Xxxx		Detail		X	X	X	X	X	X	✓	X	X	X	X
156	Xxxx		Detail		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
157	Xxxx		Detail		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
158	Xxxx		Detail		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table: User Events

Event Details					Applicable Devices											
Event ID	(Field 1) User ID	(Field 2) Special Code	(Field 3) Entry/ Exit	(Field 4) and (Field 5)	Direct Door V2	Path Contr oller	Wire less Door / Door V3	NGT Door	PVR Door	Vega Contr oller	Panel/ Panel-Lite/ Standalone Panellite	ARC	Door FMX	ARGO	Direct Door V4	
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	Detail		✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
		1=Door Not in Route														
		2=Door Not in Sequence for Smart card based Route														
		3=Door Not in Smart card based Route			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		4=Cred ential Invalid for Smart card based Route Access														
		5=Door cannot be accessed now due to time zone inside access route violation			✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
		6=Door is not assigned			✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗

Table: User Events

Event Details					Applicable Devices										
Event ID	(Field 1) User ID	(Field 2) Special Code	(Field 3) Entry/Exit	(Field 4) and (Field 5)	Direct Door V2	Path Controller	Wireless Door / Door V3	NGT Door	PVR Door	Vega Controller	Panel/ Panel-Lite/ Standalone Panellite	ARC	Door FMX	ARGO	Direct Door V4
166	Xxxx	0=Outside working hours 1=Holiday 2=Week off 3=Field Break 4=Rest Day	Detail		x	x	x	x	x	x	✓	x	x	x	x
					✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
167	Xxxx		Detail		x	x	x	x	x	x	✓	x	x	x	x
171	First four bytes of extension number	last two bytes of extension number	Detail		✓	✓	✓	✓	✓	✓	x	✓	✓	✓	✓
172	Xxxx		Detail		x	x	x	x	x	x	x	x	✓	x	x

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	✓	x
2	Official Work-OUT Marking in T&A	2	✓	x
3	Short Leave-IN Marking in T&A	3	✓	x
4	Short Leave-OUT Marking in T&A	4	✓	x
5	Clock - IN Marking in T&A	5	✓	x
6	Clock - OUT Marking in T&A	6	✓	x
7	Post Lunch-IN Marking in T&A	7	✓	x
8	Pre Lunch -OUT Marking in T&A	8	✓	x
9	Over time – IN Marking in T&A	9	✓	x

Table: Special Function Codes Reference

S.No.	Special Function Name	Special Function Code	Applicable for Allowed Events	Applicable for Denied Events
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	QR	Card 2	Card 1	BLE	Face	API	Group	Palm	Finger	Card	PIN	RFU		Time Stamp	Entry/Exit

* Applicable for Firmware Version of following doors:

ARC DC200: V01R40 and later.

ARGO: V01R42 and later.

ARGO FACE: V01R20 and later.

VEGA: V01R71 and later.

PATH V2: V01R34 and later.

Table: Information of Bit 0 and 1*

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

* Applicable for Firmware Version of following doors:

ARC DC200: V01R40 and later.

ARGO: V01R42 and later.

ARGO FACE: V01R20 and later.

VEGA: V01R71 and later.

PATH V2: V01R34 and later.

Table: Information of Bit 4 to 13*

Credential	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Value
PIN	0	0	0	0	0	0	0	0	0	0	1	1
Card 1	0	0	1	0	0	0	0	0	0	1	0	258
Card 1 + PIN	0	0	1	0	0	0	0	0	0	1	1	259
Finger	0	0	0	0	0	0	0	0	1	0	0	4
Finger + PIN	0	0	0	0	0	0	0	0	1	0	1	5
Finger + Card 1	0	0	1	0	0	0	0	0	1	1	0	262

Credential	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Value
Finger + Card 1 + PIN	0	0	1	0	0	0	0	0	1	1	1	263
Palm	0	0	0	0	0	0	0	1	0	0	0	8
PIN + Palm	0	0	0	0	0	0	0	1	0	0	1	9
Card 1 + Palm	0	0	1	0	0	0	0	1	0	1	0	266
PIN + Card 1 + Palm	0	0	1	0	0	0	0	1	0	1	1	267
Group + Palm	0	0	0	0	0	0	1	1	0	0	0	24
API	0	0	0	0	0	1	0	0	0	0	0	32
API + Finger	0	0	0	0	0	1	0	0	1	0	0	36
API + Palm	0	0	0	0	0	1	0	1	0	0	0	40
API + Face	0	0	0	0	1	1	0	0	0	0	0	96
API + PIN	0	0	0	0	0	1	0	0	0	0	1	33
Group + Finger	0	0	0	0	0	0	1	0	1	0	0	20
API + Card 1	0	0	1	0	0	1	0	0	0	1	0	290
Face	0	0	0	0	1	0	0	0	0	0	0	64
PIN + Face	0	0	0	0	1	0	0	0	0	0	1	65
Card 1 + Face	0	0	1	0	1	0	0	0	0	1	0	322
Finger + Face	0	0	0	0	1	0	0	0	1	0	0	68
Palm + Face	0	0	0	0	1	0	0	1	0	0	0	72
BLE 1	0	0	1	1	0	0	0	0	0	0	0	384
BLE 2	0	1	0	1	0	0	0	0	0	0	0	640
BLE 1 + Card 1	0	0	1	1	0	0	0	0	0	1	0	386
BLE 2 + Card 1	0	1	1	1	0	0	0	0	0	1	0	898
BLE 1 + Finger	0	0	1	1	0	0	0	0	1	0	0	388
BLE 2 + Finger	0	1	0	1	0	0	0	0	1	0	0	644
BLE 1 + Palm	0	0	1	1	0	0	0	1	0	0	0	392
BLE 2 + Palm	0	1	0	1	0	0	0	1	0	0	0	648
BLE 1 + Face	0	0	1	1	1	0	0	0	0	0	0	448
BLE 2 + Face	0	1	0	1	1	0	0	0	0	0	0	704
BLE 1 + PIN	0	0	1	1	0	0	0	0	0	0	1	385
BLE 2 + PIN	0	1	0	1	0	0	0	0	0	0	1	641
Card 2	0	1	0	0	0	0	0	0	0	1	0	514
Card 2 + PIN	0	1	0	0	0	0	0	0	0	1	1	515
Finger + Card 2	0	1	0	0	0	0	0	0	1	1	0	518
Finger + Card 2 + PIN	0	1	0	0	0	0	0	0	1	1	1	519
Card 2 + Palm	0	1	0	0	0	0	0	1	0	1	0	522

Credential	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Value
PIN + Card 2 + Palm	0	1	0	0	0	0	0	1	0	1	1	523
API + Card 2	0	1	0	0	0	1	0	0	0	1	0	546
Card 2 + Face	0	1	0	0	1	0	0	0	0	1	0	578
BLE 1 + Card 2	0	1	1	1	0	0	0	0	0	1	0	898
BEL 2 + Card 2	0	1	0	1	0	0	0	0	0	1	0	642
QR 1	1	0	1	0	0	0	0	0	0	0	0	1280
QR 1 + PIN	1	0	1	0	0	0	0	0	0	0	1	1281
API + QR 1	1	0	1	0	0	1	0	0	0	0	0	1312
QR 1 + Face	1	0	1	0	1	0	0	0	0	0	0	1344
BLE 1 + QR 1	1	0	1	1	0	0	0	0	0	0	0	1408
BLE 2 + QR 1	1	1	1	1	0	0	0	0	0	0	0	1920
QR 2	1	1	0	0	0	0	0	0	0	0	0	1536
QR 2 + PIN	1	1	0	0	0	0	0	0	0	0	1	1537
API + QR 2	1	1	0	0	0	1	0	0	0	0	0	1568
QR 2 + Face	1	1	0	0	1	0	0	0	0	0	0	1600
BLE 1 + QR 2	1	1	1	1	0	0	0	0	0	0	0	1920
BLE 2 + QR 2	1	1	0	1	0	0	0	0	0	0	0	1664

* Applicable for Firmware Version of following doors:

ARC DC200: V01R40 and later.

ARGO: V01R42 and later.

ARGO FACE: V01R20 and later.

VEGA: V01R71 and later.

PATH V2: V01R34 and later.

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				BLE	Face	API	Group	Palm	Finger	Card	PIN	RFU		RFU	Entry/Exit

* Applicable for All Doors.



Applicable for Firmware Version of following doors:

ARC DC200: Earlier than V01R40.

ARGO: Earlier than V01R42.

ARGO FACE: Earlier than V01R20.

VEGA: Earlier than V01R71.

PATH V2: Earlier than V01R34.

Table: Information of Bit 0 and Bit 1*

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

* Applicable for All Doors.



Applicable for Firmware Version of following doors:

ARC DC200: Earlier than V01R40.

ARGO: Earlier than V01R42.

ARGO FACE: Earlier than V01R20.

VEGA: Earlier than V01R71.

PATH V2: Earlier than V01R34.

Table: Information of Bit 4 to 9*

Credential	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Value
PIN	0	0	0	0	0	0	0	1	1
Card	0	0	0	0	0	0	1	0	2
Card + PIN	0	0	0	0	0	0	1	1	3
Finger	0	0	0	0	0	1	0	0	4
Finger + PIN	0	0	0	0	0	1	0	1	5
Finger + Card	0	0	0	0	0	1	1	0	6
Finger + Card + PIN	0	0	0	0	0	1	1	1	7
Palm	0	0	0	0	1	0	0	0	8
PIN + Palm	0	0	0	0	1	0	0	1	9
Card + Palm	0	0	0	0	1	0	1	0	10
PIN + Card + Palm	0	0	0	0	1	0	1	1	11
Group + Palm	0	0	0	1	1	0	0	0	24
API	0	0	1	0	0	0	0	0	32
API + Finger	0	0	1	0	0	1	0	0	36
API + Palm	0	0	1	0	1	0	0	0	40
API + PIN	0	0	1	0	0	0	0	1	33
Group + Finger	0	0	0	1	0	1	0	0	20
Face	0	1	0	0	0	0	0	0	64
Card + Face	0	1	0	0	0	0	1	0	66
PIN + Face	0	1	0	0	0	0	0	1	65
Finger + Face	0	1	0	0	0	1	0	0	68
Palm + Face	0	1	0	0	1	0	0	0	72
BLE	1	0	0	0	0	0	0	0	128
BLE + Finger	1	0	0	0	0	1	0	0	132
BLE + Palm	1	0	0	0	1	0	0	0	136
BLE + Face	1	1	0	0	0	0	0	0	192
BLE + PIN	1	0	0	0	0	0	0	1	129
BLE + Card	1	0	0	0	0	0	1	0	130

* Applicable for All Doors.



Applicable for Firmware Version of following doors:

ARC DC200: Earlier than V01R40.

ARGO: Earlier than V01R42.

ARGO FACE: Earlier than V01R20.

VEGA: Earlier than V01R71.

PATH V2: Earlier than V01R34.

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	Panel/ Panel Lite	ARGO	Direct Door V4	ARC Controller
301	Reference ID Xxxx	1 = Critical	Alarm Sequence Number		✓	✓	✓	✓	✓	✓	✗	✓	✗	✗	✗
302	Reference ID Xxxx	1 = Critical	Same as above		✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗
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		✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗

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	Panel/ Panel Lite	ARGO	Direct Door V4	ARC Controller
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	0	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		✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗

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	Panel/ Panel Lite	ARGO	Direct Door V4	ARC Controller
320	Reference ID Xxxx	2 = Major	Same as above		X	X	X	X	X	X	X	✓	X	X	X
321	Reference ID Xxxx	2 = Major	Same as above		X	X	X	X	X	X	X	✓	X	X	X
322	Reference ID Xxxx	2 = Major	Same as above		X	X	X	X	X	X	X	✓	X	X	X
323		2 = Major	Same as above		X	X	X	X	X	X	X	✓	X	X	X
324		2 = Major	Same as above		X	X	X	X	X	X	X	✓	X	X	X
325		2 = Major	Same as above		X	X	X	X	X	X	X	✓	X	X	X
326		2 = Major	Same as above		X	X	X	X	X	X	X	✓	X	X	X
327	Reference ID Xxxx	2 = Major	Same as above		X	X	X	X	X	X	X	✓	X	X	X
328	Reference ID Xxxx	2 = Major	Same as above		X	X	X	X	X	X	X	✓	X	X	X
329	Reference ID Xxxx	2 = Major	Same as above		X	X	X	X	X	X	X	✓	X	X	X
351		4 = SysInte rlock 5 = User_J eeves 6 = User_A CMS 9 = Auto	Same as above		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
352		4 = SysInte rlock 5 = User_J eeves 6 = User_A CMS 7 = Special Functio n	Same as above		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
353			Same as above		X	X	X	X	X	X	X	✓	X	X	X

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/Door V3	NGT Door	PVR Door	Vega Controller	Door FMX	Panel/Panel Lite/ Standalone Panel lite	IO Controller	ARC Controller	ARGO	Direct Door V4
401	User ID: xxxx	0= Unused (Restore User) 1=Absentee Rule 2=Unauthorized access 3=Usage count 4=Invalid PIN 5=Tail Gating 6=Man Trap Timer Violated 7=Occupancy Violated 8=Anti-Pass Back Violated	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		✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

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/Door V3	NGT Door	PVR Door	Vega Controller	Door FMX	Panel/Panel Lite/ Standalone Panel lite	IO Controller	ARC Controller	ARGO	Direct Door V4
405	reference ID: Xxxx	8= User Card 9= User Finger Propriety format 10= Special Cards 14= Palm 16= palm template with guide mode 17=User Finger - Suprema ISO format 18=User Finger - Lumidigm ISO format 19= User Face 20=Mobile Device			✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓
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		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

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/Door V3	NGT Door	PVR Door	Vega Controller	Door FMX	Panel/Panel Lite/ Standalone Panel lite	IO Controller	ARC Controller	ARGO	Direct Door V4
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/Door V3	NGT Door	PVR Door	Vega Controller	Door FMX	Panel/Panel Lite/ Standalone Panel lite	IO Controller	ARC Controller	ARGO	Direct Door V4
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	X	X	X	X	X	X	X	✓	X	X	X	X
456	1=Backup 2=Update	1=Configuration 2=Event 3=Firmware		0 = Fail 1 = Success 2 = CRC Check Fail	X	X	✓	✓	✓	✓	✓	✓	X	X	✓	✓
457				6 = from ACMS 8 = from Hardware	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
458		0 = Internal Finger Reader 1 = External Finger Reader		0 = Fail 1 = Success 2 = Not Supported	✓	✓	✓	✓	X	✓	✓	✓	X	✓	✓	✓
459	Card ID	Card ID	Card ID		✓	✓	✓	✓	✓	✓	✓	✓	X	✓	✓	✓
460	Reference ID: xxxx				X	X	✓	✓	✓	✓	✓	✓	X	X	✓	✓
461	Reference ID: xxxx	0 = Authorized 1 = Rejected		System User Index 1 to 10	X	X	X	X	X	X	X	✓	X	X	X	X

Table: Cafeteria Events

Event Details						Applicable Devices										
Event ID	(Field 1)	(Field 2)	(Field 3)	(Field 4)	(Field 5)	Direct Door V2	Path Controller	Wireless Door	NGT Door	PVR Door	Vega Controller	Door FMX	Panel/ Panel Lite	ARC Controller	ARGO	Direct Door V4
501	Reference ID Xxxx	Item Code + Menu Number + Quantity	Menu Number	Item Cost		✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
502	Reference ID Xxxx	Item Code + Menu Number + Quantity	User Level + Menu Number	Item Cost	Discount Value	✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
503	Reference ID Xxxx	Item Code + Menu Number + Quantity	Menu Number	Item Cost		✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
504	Reference ID Xxxx	Item Code + Menu Number + Quantity	User Level + Menu Number	Item Cost	Discount Value	✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
505		Item Code + Menu Number	Menu Number	Item Cost		✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
506		Item Code + Menu Number	Menu Number	Item Cost		✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
507	Reference ID Xxxx			Transaction Amount	Available Balance	✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
508	Reference ID Xxxx	0= Failure 1=Successful 2=verification failed		Previous Balance	Recharge Amount	✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
509	Reference ID Xxxx	0= Failure 1=Successful 2=verification failed		Previous Balance	Reset Value	✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
510	Reference ID Xxxx		Menu Number	Previous Balance	Transaction Amount	✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓
511	Reference ID Xxxx		Menu Number		Transaction Amount	✓	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓

Table: Cafeteria Events

Event Details						Applicable Devices										
Event ID	(Field 1)	(Field 2)	(Field 3)	(Field 4)	(Field 5)	Direct Door V2	Path Controller	Wireless Door	NGT Door	PVR Door	Vega Controller	Door FMX	Panel/ Panel Lite	ARC Controller	ARGO	Direct Door V4
512	Reference ID Xxxx	0= Consecutive transaction 1= pre-ordering				✗	✗	✓	✓	✓	✓	✓	✗	✗	✓	✓



MATRIX COMSEC PVT. LTD.

Head Office:

394-GIDC, Makarpura, Vadodara - 390010, India.

Ph.: (+91)18002587747

E-mail: Tech.Support@MatrixComSec.com

Website: www.MatrixComSec.com