HIKVISION

How to Integrate with Temperature Screen Terminal via ISAPI

What is ISAPI

The OPEN Intelligent Security Application Programming Interface (ISAPI) is a text protocol in **RESTful style** based on **HTTP** for communicating between security devices/servers (e.g., cameras, DVR, NVR, Access Control, etc.) and client software/system. It defines the communication standard between device/server and client software/system via the Internet Protocol (IP).
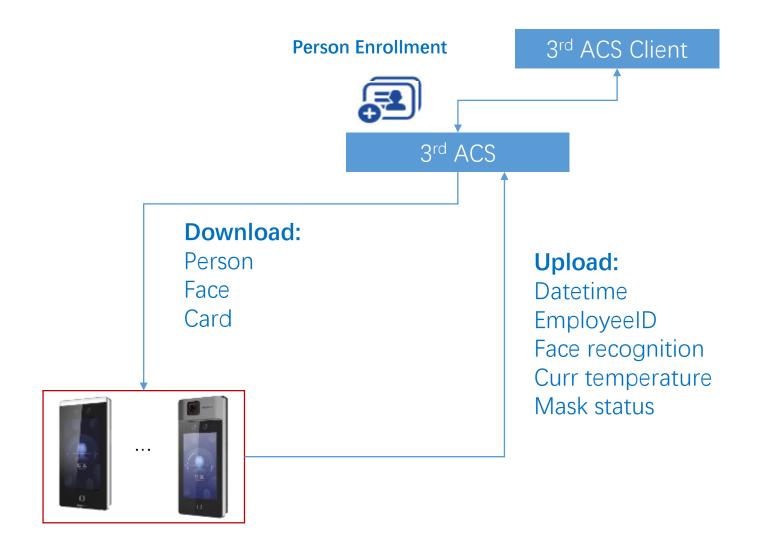
How to get ISAPI document

Our partners need to register in TPP website(https://tpp.hikvision.com/) to sign MLA or apply by OA process

Before you integrate with terminal, you need to prepare the following tools:

➢ Postman(*https://www.postman.com/downloads/*)

➢ Wireshark

➢ Notepad++(optional) (*https://notepad-plus.en.softonic.com/*)
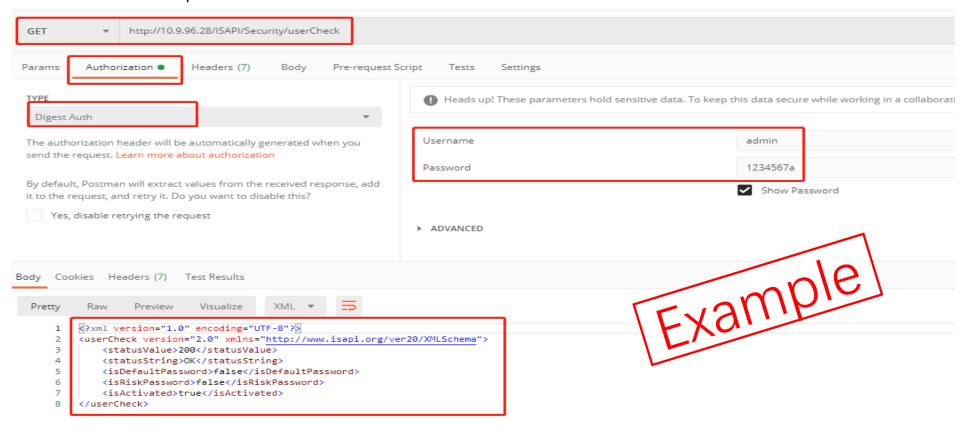
What is the terminal logic

Person Enrollment

3rd ACS Client

3rd ACS

Download:
Person
Face
Card

Upload:
Datetime
EmployeeID
Face recognition
Curr temperature
Mask status

...

How to digest login

1. You need to login the terminal by digest, you can refer to the **chapter4 Login** in **General** document.
2. About the digest authentication, you can refer to **chapter 3.1 Authentication**(General document). see *https://tools.ietf.org/html/rfc2617* for details.

You also can use postman to test as below:
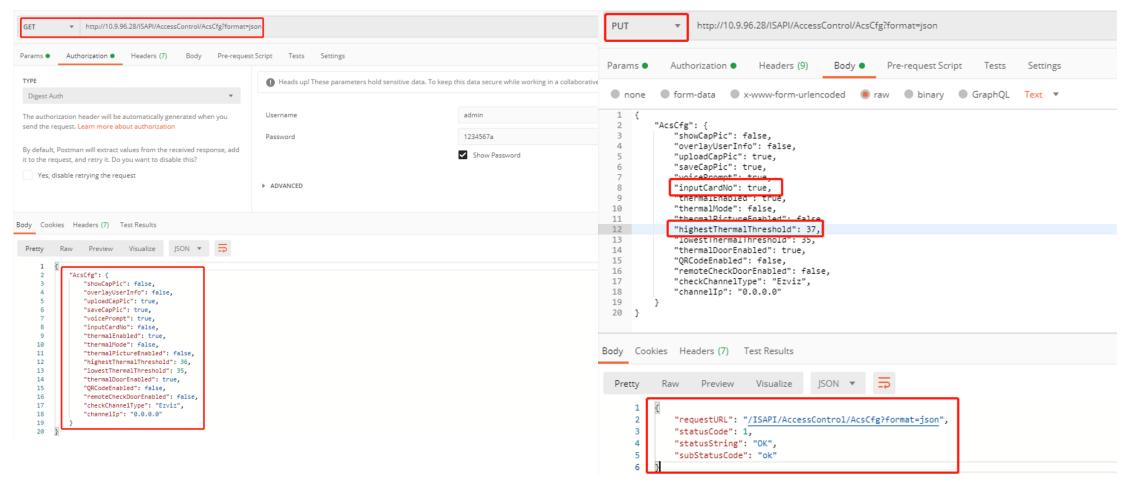
How to use postman for more functions

After login the device, we can use postman to test more functions, for example we can GET/PUT access controller parameters.

GET method:
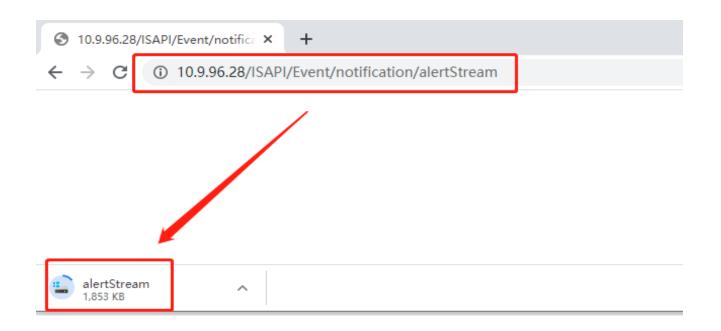
PUT method:

When you use PUT or POST method, please pay attention to the body of method in document.

How to get real time events
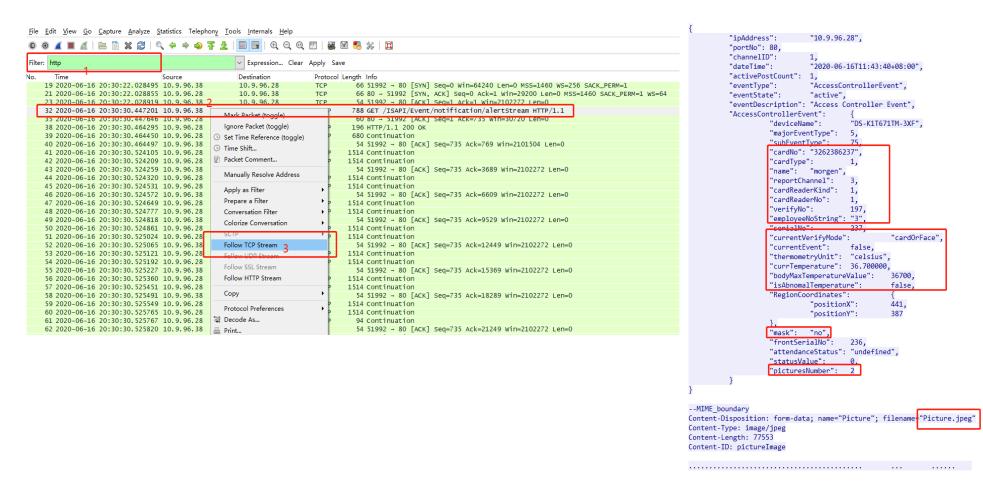
If you want to get the access events contain temperature and mask information, terminal only can use **arming mode** to get the events. You can refer to **the chapter 4.12.4 Receive Alarm/event in Arming Mode**(Access Control on Person document). You can use */ISAPI/Event/notification/alertStream* to receive alarm/event, but can't test through postman. You can use Wireshark or Chrome to test arming mode.

You can use notepad++ to open the real time data flow, it will never stop, unless you close the browser.
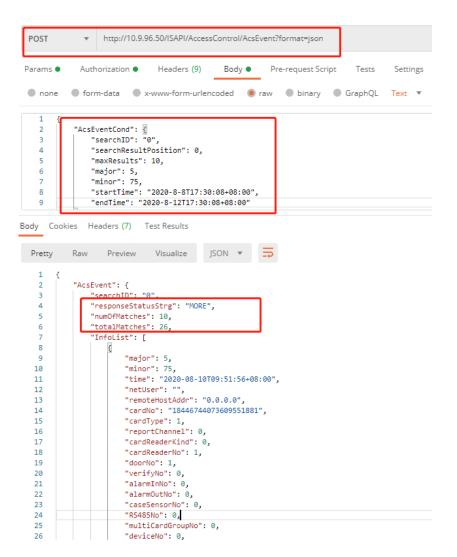
How to get real time events

You also can use Wireshark to follow TCP stream, if the person was registered, you can get name/card no /employee no and so on, device also will upload 2 pictures, one for visible light, another for thermal, both of them are binary data without encryption, you can translate it to JPEG directly. And the data upload by device in Json format

```
{
        "ipAddress":        "10.9.96.28",
        "portNo": 80,
        "channelID":        1,
        "dateTime":        "2020-06-16T11:43:40+08:00",
        "activePostCount":  1,
        "eventType":        "AccessControllerEvent",
        "eventState":       "active",
        "eventDescription": "Access Controller Event",
        "AccessControllerEvent":        {
                "deviceName":       "DS-K1T671TM-3XF",
                "majorEventType":   5,
                "subEventType":     75,
                "cardNo":   "3262386237",
                "cardType":         1,
                "name":    "morgen",
                "reportChannel":    3,
                "cardReaderKind":   1,
                "cardReaderNo":     1,
                "verifyNo":         197,
                "employeeNoString": "3",
                "serialNo":         237,
                "currentVerifyMode":        "cardOrFace",
                "currentEvent":     false,
                "thermometryUnit":  "celsius",
                "currTemperature":  36.700000,
                "bodyMaxTemperatureValue":  36700,
                "isAbnomalTemperature":     false,
                "RegionCoordinates":        {
                        "positionX":        441,
                        "positionY":        387
                },
                "mask":    "no",
                "frontSerialNo":    236,
                "attendanceStatus": "undefined",
                "statusValue":      0,
                "picturesNumber":   2
        }
}

--MIME_boundary
Content-Disposition: form-data; name="Picture"; filename="Picture.jpeg"
Content-Type: image/jpeg
Content-Length: 77553
Content-ID: pictureImage
```

How to get history events

You also can use **/ISAPI/AccessControl/AcsEvent?format=json** to get the history records, please note no all the model of terminal support this functions.
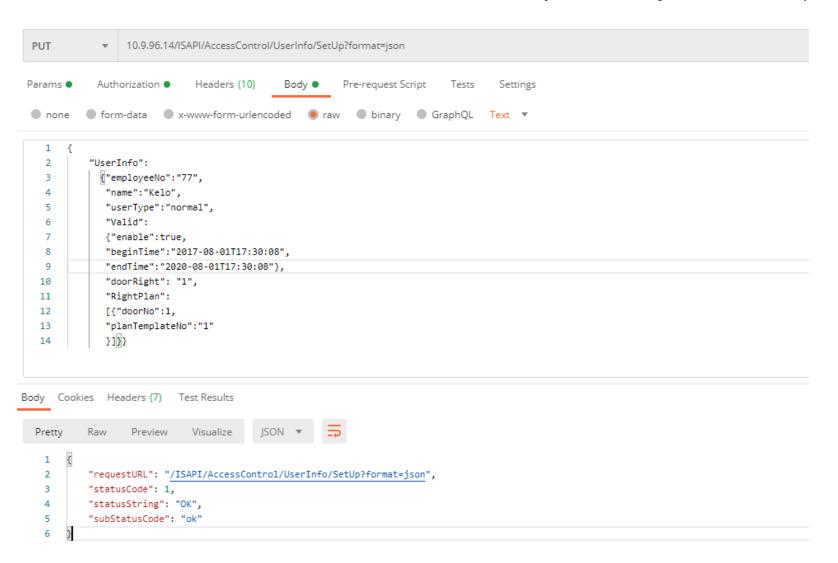
Notes:
- we recommend you set the max results = 10 in every one page.
- If there are many pages, please set the search result position when you search for other pages.
- The history records didn't contain temperature info and mask status in latest firmware.

How to add person to terminal

You can use **PUT /ISAPI/AccessControl/UserInfo/SetUp?format=json** to add a person.

PUT	▼	10.9.96.14/ISAPI/AccessControl/UserInfo/SetUp?format=json

Params ●   Authorization ●   Headers (10)   Body ●   Pre-request Script   Tests   Settings

● none   ● form-data   ● x-www-form-urlencoded   ◉ raw   ● binary   ● GraphQL   Text ▼

```
1   {
2       "UserInfo":
3       {"employeeNo":"77",
4        "name":"Kelo",
5        "userType":"normal",
6        "Valid":
7        {"enable":true,
8        "beginTime":"2017-08-01T17:30:08",
9        "endTime":"2020-08-01T17:30:08"},
10       "doorRight": "1",
11       "RightPlan":
12       [{"doorNo":1,
13        "planTemplateNo":"1"
14       }]}}
```

Body   Cookies   Headers (7)   Test Results

Pretty   Raw   Preview   Visualize   JSON ▼

```
1   {
2       "requestURL": "/ISAPI/AccessControl/UserInfo/SetUp?format=json",
3       "statusCode": 1,
4       "statusString": "OK",
5       "subStatusCode": "ok"
6   }
```

How to add face to person

You can use **/ISAPI/Intelligent/FDLib/FaceDataRecord?format=json** to add a person. You can add a face on your picture library by faceURL, or add by binary(you need transfer JPG format to binary).
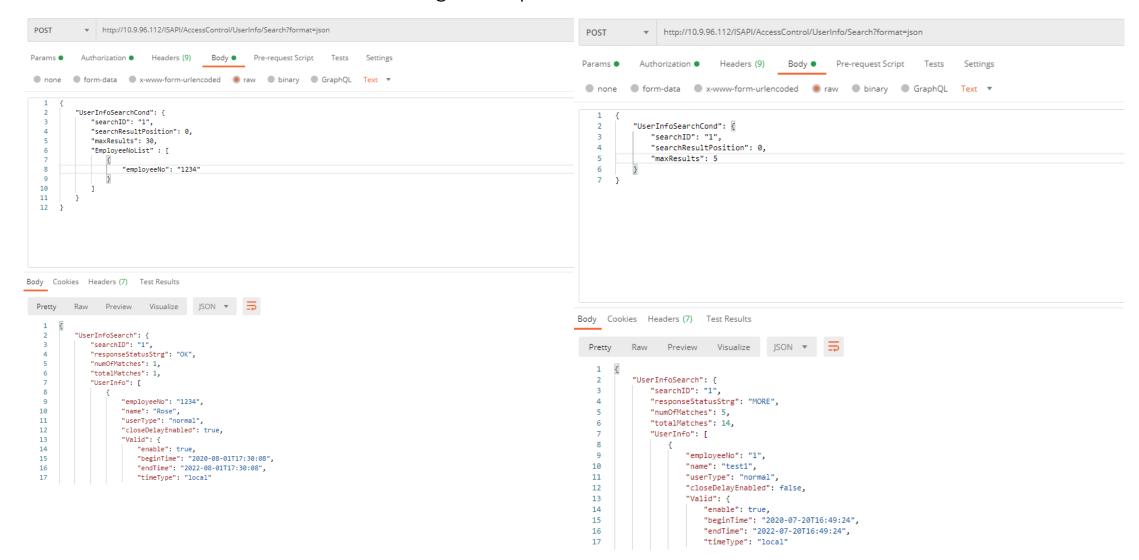
Notes:
FPID is the employeeNo.

When enrolling face pictures, it requires pupillary distance to be more than 60 pixels, up, down, left and right deflection angle of person's pose should be less than 10 degrees, format od picture should be JPG, size should be less than 200KB.

How to search one/all person

You can use **/ISAPI/AccessControl/UserInfo/Search?format=json** to search person. if the EmploeeNo not exist or is set to NULL, it indicates searching for all person information.
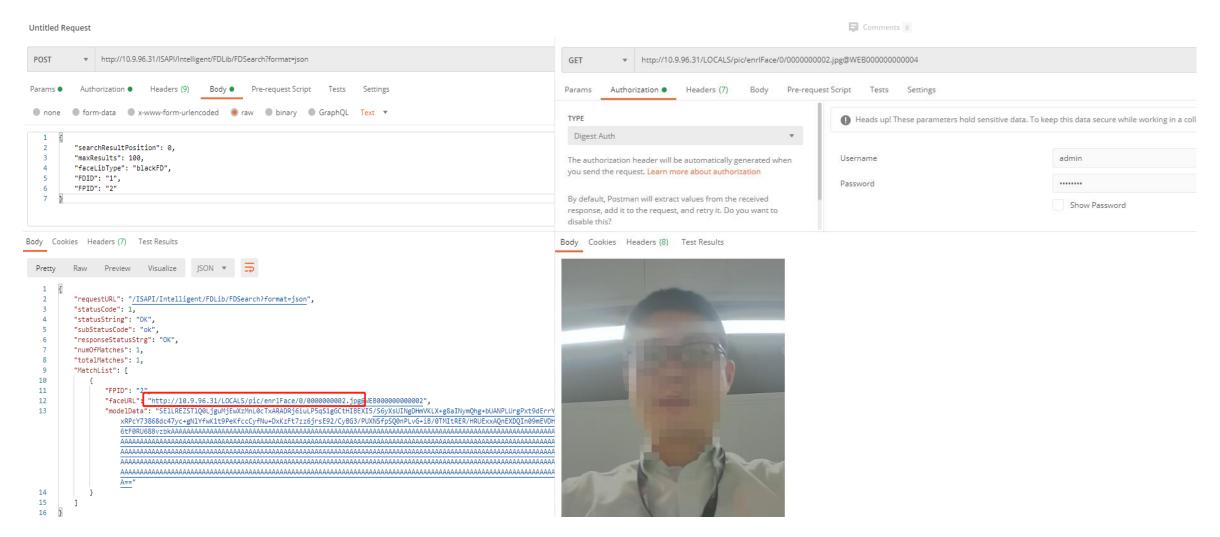
POST ▼ http://10.9.96.112/ISAPI/AccessControl/UserInfo/Search?format=json

Params ● | Authorization ● | Headers (9) | Body ● | Pre-request Script | Tests | Settings

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL | Text ▼

```
1  {
2      "UserInfoSearchCond": {
3          "searchID": "1",
4          "searchResultPosition": 0,
5          "maxResults": 30,
6          "EmployeeNoList" : [
7              {
8                  "employeeNo": "1234"
9              }
10         ]
11     }
12 }
```

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON ▼

```
1  {
2      "UserInfoSearch": {
3          "searchID": "1",
4          "responseStatusStrg": "OK",
5          "numOfMatches": 1,
6          "totalMatches": 1,
7          "UserInfo": [
8              {
9                  "employeeNo": "1234",
10                 "name": "Rose",
11                 "userType": "normal",
12                 "closeDelayEnabled": true,
13                 "Valid": {
14                     "enable": true,
15                     "beginTime": "2020-08-01T17:30:08",
16                     "endTime": "2022-08-01T17:30:08",
17                     "timeType": "local"
```

POST ▼ http://10.9.96.112/ISAPI/AccessControl/UserInfo/Search?format=json

Params ● | Authorization ● | Headers (9) | Body ● | Pre-request Script | Tests | Settings

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL | Text ▼

```
1  {
2      "UserInfoSearchCond": {
3          "searchID": "1",
4          "searchResultPosition": 0,
5          "maxResults": 5
6      }
7  }
```

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON ▼

```
1  {
2      "UserInfoSearch": {
3          "searchID": "1",
4          "responseStatusStrg": "MORE",
5          "numOfMatches": 5,
6          "totalMatches": 14,
7          "UserInfo": [
8              {
9                  "employeeNo": "1",
10                 "name": "test1",
11                 "userType": "normal",
12                 "closeDelayEnabled": false,
13                 "Valid": {
14                     "enable": true,
15                     "beginTime": "2020-07-20T16:49:24",
16                     "endTime": "2022-07-20T16:49:24",
17                     "timeType": "local"
```
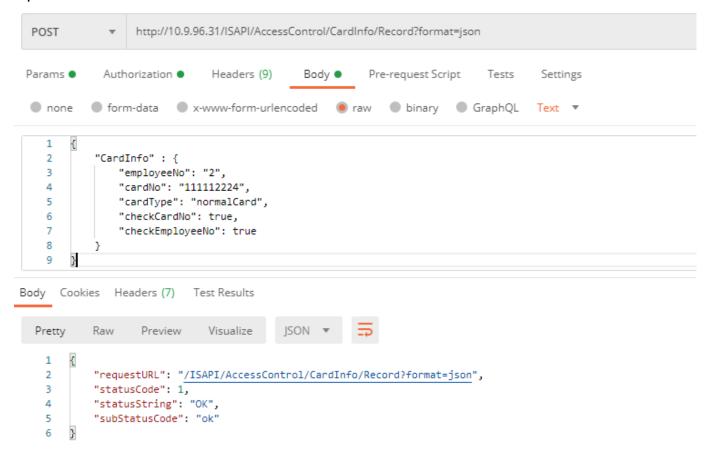
How to get face form device

You can use **/ISAPI/Intelligent/FDLib/FDSearch?format=json** to get the face in terminal.
FPID is the employeeNo, you can get the pic directly via the pic URL, if the network is well.

How to add card to person

You can use **/ISAPI/AccessControl/CardInfo/Record?format=json** to add a card. If the device is based on card device, one person only can add one card, for devices which are based on person, one person can add multiple cards.

| POST | ▼ | http://10.9.96.31/ISAPI/AccessControl/CardInfo/Record?format=json |

Params ● Authorization ● Headers (9) Body ● Pre-request Script Tests Settings

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL Text ▼

```
1  {
2      "CardInfo" : {
3          "employeeNo": "2",
4          "cardNo": "111112224",
5          "cardType": "normalCard",
6          "checkCardNo": true,
7          "checkEmployeeNo": true
8      }
9  }
```

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON ▼

```
1  {
2      "requestURL": "/ISAPI/AccessControl/CardInfo/Record?format=json",
3      "statusCode": 1,
4      "statusString": "OK",
5      "subStatusCode": "ok"
6  }
```
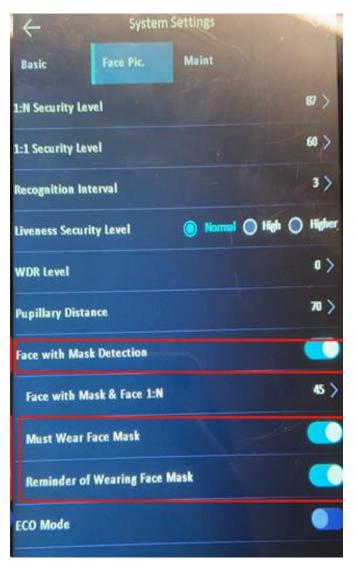
Notes:
 Delete card:
/ISAPI/AccessControl/CardInfo/Delete?format=json

How to GET/PUT mask strategy

You can use **/ISAPI/AccessControl/maskDetection?format=json** to GET/PUT the mask strategy.


```
1  {
2      "MaskDetection": {
3          "enable": true,
4          "noMaskStrategy": "tipsAndNotOpenDoor"
5      }
6  }
```

Terminal display		Command parameter
Face with mask Detection(enable)		enable:true
Must wear mask(en)	Reminder of wear(en)	tipsAndNotOpenDoor
Must wear mask(en)	Reminder of wear(dis)	tipsAndNotOpenDoor
Must wear mask(dis)	Reminder of wear(en)	tipsAndOpenDoor
Must wear mask(dis)	Reminder of wear(dis)	noTipsAndOpenDoor

How to set temperature strategy

You can use **/ISAPI/AccessControl/AcsCfg?format=json** to GET/PUT the temperature strategy.

Temperature Settings

Enable Temperature Detection — Thermal enable

Over-Temperature Alarm Threshold (Max.) 37.3 — Highest thermal threshold

Over-Temperature Alarm Threshold (Min.) 35.0 — lowest thermal threshold

Temperature Compensation 0.00

Door Not Open When Temperature is Abnormal — Thermal door enable

Temperature Measurement Only — Thermal mode

Unit ℃ ℉

Measurement Area Calibration

Measurement Area Settings

Black Body Settings

GET http://10.9.96.50/ISAPI/AccessControl/AcsCfg?format=json

Params ● | Authorization ● | Headers (7) | Body | Pre-request Script

TYPE

Digest Auth

The authorization header will be automatically generated when you send the request. Learn more about authorization

By default, Postman will extract values from the received response, add it to the request, and retry it. Do you want to disable this?

Body | Cookies | Headers (7) | Test Results

Pretty | Raw | Preview | Visualize | JSON

```
1  {
2      "AcsCfg": {
3          "showCapPic": false,
4          "overlayUserInfo": false,
5          "uploadCapPic": true,
6          "saveCapPic": true,
7          "voicePrompt": true,
8          "inputCardNo": false,
9          "thermalEnabled": true,
10         "thermalMode": false,
11         "thermalPictureEnabled": false,
12         "highestThermalThreshold": 37.300000,
13         "lowestThermalThreshold": 35,
14         "thermalDoorEnabled": true,
15         "QRCodeEnabled": false,
16         "remoteCheckDoorEnabled": false,
17         "checkChannelType": "ISAPI",
18         "channelIp": ""
19     }
```