

CGI API

ID: RK-SM-YF-383

Release Version: V1.0.1

Release Date: 2021-03-15

Security Level: ☐Top-Secret ☐Secret ☐Internal ☒Public

DISCLAIMER

THIS DOCUMENT IS PROVIDED "AS IS". ROCKCHIP ELECTRONICS CO., LTD. ("ROCKCHIP") DOES NOT PROVIDE ANY WARRANTY OF ANY KIND, EXPRESSED, IMPLIED OR OTHERWISE, WITH RESPECT TO THE ACCURACY, RELIABILITY, COMPLETENESS, MERCHANTABILITY, FITNESS FOR ANY PARTICULAR PURPOSE OR NON-INFRINGEMENT OF ANY REPRESENTATION, INFORMATION AND CONTENT IN THIS DOCUMENT. THIS DOCUMENT IS FOR REFERENCE ONLY. THIS DOCUMENT MAY BE UPDATED OR CHANGED WITHOUT ANY NOTICE AT ANY TIME DUE TO THE UPGRADES OF THE PRODUCT OR ANY OTHER REASONS.

Trademark Statement

"Rockchip", "瑞芯微", "瑞芯" shall be Rockchip's registered trademarks and owned by Rockchip. All the other trademarks or registered trademarks mentioned in this document shall be owned by their respective owners.

All rights reserved. ©2021. Rockchip Electronics Co., Ltd.

Beyond the scope of fair use, neither any entity nor individual shall extract, copy, or distribute this document in any form in whole or in part without the written approval of Rockchip.

Rockchip Electronics Co., Ltd.

No.18 Building, A District, No.89, software Boulevard Fuzhou, Fujian, PRC

Website: www.rock-chips.com

Customer service Tel: +86-4007-700-590

Customer service Fax: +86-591-83951833

Customer service e-Mail: fae@rock-chips.com

Preface

Overview

This document provides an introduction to API input and output of CGI program.

Product Version

Chipset	Kernel Version
RV1126, RV1109	Linux 4.19
RK1808, RK1806	Linux 4.4

Intended Audience

This document (this guide) is mainly intended for:

Technical support engineers

Software development engineers

Revision History

Version	Author	Date	Change Description
V1.0.0	Allen Chen	2020-08-29	Initial version
V1.0.1	Ruby Zhang	2021-03-15	Update product version

Contents

CGI API

1. Notice
2. Functions Index
3. System
 - 3.1 login
 - 3.2 device-info
 - 3.3 remain-space
 - 3.4 para
 - 3.5 firmware-upgrade
 - 3.6 reboot
 - 3.7 factory-reset
 - 3.8 export-log
 - 3.9 import-db
4. video
 - 4.1 region-clip
 - 4.2 advanced-enc
5. stream-url
6. storage
 - 6.1 hdd-list
 - 6.2 quota
 - 6.3 snap-plan
 - 6.4 current-path
 - 6.5 format
 - 6.6 search
 - 6.7 advance-para
 - 6.8 delete
7. roi
8. peripherals
 - 8.1 gate
 - 8.2 fill-light
9. osd
 - 9.1 overlays
 - 9.2 image
 - 9.3 privacy-mask
10. network
 - 10.1 lan
 - 10.2 wlan
 - 10.3 Wi-Fi
 - 10.4 wifi-list
11. network-ntp
 - 11.1 time
12. network-port
13. image
 - 13.1 id
14. event
 - 14.1 triggers
 - 14.2 schedules
 - 14.3 motion-detection
 - 14.4 regional-invasion
 - 14.5 face-list
 - 14.6 face
 - 14.7 face-config
 - 14.8 face-waiting
 - 14.9 smart
 - 14.10 get-record-status

- 14.11 last-face
- 14.12 snapshot-record
- 14.13 control-record
- 14.14 check-face
- 14.15 reset-face
- 14.16 reset-snap
- 14.17 reset-control
- 14.18 face-picture
- 14.19 take-photo
- 14.20 start-record
- 14.21 stop-record

15. audio

16. Trouble Shooting

- 16.1 401
- 16.2 500
- 16.3 501

1. Notice

1. The URLs provided in this document are all without <http://{IP}/cgi-bin/entry.cgi>, and it should be added during actual usage.
2. Except for /system/login and /system/para/webPage, access to other URLs requires login to return cookies.
3. The request provides examples of the information to be carried in the request body, and examples of the returned result is provided in the response.
4. Whether the API is valid for the device, please refer to the document "Rockchip_Instructions_Linux_Web_Configuration" in docs/Linux/ApplicationNote/, to check whether the corresponding function is valid.

2. Functions Index

Functions	Description
Login	Login and get cookies.
User Management	User registration, modification and deletion.
Get video stream	Get RTSP/RTMP URL of each stream.
Reboot	Reboot the device.
Restore factory settings	Restore the database to factory settings.
Remote Upgrade	<ol style="list-style-type: none">1. Apply for a breakpoint continual transfer id;2. Check whether the rest of capacity meets upgrade requirements;3. Upgrade file though breakpoint continual transferring;4. Send the completion signal;5. Send the query signal to check whether the upgrade is completed;6. Delete the upgrade file.
Get/Configure Video Encoding	Get and configure video encoding parameters, such as resolution, image quality, and encoding.
Get/Configure LAN	Get and configure LAN, such as IP, subnet mask.
WiFi Configuration	<ol style="list-style-type: none">1. WiFi Basic Configuration;2. WiFi Information and Connection.
Port Settings	Get and configure the device port.
Time Setting	Get and configure device time, such as time zone and device time.
ISP Configuration	Get/set isp, including image adjustment, exposure, day and night conversion, high dynamic, white balance, image enhancement, video adjustment and other settings.
OSD Characters	OSD character style, obtain and configure character content.

Funtions	Description
OSD privacy mask	Obtain and configure OSD privacy mask parameters.
OSD picture mask	<ol style="list-style-type: none"> 1. Upload the mask picture; 2. Get the parameter used to set picture masking.
Audio Settings	Obtain and set audio parameters.
Screenshot	<ol style="list-style-type: none"> 1. Screenshot of main stream in real-time; 2. Search the screenshot record.
Recording	<ol style="list-style-type: none"> 1. Start/Stop recording of main stream in real-time ; 2. Search recording; 3. Recording schedule configuration; <p>Note: If the real-time recording conflicts with the schedule, it will be stopped.</p>
Face registration	<ol style="list-style-type: none"> 1. Start face detection and face recognition; 2. Upload registration information and get the photo upload address (API: /event/face, with the note information of undone); 3. Face pictures upload; 4. Confirmation of uploading face pictures (API: /event/face, with empty note information); <p>Note: registration in large quantity requires search the number of pictures waiting for processing, when the pictures waiting for processing are greater than 10, need to upload after buff processing.</p>
Face Information Search/Management	<ol style="list-style-type: none"> 1. Registered Members List; 2. Snapshot Record 3. Control Record

3. System

3.1 login

```
1  # GET /system/login: Get login user information
2  # response iAuthLevel: Authentication level (it is not used yet),
   iUserLevel: User authority (0: Administrator, 1: Operator, 2: Ordinary
   users), id: Database id, sUserName: User name
3  [
4      {
5          "iAuthLevel": 1,
6          "iUserLevel": 0,
7          "id": 0,
8          "sUserName": "admin"
9      }
10 ]
11
12 # PUT/POST /system/login?expire: User login, the effective time of cookie is
   one hour
13 # PUT/POST /system/login?expire=day: User login, the effective time of
   cookie is one day
14 # PUT/POST /system/login?expire=week: User login, the effective time of
   cookie is one week
15 # PUT/POST /system/login?expire=month: User login, the effective time of
   cookie is one month
16 # request sUserName: Username, sPassword: Password (using base64 encryption)
17 {
18     "sUserName": "admin",
19     "sPassword": "YWRtaW4="
20 }
21 # response auth: User authority, status: Login status (greater than or equal
   to 0 represents success, -1 represents password is error, -2 represents the
   user does not exist)
22 # After successful login, the set-cookie will carry the cookie required for
   authentication
23 {
24     "auth": 0,
25     "status": 0
26 }
27
28 # PUT/POST /system/login/add: Add user password
29 # request sUserName: Administrator user name, sPassword: Administrator
   password (using base64 encryption), newUserName: New user name, newPassword:
   New user password (using base64 encryption), secondNewPw: Double check
   password (using base64 encryption), iUserLevel: User authority
30 {
31     "sUserName": "admin",
32     "sPassword": "YWRtaW4=",
33     "newUserName": "test",
34     "newPassword": "dGVzdA==",
35     "secondNewPw": "dGVzdA==",
36     "iUserLevel": 1
37 }
```



```

37 }
38 # response status: Registration status (-2 indicates registration is
    successful, >=0 indicates user name is repeated)
39 {"status":-2}
40
41 # PUT/POST /system/login/modify: User password modification
42 # request: Is the same with /system/login/add
43 {
44     "sUserName":"admin",
45     "sPassword":"YWRtaW4=",
46     "newUserName":"test",
47     "newPassword":"dGVzdA==",
48     "secondNewPw":"dGVzdA==",
49     "iUserLevel":1
50 }
51 # response status: Modification status (>=0 indicates registration is
    successful, -2 indicates username does not exist)
52 {"status":2}
53
54 # PUT/POST /system/login/delete: Delete by user (administrator cannot
    delete)
55 # request administrator user name, sPassword: Administrator password (using
    base64 encryption), newUserName: Delete user name
56 {
57     "sUserName":"admin",
58     "sPassword":"YWRedWI=",
59     "newUserName":"test"
60 }
61 # response status: Modification status (>0 indicates deleted successfully,
    -2 indicates username does not exist)
62 {"status":2}

```

3.2 device-info

```

1 # GET /system/device-info: Get device information
2 # response name: The name of the information, value: The value of the
    information , ro: Read-only attribute
3 [
4     {
5         "id": 0,
6         "name": "deviceName",
7         "ro": "false",
8         "value": "RK IP Camera"
9     },
10    {
11        "id": 1,
12        "name": "telecontrolID",
13        "ro": "false",
14        "value": "88"
15    },
16    {
17        "id": 2,
18        "name": "model",
19        "ro": "true",
20        "value": "RK-003"

```

```
21     },
22     {
23         "id": 3,
24         "name": "serialNumber",
25         "ro": "true",
26         "value": "RK-003-A"
27     },
28     {
29         "id": 4,
30         "name": "firmwareVersion",
31         "ro": "true",
32         "value": "V0.2.6 build 200413"
33     },
34     {
35         "id": 5,
36         "name": "encoderVersion",
37         "ro": "true",
38         "value": "V1.0 build 200413"
39     },
40     {
41         "id": 6,
42         "name": "webVersion",
43         "ro": "true",
44         "value": "V1.12.2 build 200413"
45     },
46     {
47         "id": 7,
48         "name": "pluginVersion",
49         "ro": "true",
50         "value": "V1.0.0.0"
51     },
52     {
53         "id": 8,
54         "name": "channelsNumber",
55         "ro": "true",
56         "value": "1"
57     },
58     {
59         "id": 9,
60         "name": "hardDisksNumber",
61         "ro": "true",
62         "value": "1"
63     },
64     {
65         "id": 10,
66         "name": "alarmInputsNumber",
67         "ro": "true",
68         "value": "0"
69     },
70     {
71         "id": 11,
72         "name": "alarmOutputsNumber",
73         "ro": "true",
74         "value": "0"
75     },
76     {
77         "id": 12,
78         "name": "firmwareVersionInfo",
```

```

79         "ro": "true",
80         "value": "CP-3-B"
81     },
82     {
83         "id": 13,
84         "name": "manufacturer",
85         "ro": "true",
86         "value": "Rockchip"
87     },
88     {
89         "id": 14,
90         "name": "hardwareId",
91         "ro": "true",
92         "value": "c3d9b8674f4b94f6"
93     }
94 ]
95
96 # PUT/POST /system/device-info: Modify device information
97 # request: Modify data unit information
98 {
99     "id":1,
100     "name":"telecontrolID",
101     "value":"88",
102     "ro":"false"
103 }
104 # response: data unit information after modification
105 {
106     "id":1,
107     "name":"telecontrolID",
108     "value":"88",
109     "ro":"false"
110 }

```

3.3 remain-space

```

1 # GET /system/remain-space: Get the rest of space in userdata of device
2 # response: The rest of space in bytes
3 {
4     "availableDisk": 1026608128
5 }

```

3.4 para

```

1 # GET /system/remain-space/key: Get device capability set, key is the name in
  the database SystemPara
2 # response: Is the ability set json characters (that is, the para in the
  database SystemPara), which can be converted to json objects

```

3.5 firmware-upgrade

```
1  # GET /system/firmware-upgrade?upload-type=resumable: Breakpoint continual
   transferring id application
2  # reueset
3  null
4  # response the id requested in Headers/X-Location
5  Headers: X-Location: http://{IP}}/cgi-bin/entry.cgi/system/firmware-
   upgrade?id=0
6
7  # POST/PUT /system/firmware-upgrade?id=0: Upgrade breakpoint continual
   transferring, id is the file number
8  # reueset
9  Headers: Content-Range: bytes 524288-1048575 # Start and end position of
   file
10 Headers: Content-Type: text/plain
11 Body: text/plain data, the size is less than 1M
12 # response: Write status of current file
13 {"range":"bytes 0-1572863"}
14
15 # POST/PUT /system/firmware-upgrade?start=id: The id is a number,
   representing the breakpoint continual transferring id of the upgrade file,
   and start system upgrading
16 # reueset
17 null
18 # response: There will be no response, after the timeout, wait for the
   upgrade to complete
19
20 # DELETE /system/firmware-upgrade?id=0: Delete the upgrade file
```

3.6 reboot

```
1  # POST/PUT /system/reboot: Reboot
2  # reueset
3  null
```

3.7 factory-reset

```
1  # POST/PUT /system/factory-reset: Restore factory settings
2  # reueset
3  null
```

3.8 export-log

```

1 # POST/PUT /system/export-log: Export log
2 # reueset
3 null
4 # response log address
5 {"location":"http://172.16.21.106/userdata/export.log"}

```

3.9 import-db

```

1 # POST/PUT /system/import-db: Upload database
2 # reueset: the data type is multipart/form-data
3 Header:content-Type: multipart/form-data
4 Form Data: file data
5 # response
6 {}
7
8 # POST/PUT /system/import-db?start=1: Start importing the database and
  restart
9 # reueset
10 null

```

4. video

```

1 # GET /video: Get all video encoding configuration
2 #response iGOP: I frame interval, iMaxRate: The maximum vaule of bit rate,
  iMinRate: The minimum vaule of bit rate, iStreamSmooth: Stream smoothing,
  TargetRate: Target bit rate, sFrameRate: Video frame rate, sFrameRateIn:
  Input frame rate, sH264Profile: Encoding complexity, sOutputDataType: Video
  encoding, sRCMode: Bit rate type, sRCQuality: Image quality, sResolution:
  Resolution, SVC: SVC switch, Smart: Smart switch, sStreamType: Stream type,
  sVideoType: Video type, the rest are data tokens
3 [
4   {
5     "iGOP": 50,
6     "iMaxRate": 8192,
7     "iMinRate": 0,
8     "iStreamSmooth": 50,
9     "iTargetRate": 0,
10    "id": 0,
11    "sFrameRate": "25",
12    "sFrameRateIn": "25",
13    "sH264Profile": "high",
14    "sOutputDataType": "H.265",
15    "sRCMode": "CBR",
16    "sRCQuality": "high",
17    "sResolution": "2688*1520",
18    "sSVC": "close",
19    "Smart": "close",
20    "sStreamType": "mainStream",
21    "sVideoEncoderConfigurationName": "VideoEncoder_0",
22    "sVideoEncoderConfigurationToken": "VideoEncoderToken_0",

```

```

23         "sVideoSourceToken": "VideoSource_0",
24         "sVideoType": "compositeStream"
25     },
26     {
27         "iGOP": 50,
28         "iMaxRate": 1024,
29         "iMinRate": 0,
30         "iStreamSmooth": 50,
31         "iTargetRate": 0,
32         "id": 1,
33         "sFrameRate": "25",
34         "sFrameRateIn": "25",
35         "sH264Profile": "high",
36         "sOutputDataType": "H.264",
37         "sRCMode": "CBR",
38         "sRCQuality": "high",
39         "sResolution": "640*480",
40         "sSVC": "close",
41         "sSmart": "close",
42         "sStreamType": "subStream",
43         "sVideoEncoderConfigurationName": "VideoEncoder_1",
44         "sVideoEncoderConfigurationToken": "VideoEncoderToken_1",
45         "sVideoSourceToken": "VideoSource_0",
46         "sVideoType": "compositeStream"
47     },
48     {
49         "iGOP": 50,
50         "iMaxRate": 2048,
51         "iMinRate": 0,
52         "iStreamSmooth": 50,
53         "iTargetRate": 0,
54         "id": 2,
55         "sFrameRate": "25",
56         "sFrameRateIn": "25",
57         "sH264Profile": "high",
58         "sOutputDataType": "H.265",
59         "sRCMode": "CBR",
60         "sRCQuality": "high",
61         "sResolution": "1920*1080",
62         "sSVC": "close",
63         "sSmart": "close",
64         "sStreamType": "thirdStream",
65         "sVideoEncoderConfigurationName": "VideoEncoder_2",
66         "sVideoEncoderConfigurationToken": "VideoEncoderToken_2",
67         "sVideoSourceToken": "VideoSource_0",
68         "sVideoType": "compositeStream"
69     }
70 ]
71
72 # GET /video/id: Get the video encoding configuration of corresponding id,
the id must be a number in 0-2
73 # response: Return single id data in GET /video
74
75 # POST/PUT /video/id: Configure the corresponding id video encoding
parameters, id must be a number in 0-2
76 # request: Pass in the json of above data unit
77 # response: Get the json of data unit after setting

```

4.1 region-clip

```
1  # GET /video/2/region-clip: Get the region cropping configuration
2  # response normalizedScreenSize: Normalized size, regionClip: Region
   cropping parameters
3  {
4      "normalizedScreenSize": {
5          "iNormalizedScreenHeight": 480,
6          "iNormalizedScreenWidth": 704
7      },
8      "regionClip": {
9          "iHeight": 480,
10         "iPositionX": 0,
11         "iPositionY": 0,
12         "iRegionClipEnabled": 0,
13         "iWidth": 640
14     }
15 }
```

4.2 advanced-enc

```
1  # GET /video/0/advanced-enc: Get the default value of advanced encoding
   parameters
2  # response
3  [
4      {
5          "id": 0,
6          "sFunction": "qp",
7          "sParameters": "{\
            \"qp_init\":24,\
            \"qp_step\":4,\
            \"qp_min\":12,\
            \"qp_max\":48,\
            \"min_i_qp\":10,\
            \"max_i_qp\":20\
        }",
8          "sStreamType": "mainStream"
9      },
10     {
11         "id": 1,
12         "sFunction": "split",
13         "sParameters": "{\
            \"mode\":0,\
            \"size\":1024\
        }",
14         "sStreamType": "mainStream"
15     }
16 ]
```

5. stream-url

```
1  # GET /video/stream-url: Get the address of the RTSP/RTMP video stream
2  # response
3  [
4      {
5          "id":0,
6          "sStreamProtocol":"RTSP",
```

```

7         "sURL":"rtsp://172.16.21.106:554/mainstream"
8     },
9     {
10         "id":1,
11         "sStreamProtocol":"RTMP",
12         "sURL":"rtmp://172.16.21.106:1935/live/substream"
13     },
14     {
15         "id":2,
16         "sStreamProtocol":"RTMP",
17         "sURL":"rtmp://172.16.21.106:1935/live/thirdstream"
18     }
19 ]
20
21 # POST/PUT /video/stream-url/id: Set video streaming protocol
22 # request
23 {
24     "sStreamProtocol":"RTSP",
25 }
26 # response
27 {
28     "id":0,
29     "sStreamProtocol":"RTSP",
30     "sURL":"rtsp://172.16.21.106:554/mainstream"
31 }

```

6. storage

6.1 hdd-list

```

1 # GET /storage/hdd-list: Get all disk information
2 # response
3 [
4     {
5         "iFormatProg":0,
6         "iFormatStatus":0,
7         "iFreeSize":0,
8         "iMediaSize":0,
9         "iTotalSize":0,
10        "id":1,
11        "sDev":"",
12        "sFormatErr":"",
13        "sMountPath":"/mnt/sdcard",
14        "sName":"SD Card",
15        "sStatus":"unmounted",
16        "sType":""
17    },
18    {
19        "iFormatProg":0,
20        "iFormatStatus":0,
21        "iFreeSize":12.0438613891602,
22        "iMediaSize":60972,

```



```

23         "iTotalSize":12.1327171325684,
24         "id":3,
25         "sAttributes":"rw",
26         "sDev":"/dev/block/by-name/media",
27         "sFormatErr":"",
28         "sMountPath":"/userdata/media",
29         "sName":"Emmc",
30         "sStatus":"mounted",
31         "sType":"ext2"
32     },
33     {
34         "iFormatProg":0,
35         "iFormatStatus":0,
36         "iFreeSize":0,
37         "iMediaSize":0,
38         "iTotalSize":0,
39         "id":2,
40         "sDev":"",
41         "sFormatErr":"",
42         "sMountPath":"/media/usb0",
43         "sName":"U Disk",
44         "sStatus":"unmounted",
45         "sType":""
46     }
47 ]
48
49 # GET /storage/hdd-list/id: The id is a number, get the disk information
with the corresponding id
50 # response
51 {
52     "iFormatProg":0,
53     "iFormatStatus":0,
54     "iFreeSize":0,
55     "iMediaSize":0,
56     "iTotalSize":0,
57     "id":2,
58     "sDev":"",
59     "sFormatErr":"",
60     "sMountPath":"/media/usb0",
61     "sName":"U Disk",
62     "sStatus":"unmounted",
63     "sType":""
64 }

```

6.2 quota

```

1  # GET /storage/quota: Get disk quota information
2  # response
3  [
4      {
5          "iFreePictureQuota": 0.0,
6          "iFreeVideoQuota": 0.0,
7          "iPictureQuotaRatio": 5,
8          "iTotalPictureVolume": 0.0,
9          "iTotalVideoVolume": 0.0,

```

```

10         "iVideoQuotaRatio": 45,
11         "id": 1
12     },
13     {
14         "iFreePictureQuota": 0.595100224018097,
15         "iFreeVideoQuota": 5.42329835891724,
16         "iPictureQuotaRatio": 5,
17         "iTotalPictureVolume": 0.606635868549347,
18         "iTotalVideoVolume": 5.45972299575806,
19         "iVideoQuotaRatio": 45,
20         "id": 2
21     },
22     {
23         "iFreePictureQuota": 0.0,
24         "iFreeVideoQuota": 0.0,
25         "iPictureQuotaRatio": 5,
26         "iTotalPictureVolume": 0.0,
27         "iTotalVideoVolume": 0.0,
28         "iVideoQuotaRatio": 45,
29         "id": 3
30     }
31 ]
32
33 # GET /storage/quota/id: The id is a number, get the disk quota information
34 # with the corresponding id
35 # response
36 {
37     "iFreePictureQuota": 0.595100224018097,
38     "iFreeVideoQuota": 5.42329835891724,
39     "iPictureQuotaRatio": 5,
40     "iTotalPictureVolume": 0.606635868549347,
41     "iTotalVideoVolume": 5.45972299575806,
42     "iVideoQuotaRatio": 45,
43     "id": 2
44 }
45
46 # POST/PUT /storage/quota/id: The id is a number, set the disk quota
47 # information with the corresponding id, and switch the storage disk to the
48 # disk with the corresponding id
49 # request
50 {
51     "id":3,
52     "iPictureQuotaRatio":5,
53     "iVideoQuotaRatio":45
54 }
55 # response
56 {
57     "iFreePictureQuota":0,
58     "iFreeVideoQuota":0,
59     "iPictureQuotaRatio":5,
60     "iTotalPictureVolume":0,
61     "iTotalVideoVolume":0,
62     "iVideoQuotaRatio":45,
63     "id":3
64 }

```

6.3 snap-plan

```
1  # GET /storage/snap-plan/id: The id is a number, get the plan snapshot
   parameters of the id
2  # response
3  {
4      "iEnabled": 0,
5      "iImageQuality": 10,
6      "iShotInterval": 1000,
7      "iShotNumber": 4,
8      "sImageType": "JPEG",
9      "sResolution": "2688*1520"
10 }
11
12 # POST/PUT /storage/snap-plan/id: The id is a number, configure the plan
   snapshot parameters of the id
13 # request
14 {
15     "iEnabled":0,
16     "sImageType":"JPEG",
17     "sResolution":"2688*1520",
18     "iImageQuality":10,
19     "iShotInterval":10000
20 }
21 # response
22 {
23     "iEnabled":0,
24     "iImageQuality":10,
25     "iShotInterval":10000,
26     "sImageType":"JPEG",
27     "sResolution":"2688*1520"
28 }
```

6.4 current-path

```
1  # GET /storage/current-path: Get the current storage path
2  # response
3  {
4      "sMountPath": "/userdata/media"
5  }
```

6.5 format

```
1  # POST/PUT /storage/format/id: id is a number, format the disk with the id
2  # request
3  null
4  # response
5  {}
```

6.6 search

```
1  # POST/PUT /storage/search: Search the storage records of video/snapshot
2  # request: Query conditions, maxResults: Return the maximum number of query
   results, searchResultPosition: Return the starting position of the query
   results, order: 0 represents query in positive order, 1 represents query in
   reverse order
3  {
4      "searchType":"video0",
5      "startTime":"1970-01-01T00:00:00",
6      "endTime":"2020-08-29T23:59:59",
7      "maxResults":20,
8      "searchResultPosition: return the starting position of the query
   result, ":0,
9      "order":0
10 }
11 # response
12 {
13     "matchList":[
14         {
15             "fileAddress":"http://172.16.21.106//main_20200715200714_1.mp4",
16             "fileId":0,
17             "fileName":"main_20200715200714_1.mp4",
18             "fileSize":0.82734203338623,
19             "fileTime":"2020-07-15T20:07:16"
20         },
21         {
22             "fileAddress":"http://172.16.21.106/main_20200715200335_6.mp4",
23             "fileId":1,
24             "fileName":"main_20200715200335_6.mp4",
25             "fileSize":0.00182342529296875,
26             "fileTime":"2020-07-15T20:03:35"
27         }
28     ],
29     "numOfMatches":2
30 }
```

6.7 advance-para

```
1  # GET /storage/advance-para/0: Get scheduled recording configuration
   parameters
2  {
3      "iEnabled": 0,
4      "id": 0
5  }
6
7  # POST/PUT /storage/advance-para/0: Configure scheduled recording parameters
8  # request
9  {
10     "iEnabled": 1,
11     "id": 0
12 }
13 # response
```

```

14  {
15      "iEnabled": 1,
16      "id": 0
17  }

```

6.8 delete

```

1  # POST/PUT /storage/delete: Delete the specified type of snapshot/recording
2  # request
3  {
4      "type":"photo0",
5      "name":[
6          "main_19700101_085333_2.jpeg",
7          "main_19700101_080219_1.jpeg"
8      ]
9  }
10 # response :1 represents successfully deleted, 0 represents failing to
    delete
11 {"rst":1}

```

7. roi

```

1  # GET /roi: Get roi information of all code streams
2  # response
3  {
4      "ROIRegionList": [
5          {
6              "iHeight": 0,
7              "iPositionX": 0,
8              "iPositionY": 0,
9              "iQualityLevelOfROI": 3,
10             "iROIEnabled": 0,
11             "iROIId": 1,
12             "iStreamEnabled": 0,
13             "iWidth": 0,
14             "sName": "test",
15             "sStreamType": "mainStream"
16         },
17         {
18             "iHeight": 0,
19             "iPositionX": 0,
20             "iPositionY": 0,
21             "iQualityLevelOfROI": 3,
22             "iROIEnabled": 0,
23             "iROIId": 2,
24             "iStreamEnabled": 0,
25             "iWidth": 0,
26             "sName": "test",
27             "sStreamType": "mainStream"
28         },
29         {

```

```

30         "iHeight": 0,
31         "iPositionX": 0,
32         "iPositionY": 0,
33         "iQualityLevelOfROI": 3,
34         "iROIEnabled": 0,
35         "iROIId": 1,
36         "iStreamEnabled": 0,
37         "iWidth": 0,
38         "sName": "test",
39         "sStreamType": "subStream"
40     },
41     {
42         "iHeight": 0,
43         "iPositionX": 0,
44         "iPositionY": 0,
45         "iQualityLevelOfROI": 3,
46         "iROIEnabled": 0,
47         "iROIId": 2,
48         "iStreamEnabled": 0,
49         "iWidth": 0,
50         "sName": "test",
51         "sStreamType": "subStream"
52     },
53     {
54         "iHeight": 0,
55         "iPositionX": 0,
56         "iPositionY": 0,
57         "iQualityLevelOfROI": 3,
58         "iROIEnabled": 0,
59         "iROIId": 1,
60         "iStreamEnabled": 0,
61         "iWidth": 0,
62         "sName": "test",
63         "sStreamType": "thirdStream"
64     },
65     {
66         "iHeight": 0,
67         "iPositionX": 0,
68         "iPositionY": 0,
69         "iQualityLevelOfROI": 3,
70         "iROIEnabled": 0,
71         "iROIId": 2,
72         "iStreamEnabled": 0,
73         "iWidth": 0,
74         "sName": "test",
75         "sStreamType": "thirdStream"
76     }
77 ],
78     "normalizedScreenSize": {
79         "iNormalizedScreenHeight": 480,
80         "iNormalizedScreenWidth": 704
81     }
82 }
83
84 # GET /roi/main-stream: Get the roi information of the main stream
85 # GET /roi/subStream: Get the roi information of the substream
86 # GET /roi/thirdStream: Get the roi information of the third stream
87 # response

```

```

88  [
89      {
90          "iHeight": 0,
91          "iPositionX": 0,
92          "iPositionY": 0,
93          "iQualityLevelOfROI": 3,
94          "iROIEnabled": 0,
95          "iROIId": 1,
96          "iStreamEnabled": 0,
97          "iWidth": 0,
98          "sName": "test",
99          "sStreamType": "mainStream"
100     },
101     {
102         "iHeight": 0,
103         "iPositionX": 0,
104         "iPositionY": 0,
105         "iQualityLevelOfROI": 3,
106         "iROIEnabled": 0,
107         "iROIId": 2,
108         "iStreamEnabled": 0,
109         "iWidth": 0,
110         "sName": "test",
111         "sStreamType": "mainStream"
112     }
113 ]
114
115 # GET /roi/main-stream/id: Get the roi information with the id
corresponding to the main stream. For other streams, please refer to this
URL
116 # response
117 {
118     "iHeight": 0,
119     "iPositionX": 0,
120     "iPositionY": 0,
121     "iQualityLevelOfROI": 3,
122     "iROIEnabled": 0,
123     "iROIId": 1,
124     "iStreamEnabled": 0,
125     "iWidth": 0,
126     "sName": "test",
127     "sStreamType": "mainStream"
128 }
129
130 # PUT/POST /roi/main-stream/id: Configure the roi corresponding to the id
of the main stream. For other streams, please refer to this URL
131 # request
132 {
133     "iHeight": 0,
134     "iPositionX": 0,
135     "iPositionY": 0,
136     "iQualityLevelOfROI": 3,
137     "iROIEnabled": 1,
138     "iROIId": 1,
139     "iStreamEnabled": 0,
140     "iWidth": 0,
141     "sName": "test",
142     "sStreamType": "mainStream"

```

```

143 }
144 # response
145 {
146     "iHeight": 0,
147     "iPositionX": 0,
148     "iPositionY": 0,
149     "iQualityLevelOfROI": 3,
150     "iROIEnabled": 1,
151     "iROIId": 1,
152     "iStreamEnabled": 0,
153     "iWidth": 0,
154     "sName": "test",
155     "sStreamType": "mainStream"
156 }

```

8. peripherals

8.1 gate

```

1  # GET /peripherals/gate: Get the peripheral configuration of the gate/access
   control product
2  # response
3  {
4      "relay": {
5          "iDuration": 500,
6          "iEnable": 0,
7          "iIOIndex": 0,
8          "iValidLevel": 1,
9          "id": 0
10     },
11     "weigen": {
12         "iDuration": 0,
13         "iEnable": 0,
14         "iWiegandBit": 26,
15         "id": 0
16     }
17 }
18
19 # POST/PUT /peripherals/gate: Configure parameters of the gate/access
   control product
20 # request
21 {
22     "relay": {
23         "iDuration": 500,
24         "iEnable": 0,
25         "iIOIndex": 0,
26         "iValidLevel": 1,
27         "id": 0
28     },
29     "weigen": {
30         "iDuration": 0,
31         "iEnable": 0,

```



```

32         "iWiegandBit": 26,
33         "id": 0
34     }
35 }
36 # response
37 {
38     "relay": {
39         "iDuration": 500,
40         "iEnable": 0,
41         "iIOIndex": 0,
42         "iValidLevel": 1,
43         "id": 0
44     },
45     "weigen": {
46         "iDuration": 0,
47         "iEnable": 0,
48         "iWiegandBit": 26,
49         "id": 0
50     }
51 }

```

8.2 fill-light

```

1  # GET /peripherals/fill-light: Get the peripheral configuration of the fill
   light
2  # response
3  {
4      "iNormalBrightness": 50,
5      "iSaveEnergyBrightness": 50,
6      "iSaveEnergyEnable": 0,
7      "id": 0
8  }
9
10 # POST/PUT /peripherals/fill-light: Configure fill light parameters
11 # request
12 {
13     "iNormalBrightness": 50,
14     "iSaveEnergyBrightness": 50,
15     "iSaveEnergyEnable": 0,
16     "id": 0
17 }
18 # response
19 {
20     "iNormalBrightness": 50,
21     "iSaveEnergyBrightness": 50,
22     "iSaveEnergyEnable": 0,
23     "id": 0
24 }

```

9. osd

9.1 overlays

```
1  # GET /osd/overlays: Get OSD overlay configuration
2  # response
3  {
4      "attribute": {
5          "iBoundary": 0,
6          "sAlignment": "customize",
7          "sOSDAttribute": "transparent/not-flashing",
8          "sOSDFontSize": "32*32",
9          "sOSDFrontColor": "fff799",
10         "sOSDFrontColorMode": "customize"
11     },
12     "channelNameOverlay": {
13         "iChannelNameOverlayEnabled": 1,
14         "iPositionX": 560,
15         "iPositionY": 432,
16         "sChannelName": "Camera 01"
17     },
18     "characterOverlay": [
19         {
20             "iPositionX": 0,
21             "iPositionY": 0,
22             "iTextOverlayEnabled": 0,
23             "id": 0,
24             "sDisplayText": "",
25             "sIsPersistentText": "true"
26         },
27         {
28             "iPositionX": 0,
29             "iPositionY": 0,
30             "iTextOverlayEnabled": 0,
31             "id": 1,
32             "sDisplayText": "",
33             "sIsPersistentText": "true"
34         },
35         {
36             "iPositionX": 0,
37             "iPositionY": 0,
38             "iTextOverlayEnabled": 0,
39             "id": 2,
40             "sDisplayText": "",
41             "sIsPersistentText": "true"
42         },
43         {
44             "iPositionX": 0,
45             "iPositionY": 0,
46             "iTextOverlayEnabled": 0,
47             "id": 3,
48             "sDisplayText": "",
49             "sIsPersistentText": "true"
50         },
51         {
52             "iPositionX": 0,
53             "iPositionY": 0,
54             "iTextOverlayEnabled": 0,
```

```

55         "id": 4,
56         "sDisplayText": "",
57         "sIsPersistentText": "true"
58     },
59     {
60         "iPositionX": 0,
61         "iPositionY": 0,
62         "iTextOverlayEnabled": 0,
63         "id": 5,
64         "sDisplayText": "",
65         "sIsPersistentText": "true"
66     },
67     {
68         "iPositionX": 0,
69         "iPositionY": 0,
70         "iTextOverlayEnabled": 0,
71         "id": 6,
72         "sDisplayText": "",
73         "sIsPersistentText": "true"
74     },
75     {
76         "iPositionX": 0,
77         "iPositionY": 0,
78         "iTextOverlayEnabled": 0,
79         "id": 7,
80         "sDisplayText": "",
81         "sIsPersistentText": "true"
82     }
83 ],
84 "dateTimeOverlay": {
85     "iDateTimeOverlayEnabled": 1,
86     "iDisplayWeekEnabled": 1,
87     "iPositionX": 16,
88     "iPositionY": 16,
89     "sDateStyle": "CHR-YYYY-MM-DD",
90     "sTimeStyle": "24hour"
91 },
92 "normalizedScreenSize": {
93     "iNormalizedScreenHeight": 480,
94     "iNormalizedScreenWidth": 704
95 }
96 }
97
98 # POST/PUT /osd/overlays: Configure OSD overlay
99 # request
100 {
101     "attribute": {
102         "iBoundary": 0,
103         "sAlignment": "customize",
104         "sOSDAttribute": "transparent/not-flashing",
105         "sOSDFontSize": "32*32",
106         "sOSDFrontColor": "fff799",
107         "sOSDFrontColorMode": "customize"
108     },
109     "channelNameOverlay": {
110         "iChannelNameOverlayEnabled": 1,
111         "iPositionX": 560,
112         "iPositionY": 432,

```

```
113         "sChannelName": "Camera 01"
114     },
115     "characterOverlay": [
116         {
117             "iPositionX": 0,
118             "iPositionY": 0,
119             "iTextOverlayEnabled": 0,
120             "id": 0,
121             "sDisplayText": "",
122             "sIsPersistentText": "true"
123         },
124         {
125             "iPositionX": 0,
126             "iPositionY": 0,
127             "iTextOverlayEnabled": 0,
128             "id": 1,
129             "sDisplayText": "",
130             "sIsPersistentText": "true"
131         },
132         {
133             "iPositionX": 0,
134             "iPositionY": 0,
135             "iTextOverlayEnabled": 0,
136             "id": 2,
137             "sDisplayText": "",
138             "sIsPersistentText": "true"
139         },
140         {
141             "iPositionX": 0,
142             "iPositionY": 0,
143             "iTextOverlayEnabled": 0,
144             "id": 3,
145             "sDisplayText": "",
146             "sIsPersistentText": "true"
147         },
148         {
149             "iPositionX": 0,
150             "iPositionY": 0,
151             "iTextOverlayEnabled": 0,
152             "id": 4,
153             "sDisplayText": "",
154             "sIsPersistentText": "true"
155         },
156         {
157             "iPositionX": 0,
158             "iPositionY": 0,
159             "iTextOverlayEnabled": 0,
160             "id": 5,
161             "sDisplayText": "",
162             "sIsPersistentText": "true"
163         },
164         {
165             "iPositionX": 0,
166             "iPositionY": 0,
167             "iTextOverlayEnabled": 0,
168             "id": 6,
169             "sDisplayText": "",
170             "sIsPersistentText": "true"
```

```

171     },
172     {
173         "iPositionX": 0,
174         "iPositionY": 0,
175         "iTextOverlayEnabled": 0,
176         "id": 7,
177         "sDisplayText": "",
178         "sIsPersistentText": "true"
179     }
180 ],
181 "dateTimeOverlay": {
182     "iDateTimeOverlayEnabled": 1,
183     "iDisplayWeekEnabled": 1,
184     "iPositionX": 16,
185     "iPositionY": 16,
186     "sDateStyle": "CHR-YYYY-MM-DD",
187     "sTimeStyle": "24hour"
188 },
189 "normalizedScreenSize": {
190     "iNormalizedScreenHeight": 480,
191     "iNormalizedScreenWidth": 704
192 }
193 }
194 # response
195 {
196     "attribute": {
197         "iBoundary": 0,
198         "sAlignment": "customize",
199         "sOSDAttribute": "transparent/not-flashing",
200         "sOSDFontSize": "32*32",
201         "sOSDFrontColor": "fff799",
202         "sOSDFrontColorMode": "customize"
203     },
204     "channelNameOverlay": {
205         "iChannelNameOverlayEnabled": 1,
206         "iPositionX": 560,
207         "iPositionY": 432,
208         "sChannelName": "Camera 01"
209     },
210     "characterOverlay": [
211         {
212             "iPositionX": 0,
213             "iPositionY": 0,
214             "iTextOverlayEnabled": 0,
215             "id": 0,
216             "sDisplayText": "",
217             "sIsPersistentText": "true"
218         },
219         {
220             "iPositionX": 0,
221             "iPositionY": 0,
222             "iTextOverlayEnabled": 0,
223             "id": 1,
224             "sDisplayText": "",
225             "sIsPersistentText": "true"
226         },
227         {
228             "iPositionX": 0,

```

```

229         "iPositionY": 0,
230         "iTextOverlayEnabled": 0,
231         "id": 2,
232         "sDisplayText": "",
233         "sIsPersistentText": "true"
234     },
235     {
236         "iPositionX": 0,
237         "iPositionY": 0,
238         "iTextOverlayEnabled": 0,
239         "id": 3,
240         "sDisplayText": "",
241         "sIsPersistentText": "true"
242     },
243     {
244         "iPositionX": 0,
245         "iPositionY": 0,
246         "iTextOverlayEnabled": 0,
247         "id": 4,
248         "sDisplayText": "",
249         "sIsPersistentText": "true"
250     },
251     {
252         "iPositionX": 0,
253         "iPositionY": 0,
254         "iTextOverlayEnabled": 0,
255         "id": 5,
256         "sDisplayText": "",
257         "sIsPersistentText": "true"
258     },
259     {
260         "iPositionX": 0,
261         "iPositionY": 0,
262         "iTextOverlayEnabled": 0,
263         "id": 6,
264         "sDisplayText": "",
265         "sIsPersistentText": "true"
266     },
267     {
268         "iPositionX": 0,
269         "iPositionY": 0,
270         "iTextOverlayEnabled": 0,
271         "id": 7,
272         "sDisplayText": "",
273         "sIsPersistentText": "true"
274     }
275 ],
276 "dateTimeOverlay": {
277     "iDateTimeOverlayEnabled": 1,
278     "iDisplayWeekEnabled": 1,
279     "iPositionX": 16,
280     "iPositionY": 16,
281     "sDateStyle": "CHR-YYYY-MM-DD",
282     "sTimeStyle": "24hour"
283 },
284 "normalizedScreenSize": {
285     "iNormalizedScreenHeight": 480,
286     "iNormalizedScreenWidth": 704

```

```
287     }
288 }
```

9.2 image

```
1  # GET /osd/image: Get image masking configuration
2  # response
3  {
4      "imageOverlay": {
5          "iImageHeight": 80,
6          "iImageOverlayEnabled": 1,
7          "iImageWidth": 160,
8          "iPositionX": 16,
9          "iPositionY": 388,
10         "iTransparentColorEnabled": 0
11     },
12     "normalizedScreenSize": {
13         "iNormalizedScreenHeight": 480,
14         "iNormalizedScreenWidth": 704
15     }
16 }
17
18 # POST/PUT /osd/image: Configure image masking
19 # request
20 {
21     "imageOverlay": {
22         "iImageHeight": 80,
23         "iImageOverlayEnabled": 1,
24         "iImageWidth": 160,
25         "iPositionX": 16,
26         "iPositionY": 388,
27         "iTransparentColorEnabled": 0
28     },
29     "normalizedScreenSize": {
30         "iNormalizedScreenHeight": 480,
31         "iNormalizedScreenWidth": 704
32     }
33 }
34 # response
35 {
36     "imageOverlay": {
37         "iImageHeight": 80,
38         "iImageOverlayEnabled": 1,
39         "iImageWidth": 160,
40         "iPositionX": 16,
41         "iPositionY": 388,
42         "iTransparentColorEnabled": 0
43     },
44     "normalizedScreenSize": {
45         "iNormalizedScreenHeight": 480,
46         "iNormalizedScreenWidth": 704
47     }
48 }
49
50 # POST/PUT image/picture: Set the masking image
```

```

51 # request: The image must be 64bit, width and height are 16 aligned, bmp
    images with a size less than 256KB, and the transmission format is as
    follows
52 Content-Type: multipart/form-data
53 # response
54 {}

```

9.3 privacy-mask

```

1  # GET /osd/privacy-mask: Get privacy mask configuration
2  # response
3  {
4      "normalizedScreenSize": {
5          "iNormalizedScreenHeight": 480,
6          "iNormalizedScreenWidth": 704
7      },
8      "privacyMask": [
9          {
10             "iMaskHeight": 0,
11             "iMaskWidth": 0,
12             "iPositionX": 0,
13             "iPositionY": 0,
14             "iPrivacyMaskEnabled": 0,
15             "id": 0
16         },
17         {
18             "iMaskHeight": 0,
19             "iMaskWidth": 0,
20             "iPositionX": 0,
21             "iPositionY": 0,
22             "iPrivacyMaskEnabled": 0,
23             "id": 1
24         },
25         {
26             "iMaskHeight": 0,
27             "iMaskWidth": 0,
28             "iPositionX": 0,
29             "iPositionY": 0,
30             "iPrivacyMaskEnabled": 0,
31             "id": 2
32         },
33         {
34             "iMaskHeight": 0,
35             "iMaskWidth": 0,
36             "iPositionX": 0,
37             "iPositionY": 0,
38             "iPrivacyMaskEnabled": 0,
39             "id": 3
40         }
41     ]
42 }
43
44 # POST/PUT /osd/privacy-mask: Configure privacy mask
45 # request
46 {

```



```
47     "normalizedScreenSize": {
48         "iNormalizedScreenHeight": 480,
49         "iNormalizedScreenWidth": 704
50     },
51     "privacyMask": [
52         {
53             "iMaskHeight": 0,
54             "iMaskWidth": 0,
55             "iPositionX": 0,
56             "iPositionY": 0,
57             "iPrivacyMaskEnabled": 0,
58             "id": 0
59         },
60         {
61             "iMaskHeight": 0,
62             "iMaskWidth": 0,
63             "iPositionX": 0,
64             "iPositionY": 0,
65             "iPrivacyMaskEnabled": 0,
66             "id": 1
67         },
68         {
69             "iMaskHeight": 0,
70             "iMaskWidth": 0,
71             "iPositionX": 0,
72             "iPositionY": 0,
73             "iPrivacyMaskEnabled": 0,
74             "id": 2
75         },
76         {
77             "iMaskHeight": 0,
78             "iMaskWidth": 0,
79             "iPositionX": 0,
80             "iPositionY": 0,
81             "iPrivacyMaskEnabled": 0,
82             "id": 3
83         }
84     ]
85 }
86 # response
87 {
88     "normalizedScreenSize": {
89         "iNormalizedScreenHeight": 480,
90         "iNormalizedScreenWidth": 704
91     },
92     "privacyMask": [
93         {
94             "iMaskHeight": 0,
95             "iMaskWidth": 0,
96             "iPositionX": 0,
97             "iPositionY": 0,
98             "iPrivacyMaskEnabled": 0,
99             "id": 0
100         },
101         {
102             "iMaskHeight": 0,
103             "iMaskWidth": 0,
104             "iPositionX": 0,
```

```

105         "iPositionY": 0,
106         "iPrivacyMaskEnabled": 0,
107         "id": 1
108     },
109     {
110         "iMaskHeight": 0,
111         "iMaskWidth": 0,
112         "iPositionX": 0,
113         "iPositionY": 0,
114         "iPrivacyMaskEnabled": 0,
115         "id": 2
116     },
117     {
118         "iMaskHeight": 0,
119         "iMaskWidth": 0,
120         "iPositionX": 0,
121         "iPositionY": 0,
122         "iPrivacyMaskEnabled": 0,
123         "id": 3
124     }
125 ]
126 }

```

10. network

10.1 lan

```

1  # GET /network/lan: Get LAN configuration
2  # response
3  {
4      "ipv4":{
5          "sV4Address":"172.16.21.106",
6          "sV4Gateway":"172.16.21.1",
7          "sV4Method":"dhcp",
8          "sV4Netmask":"255.255.255.0"
9      },
10     "link":{
11         "iDuplex":1,
12         "iNicSpeed":1000,
13         "iPower":1,
14         "sAddress":"fa:40:a4:8b:ad:57",
15         "sDNS1":"10.10.10.188",
16         "sDNS2":"58.22.96.66",
17         "sInterface":"eth0",
18         "sNicSpeed":"Auto",
19         "sNicSpeedSupport":"Auto 10baseT/Half 10baseT/Full 100baseT/Half
20         100baseT/Full 1000baseT/Full "
21     }
22 }
23 # POST/PUT /network/lan: Configure LAN, get IP automatically
24 # request

```

```

25 {
26     "ipv4":{
27         "sV4Method":"dhcp"
28     },
29     "link":{
30         "sNicSpeed":"Auto",
31         "sDNS1":"10.10.10.188",
32         "sDNS2":"58.22.96.66",
33     }
34 }
35 # response
36 {
37     "ipv4":{
38         "sV4Address":"172.16.21.106",
39         "sV4Gateway":"172.16.21.1",
40         "sV4Method":"dhcp",
41         "sV4Netmask":"255.255.255.0"
42     },
43     "link":{
44         "iDuplex":1,
45         "iNicSpeed":1000,
46         "iPower":1,
47         "sAddress":"fa:40:a4:8b:ad:57",
48         "sDNS1":"10.10.10.188",
49         "sDNS2":"58.22.96.66",
50         "sInterface":"eth0",
51         "sNicSpeed":"Auto",
52         "sNicSpeedSupport":"Auto 10baseT/Half 10baseT/Full 100baseT/Half
100baseT/Full 1000baseT/Full "
53     }
54 }
55
56 # POST/PUT /network/lan: Configure LAN, set IP manually
57 # request
58 {
59     "ipv4":{
60         "sV4Address":"172.16.21.106",
61         "sV4Gateway":"172.16.21.1",
62         "sV4Method":"manual",
63         "sV4Netmask":"255.255.255.0"
64     },
65     "link":{
66         "sNicSpeed":"Auto",
67         "sDNS1":"10.10.10.188",
68         "sDNS2":"58.22.96.66",
69     }
70 }
71 # response
72 {
73     "ipv4":{
74         "sV4Address":"172.16.21.106",
75         "sV4Gateway":"172.16.21.1",
76         "sV4Method":"manual",
77         "sV4Netmask":"255.255.255.0"
78     },
79     "link":{
80         "iDuplex":1,
81         "iNicSpeed":1000,

```

```

82         "iPower":1,
83         "sAddress":"fa:40:a4:8b:ad:57",
84         "sDNS1":"10.10.10.188",
85         "sDNS2":"58.22.96.66",
86         "sInterface":"eth0",
87         "sNicSpeed":"Auto",
88         "sNicSpeedSupport":"Auto 10baseT/Half 10baseT/Full 100baseT/Half
100baseT/Full 1000baseT/Full "
89     }
90 }

```

10.2 wlan

```

1  # GET /network/wlan: Get the wireless LAN configuration, the example is the
   configuration when Wi-Fi is not connected
2  # response
3  {
4      "ipv4":{
5          "sV4Address":"",
6          "sV4Gateway":"",
7          "sV4Method":"dhcp",
8          "sV4Netmask":""
9      },
10     "link":{
11         "iDuplex":-1,
12         "iNicSpeed":-1,
13         "iPower":0,
14         "sAddress":"c0:84:7d:e1:ce:00",
15         "sDNS1":"",
16         "sDNS2":"",
17         "sInterface":"wlan0",
18         "sNicSpeed":""
19     }
20 }
21
22 # POST/PUT /network/wlan: Configure LAN, get IP automatically
23 # request
24 {
25     "ipv4":{
26         "sV4Method":"dhcp"
27     },
28     "link":{
29         "sDNS1":"",
30         "sDNS2":"",
31     }
32 }
33 # response: Is the configuration when Wi-Fi is not connected
34 {
35     "ipv4":{
36         "sV4Address":"",
37         "sV4Gateway":"",
38         "sV4Method":"dhcp",
39         "sV4Netmask":""
40     },
41     "link":{

```

```

42         "iDuplex":-1,
43         "iNicSpeed":-1,
44         "iPower":0,
45         "sAddress":"c0:84:7d:e1:ce:00",
46         "sDNS1":"",
47         "sDNS2":"",
48         "sInterface":"wlan0",
49         "sNicSpeed":""
50     }
51 }
52
53 # POST/PUT /network/wlan: Configure LAN, set IP manually
54 # request
55 {
56     "ipv4":{
57         "sV4Address":"172.16.21.106",
58         "sV4Gateway":"172.16.21.1",
59         "sV4Method":"manual",
60         "sV4Netmask":"255.255.255.0"
61     },
62     "link":{
63         "sDNS1":"10.10.10.188",
64         "sDNS2":"58.22.96.66",
65     }
66 }
67 # response: Is the configuration when Wi-Fi is not connected
68 {
69     "ipv4":{
70         "sV4Address":"172.16.21.106",
71         "sV4Gateway":"172.16.21.1",
72         "sV4Method":"manual",
73         "sV4Netmask":"255.255.255.0"
74     },
75     "link":{
76         "iDuplex":-1,
77         "iNicSpeed":-1,
78         "iPower":0,
79         "sAddress":"c0:84:7d:e1:ce:00",
80         "sDNS1":"10.10.10.188",
81         "sDNS2":"58.22.96.66",
82         "sInterface":"wlan0",
83         "sNicSpeed":""
84     }
85 }

```

10.3 Wi-Fi

```

1  # GET /network/wifi: Get Wi-Fi configuration
2  # response
3  {
4      "iPower":0,
5      "id":1,
6      "sType":"wifi"
7  }
8

```

```

 9  # POST/PUT /network/wifi?power=on: Power on Wi-Fi
10  # request
11  null
12  # response
13  {
14      "iPower":1,
15      "id":1,
16      "sType":"wifi"
17  }
18
19  # POST/PUT /network/wifi?power=off: Power off Wi-Fi
20  # request
21  null
22  # response
23  {
24      "iPower":0,
25      "id":1,
26      "sType":"wifi"
27  }
28
29  # POST/PUT /network/wifi: Connect to Wi-Fi
30  # request
31  {
32      "sName": "test",
33      "sService": "sadgwegwe_sdgas",
34      "sPassword": "test",
35      "iFavorite": 1,
36      "iAutoconnect": 1,
37      "sState": "ready",
38  }
39  # response
40  {}
41
42  # DELETE /network/wifi?service=sadgwegwe_sdgas: Delete Wi-Fi connection
    settings
43  # response
44  {}

```

10.4 wifi-list

```

 1  # GET /network/wifi-list: Get the scanned Wi-Fi list
 2  # response
 3  [
 4      {
 5          "Favorite": 1,
 6          "Strength": 90,
 7          "sName": "test"
 8          "sSecurity": "psk";
 9          "sService": "sadgwegwe_sdgas";
10          "sState": "ready";
11          "sType": "wifi";
12      }
13  ]

```

11. network-ntp

```
1  # GET /network-ntp: Get time settings
2  # response
3  {
4      "iAutoDst":0,
5      "iAutoMode":1,
6      "iRefreshTime":60,
7      "id":0,
8      "sNtpServers":"122.224.9.29 94.130.49.186",
9      "sTimeZone":"ChinaStandardTime-8",
10     "sTimeZoneFile":"posix/Etc/GMT-8",
11     "sTimeZoneFileDst":"posix/Asia/Shanghai"
12 }
13
14 # PUT/POST /network-ntp: Configure time parameters
15 # request
16 {
17     "iAutoDst":0,
18     "iAutoMode":1,
19     "iRefreshTime":60,
20     "id":0,
21     "sNtpServers":"122.224.9.29 94.130.49.186",
22     "sTimeZone":"ChinaStandardTime-8",
23     "sTimeZoneFile":"posix/Etc/GMT-8",
24     "sTimeZoneFileDst":"posix/Asia/Shanghai"
25 }
26 # response
27 {
28     "iAutoDst":0,
29     "iAutoMode":1,
30     "iRefreshTime":60,
31     "id":0,
32     "sNtpServers":"122.224.9.29 94.130.49.186",
33     "sTimeZone":"ChinaStandardTime-8",
34     "sTimeZoneFile":"posix/Etc/GMT-8",
35     "sTimeZoneFileDst":"posix/Asia/Shanghai"
36 }
```

11.1 time

```
1  # GET /network-ntp/time: Get device time
2  # response
3  {
4      "time":"2020-09-01T08:38:08"
5  }
6
7  # PUT/POST /network-ntp/time; Set device time, which is no need to set when
  in automatic mode
8  # request
9  {
10     "time":"2020-09-01T08:38:08"
11 }
```

```
12 # response
13 {
14     "time":"2020-09-01T08:38:08"
15 }
```

12. network-port

```
1 # GET /network-port: Get device port
2 # response
3 [
4     {
5         "iPortNo":80,
6         "id":0,
7         "sProtocol":"HTTP"
8     },
9     {
10        "iPortNo":443,
11        "id":1,
12        "sProtocol":"HTTPS"
13    },
14    {
15        "iPortNo":8080,
16        "id":2,
17        "sProtocol":"DEV_MANAGE"
18    },
19    {
20        "iPortNo":554,
21        "id":3,
22        "sProtocol":"RTSP"
23    },
24    {
25        "iPortNo":1935,
26        "id":4,
27        "sProtocol":"RTMP"
28    }
29 ]
30
31 # PUT/POST /network-port/id: The id is a number, corresponding to the
32 # request
33 # request
34 {
35     "iPortNo":80,
36     "id":0,
37     "sProtocol":"HTTP"
38 }
39 # response
40 {
41     "iPortNo":80,
42     "id":0,
43     "sProtocol":"HTTP"
44 }
```


13. image

```
1  # GET /image: Get all ISP configuration information
2  # response
3  [
4      {
5          "BLC": {
6              "iBLCRegionHeight": 92,
7              "iBLCRegionWidth": 120,
8              "iHDRLevel": 50,
9              "iHLCLevel": 0,
10             "iPositionX": 0,
11             "iPositionY": 0,
12             "iWDRLevel": 0,
13             "sBLCRegion": "close",
14             "sHDR": "open",
15             "sHLC": "close",
16             "sWDR": "close"
17         },
18         "exposure": {
19             "iAutoIrisLevel": 5,
20             "iExposureGain": 1,
21             "sExposureTime": "1/6",
22             "sIrisType": "auto"
23         },
24         "id": 0,
25         "imageAdjustment": {
26             "iBrightness": 50,
27             "iContrast": 50,
28             "iSaturation": 50,
29             "iSharpness": 50
30         },
31         "imageEnhancement": {
32             "iDehazeLevel": 0,
33             "iDenoiseLevel": 0,
34             "iImageRotation": 0,
35             "iSpatialDenoiseLevel": 0,
36             "iTemporalDenoiseLevel": 0,
37             "sDIS": "close",
38             "sDehaze": "close",
39             "sFEC": "close",
40             "sGrayScaleMode": "[0-255]",
41             "sNoiseReduceMode": "general"
42         },
43         "nightToDay": {
44             "iDistanceLevel": 1,
45             "iLightBrightness": 1,
46             "iNightToDayFilterLevel": 5,
47             "iNightToDayFilterTime": 5,
48             "sBeginTime": "07:00:00",
49             "sBrightnessAdjustmentMode": "auto",
50             "sEndTime": "18:00:00",
51             "sFillLightMode": "IR",
52             "sIrcutFilterAction": "day",
53             "sNightToDay": "auto",
54             "sOverexposeSuppress": "open",
```

```

55         "sOverexposeSuppresType": "auto"
56     },
57     "videoAdjustment": {
58         "sImageFlip": "close",
59         "sPowerLineFrequencyMode": "PAL(50HZ) ",
60         "sSceneMode": "indoor"
61     },
62     "whiteBlance": {
63         "iWhiteBalanceBlue": 50,
64         "iWhiteBalanceRed": 50,
65         "sWhiteBalanceStyle": "autoWhiteBalance"
66     }
67 }
68 ]

```

13.1 id

```

1  # GET /image/id: The id is a number, get the ISP configuration information
  of the corresponding channel
2  # response
3  {
4      "BLC": {
5          "iBLCRegionHeight": 92,
6          "iBLCRegionWidth": 120,
7          "iHDRLevel": 50,
8          "iHLCLevel": 0,
9          "iPositionX": 0,
10         "iPositionY": 0,
11         "iWDRLevel": 0,
12         "sBLCRegion": "close",
13         "sHDR": "open",
14         "sHLC": "close",
15         "sWDR": "close"
16     },
17     "exposure": {
18         "iAutoIrisLevel": 5,
19         "iExposureGain": 1,
20         "sExposureTime": "1/6",
21         "sIrisType": "auto"
22     },
23     "id": 0,
24     "imageAdjustment": {
25         "iBrightness": 50,
26         "iContrast": 50,
27         "iSaturation": 50,
28         "iSharpness": 50
29     },
30     "imageEnhancement": {
31         "iDehazeLevel": 0,
32         "iDenoiseLevel": 0,
33         "iImageRotation": 0,
34         "iSpatialDenoiseLevel": 0,
35         "iTemporalDenoiseLevel": 0,
36         "sDIS": "close",
37         "sDehaze": "close",

```

```
38     "sFEC": "close",
39     "sGrayScaleMode": "[0-255]",
40     "sNoiseReduceMode": "general"
41 },
42 "nightToDay": {
43     "iDistanceLevel": 1,
44     "iLightBrightness": 1,
45     "iNightToDayFilterLevel": 5,
46     "iNightToDayFilterTime": 5,
47     "sBeginTime": "07:00:00",
48     "sBrightnessAdjustmentMode": "auto",
49     "sEndTime": "18:00:00",
50     "sFillLightMode": "IR",
51     "sIrcutFilterAction": "day",
52     "sNightToDay": "auto",
53     "sOverexposeSuppress": "open",
54     "sOverexposeSuppressType": "auto"
55 },
56 "videoAdjustment": {
57     "sImageFlip": "close",
58     "sPowerLineFrequencyMode": "PAL(50HZ)",
59     "sSceneMode": "indoor"
60 },
61 "whiteBalance": {
62     "iWhiteBalanceBlue": 50,
63     "iWhiteBalanceRed": 50,
64     "sWhiteBalanceStyle": "autoWhiteBalance"
65 }
66 }
67
68 # POST/PUT /image/id: The id is a number, configure the ISP information of
69 # the corresponding channel
70 # request
71 {
72     "BLC": {
73         "iBLCRegionHeight": 92,
74         "iBLCRegionWidth": 120,
75         "iHDRLevel": 50,
76         "iHLCLLevel": 0,
77         "iPositionX": 0,
78         "iPositionY": 0,
79         "iWDRLevel": 0,
80         "sBLCRegion": "close",
81         "sHDR": "open",
82         "sHLC": "close",
83         "sWDR": "close"
84     },
85     "exposure": {
86         "iAutoIrisLevel": 5,
87         "iExposureGain": 1,
88         "sExposureTime": "1/6",
89         "sIrisType": "auto"
90     },
91     "id": 0,
92     "imageAdjustment": {
93         "iBrightness": 50,
94         "iContrast": 50,
95         "iSaturation": 50,
```

```
95         "iSharpness": 50
96     },
97     "imageEnhancement": {
98         "iDehazeLevel": 0,
99         "iDenoiseLevel": 0,
100        "iImageRotation": 0,
101        "iSpatialDenoiseLevel": 0,
102        "iTemporalDenoiseLevel": 0,
103        "sDIS": "close",
104        "sDehaze": "close",
105        "sFEC": "close",
106        "sGrayScaleMode": "[0-255]",
107        "sNoiseReduceMode": "general"
108    },
109    "nightToDay": {
110        "iDistanceLevel": 1,
111        "iLightBrightness": 1,
112        "iNightToDayFilterLevel": 5,
113        "iNightToDayFilterTime": 5,
114        "sBeginTime": "07:00:00",
115        "sBrightnessAdjustmentMode": "auto",
116        "sEndTime": "18:00:00",
117        "sFillLightMode": "IR",
118        "sIrcutFilterAction": "day",
119        "sNightToDay": "auto",
120        "sOverexposeSuppress": "open",
121        "sOverexposeSuppressType": "auto"
122    },
123    "videoAdjustment": {
124        "sImageFlip": "close",
125        "sPowerLineFrequencyMode": "PAL(50HZ)",
126        "sSceneMode": "indoor"
127    },
128    "whiteBalance": {
129        "iWhiteBalanceBlue": 50,
130        "iWhiteBalanceRed": 50,
131        "sWhiteBalanceStyle": "autoWhiteBalance"
132    }
133 }
134 # response
135 {
136     "BLC": {
137         "iBLCRegionHeight": 92,
138         "iBLCRegionWidth": 120,
139         "iHDRLevel": 50,
140         "iHLCLLevel": 0,
141         "iPositionX": 0,
142         "iPositionY": 0,
143         "iWDRLevel": 0,
144         "sBLCRegion": "close",
145         "sHDR": "open",
146         "sHLC": "close",
147         "sWDR": "close"
148     },
149     "exposure": {
150         "iAutoIrisLevel": 5,
151         "iExposureGain": 1,
152         "sExposureTime": "1/6",
```

```

153         "sIrisType": "auto"
154     },
155     "id": 0,
156     "imageAdjustment": {
157         "iBrightness": 50,
158         "iContrast": 50,
159         "iSaturation": 50,
160         "iSharpness": 50
161     },
162     "imageEnhancement": {
163         "iDehazeLevel": 0,
164         "iDenoiseLevel": 0,
165         "iImageRotation": 0,
166         "iSpatialDenoiseLevel": 0,
167         "iTemporalDenoiseLevel": 0,
168         "sDIS": "close",
169         "sDehaze": "close",
170         "sFEC": "close",
171         "sGrayScaleMode": "[0-255]",
172         "sNoiseReduceMode": "general"
173     },
174     "nightToDay": {
175         "iDistanceLevel": 1,
176         "iLightBrightness": 1,
177         "iNightToDayFilterLevel": 5,
178         "iNightToDayFilterTime": 5,
179         "sBeginTime": "07:00:00",
180         "sBrightnessAdjustmentMode": "auto",
181         "sEndTime": "18:00:00",
182         "sFillLightMode": "IR",
183         "sIrcutFilterAction": "day",
184         "sNightToDay": "auto",
185         "sOverexposeSuppress": "open",
186         "sOverexposeSuppressType": "auto"
187     },
188     "videoAdjustment": {
189         "sImageFlip": "close",
190         "sPowerLineFrequencyMode": "PAL(50HZ)",
191         "sSceneMode": "indoor"
192     },
193     "whiteBalance": {
194         "iWhiteBalanceBlue": 50,
195         "iWhiteBalanceRed": 50,
196         "sWhiteBalanceStyle": "autoWhiteBalance"
197     }
198 }
199
200 # GET /image/id/type: The id is a number, to get the type information of
the ISP configuration of the corresponding channel, take /image/0/blc as an
example
201 # response
202 {
203     "iBLCRegionHeight": 92,
204     "iBLCRegionWidth": 120,
205     "iHDRLevel": 50,
206     "iHLCLLevel": 0,
207     "iPositionX": 0,
208     "iPositionY": 0,

```

```

209     "iWDRLevel": 0,
210     "sBLCRegion": "close",
211     "sHDR": "open",
212     "sHLC": "close",
213     "sWDR": "close"
214 }
215
216 # POST/PUT /image/id/type: The id is a number, configure the ISP type
information of the corresponding channel, take /image/0/blc as an example
217 # request
218 {
219     "iBLCRegionHeight": 92,
220     "iBLCRegionWidth": 120,
221     "iHDRLevel": 50,
222     "iHLCLevel": 0,
223     "iPositionX": 0,
224     "iPositionY": 0,
225     "iWDRLevel": 0,
226     "sBLCRegion": "close",
227     "sHDR": "open",
228     "sHLC": "close",
229     "sWDR": "close"
230 }
231 # response
232 {
233     "iBLCRegionHeight": 92,
234     "iBLCRegionWidth": 120,
235     "iHDRLevel": 50,
236     "iHLCLevel": 0,
237     "iPositionX": 0,
238     "iPositionY": 0,
239     "iWDRLevel": 0,
240     "sBLCRegion": "close",
241     "sHDR": "open",
242     "sHLC": "close",
243     "sWDR": "close"
244 }

```

14. event

14.1 triggers

```

1  # GET /event/triggers/vmd_0: Get motion detection linkage mode configuration
2  # response
3  {
4      "iNotificationCenterEnabled": 0,
5      "iNotificationEmailEnabled": 0,
6      "iNotificationFTPEnabled": 0,
7      "iNotificationIO1Enabled": 0,
8      "iNotificationRecord1Enabled": 0,
9      "iVideoInputChannelID": 0,
10     "id": 0,

```

```
11     "sEventType": "VMD"
12 }
13
14 # POST/PUT /event/triggers/vmd_0: Configure motion detection linkage mode
15 # request
16 {
17     "iNotificationCenterEnabled": 0,
18     "iNotificationEmailEnabled": 0,
19     "iNotificationFTPEEnabled": 0,
20     "iNotificationIOIEnabled": 0,
21     "iNotificationRecord1Enabled": 0,
22     "iVideoInputChannelID": 0,
23     "id": 0,
24     "sEventType": "VMD"
25 }
26 # response
27 {
28     "iNotificationCenterEnabled": 0,
29     "iNotificationEmailEnabled": 0,
30     "iNotificationFTPEEnabled": 0,
31     "iNotificationIOIEnabled": 0,
32     "iNotificationRecord1Enabled": 0,
33     "iVideoInputChannelID": 0,
34     "id": 0,
35     "sEventType": "VMD"
36 }
37
38 # GET /event/triggers/vri_0: Get regional intrusion linkage mode
39 # configuration
40 # response
41 {
42     "iNotificationCenterEnabled": 0,
43     "iNotificationEmailEnabled": 0,
44     "iNotificationFTPEEnabled": 0,
45     "iNotificationIOIEnabled": 0,
46     "iNotificationRecord1Enabled": 0,
47     "iVideoInputChannelID": 0,
48     "id": 1,
49     "sEventType": "VRI"
50 }
51 # POST/PUT /event/triggers/vri_0: Configure regional intrusion linkage mode
52 # request
53 {
54     "iNotificationCenterEnabled": 0,
55     "iNotificationEmailEnabled": 0,
56     "iNotificationFTPEEnabled": 0,
57     "iNotificationIOIEnabled": 0,
58     "iNotificationRecord1Enabled": 0,
59     "iVideoInputChannelID": 0,
60     "id": 0,
61     "sEventType": "VMD"
62 }
63 # response
64 {
65     "iNotificationCenterEnabled": 0,
66     "iNotificationEmailEnabled": 0,
67     "iNotificationFTPEEnabled": 0,
```

```

68     "iNotificationIO1Enabled": 0,
69     "iNotificationRecord1Enabled": 0,
70     "iVideoInputChannelID": 0,
71     "id": 0,
72     "sEventType": "VMD"
73 }

```

14.2 schedules

```

1  # GET /event/schedules/motion: Get defense schedule configuration of motion
   detection
2  # response: json string
3  {
4      "sSchedulesJson": "[[],[],[],[],[],[],[]]"
5  }
6
7  # POST/PUT /event/schedules/motion: Configure defense schedule of motion
   detection
8  # request json string
9  {
10     "sSchedulesJson": "[[],[],[],[],[],[],[]]"
11 }
12 # response json string
13 {
14     "sSchedulesJson": "[[],[],[],[],[],[],[]]"
15 }
16
17 # GET /event/schedules/intrusion: Get defense schedule configuration of
   regional intrusion
18 # response json string
19 {
20     "sSchedulesJson": "[[],[],[],[],[],[],[]]"
21 }
22
23 # POST/PUT /event/schedules/motion: Configure defense schedule of regional
   intrusion
24 # request json string
25 {
26     "sSchedulesJson": "[[],[],[],[],[],[],[]]"
27 }
28 # response json string
29 {
30     "sSchedulesJson": "[[],[],[],[],[],[],[]]"
31 }
32
33 # GET /event/schedules/video-plan: Get defense schedule configuration of the
   recording plan
34 # response json string
35 {
36     "sSchedulesJson": "[[],[],[],[],[],[],[]]"
37 }
38
39 # POST/PUT /event/schedules/motion: Configure defense schedule of the
   recording plan
40 # request json string

```



```

41 {
42     "sSchedulesJson": "[[], [], [], [], [], [], []]"
43 }
44 # response json string
45 {
46     "sSchedulesJson": "[[], [], [], [], [], [], []]"
47 }
48
49 # GET /event/schedules/screenshot: Get defense schedule configuration of
snapshot plan
50 # response json string
51 {
52     "sSchedulesJson": "[[], [], [], [], [], [], []]"
53 }
54
55 # POST/PUT /event/schedules/motion: Configure defense schedule of snapshot
plan
56 # request json string
57 {
58     "sSchedulesJson": "[[], [], [], [], [], [], []]"
59 }
60 # response json string
61 {
62     "sSchedulesJson": "[[], [], [], [], [], [], []]"
63 }

```

The following is the parsing method of sSchedulesJson:

eg: [{"start":0.3134548611111111,"end":0.6328993055555551,"type":"timing"}],[[],[],[],[],[],[]]"

1. The whole body is a json array containing 7 small json arrays, and the small arrays represent the defense schedule configuration from Monday to Sunday in order;
2. Each small array contains up to 8 dictionaries, which are defense units. The floating point number in the unit is calculated from the defense time in seconds/ seconds converted from 24 hours. If the start time is 8 o'clock in the morning, then start is $(8 \times 60 \times 60) / (24 \times 60 \times 60) = 0.3333333333333333$;
3. Type is the type of defense, except for the video plan/capture plan, other defense units are without type;

14.3 motion-detection

```

1  # GET /event/motion-detection/0: Get motion detection configuration
2  # response
3  {
4      "iColumnGranularity": 22,
5      "iEndTriggerTime": 500,
6      "iHighlightEnabled": 0,
7      "iMotionDetectionEnabled": 0,
8      "iRowGranularity": 18,
9      "iSamplingInterval": 2,
10     "iSensitivityLevel": 1,
11     "iStartTriggerTime": 500,
12     "id": 0,
13     "sGridMap": "",
14     "sRegionType": "grid"
15 }
16

```

[illegible]

14.4 regional-invasion

```
1 # GET /event/regional-invasion/0: Get regional invasion configuration
2 # response
3 {
4     "normalizedScreenSize": {
5         "iNormalizedScreenHeight": 480,
6         "iNormalizedScreenWidth": 704
7     },
8     "regionalInvasion": {
9         "iEnabled": 0,
10        "iHeight": 0,
11        "iPositionX": 0,
12        "iPositionY": 0,
13        "iProportion": 0,
14        "iSensitivityLevel": 50,
15        "iTimeThreshold": 0,
16        "iWidth": 0
17    }
18 }
```

```

18  }
19
20  # POST/PUT /event/regional-invasion/0: Cconfigure regional invasion
21  # request
22  {
23      "normalizedScreenSize": {
24          "iNormalizedScreenHeight": 480,
25          "iNormalizedScreenWidth": 704
26      },
27      "regionalInvasion": {
28          "iEnabled": 0,
29          "iHeight": 0,
30          "iPositionX": 0,
31          "iPositionY": 0,
32          "iProportion": 0,
33          "iSensitivityLevel": 50,
34          "iTimeThreshold": 0,
35          "iWidth": 0
36      }
37  }
38  # response
39  {
40      "normalizedScreenSize": {
41          "iNormalizedScreenHeight": 480,
42          "iNormalizedScreenWidth": 704
43      },
44      "regionalInvasion": {
45          "iEnabled": 0,
46          "iHeight": 0,
47          "iPositionX": 0,
48          "iPositionY": 0,
49          "iProportion": 0,
50          "iSensitivityLevel": 50,
51          "iTimeThreshold": 0,
52          "iWidth": 0
53      }
54  }

```

14.5 face-list

```

1  # GET /event/face-list: Get all registered face information
2  # response
3  [
4      {
5          "iAccessCardNumber": 11,
6          "iAge": 50,
7          "iFaceDBId": 1,
8          "iLoadCompleted": 1,
9          "id": 0,
10         "sAddress": "",
11         "sBirthday": "1970-01-01",
12         "sCertificateNumber": "",
13         "sCertificateType": "identityCard",
14         "sGender": "male",
15         "sHometown": "",

```

```

16         "sListType": "permanent",
17         "sName": "test",
18         "sNation": "汉族",
19         "sNote": "",
20         "sPicturePath": "/userdata/media/white_list/test_0.jpg",
21         "sRegistrationTime": "2020-08-27T16:11:06",
22         "sTelephoneNumber": "",
23         "sType": "whiteList"
24     }
25 ]
26
27 # POST/PUT /event/face-config?search=condition: Search for registered face
information according to conditions
28 # request
29 {
30     "beginTime": "1970-01-01T00:00:00",
31     "endTime": "2020-08-29T23:59:59",
32     "type": "all",
33     "gender": "all",
34     "minAge": 0,
35     "maxAge": 100,
36     "accessCardNumber": 0,
37     "beginPosition": 0,
38     "endPosition": 19
39 }
40 # response
41 {
42     "matchList": [
43         {
44             "iAccessCardNumber": 11,
45             "iAge": 50,
46             "iFaceDBId": 1,
47             "iLoadCompleted": 1,
48             "id": 0,
49             "sAddress": "",
50             "sBirthday": "1970-01-01",
51             "sCertificateNumber": "",
52             "sCertificateType": "identityCard",
53             "sGender": "male",
54             "sHometown": "",
55             "sListType": "permanent",
56             "sName": "test",
57             "sNation": "汉族",
58             "sNote": "",
59             "sPicturePath": "/userdata/media/white_list/test_0.jpg",
60             "sRegistrationTime": "2020-08-27T16:11:06",
61             "sTelephoneNumber": "",
62             "sType": "whiteList"
63         }
64     ],
65     "numOfMatches": 1
66 }
67
68 # POST/PUT /event/face-config?search=name: Fuzzy search for registered face
information with name
69 # request
70 {
71     "name": "t",

```

```

72     "beginPosition":0,
73     "endPosition":19
74 }
75 # response
76 {
77     "matchList":[
78         {
79             "iAccessCardNumber": 11,
80             "iAge": 50,
81             "iFaceDBId": 1,
82             "iLoadCompleted": 1,
83             "id": 0,
84             "sAddress": "",
85             "sBirthday": "1970-01-01",
86             "sCertificateNumber": "",
87             "sCertificateType": "identityCard",
88             "sGender": "male",
89             "sHometown": "",
90             "sListType": "permanent",
91             "sName": "test",
92             "sNation": "汉族",
93             "sNote": "",
94             "sPicturePath": "/userdata/media/white_list/test_0.jpg",
95             "sRegistrationTime": "2020-08-27T16:11:06",
96             "sTelephoneNumber": "",
97             "sType": "whiteList"
98         }
99     ],
100     "numOfMatches":1
101 }

```

14.6 face

```

1  # GET /event/face/id: The id is a number, get the face information
   corresponding to the id
2  # response
3  {
4      "iAccessCardNumber": 11,
5      "iAge": 50,
6      "iFaceDBId": 1,
7      "iLoadCompleted": 1,
8      "id": 0,
9      "sAddress": "",
10     "sBirthday": "1970-01-01",
11     "sCertificateNumber": "",
12     "sCertificateType": "identityCard",
13     "sGender": "male",
14     "sHometown": "",
15     "sListType": "permanent",
16     "sName": "test",
17     "sNation": "汉族",
18     "sNote": "",
19     "sPicturePath": "/userdata/media/white_list/test_0.jpg",
20     "sRegistrationTime": "2020-08-27T16:11:06",
21     "sTelephoneNumber": "",

```

```
22     "sType": "whiteList"
23 }
24
25 # POST/PUT /event/face: Face registration, return the upload address of face
photos
26 # request
27 {
28     "iAccessCardNumber":0,
29     "sTelephoneNumber":"","
30     "sAddress":"","
31     "sBirthday":"1970-01-01",
32     "sCertificateNumber":"","
33     "sCertificateType":"identityCard",
34     "sGender":"male",
35     "sHometown":"","
36     "sListType":"permanent",
37     "sName":"test",
38     "sNation":"汉族",
39     "sNote":"undone",
40     "sType":"whiteList",
41     "iFaceDBId":-2
42 }
43 # response
44 {
45     "id":2,
46     "sPicturePath":"/userdata/media/white_list/test_2.jpg"
47 }
48
49 # POST/PUT /event/face/id: The id is a number, modify the face information
corresponding to the id
50 # request
51 {
52     "iAccessCardNumber":0,
53     "sTelephoneNumber":"","
54     "sAddress":"","
55     "sBirthday":"1970-01-01",
56     "sCertificateNumber":"","
57     "sCertificateType":"identityCard",
58     "sGender":"male",
59     "sHometown":"","
60     "sListType":"permanent",
61     "sName":"test",
62     "sNation":"汉族",
63     "sNote":"undone",
64     "sType":"whiteList"
65 }
66 # response
67 {
68     "iAccessCardNumber":0,
69     "iAge":50,
70     "iFaceDBId":2,
71     "iLoadCompleted":1,
72     "id":1,
73     "sAddress":"","
74     "sBirthday":"1970-01-01",
75     "sCertificateNumber":"","
76     "sCertificateType":"identityCard",
77     "sGender":"male",
```

```

78     "sHometown":"","
79     "sListType":"permanent",
80     "sName":"test",
81     "sNation":"汉族",
82     "sNote":"undone",
83     "sPicturePath":"/userdata/media/white_list/test_1.jpg",
84     "sRegistrationTime":"2020-08-29T16:06:52",
85     "sTelephoneNumber":"","
86     "sType":"whiteList"
87 }
88
89 # DELETE /event/face/id: The id is a number, delete the face information
    corresponding to the id
90 # response
91 {}

```

14.7 face-config

```

1  # GET /event/face-config: Get face parameter configuration (gate/acess
    control products)
2  # response
3  {
4      "iDetectHeight": 1280,
5      "iDetectWidth": 720,
6      "iFaceDetectionThreshold": 55,
7      "iFaceMinPixel": 144,
8      "iFaceRecognitionThreshold": 50,
9      "iLeftCornerX": 0,
10     "iLeftCornerY": 0,
11     "iLiveDetectThreshold": 50,
12     "iNormalizedHeight": 1280,
13     "iNormalizedWidth": 720,
14     "iPromptVolume": 50,
15     "id": 0,
16     "sLiveDetect": "open",
17     "sLiveDetectBeginTime": "00:00",
18     "sLiveDetectEndTime": "23:59"
19 }
20
21 # POST/PUT /event/face-config: Configure face parameters
22 # request
23 {
24     "iDetectHeight": 1280,
25     "iDetectWidth": 720,
26     "iFaceDetectionThreshold": 55,
27     "iFaceMinPixel": 144,
28     "iFaceRecognitionThreshold": 50,
29     "iLeftCornerX": 0,
30     "iLeftCornerY": 0,
31     "iLiveDetectThreshold": 50,
32     "iNormalizedHeight": 1280,
33     "iNormalizedWidth": 720,
34     "iPromptVolume": 50,
35     "id": 0,
36     "sLiveDetect": "open",

```

```

37     "sLiveDetectBeginTime": "00:00",
38     "sLiveDetectEndTime": "23:59"
39 }
40 # response
41 {
42     "iDetectHeight": 1280,
43     "iDetectWidth": 720,
44     "iFaceDetectionThreshold": 55,
45     "iFaceMinPixel": 144,
46     "iFaceRecognitionThreshold": 50,
47     "iLeftCornerX": 0,
48     "iLeftCornerY": 0,
49     "iLiveDetectThreshold": 50,
50     "iNormalizedHeight": 1280,
51     "iNormalizedWidth": 720,
52     "iPromptVolume": 50,
53     "id": 0,
54     "sLiveDetect": "open",
55     "sLiveDetectBeginTime": "00:00",
56     "sLiveDetectEndTime": "23:59"
57 }

```

14.8 face-waiting

```

1 # GET /event/face-waiting: Get the number of Registered photos
2 # response
3 {
4     "numOfWaiting": 0
5 }

```

14.9 smart

```

1 # GET /event/smart/cover: Get face configuration (IPC)
2 # response iFaceEnabled: Face detection is enabled,
   iFaceRecognitionEnabled: Face recognition is enabled, iImageOverlayEnabled:
   Alarm snapshot overlay is enabled, iStreamOverlayEnabled: Stream overlay is
   enabled
3 {
4     "iBodyHeightRatio": 100,
5     "iFaceEnabled": 0,
6     "iFaceHeightRatio": 100,
7     "iFaceRecognitionEnabled": 0,
8     "iImageOverlayEnabled": 0,
9     "iInfoOverlayEnabled": 0,
10    "iStreamOverlayEnabled": 0,
11    "iWidthRatio": 100,
12    "id": 0,
13    "infoOverlay": [
14        {
15            "iEnabled": 0,
16            "iOrder": 0,
17            "id": 0,

```



```

18         "sInfo": "",
19         "sName": "deviceNum"
20     },
21     {
22         "iEnabled": 0,
23         "iOrder": 1,
24         "id": 1,
25         "sInfo": "",
26         "sName": "snapTime"
27     },
28     {
29         "iEnabled": 0,
30         "iOrder": 2,
31         "id": 2,
32         "sInfo": "",
33         "sName": "positonInfo"
34     }
35 ],
36 "sImageQuality": "good",
37 "sTargetImageType": "head"
38 }
39
40 # POST/PUT /event/smart/cover: Configure face (IPC)
41 # request
42 {
43     "iBodyHeightRatio": 100,
44     "iFaceEnabled": 0,
45     "iFaceHeightRatio": 100,
46     "iFaceRecognitionEnabled": 0,
47     "iImageOverlayEnabled": 0,
48     "iInfoOverlayEnabled": 0,
49     "iStreamOverlayEnabled": 0,
50     "iWidthRatio": 100,
51     "id": 0,
52     "infoOverlay": [
53         {
54             "iEnabled": 0,
55             "iOrder": 0,
56             "id": 0,
57             "sInfo": "",
58             "sName": "deviceNum"
59         },
60         {
61             "iEnabled": 0,
62             "iOrder": 1,
63             "id": 1,
64             "sInfo": "",
65             "sName": "snapTime"
66         },
67         {
68             "iEnabled": 0,
69             "iOrder": 2,
70             "id": 2,
71             "sInfo": "",
72             "sName": "positonInfo"
73         }
74     ],
75     "sImageQuality": "good",

```

```

76     "sTargetImageType": "head"
77 }
78 # response
79 {
80     "iBodyHeightRatio": 100,
81     "iFaceEnabled": 0,
82     "iFaceHeightRatio": 100,
83     "iFaceRecognitionEnabled": 0,
84     "iImageOverlayEnabled": 0,
85     "iInfoOverlayEnabled": 0,
86     "iStreamOverlayEnabled": 0,
87     "iWidthRatio": 100,
88     "id": 0,
89     "infoOverlay": [
90         {
91             "iEnabled": 0,
92             "iOrder": 0,
93             "id": 0,
94             "sInfo": "",
95             "sName": "deviceNum"
96         },
97         {
98             "iEnabled": 0,
99             "iOrder": 1,
100            "id": 1,
101            "sInfo": "",
102            "sName": "snapTime"
103        },
104        {
105            "iEnabled": 0,
106            "iOrder": 2,
107            "id": 2,
108            "sInfo": "",
109            "sName": "positonInfo"
110        }
111    ],
112    "sImageQuality": "good",
113    "sTargetImageType": "head"
114 }

```

14.10 get-record-status

```

1 # GET /event/get-record-status: Get record status
2 # response 0 represents not recording, 1 represents in recording
3 0

```

14.11 last-face

```

1  # GET /event/last-face: Get the last id of the registered face information
2  # response
3  {
4      "id": 0
5  }

```

14.12 snapshot-record

```

1  # POST/PUT /event/snapshot-record: Query snapshot record by conditions
2  # request
3  {
4      "beginTime":"1970-01-01T00:00:00",
5      "endTime":"2020-08-29T23:59:59",
6      "beginPosition":0,
7      "endPosition":19
8  }
9  # response
10 {
11     "matchList":[
12         "id": 0,
13         "sNote": "",
14         "sPicturePath": "test.jpg",
15         "sStatus": "process",
16         "sTime": "2020-08-27T16:11:06",
17         "sSnapshotName": "test"
18     ],
19     "numOfMatches":1
20 }
21
22 # DELETE /event/snapshot-record/id: The id is a number, delete the snapshot
    record corresponding to the id
23 # response
24 {}

```

14.13 control-record

```

1  # POST/PUT /event/control-record?search=condition: Conditional query control
    record
2  # request
3  {
4      "beginTime":"1970-01-01T00:00:00",
5      "endTime":"2020-08-29T23:59:59",
6      "type":"all",
7      "gender":"all",
8      "minAge":0,
9      "maxAge":100,
10     "accessCardNumber":0,
11     "beginPosition":0,
12     "endPosition":19
13 }
14 # response
15 {

```

```

16     "matchList":[
17         {
18             "iAccessCardNumber":0,
19             "iAge":50,
20             "iFaceDBId":2,
21             "iFaceId":2,
22             "iLoadCompleted":1,
23             "id":1,
24             "sAddress":"",
25             "sBirthday":"1970-01-01",
26             "sCertificateNumber":"",
27             "sCertificateType":"identityCard",
28             "sGender":"male",
29             "sHometown":"",
30             "sListType":"permanent",
31             "sName":"test",
32             "sNation":"汉族",
33             "sNote":"",
34
35             "sPicturePath":"http://172.16.21.106/userdata/white_list/test_1.jpg",
36             "sRegistrationTime":"2020-08-29T15:33:41",
37             "sSimilarity":"0.4",
38             "sSnapshotName":"",
39             "sSnapshotPath":"http://172.16.21.106",
40             "sStatus":"Processed",
41             "sTelephoneNumber":"",
42             "sTime":"2020-08-29T15:33:46",
43             "sType":"whiteList"
44         }
45     ],
46     "numOfMatches":1
47 }
48 # POST/PUT /event/control-record?search=name: Fuzzy query control record
49 # request
50 {
51     "name":"t",
52     "beginPosition":0,
53     "endPosition":19
54 }
55 # response
56 {
57     "matchList":[
58         {
59             "iAccessCardNumber":0,
60             "iAge":50,
61             "iFaceDBId":2,
62             "iFaceId":2,
63             "iLoadCompleted":1,
64             "id":1,
65             "sAddress":"",
66             "sBirthday":"1970-01-01",
67             "sCertificateNumber":"",
68             "sCertificateType":"identityCard",
69             "sGender":"male",
70             "sHometown":"",
71             "sListType":"permanent",

```

```

72         "sName":"test",
73         "sNation":"汉族",
74         "sNote":"",
75
76         "sPicturePath":"http://172.16.21.106/userdata/white_list/test_1.jpg",
77         "sRegistrationTime":"2020-08-29T15:33:41",
78         "sSimilarity":"0.4",
79         "sSnapshotName":"",
80         "sSnapshotPath":"http://172.16.21.106",
81         "sStatus":"Processed",
82         "sTelephoneNumber":"",
83         "sTime":"2020-08-29T15:33:46",
84         "sType":"whiteList"
85     },
86     "numOfMatches":1
87 }
88
89 # DELETE /event/control-record/id: The id is a number, delete the control
90 # response
91 {}

```

14.14 check-face

```

1  # POST/PUT /event/check-face?id=0: Face data checking, to check whether the
   # face registration with the id greater than 0 is successful, if there is a
   # failure, delete it and return the failure result
2  # request
3  null
4  # response
5  [
6      {
7          "iLoadCompleted":-1,
8          "sName":"test",
9
10         "sPicturePath":"http://172.16.21.106/userdata/white_list/test_1.jpg",
11     }
12 ]

```

14.15 reset-face

```

1  # POST/PUT /event/reset-face: Reset the face database
2  # request
3  null
4  # response
5  {}

```

14.16 reset-snap

```
1 # POST/PUT /event/reset-snap: Reset the snapshot record
2 # request
3 null
4 # response
5 {}
```

14.17 reset-control

```
1 # POST/PUT /event/reset-control: Reset control record
2 # request
3 null
4 # response
5 {}
```

14.18 face-picture

```
1 # POST/PUT /event/face-picture?path=address: Address string, upload face
  image
2 # request
3 Content-Type: multipart/form-data
4 Form Data: Photo data
5 # response
6 {}
7
8 # POST/PUT /event/face-picture?copy-path=address: Address string, copy face
  image to the address
9 # request
10 {
11     "path": "old_address"
12 }
13 # response
14 {}
```

14.19 take-photo

```
1 # POST/PUT /event/take-photo: snapshot of the main stream
2 # request
3 null
4 # response
5 {}
```

14.20 start-record

```
1 # POST/PUT /event/start-record: Start recording
2 # request
3 null
4 # response
5 {}
```

14.21 stop-record

```
1 # POST/PUT /event/stop-record: Stop recording
2 # request
3 null
4 # response
5 {}
```

15. audio

```
1 # GET audio/0: Get audio configuration
2 # response
3 {
4     "iBitRate":32000,
5     "iSampleRate":16000,
6     "iVolume":50,
7     "id":0,
8     "sANS":"close",
9     "sEncodeType":"MP2",
10    "sInput":"micIn"
11 }
12
13 # POST/PUT audio/0: Configure audio parameters
14 # request
15 {
16     "iBitRate":32000,
17     "iSampleRate":16000,
18     "iVolume":50,
19     "id":0,
20     "sANS":"close",
21     "sEncodeType":"MP2",
22     "sInput":"micIn"
23 }
24 # response
25 {
26     "iBitRate":32000,
27     "iSampleRate":16000,
28     "iVolume":50,
29     "id":0,
30     "sANS":"close",
31     "sEncodeType":"MP2",
```

```
32     "sInput": "micIn"
33 }
```

16. Trouble Shooting

16.1 401

```
1  1.Cookie expired, you need to log in again to apply for a new cookie
2  {
3      "error": {
4          "code": 401,
5          "message": "token verification failed: not found cookie"
6      }
7  }
8
9  2.The login information contained in the cookie is wrong, you need to log in
  again to apply for a new cookie
10 {
11     "error": {
12         "code": 401,
13         "message": "Unauthorized"
14     }
15 }
```

16.2 500

```
1  1.The json carried in the request has a data type error
2  {
3      "error": {
4          "code": 500,
5          "message": "json.exception.type_error.304"
6      }
7  }
8
9  2.The requested id or key is not within the specified range
10 {
11     "error": {
12         "code": 500,
13         "message": "json.exception.out_of_range.403"
14     }
15 }
16
```


16.3 501

```
1 1. Invalid URL
2 {
3     "error": {
4         "code": 501,
5         "message": "Not Implemented"
6     }
7 }
```