

1、Preface

IPC External SDK API Documentation

2、Modification records

-Feb 2021

Date	Modified by	Interfaces involved	Modified content
Feb 26th	lms	First Addition	First Addition

3、Global error codes

[Error Code Description]

- 0x40000000 The first 32 bit is 0 for int and unsigned int compatibility, the first 32 bit is not used, 1 from 31 bit indicates error. The first four bits are the module name and the last four bits are the error code.

[Module List]

Module Code	Module Description
0x4000	General Module, basic Module
0x4001	Face Module

[Error Code Description]

Error Code	Description
0	Normal acceptance
101	Not logged in or session disabled
102	No permission
103	Session timeout
202	Interface call failure
203	Too many connected users
204	Upgrade package error
210	The logged-in user does not have enough privileges to perform modification operations

Error Code	Description
211	Login user password is wrong, can't do the modification
220	Password security problems verification error
225	New password setting can't be empty
230	Key verification error
302	ws End of push
303	Parsing audio and video recording file errors
0x40010001	Face module, module not initialized
0x40010002	Face module, parameter error
0x40010003	Face module, memory allocation failure
0x40010010	Face module, face gallery file does not exist
0x40010011	Face module, face gallery file is corrupted
0x40010012	Face module, database does not exist
0x40010013	Face module, file open failure
0x40010014	Face module, file open failure
0x40010015	Face module, face database data error
0x40010016	Face module, picture error
0x40010017	Face module, the person information error or the face picture exceeds the maximum.
0x40010018	Face module, face picture duplicated
0x40010019	Face module, face gallery name duplicated
0x4001001A	Face module, wrong image format
0x4001001B	Face module, face gallery personnel information is full
0x4001001C	Face module, Eigenvalues are being extracted, operation is prohibited
0x4001001D	Face module, database operation failed, please operate again, this error generally has more than one link together to operate the database
0x4001001E	Face module, no face detected

4、General Enumeration List

• Cloud Storage Type Enumeration Values

Type	Type Description	Corresponding enumeration values
CLOUD_BRAND_BAIDU	Baidu	16
CLOUD_BRAND_GOOGLE	Google	17
CLOUD_BRAND_DROPBOX	dropbox	18

• Video Storage Enumeration Values

Type	Type Description	Corresponding enumeration values
STORAGE_TYPE_SDCARE	hd card	0x01

Type	Type Description	Corresponding enumeration values
STORAGE_TYPE_USBDISK	usb	0x02
STORAGE_TYPE_HARDDISK	Hard Disk	0x04
STORAGE_TYPE_FTP	ftp	0x08
STORAGE_TYPE_SMTP	smtp	0x10
STORAGE_TYPE_CLOUD	Cloud	0x20
STORAGE_TYPE_ALL	All	0xff

- Recording time storage enumeration

Type	Type Description	Corresponding enumeration values
TIME_SCOPE_MODE_NULL	No time period	0
TIME_SCOPE_MODE_ALL	Full day time period	1
TIME_SCOPE_MODE_SAME	By the first time period	2
TIME_SCOPE_MODE_DIFF	By each specific time period	3

- Image type enumeration values

Type	Type Description	Corresponding enumeration values
PICTURE_CAPTURE_NULL	Invalid image types	0
PICTURE_CAPTURE_MANUAL	Manual capture	0x01
PICTURE_CAPTURE_MOTION	Motion detection capture	0x02
PICTURE_CAPTURE_ALARM	Alarm input capture	0x04
PICTURE_CAPTURE_MOTION_OR_ALARM	Motion detection or alarm input trigger capture	0x08
PICTURE_CAPTURE_MOTION_AND_ALARM	Motion detection and alarm input trigger simultaneously capture	0x10
PICTURE_CAPTURE_FACE	Face Detection Capture	0x20
PICTURE_CAPTURE_INTRUSION	Regional intrusion capture	0x40
PICTURE_CAPTURE_CROSS_BORDER	Cross-border capture	0x80
PICTURE_CAPTURE_SCENE_CHANGE	Scene change capture	0x100
PICTURE_CAPTURE_GOODS_LEGACY	Goods legacy detection capture	0x200
PICTURE_CAPTURE_GOODS_TAKE	Goods take detection capture	0x400
PICTURE_FACE_COMPARE	Face comparison capture	0x800
PICTURE_CAPTURE_PEOPLE_STAY	People wandering capture	0x1000
PICTURE_CAPTURE_PEOPLE_GATHERING	People gathering capture	0x2000
PICTURE_CAPTURE_SMOKE	Smoke sensor alarm capture	0x4000
PICTURE_CAPTURE_INFRARE	Human infrared capture	0x8000
...
PICTURE_CAPTURE_ALL	Query all types of images	0xFFFFFFFF

*The data type of the enumeration value is hexadecimal. If it is sent in decimal, it needs to be converted.

- Recording time enumeration value

Type	Type Description	Corresponding enumeration values
RECORD_TYPE_NORMAL	Timed	0x01
RECORD_TYPE_MOTION	Motion detection recording	0x02
RECORD_TYPE_ALARM	Alarm capture	0x04
RECORD_TYPE_M_AND_A	Motion detection and alarm recording	0x08
RECORD_TYPE_SMART_EVENT	Search all smart types of recording videos	0x80000000
...
RECORD_TYPE_ALL	Search all types of videos	0xFFFFFFFF

*The data type of the enumeration value is hexadecimal. If it is sent in decimal, it needs to be converted.

- Pre-recorded time type enumeration values

Type	Type Description	Corresponding enumeration values
STORAGE_PRERECORDTIME_NULL	No pre-recording	0
STORAGE_PRERECORDTIME_5S		1
STORAGE_PRERECORDTIME_10s		2
STORAGE_PRERECORDTIME_15s		3
STORAGE_PRERECORDTIME_20s		4
STORAGE_PRERECORDTIME_25s		5
STORAGE_PRERECORDTIME_30s		6
STORAGE_PRERECORDTIME_AUTO		7

- Recording delay type enumeration value

Type	Type Description	Corresponding enumeration values
STORAGE_RECORDERDELAY_NULL	None	0
STORAGE_RECORDERDELAY_5S		1
STORAGE_RECORDERDELAY_10S		2
STORAGE_RECORDERDELAY_30S		3
STORAGE_RECORDERDELAY_60S		4
STORAGE_RECORDERDELAY_120S		5
STORAGE_RECORDERDELAY_300S		6
STORAGE_RECORDERDELAY_600S		7

- Cloud control enumeration values

Type	Type Description	Corresponding enumeration values
PTZ_CONTROL_CMD_STOP	Stop	0
PTZ_CONTROL_CMD_UP	Up	10
PTZ_CONTROL_CMD_DOWN	Down	11
PTZ_CONTROL_CMD_LEFT	Left	12
PTZ_CONTROL_CMD_RIGHT	Right	13
PTZ_CONTROL_CMD_LEFTUP	Left up	14
PTZ_CONTROL_CMD_LEFTDOWN	Left down	15
PTZ_CONTROL_CMD_RIGHTUP	Right up	16
PTZ_CONTROL_CMD_RIGHTDOWN	Right down	17
PTZ_CONTROL_CMD_ADDZOOM	Zoom +	20
PTZ_CONTROL_CMD_DECZOOM	Zoom -	21
PTZ_CONTROL_CMD_ADDFOCUS	Focus +	22
PTZ_CONTROL_CMD_DECFOCUS	Focus -	23
PTZ_CONTROL_CMD_ADDAPERTURE	Aperture +	24
PTZ_CONTROL_CMD_DECAPERTURE	Aperture -	25
PTZ_CONTROL_CMD_ZOOM_STOP	Zoom stop	26
PTZ_CONTROL_CMD_FOCUS_STOP	Focus stop	27
PTZ_CONTROL_CMD_ONEKEYFOCUS	One key focus	31
PTZ_CONTROL_CMD_INITLENS	Initializing the lens	32
PTZ_CONTROL_CMD_LENS_CORRECITON	Lens Correction	33
PTZ_CONTROL_CMD_AUTOPAN	Continuous horizontal rotation	34
PTZ_CONTROL_CMD_AUTO_POSITION3D_START	Turn on automatic 3D positioning	35
PTZ_CONTROL_CMD_AUTO_POSITION3D_STOP	Turn off automatic 3D positioning	36
PTZ_CONTROL_CMD_MANUAL_POSITION3D	Manual 3D positioning	37
PTZ_CONTROL_CMD_MOVEPOINT	Move to preset point	40
PTZ_CONTROL_CMD_CRUISE_START	Start cruising	41
PTZ_CONTROL_CMD_PRESET_SET	Set preset points	44
PTZ_CONTROL_CMD_PRESET_DEL	Delete preset points	45
PTZ_CONTROL_CMD_CRUISE_STOP	Stop Cruising	46

5、Data format description

Data format The external SDK data transfer adopts the format of data header + data content The data header is in the HTTP

encapsulation format Data content is in the JSON format **Data header encapsulation format Request header:**

```
POST /api/login-capabilities HTTP/1.1
Accept: */*
Host: 172.18.195.188
Content-Type: application/json; charset=UTF-8
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Content-Length: 0
Api-Version: v1.2.0
```

Return header:

```
HTTP/1.1 200 OK
Date: Fri, 05 Mar 2021 02:40:45 GMT
Server: webserver
Content-Length: 392
Connection: keep-alive
Keep-Alive: timeout=10, max=99
Content-Type: application/json
Api-Version: v1.2.0
```

Data content format

```
Using json data format:{ key: data }
For example:{
  "action": "set",
  "data": NULL
}
```

Data type description:

Name	Type
String	string
Integer	int
Floating point	float
Object	object (JSON object)
Arrays	array
Boolean	bool: true, false
Null value	null

Request Format:

```
{
  "action": "...",
  "data": { ... }
}
```

Return Format:

```
{
  "code": 0,
  "msg": null,
  "data": { ... }
}
```

Parameter formatting conventions

For switch type parameters, use bool type uniformly
 For parameters that allow multi-channel selection, the string type is used uniformly, and channels are calculated in binary by bit offset
 For parameters with long values such as timestamps, the string type is used uniformly

Field Description

Parameter Name	Type	Example values	Description
action	string	"set"	Module operation items, "get":get; "set":set, this item is a mandatory item
code	int	0	error code, 0: request success, 1: request parameter content error, 2: the corresponding return parameter content was not found
msg	string	null	Add description of error code information for future expansion

Protocol version number

Add version number control, and the version number is placed in the HTTP protocol header in the following format:
 Master version number: When you make incompatible API changes,
 Minor version number: When you make a backward-compatible functional addition,
 The Revision number: When you do a backward compatible bug fix.
 Example: Api-Version: v1.2.0

Other

API protocols use all lowercase, middle-strike distinction, and names are nouns: example: http://192.168.1.168/api/param-config/motion

6、The description of the parameter configuration interface

parameters configuration is API protocol, the default first-level path is api, the second-level interface directory is the module category, and the third-level interface directory is the specific functional module to be operated.

- The list of corresponding module categories in the secondary directory is as follows:

Catalog Name	Description
system	System Configuration Module
network	Network Configuration Module
video	Audio and video encoding configuration module
image	Image Configuration Module
storage	Storage Configuration Module
event	Event Configuration Module
face	Face Configuration Module

- The list of corresponding modules in the three-level directory is as follows:

Catalog Name	Description
motion	Motion detection function module

Example: When operating a motion detection configuration, api is the first-level directory field, event is the second-level directory field, and motion is the third-level directory field, the URL is [api/event/motion]

- The data body is assembled by using the json format, and both set and get need to contain the following operational fields, which have the following field format:

Parameter Name	Type	Example values	Description
action	String	"set"	Module operation items, "get": get; "set": set, this item is mandatory

Special instructions for sliders or drop-down boxes that need to be limited in scope: For parameters that need to be given a range by the server, you need to add the xxMin and xxMax fields to specify the setting range For parameters that need to be given the drop-down content by the server, you need to add the xxList field to specify the drop-down content, the format is array.

- When setting parameters, you can send all the parameter settings or just a single one

7、Interface example

7.1、 device session

7.1.1、 Login process

Request method:

- POST

step:

- The client sends the plaintext user name, encrypted password and sessionid to the server.
- If the password is correct, the server adds a set cookie in the HTTP header and sends it to the client
- The client records its own cookie information
- The client needs to bring its own cookie information when making a request next time
- The server parses the cookie information to obtain the relevant information of the last access of the client
- Response client

Login Request URL:

- /api/session/login

Send example

```
{
  "action": "set",
  "data": {
    "username": "admin",
    "loginEncryptionType": "sha256-1",
    "password": "422EnnrG2puKTxVWm0gzCQ==",
    "sessionID": "e25ffee55bf17016f1c8f6047eb4ccda",
    "datetime": "2021-03-05T17:50:02"
  }
}
```

Return example:

```
{
  "code": 0,
  "data": {
    "cookie": "sessionID=a854e2311ab87eb06c16ffd06fd3ebf7"
  }
}
```

Parameter description

Parameter name	type	example value	description
username	string	username	
loginEncryptionType	string	"sha256-1"	login encryption type
password	string	"422EnnrG2puKTxVWm0gzCQ=="	encryption password 32 bit, hexadecimal, 64 characters
code	int		return error code of interface call
cookie	string	"093e73287b7ba6d7a873bcad2db"	text information of recording client status, 32 characters
sessionID	string	"093e73287b7ba6d7a873bcad2db"	conversation ID, 32 characters
datetime	string	"2021-03-05T17:50:02"	Date Time, format:YYYY-MM-DDThh:mm:ss

encryption process sha256 encryption algorithm

1) first round encryption

Date base64 encryption, date format: YYYY-MM-DDThh:mm:ss get DateBase64 character string

2) second round encryption

Encrypted character: username+LICENSE_KEY+DateBase64+password, encryption using sha256 encryption algorithm, return encrypted ciphertext.

tip: authority LICENSE_KEY, it is necessary to separate apply for every client connecting.

7.1.2. Logout login

brief description:

- device logout

Reuquest method:

- POST

Request URL:

- /api/session/logout

Send example:

```
{
  "action": "set",
  "data": {
    "cookie": "sessionID=a854e2311ab87eb06c16ffd06fd3ebf7"
```

```
}  
}
```

Return example:

```
{  
  "code": 0,  
  "msg": null  
  "data": null  
}
```

parameter description:

parameter name	type	example value	description
code	int	0	return error code
action	string	"set"	set operation method
cookie	string	"sessionID=a854e2311ab87eb06c16ffd06fd3ebf7"	cookie which was received in login

7.1.3、 session heartbeat mechanism

brief description:

- The client actively sends heartbeat packets to the server, with a heartbeat interval of 30s

Request method:

- POST

Request URL:

- /api/session/heart-beat

Heartbeat parameter return: { "code": 0 }

parameter description:

parameter name	type	example value	description
code	int	0	return error code

7.2、 System parameters configuration

7.2.1、 platform capability set attributes

Short description:

- Get a list of platform capability support

Request URL:

- /api/system/platform-capabilities

Request Method:

- POST

Get Example Get Parameter Request:

```
{
  "action": "get",
  "data": {
  }
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enableProductType": 0,
  }
}
```

Get parameter description:

Parameter Name	Type	Example values	Description
enableProductType	int		0:common ipc products 1:module ipc products 2:common ipc new naming rules

Note: This interface does not require to login device

7.2.2、device capability set attributes

Short description:

- Get a list of device capability support

Request URL:

- /api/system/capabilities

Request Method:

- POST

Get example: Get parameter request: type is 0

```
{
  "action": "get",
  "data": {
    "type": 0
  }
}
```

Get parameters returned: **capability set parameters as first class objects**

```
{
  "code": 0,
  "data": {
    "enableDefog": 1,
    "enableRotateAngle": 1,
    "enableBackLight": 1,
    "enableExposure": 1,
    "enableH265": 1,
    "enableWdr": 0,
    "enableSLInhibition": 1,
    "enableLdc": 1,
    "enableWaterMark": 1,
    "enableMirrorAndFlip": 0,
    "enableWhiteBalance": 0,
    "enableVideoAdjust": 0,
    "enableOsdEx": 0,
    "enableSmartir": 1,
    "enableIsoMode": 0,
    "enableHdr": 0,
    "enablePtz": 2,
    "enablePtzConfig": 0,
    "enableWdr3to1Show": 1,
    "enableElectronicAntiShake": 0,
    "enableFocus": 1,

    "deviceType": 231,
    "enablePasswdEncrypt": 1,
    "enableFactorySetting": 1,
    "enableNtpAutoCorrection": 1,
    "enableAbHs": 1,
    "enableRtspEncrypt": 1,

    "enableIpv6": 1,
    "enablePort": 1,
    "enableGb28181": 0,
    "enableBitdogPort": 1,
    "enableWifi": 0,
    "enableWifiNetworkManagement": 0,
    "enablePppoe": 1,
    "enableSnmp": 1,

    "enableAudio": 1,
    "enableAudioOut": 0,
    "enableTalk": 0,
  }
}
```

```
"enableAacHide": 1,
"enableSmartVideoRestart": 0,
"enableFish": 0,
"enableSmartVideoDisplay": 1,

"enableSdCard": 1,
"enableRec": 2,
"enableIpeye": 1,
"enableLimitRecFrameRate": 0,
"enableJpegCaptureHide": 1,
"enableRecDevTypeHide": 1,

"enableIo": 0,
"enableIoInput": 1,
"enablePrivacyMask": 0,
"enabelRoi": 1,
"enableRoiFrameRateEnable": 1,
"enableVideoTamper": 0,

"enableFace": 0,
"enableRegionalIntrusionDetectionus": 0,
"enableCrossBorderDetection": 0,
"enableSceneChangeDetection": 0,
"enableItemLegacyDetection": 0,
"enableItemPickingDetection": 0,
"enablePeopleStayDetection": 0,
"enablePeopleGatheringDetection": 0,
"enableSmartMotionDetection": 0,
"enableSmartPicture": 0,
"enableSmartModeHide": 1,
"enableMdRecStreamType": 0,
"enableNoUpload": 1,
}
}
```

Get parameter request: **type is not 0**

```
{
  "action": "get",
  "data": {
    "type": 1
  }
}
```

Get parameters returned: **capability set parameters as secondary objects**

```
{
  "code": 0,
  "data": {
    "image": {
      "enableDefog": 1,
      "enableRotateAngle": 1,
      "enableBackLight": 1,
      "enableExposure": 1,
      "enableH265": 1,
      "enableWdr": 0,
    }
  }
}
```

```

        "enableSLInhibition": 1,
        "enableLdc": 1,
        "enableWaterMark": 1,
        "enableMirrorAndFlip": 0,
        "enableWhiteBalance": 0,
        "enableVideoAdjust": 0,
        "enableOsdEx": 0,
        "enableSmartir": 1,
        "enableIsoMode": 0,
        "enableHdr": 0,
        "enablePtz": 2,
        "enablePtzConfig": 0
        "enableRotateOrLdc": 0
    }
}
}

```

Request parameter description:

Parameter Name	Type	Example values	Description
type	int		Request capability set type 0: All types of parameters 1: Image configuration parameters 2: System configuration parameters 3: Network configuration parameters 4: Audio and video configuration parameters 5: Storage configuration parameters 6: Event configuration parameters 7: smart event configuration parameters

Note: When type is 0, the capability set parameter is the first-level object; when type is not 0, the capability set parameter is the second-level object, and the object names of various configuration parameters are as follows.

- The list of corresponding module categories in the secondary directory is as follows:

Catalog Name	Description
system	System Configuration Module
network	Network Configuration Module
video	Audio and video encoding configuration module
image	Image Configuration Module
storage	Storage Configuration Module
event	Event Configuration Module
face	Face Configuration Module

Return parameter description:

- Image configuration parameters:

Parameter Name	Type	Example values	Description
----------------	------	----------------	-------------

Parameter Name	Type	Example values	Description
enableDefog	int		Defog enable 1:enable 0:disable
enableRotateAngle	int		Initialize corridor mode 1:enable 0:disable
enableBackLight	int		Initialize backlight 0:Remove backlight amount 1:Show backlight amount 2:Remove backlight
enableIsoMode	int		Initialize iISO mode 1:enable 0:disable
enableWhiteBalance	int		Initialize white balance (original variable: ucGreenGain) 0:remove green gain 1:show green gain 2:remove white balance
enableExposure	int		Initialize exposure 0:Close exposure 1:Open exposure (with auto gain) 3:Open exposure (without auto gain)
enableHdr	int		Initialize hdr 1:enable 0:disable
enableH265	int		Initialize h265+/h264+ 0:turn off smart+, turn off I frame interval 1:turn on smart + 2:turn off smart+
enableOsdEx	int		Initialize osd extensions Smart Platform
enableWdr	int		Remove wide dynamic function 0:Show 1:Don't show
enableLdc	int		Distortion correction 0:disable 1:enable
enableSLInhibition	int		New add HLC function 0:disable 1:enable
enablePtz	int		Initialize ptz 0:off 1:rotate+zoom 2:zoom 3:zoom 4:preset point+cruise
enablePtzConfig	int		Initialize network PTZ 1:enable 0:disable
enableVideoAdjust	int		Initialize video adjustment (original variable: 2DDNR) 0:Remove digital 2DNR 1:Show digital 2DNR 2:Remove video adjustment
enableSmartir	int		Initialize Smartir 0: Displays IRCUT.1 : Displays day/night switch, enables light mode. 2: No IRCUT, no Smartir.3 : Displays day/night switch, displays fill light mode.4 : Show day/night mode, show day/night switch.5 : Display day-night switch, hide light mode, hide light brightness. 6: basically added maintain full color
enableWaterMark	int		Watermark 0:disable 1:enable
enableMirrorAndFlip	int		Remove mirror function 0:Show 1:Don't show
enableElectronicAntiShake	int		Initialize electronic anti-shake 1:enable 0:disable
enableWdr3to1Show	int		Show three frames in one option 1:hide 0:show
enableFocus	int		Initialize focus mode 1:enable 0:disable
enableRotateOrLdc	int		Distortion correction and corridor mutually exclusive 0:not mutually exclusive 1:mutually exclusive

- System configuration parameters:

Parameter Name	Type	Example values	Description
deviceType	int		Device type Fill in ARG_PLATFORM_TYPE
enablePasswdEncrypt	int		Initialize key retrieval password 0:disable 1:enable

Parameter Name	Type	Example values	Description
enableFactorySetting	int		Restore factory settings with new full recovery option 0:disable 1:enable
enableNtpAutoCorrection	int		Initialize ucNtpAutoCorrection 0:disable 1:enable
enableAbHs	int		Initialize the online upgrade device type 0:Ambarella 1:Hisilicon 2:zoom 3:Fullhan 4:RK 5:MSTAR 6:GK
enableRtspEncrypt	int		Initialize onvif/rtsp encryption authentication 1:enable 0:disable
enableApp	int		Initialize app 0:bivision 1:ivs365

- Network configuration parameters:

Parameter Name	Type	Example values	Description
enableIpv6	int		IPV6 support 1:enable 0:disable
enableGb28181	int		Initialize GB28181 1:enable 0:disable
enableBitdogPort	int		Initialize BitdogPort 1:enable 0:disable 2:ALIOT
enablePort	int		HIKVISION XM port 0:Show HIKVISION + XM port 1:HIKVISION + XM 2:Show HIKVISION port
enableWifi	int		Initialize WIFI 1:enable 0:disable
enableWifiNetworkManagement	int		WIFI network management list 0:disable 1:enable
enablePppoe	int		Initialize Pppoe 1:enable 0:disable
enableFtp	int		Initialize Ftp 1:enable 0:disable
enableSntp	int		Initialize Sntp 1:enable 0:disable
enableDdns	int		Initialize Ddns 1:enable 0:disable
enableSnmp	int		SNMP enable 0:disable 1:enable

- Audio and video configuration parameters:

Parameter Name	Type	Example values	Description
enableAudio	int		Audio 1:enable 0:disable
enableTalk	int		Initialize intercom 1:enable 0:disable
enableAudioOut	int		Initialize AudioOut 0:input mic output no 1:input mic/line output external/internal 2:input line(initial 0) output external(initial 0) 3:input mic/line output no 4:hide audio input, audio output
enableSmartVideoRestart	int		Configure the video interface to display the switch resolution restart 0:disable 1:enable
enableAacHide	int		Hide AAC audio option 1:hide 0:show
enableFish	int		Initialize fish eye 1:enable 0:disable
enableSmartVideoDisplay	int		Configure the video interface display type currently 0 : Normal IPC video display interface 1 : Smart IPC

Parameter Name	Type	Example values	Description
			video display interface
enableRoi	int		ROI 1:enable 0:disable
enableRoiFrameRateEnable	int		Graying ROI frame rate setting 1:enable 0:disable
enableDigitalZoom	int		DigitalZoom 1:enable 0:disable
enableAudioAlarm	int		Voice alarm 1:enable 0:disable

- Storage configuration parameters:

Parameter Name	Type	Example values	Description
enableSdCard	int		Initialize sd card (used with ucREC) 1:enable 0:disable
enableRec	int		Initialize recording (used with ucSDCard) 1:enable 0:disable
enableIpeye	int		Initialize IPEYE 1:enable 0:disable
enableLimitRecFrameRate	int		Web recording frame rate limit 0:disable 1:enable
enableJpegCaptureHide	int		JPEG capture options 0:Show all options 1:Hide image format, resolution, start event capture, capture number, show capture interval 2:Hide all capture options
enableRecDevTypeHide	int		Remove storage device type 1:Show 0:Do not show
enableNoUpload	int		Upload not supported
enableCloudStorage	int		Cloud storage, 0:disable 1:enable
enableSnapshotPicRecording	int		Capture video 0:disable 1:enable

- Event configuration parameters:

Parameter Name	Type	Example values	Description
enableIo	int		Initialize alarm input 0:disable 1:alarm input + alarm output 2:alarm input
enableVideoTamper	int		Initialize masking alarm 1:enable 0:disable
enablePrivacyMask	int		Remove privacy masking 0:Show 1:Don't show
enableIoInput	int		Alarm recording and motion detection + alarm recording 1:enable 0:disable
enableMdRecStreamType	int		Initialize ucMDVideoStreamType

- Smart event configuration parameters:

Parameter Name	Type	Example values	Description
enableFace	int		Initial face recognition 0:None 1: Capture 2: Comparison 3: Thermal imaging
enableRegionalIntrusionDetectionus	int		Area intrusion detection 1:enable

Parameter Name	Type	Example values	Description
			0:disable
enableCrossBorderDetection	int		Cross border detection 1:enable 0:disable
enableSceneChangeDetection	int		Scene change detection 1:enable 0:disable
enableItemLegacyDetection	int		Item legacy detection 1:enable 0:disable
enableItemPickingDetection	int		Item picking detection 1:enable 0:disable
enablePeopleStayDetection	int		People stay detection 1:enable 0:disable
enablePeopleGatheringDetection	int		People Wandering detection 1:enable 0:disable
enableSmartMotionDetection	int		Configure the humanoid warning function 0:disable 1:enable
enableSmartPicture	int		Configure picture function 0:disable 1:enable
enableSmartModeHide	int		Smart detection mode 1:Display 0:No display
enableSmartEventDisplay	int		Smart event 0:disable 1:enable
enableSmartResource	int		Smart resource allocation 0:disable 1:enable

7.2.3、Device information attribute

Brief description:

- Interface to obtain device information parameters

Request URL:

- /api/system/device-info

Request Method:

- POST

Get example Get parameter request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
```

```
"code": 0,
"data": {
  "devType": 0,
  "platformType": 182,
  "devName": "IPC",
  "devId": "8888888888888888",
  "manufacturers": "",
  "softwareVersion": "0.1.12.6",
  "firmwareVersion": "3516CV500_IMX307_X10T1A1M0C0P1_W_20.1.12.6",
  "platform": "16CV500",
  "devSensor": "307"
}
}
```

Post example setting up a parameter request:

```
{
  "action": "set",
  "data": {
    "devName": "IPC"
  }
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter Description:

Parameter Name	Type	Example values	Description
devType	int		Device Type
platformType	int		Platform type Fill in ARG_PLATFORM_TYPE
devName	string		Device name
devId	string		Device ID
manufacturers	string	Manufacturer	
platform	string		Platform Type
devSensor	string	Sensor Type	
firmwareVersion	string		Firmware Version

7.2.4. System time parameters configuration attribute

Brief description:

- System time parameter configuration attribute

Request URL:

- /api/system/system-time

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "year": 2020,
    "month": 4,
    "day": 16,
    "hour": 4,
    "minute": 18,
    "second": 38,
    "model": 0,
    "timeZone": 24,
    "nvrModify": 0,
    "ntpServerAddr": "time.windows.com",
    "ntpAutomatic": 0,
    "ntpInterval": 0
  }
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "year": 2020,
    "month": 4,
    "day": 16,
    "hour": 4,
    "minute": 18,
    "second": 38,
    "model": 0,
    "timeZone": 24,
    "nvrModify": 0,
    "ntpServerAddr": "time.windows.com",
    "ntpAutomatic": 0,
    "ntpInterval": 0
  }
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter description:

Parameter Name	Type	Example values	Description
year	int		Year
month	int		Month
day	int		Day
hour	int		Hour
minute	int	minute	
second	int		second
model	int		Time mode, 0 manual setting, 1 synchronization with PC, 2 NTP service
timeZone	int		Time zone, (-12.0)~(12.0) Default is East 8
nvrModify	int	(ipc only) whether to allow nvr to modify ipc time, 0 no 1 yes	
ntpAutomatic	bool		NTP auto time switch, 0 off 1 on
ntpInterval	int		NTP Automatic time interval
ntpServerAddr	string	"time.windows.com"	NTP server address
code	int		Return error code

7.2.5、Automatic maintenance parameters configuration attribute

Brief description:

- Automatic maintenance of parameter configuration attribute

Request URL:

- /api/system/auto-maintain

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
}
```

```
"data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": 1
    "type": 2,
    "week": 3,
    "day": 3,
    "hour": 3,
    "minute": 3,
    "second": 3,
  }
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "enable": 1
    "type": 2,
    "week": 3,
    "day": 3,
    "hour": 3,
    "minute": 3,
    "second": 3,
  }
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter description:

Parameter Name	Type	Example values	Description
enable	bool		Whether to turn on timed restart, false off, true on
type	int		Restart mode, 0 daily, 1 weekly, 2 monthly
day	int		Restart time, unit days
hour	int		Restart time, unit hours
minute	int		Restart time, unit minutes
second	int		Restart time, in seconds
week	int		0-6 Sunday to Saturday, respectively
code	int		Return error code

7.2.6、Daylight saving time parameters configuration properties

Short description:

- Get the daylight saving time parameter configuration attribute

Request URL:

- /api/system/summer-time

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": 1,
    "start": {
      "month": 3,
      "weekNumber": 1,
      "week": 0,
      "hour": 2,
      "day": 0
    },
    "end": {
      "month": 9,
      "weekNumber": 5,
      "week": 0,
      "hour": 2,
      "day": 0
    },
    "offsetTime": 30,
    "dstType": 1
  }
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
```



```

"enable": 1,
"start": {
  "month": 3,
  "weekNumber": 1,
  "week": 0,
  "hour": 2,
  "day": 0
},
"end": {
  "month": 9,
  "weekNumber": 5,
  "week": 0,
  "hour": 2,
  "day": 0
},
"offsetTime": 30,
"dstType": 1
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter Description:

Parameter Name	Type	Example values	Description
enable	bool		Daylight saving time switch, true:on false:off
dstType	int		Daylight Saving Time mode, 0 dates 1 week
month	int		Month
day	int		Day
hour	int		Hour
week	int		Week
weekNumber	int		Week Serial Number
offsetTime	int		Offset time, default is one hour

7.2.7、Reboot device attribute

Brief description:

- Reboot device attribute

Request URL:

- /api/system/reboot

Request Method:

- POST

Sending example

```
{
  "action": "set",
  "data": null
}
```

Return Example

```
{
  "code": 0,
  "data": {}
}
```

Parameter Name	Type	Example values	Description
code	int		Return Code

7.2.8、Restore the default value attribute

Brief description:

- Restore default attribute

Request URL:

- /api/system/default

Request Method:

- POST

Sending example

```
{
  "action": "set",
  "data": {
    "type": 0
  }
}
```

Return Example

```
{
  "code": 0,
  "data": {}
}
```

Description of sending parameters

Parameter Name	Type	Example values	Description
type	int		Restore device parameters type (0: simple recovery 1: full recovery to factory settings)
code	int		Return error code

7.2.9. Export parameters save path

Brief description:

- Save path for exported parameters

Request URL:

- `/api/system/export-params`

Request Method:

- POST

Sending example

```
{
  "action": "get",
  "data": null
}
```

Return Example

```
{
  "code": 0,
  "data": {
    "url": "/filedir/3516DV200_IMX335_B1T1A1M0C0P1_W_20.1.2.3.zip",
    "filename": "3516DV200_IMX335_B1T1A1M0C0P1_W_20.1.2.3.zip"
  }
}
```

Get the parameter file *Request URL:**

- `/filedir/3516DV200_IMX335_B1T1A1M0C0P1_W_20.1.2.3.zip`

Request Method:

- GET

Get parameter file returns:

Get parameter description:

Parameter Name	Type	Example values	Description
url	string	"/filedir/3516DV200_IMX335_B1T1A1MOCOP1_W_20.1.2.3.zip"	Paths
filename	string	"3516DV200_IMX335_B1T1A1MOCOP1_W_20.1.2.3.zip"	File name

7.2.10、Log management parameters configuration attribute

Brief description:

- Log Management

Request URL:

- /api/system/log-manager

Request Method:

- POST

Request example: Log search request:

```
{
  "action": "get",
  "data": {
    "startTime": "1586995200",
    "endTime": "1587081599",
  }
}
```

Log search returns:

```
{
  "code": 0,
  "data": {
    "logList": [
      {
        "idx": 1,
        "time": "1587010286",
        "type": 1,
        "message": "Clear system log"
      },
      {
        "idx": 2,
        "time": "1587010173",
        "type": 1,
        "message": "Clear system log"
      }
    ]
  }
}
```

Delete log request:

```
{
  "action": "delete",
  "data": {}
}
```

Return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter description:

Parameter Name	Type	Example values	Description
startTime	string		Start time
endTime	string		Ending time
idx	int		Index
time	string		1970 to present Seconds
message	string		Event Description
type	int		Type

7.3、 Network parameters configuration

7.3.1、 TCP/IP parameters configuration attribute

Brief description:

- TCP/IP parameters configuration attribute

Request URL:

- /api/network/interfaces

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```

{
  "code": 0,
  "data": {
    "enableDhcp": false,
    "enableAllNetConnect": false,
    "ipAddrStatus": 0,
    "ip": "172.18.193.180",
    "netMask": "255.255.248.0",
    "gateway": "172.18.192.1",
    "dns1": "172.18.192.1",
    "dns2": "172.18.192.2",
    "ipv6Mode": 0,
    "maxTransUnit": 0,
    "ipv6": "",
    "subNetMaskV6": 0,
    "gatewayV6": "",
    "multicast": "",
    "mac": "B2:4E:B1:3D:42:C6"
  }
}

```

Setting Example: Set the parameter request:

```

{
  "action": "set",
  "data": {
    "enableDhcp": false,
    "enableAllNetConnect": false,
    "enableIpAddrTest": 0,
    "ip": "192.168.1.168",
    "netMask": "255.255.255,0",
    "gateway": "192.168.1.1",
    "dns1": "192.168.1.1",
    "dns2": "192.168.1.1",
    "ipv6Mode": 0,
    "maxTransUnit": 0,
    "ipv6": "",
    "subNetMaskV6": 0,
    "gatewayV6": "",
    "multicast": "",
    "mac": "11:22:33:44:55:66"
  }
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter Description

Parameter Name	Type	Example values	Description
enableDhcp	bool		enable dhcp, false off , true on

Parameter Name	Type	Example values	Description
enableAllNetConnect	bool		enable all-net connect, false off , true on
enableIpAddrTest	int		Enable IP address test, 0:off 1:on
ipAddrStatus	int		IP address status,0:available 1:not available
ip	string		IPV4 address
netMask	string		IPV4 Subnet Mask
gateway	string		IPV4 Gateway
dns1	string		dns server 1
dns2	string		dns server 2
ipv6Mode	int		IPV6 mode:fill in ARG_IPV6MODE_TYPE
maxTransUnit	int		MTU
ipv6	string		IPV6 address
subNetMaskV6	int		IPV6 Subnet Mask
gatewayV6	string		IPV6 Gateway
multicast	string		Multicast Address
mac	string		Mac Server Address

7.3.2、Protocol port configuration attribute

Brief description:

- Protocol port configuration attribute

Request URL:

- /api/network/port

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": {
    "type": 255
  }
}
```

Get parameter return:

```
{
  "code": 0,
  "data": [
    {

```

```

        "type":1,
        "enable":true,
        "property":{"rtspProtocol ":{ "port":554,"encrypt":0}}
    },
    {
        "type":2,
        "enable":true,
        "property":{"hkProtocol ":{ "port":0,"enable":true}}
    },
    {
        "type":8,
        "enable":true,
        "property":{"onvifProtocol ":{ "authentication":1,"port":8
999,"enable":true}}
    },
    {
        "type":16,
        "enable":true,
        "property":{"httpProtocol ":{ "port":80}}
    },
    {
        "type":64,
        "enable":true,
        "property":{"httpsProtocol ":{ "port":443}}
    },
    {
        "type":32,
        "enable":true,
        "property":{"
            "xmProtocol":{"port":0,"enable":true}}
    },
    {
        "type":72,
        "enable":true,
        "property":{"bitdogProtocol ":{ "port":6000,"encrypt":0}}
    }
]
}

```

Setting Example: Set the parameter request:

```

{
  "action":"set",
  "data":[
    {
      "type":1,
      "enable":true,
      "property":{"rtspProtocol ":{ "port":554,"encrypt":0}}
    },
    {
      "type":2,
      "enable":true,
      "property":{"hkProtocol ":{ "port":8000,"enable":true}}
    },
    {
      "type":8,
      "enable":true,
      "property":{"onvifProtocol ":{ "authentication":1,"port":8
999,"enable":true}}
    }
  ]
}

```



```

    },
    {
      "type":16,
      "enable":true,
      "property":{"httpProtocol ":{ "port":80}}
    },
    {
      "type":64,
      "enable":true,
      "property":{"httpsProtocol ":{ "port":443}}
    },
    {
      "type":32,
      "enable":true,
      "property":{"xmProtocol":{ "port":34576,"enable":true}
    }
  },
  {
    "argType":72,
    "enable":true,
    "property":{"bitdogProtocol ":{ "port":6000,"encrypt":0}}
  }
]
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter Description

Parameter Name	Type	Example values	Description
enable	bool		Enabled or not false off true enabled
type	int		Protocol type: Fill in ARG_AGREEMENT_TYPE
rtspProtocol	object	{ "port": 554, "encrypt": 0 }	rtsp protocol
onvifProtocol	object	{ "authentication": 1, "port": 8999, "enable": true }	onvif Protocol
httpProtocol	object	{ "port": 80 }	http protocol
hkProtocol	object	{ "port": 0, "enable": false }	hk agreement
xmProtocol	object	{ "port": 0, "enable": false }	xm protocol
port	int		Protocol Port
encrypt	int		Bitdog Ssl encryption 1:enable 0:disable
authentication	int	Whether to open ID authentication: 0 off 1 on	

7.3.3、DDNS parameters configuration attribute

Brief description:

- DDNS parameter configuration attribute, you need to get the encryption capability set first, and then get the parameters.

Request URL:

- /api/network/ddns

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": {
    "loginName": "admin",
    "encryptionType": "aes128-1",
    "datetime": "2021-03-06T10:47:05",
    "iv": "afe13ds34cdjk08c"
  }
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": false,
    "serverType": 0,
    "status": 0,
    "serverIp": "",
    "serverPort": 0,
    "domain": "",
    "defaultDomain": "",
    "username": "422EnnrG2puKTxVWm0gzCQ==",
    "alias": "",
    "alivePeriod": 0,
    "serviceType": 0
  }
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "enable": false,
    "serverType": 0,
  }
}
```

```

    "status": 0,
    "serverIp": "",
    "serverPort": 0,
    "domain": "",
    "defaultDomain": "",
    "username": "422EnnrG2puKTxVWm0gzCQ==",
    "password": "422EnnrG2puKTxVWm0gzCQ==",
    "alias": "",
    "keepAlive": 0,
    "serviceType": 0,
    "encryptionType": "aes128-1",
    "datetime": "2021-03-06T10:47:05",
    "iv": "abcde1234567890"
  }
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter Description

Parameter Name	Type	Example values	Description
loginName	string	admin	Login User Name
enable	bool		Whether to enable, false close , true enable
serverType	int		Server Type 0-Peanut Shell, 1-NOIP, 2-Dyn, 8-Planet Dynamic DNS, 9-Planet Easy DDNS
serverIp	string		Server ip
serverPort	int		Server port
domain	string		Dynamic Domain
defaultDomain	string		Default Domain Name
username	string		User name, encryption, for AES128 encryption type, encrypted data for base64 encoding
password	string		Password, encryption, for AES128 encryption type, encrypted data for base64 encoding
alias	string		Alias
keepAlive	int		Keep alive time
serviceType	int		Service Type: 0 - Normal User 1 - Administrator
encryptionType	string	"aes128-1"	Encryption type.
status	int	0	Login status 0: Successful 1: Login in progress
iv	string	"abcde1234567890"	Offset ,16in characters, for AES encryption type
datetime	string	"abcde1234567890"	Date time. Format: "YYYY-MM-DDThh:mm:ss"

Encryption instructions

The transmission of sensitive information such as user name and password is encrypted by ase128 CBC algorithm, encryption padding zeropadding, encryption key and offset iv. The

Encryption instructions

encryption key is generated by the client and the server by their own calculation of the agreed keyType, and is not passed. The offset iv and time are passed to the server in the parameters when the content is requested.

7.3.4、FTP parameters configuration attribute

Brief description:

- FTP parameter configuration attribute, you need to get the encryption capability set first, and then get the parameters.

Request URL:

- /api/network/ftp

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": {
    "loginName": "admin",
    "encryptionType": "aes128-1",
    "datetime": "2021-03-06T10:47:05",
    "iv": "abcde1234567890"
  }
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": 1,
    "serverIp": "192.168.1.1",
    "serverPort": 21,
    "username": "422EnnrG2puKTxVWm0gzCQ==",
    "enableAnonymous": 0,
    "path": "Default_Folder",
    "enableAutoCover": 0,
    "attfileFormat": 1,
    "firstDir": 0,
    "secondDir": 0
  }
}
```

Setting Example: Set the parameter request:

```

{
  "action": "set",
  "data": {
    "enable": 1,
    "serverIp": "192.168.1.1",
    "serverPort": 21,
    "username": "422EnnrG2puKTxVWm0gzCQ==",
    "enableAnonymous": 0,
    "password": "422EnnrG2puKTxVWm0gzCQ==",
    "path": "Default_Folder",
    "enableAutoCover": 0,
    "attfileFormat": 1,
    "enableFtpTest": 0,
    "firstDir": 0,
    "secondDir": 0,
    "securityType": "aes128-1",
    "datetime": "2021-03-06T10:47:05",
    "iv": "abcde1234567890"
  }
}

```

Set parameter return (ftpTestEnable = 0):

```

{
  "code": 0,
  "data": {}
}

```

Set parameter return (ftpTestEnable = 1):

```

{
  "code": 0,
  "data": {
    "ftpTestStatus": 1
  }
}

```

Parameter Description

Parameter Name	Type	Example values	Description
loginName	string	admin	Login User Name
enable	bool		Whether to turn on false not required, true enable
attfileFormat	int		Upload attachment format, 0:Message 1:JPEG 2:AVI
enableAnonymous	int		Anonymous 0:Not enabled 1:Enabled
serverIp	string	"192.168.1.1"	Server Address
serverPort	int	21	Port
username	string	"admin"	User name, need to encrypt the transmission, for AES128 encryption type, encrypted data for base64 encoding
password	string	"admin"	Password, need to encrypt the transmission,

Parameter Name	Type	Example values	Description
			for AES128 encryption type, encrypted data for base64 encoding
path	string	"Default_Folder"	Paths
enableAutoCover	int		Auto cover 0:Not enabled 1:Enabled
enableFtpTest	int		FTP test 0:Not enabled 1:Enabled
ftpTestStatus	int		FTP test status 0:successful 1:unsuccessful
firstDir	int		FTP first level directory 0:IP address 1:OSD 2:Custom (using data from parameter path)
secondDir	int		FTP secondary directory 0:Chinese 1:English
encryptionType	string	"aes128-1"	Encryption type. Get at the encryption capability set
iv	string	"abcde1234567890"	Offset ,16in characters, for AES encryption type
datetime	string	Date time. Format: "YYYY-MM-DDThh:mm:ss"	

Encryption instructions

The transmission of sensitive information such as user name and password is encrypted by aes128 CBC algorithm, encryption padding zeropadding, encryption key and offset iv. The encryption key is generated by the client and the server by their own calculation of the agreed keyType, and is not passed. The offset iv and time are passed to the server in the parameters when the content is requested.

7.3.5、Smtip parameters configuration attribute

Brief description:

- Smtip parameter configuration attribute, you need to get the encryption capability set first, and then get the parameters.

Request URL:

- /api/network/smtip

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": {
    "loginName": "admin",
    "datetime": "2021-03-06T10:47:05",
    "iv": "abcde1234567890",
    "encryptionType": "aes128-1"
  }
}
```

```
}  
}
```

Get parameter return:

```
{  
  "code": 0,  
  "data": {  
    "enable": false,  
    "attfileformat": 0,  
    "alarmDuration": 0,  
    "serverPort": 25,  
    "anonymity": false,  
    "smtpEncryptionType": 0,  
    "time": 100,  
    "accessory": 100,  
    "accessoryType": 100,  
    "username": "404561426@qq.com",  
    "server": "SMTP.domain.com",  
    "sender": "User@domain.com",  
    "receiver1": "404561426@qq.com",  
    "receiver2": "404561426@qq.com",  
    "receiver3": "404561426@qq.com",  
    "theme": "IP Camera: Motion Detected"  
  }  
}
```

Setting Example: Set the parameter request:

```
{  
  "action": "set",  
  "data": {  
    "enable": true,  
    "sender": "User@domain.com",  
    "server": "SMTP.domain.com",  
    "serverPort": 25,  
    "attfileformat": 0,  
    "alarmDuration": 0,  
    "smtpEncryptionType": 0,  
    "username": "404561426@qq.com",  
    "password": "009EnnrG2puKTxVWm0gzCQ==",  
    "receiver1": "404561426@qq.com",  
    "receiver2": "404561426@qq.com",  
    "receiver3": "404561426@qq.com",  
    "anonymity": false,  
    "time": 100,  
    "accessory": 100,  
    "accessoryType": 100,  
    "theme": "IP Camera: Motion Detected",  
    "enableSmtptest": 1,  
    "loginName": "admin",  
    "datetime": "2021-03-06T10:47:05",  
    "iv": "abcde1234567890",  
    "securityType": "aes128-1"  
  }  
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {
    "smtpTestStatus": 0
  }
}
```

Parameter Description

Parameter Name	Type	Example values	Description
enable	bool		Whether to enable false not required, true enable
attfileformat	int		Upload attachment format, 0:Message 1:JPEG 2:AVI
alarmDuration	int		Alarm upload interval, 0:10 seconds 1:30 seconds 2:1 minute 3:5 minutes 4:10 minutes 5:always valid 6:1 second
serverPort	int		SMTP port
anonymity	bool		Anonymous 0 Anonymous ,1 not anonymous
smtpEncryptionType	int		Email encryption type 0 no encryption, 1 SSL
time	int	Sending time	
accessory	int		Whether to add accessories, 0 add, 1 don't add
accessoryType	int		accessory Type
username	string	"User@domain.com"	User Name
password	string	"12345678"	Password, for AES128 encryption type, encrypted data for base64 encoding
server	string	"SMTP.domain.com"	Server Address
sender	string	"User@domain.com"	Send Address
receiver1	string	"459EnnrG2puKTxVWm0gzCQ=="	Receiving Address 1
receiver2	string		Receiving Address 2
receiver3	string		Receiving Address 3
theme	string	"IP Camera: Motion Detected"	theme
enableSmtptest	int	1	Mailbox test 0: No enable 1: Enable receiving Address 1 2: Enable receiving Address 2 3: Enable receiving Address 3
smtpTestStatus	int		Mailbox test success or failure 0: test success 1: test failure
encryptionType	string	"aes128-1"	Encryption type
iv	string	"abcde1234567890"	Offset ,16in characters, for AES encryption type

Parameter Name	Type	Example values	Description
datetime	string	Date and time, format: "YYYY-MM-DDThh:mm:ss"	
Encryption instructions			
The transmission of sensitive information such as user name and password is encrypted by aes128 CBC algorithm, encryption padding zeropadding, encryption key and offset iv. The encryption key is generated by the client and the server by their own calculation of the agreed keyType, and is not passed. The offset iv and time are passed to the server in the parameters when the content is requested.			

7.3.6、P2P parameters configuration attribute

Brief description:

- P2P parameters configuration attribute

Request URL:

- /api/network/p2p

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": true,
    "serialNumber": "88888888888888",
    "iosAppAddr": "https://itunes.apple.com/app/id1419463554",
    "androidAppAddr": "https://play.google.com/store/apps/details?id=com.gzch.lsplat.bitdog",
  }
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "enable": true
  }
}
```

```
}
```

Set the parameters to return:

```
{  
  "code": 0,  
  "data": {}  
}
```

Parameter Description

Parameter Name	Type	Example values	Description
enable	bool		Whether to enable freeip
serialNumber	string	"88888888888888"	Serial number
iosAppAddr	string	" https://itunes.apple.com/app/id1419463554 "	Apple app address
androidAppAddr	string	" https://play.google.com/store/apps/details?id=com.gz.ch.lsplat.bitdog "	Android app address

7.3.7、P2P status parameters configuration attribute

Brief description:

- P2P status parameters configuration attribute

Request URL:

- /api/network/p2p-status

Request Method:

- POST

Get Example: Get Parameter Request:

```
{  
  "action": "get",  
  "data": null  
}
```

Get parameter return:

```
{  
  "code": 0,  
  "data": {  
    "status": 0,  
  }  
}
```

```
}

```

Parameter Description

Parameter Name	Type	Example values	Description
status	int		p2p status: 0 offline 1 online

7.3.8、PPPOE parameters configuration attribute

Brief description:

- PPPOE configuration parameters interface, you need to get the encryption capability set first, and then get the parameters.

Request URL:

- /api/network/pppoe

Request Method:

- POST

Get example: Get parameter request:

```
{
  "ation": "get",
  "data": {
    "loginName": "admin",
    "datetime": "2021-03-06T10:47:05",
    "iv": "abcde1234567890",
    "encryptionType ": "aes128-1"
  }
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": true,
    "username": "422EnnrG2puKTxVWm0gzCQ==",
    "ip": "0.0.0.0"
  }
}
```

Setting Example: Set the parameter request:

```
{
  "ation": "set",

```

```

    "data": {
      "encryptionType": "aes128-1",
      "datetime": "2021-03-06T10:47:05",
      "iv": "abcde1234567890",
      "enable": true,
      "username": "422EnnrG2puKTxVWm0gzCQ==",
      "password": "422EnnrG2puKTxVWm0gzCQ=="
    }
  }
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": null
}

```

Parameter description:

Parameter Name	Type	Example values	Description
enable	bool	Enabled, false off, true on	
username	string	"422EnnrG2puKTxVWm0gzCQ=="	User name, need to encrypt, for AES128 encryption type, encrypted data for base64 encoding
password	string	"422EnnrG2puKTxVWm0gzCQ=="	Password, need to encrypt, for AES128 encryption type, encrypted data for base64 encoding
encryptionType	string	"aes128-1"	Encryption type.
iv	string	"abcde1234567890"	Offset ,16in characters, for AES encryption type
datetime	string	Date time. Format: "YYYY-MM-DDThh:mm:ss"	
enable	bool		Enabled, false off, true on
ip	string	"0.0.0.0"	Device Dynamic IP
code	int		Return error code

Encryption instructions

The transmission of sensitive information such as user name and password is encrypted by aese128 CBC algorithm, encryption padding zeropadding, encryption key and offset iv. The encryption key is generated by the client and the server by their own calculation of the agreed keyType, and is not passed. The offset iv and time are passed to the server in the parameters when the content is requested.

7.3.9、GB28181 parameters configuration attribute

Brief description:

- GB28181 parameter interface, you need to get the encryption capability set first, and then get the parameters.

Request URL:

- /api/network/gb28181

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
  "data": {
    "loginName": "admin",
    "encryptionType": "aes128-1",
    "dateTime": "2021-03-06T10:47:05",
    "iv": "abcde1234567890"
  }
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": true,
    "version": 1,
    "sipServerId": "34020000002000000001",
    "serverDomainName": "3402000000",
    "sipServerIp": "192.168.1.1",
    "sipServerPort": 5060,
    "sipLocalPort": 5060,
    "deviceId": "34020000001320000001",
    "deviceCheckId": "34020000001320000001",
    "devicePwd": "admin123456",
    "civiilCode": "34020000001320000001",
    "resignerState": 0,
    "steamType": 1,
    "transportProtocol": 0,
    "registerExpiry": 3600,
    "keepAliveCycleTime": 60,
    "resignerCycleTime": 60,
    "maxKeepAliveTimes": 3,
    "sipAlarmChannelId": "34020000001320000001",
    "sipVideoChannelId": "34020000001320000001",
    "sipAudioOutChannelId": "34020000001320000001",
  }
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "loginName": "admin",
  }
}
```

```
"encryptionType": "aes128-1",
"dateTime": "2021-03-06T10:47:05",
"iv": "abcde1234567890"
"enable": true,
"version": 1,
"sipServerId": "34020000002000000001",
"serverDomainName": "3402000000",
"sipServerIp": "192.168.1.1",
"sipServerPort": 5060,
"sipLocalPort": 5060,
"deviceId": "34020000001320000001",
"deviceCheckId": "34020000001320000001",
"devicePwd": "admin123456",
"civilCode": "34020000001320000001",
"registerStatus": 0,
"steamType": 1,
"transportProtocol": 0,
"registerExpiry": 3600,
"keepAliveCycleTime": 60,
"registerCycleTime": 60,
"maxKeepAliveTimes": 3,
"sipAlarmChannelId": "34020000001320000001",
"sipVideoChannelId": "34020000001320000001",
"sipAudioOutChannelId": "34020000001320000001",

}
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Return parameter description

Parameter Name	Type	Example values	Description
enable	bool		GB28181 function enable false:off true:on
version	int		Protocol Version 0:GB/T28181-2016
serverId	string	"34020000002000000001"	SIP Server ID
serverDomainName	string	"3402000000"	SIP server domain
serverIp	string	"192.168.1.1"	SIP server ip address
serverPort	int	5060	SIP server port
sipLocalPort	int	5060	Local SIP port
deviceId	string	"34020000001320000001"	SIP user (device) name
deviceCheckId	string	"34020000001320000001"	SIP user (device) ID authentication
devicePwd	string	"admin123456"	SIP user password, encryption, for AES128 encryption type, encrypted data for base64 encoding
civilCode	string	"34020000001320000001"	Administrative district code
registerStatus	int		Registration status 0:offline 1:Online

Parameter Name	Type	Example values	Description
steamType	int		28181Code stream index 0:main stream 1:sub stream 2:third stream
transportProtocol	int		Transport Protocol 0:TCP 1:UDP
registerExpiry	int	3600	Registration expiry 3600 seconds
keepAliveCycleTime	int	60	Keep alive cycle 60 seconds
registerCycleTime	int	60	Registration interval 60 seconds
maxKeepAliveTimes	int		Max keep alive 3
sipAlarmChannelId	string	"34020000001320000001"	Alarm input code ID
sipVideoChannelId	string	"34020000001320000001"	Video channel code ID
sipAudioOutChannelId	string	"34020000001320000001"	Voice output channel code ID
encryptionType	string	"aes128-1"	Encryption type.
iv	string	"abcde1234567890"	Offset ,16in characters, for AES encryption type
dateTime	string	"abcde1234567890"	Date and time, format: "YYYY-MM-DDThh:mm:ss"

Remarks

- For more return error codes, please see the error code descriptions on the home page.

Encryption instructions

The transmission of sensitive information such as user name and password is encrypted by aese128 CBC algorithm, encryption padding zeropadding, encryption key and offset iv. The encryption key is generated by the client and the server by their own calculation of the agreed keyType, and is not passed. The offset iv and time are passed to the server in the parameters when the content is requested.

7.3.10、WIFI parameters configuration attribute

Brief description:

- WIFI parameters configuration attribute

Request URL:

- /api/network/wifi

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
```

```
"action":"get",
"data":null
}
```

Get parameter return:

```
{
  "code":0,
  "data":{
    "wifiInfo":[,
      {
        "ssid":"03WiFi",
        "channel": 0,
        "password": "",
        "ssidSafeKey": "",
        "ssidType": 0,
        "ssidSafeType": 0,
        "strength": 40,
        "isScane": 0,
        "passwordType": 1,
        "connectStatus": 0,
        "enableDhcp": 0,
        "ip": "",
        "netMask": "",
        "gateway": "",
        "dns1": "",
        "dns2": "",
        "mac": ""
      },
      {
        "ssid":"TP-LINK_48E708",
        "channel": 0,
        "password": "",
        "ssidSafeKey": "",
        "ssidType": 0,
        "ssidSafeType": 0,
        "strength": 40,
        "isScane": 0,
        "passwordType": 1,
        "connectStatus": 0,
        "enableDhcp": 0,
        "ip": "",
        "netMask": "",
        "gateway": "",
        "dns1": "",
        "dns2": "",
        "mac": ""
      }
    ],
  }
}
```

Setting Example: Set the parameter request:

```
{
  "action":"set",
  "data":{
    "connectStatus": 1,
  }
}
```



```

"ssid": "MERCURY_CBDD60-123",
"password": "123456789",
"passwordType": 1,
"enableDhcp": 1,
"ip": "192.168.1.102",
"netMask": "255.255.255.0",
"gateway": "192.168.1.1",
"dns1": "211.136.192.6",
"dns2": "",
"channel": 0,
"netCardNum": 0,
"ssidSafeKey": "",
"ssidType": 0,
"ssidSafeType": 0,
"strength": 100,
"isScane": 0,
"mac": "",
"opertateType": 2,
"loginName": "admin",
"encryptionType": "aes128-1",
"datetime": "2021-03-06T10:47:05",
"iv": "abcde1234567890"
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter Description

Parameter Name	Type	Example values	Description
loginName	string	admin	Login User Name
wifiInfo	array		Information on connectable WiFi
connectStatus	int		Connection status, 0:Not connected, 1:Connected
ssid	string	"MERCURY_CBDD60-123"	wifi name
password	string	"123456789"	Encryption password, for AES128 encryption type, encrypted data for base64 encoding, length range 8-15 bytes
passwordType	int		Password type, 0:no password, 1:with password
enableDhcp	int		DHCP enable, 0:off, 1:enable
ip	string	"192.168.1.102"	IP Address
netMask	string	"255.255.255.0"	subnet mask
gateway	string	"192.168.1.1"	Gateway
dns1	string	"211.136.192.6"	dns server 1
dns2	string		dns server 2

Parameter Name	Type	Example values	Description
channel	int		Channel number
ssidSafeKey	string		Security Network Types
ssidType	int		Encryption Type
ssidSafeType	int		Security Type
strength	int	100	0-100 100 strongest
isScane	int		0 Scan to get, 1 manually added
mac	string		Mac server address: fill in the string e.g. "A1:B2:C3:D4:E5:F6"
opertateType	int	0 (add) 、 1 (del) 、 2 (connect) 、 3 (disconnect)	WiFi operation type, fill in HS_WIFI_OPERATE_TYPE
encryptionType	string	"aes128-1"	Encryption type. Get at the encryption capability set
iv	string	"abcde1234567890"	Offset ,16 characters, for AES encryption type
datetime	string		Date time. Format: "YYYY-MM-DDThh:mm:ss"

Encryption instructions

The transmission of sensitive information such as user name and password is encrypted by aes128 CBC algorithm, encryption padding zeropadding, encryption key and offset iv. The encryption key is generated by the client and the server by their own calculation of the agreed key Type, and is not passed. The offset iv and time are passed to the server in the parameters when the content is requested.

7.3.11、Snm parameters configuration attribute

Brief description:

- Snmp parameter configuration attribute, you need to get the encryption capability set first, and then get the parameters

Request URL:

- `api/network/snmp`

Request Method:

- POST

Get Example Get Parameter Request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": false,
    "enableType": 1,
    "serverPort": 161,
    "readCommunity": "public",
    "writeCommunity": "private",
    "trapHostIP": "",
    "trapHostPort": 162,
    "trapName": "public",
    "roUser": {
      "username": "public",
      "secLevel": 1,
      "authType": 1,
      "privType": 1,
    }
  }
}
```

Setting example Setting parameter request:

```
{
  "action": "set",
  "data": {
    "enable": false,
    "enableType": 1,
    "serverPort": 161,
    "readCommunity": "public",
    "writeCommunity": "private",
    "trapHostIP": "",
    "trapHostPort": 162,
    "trapName": "public",
    "roUser": {
      "username": "public",
      "secLevel": 1,
      "authType": 1,
      "privType": 1,
      "authpass": "12345678",
      "privpass": "12345678"
    },
    "loginName": "admin",
    "encryptionType": "aes128-1",
    "datetime": "2021-03-06T10:47:05",
    "iv": "abcde1234567890"
  }
}
```

Set parameters to return

```

{
  "code": 0,
  "data": {}
}

```

Parameters:

Parameter Name	Type	Example values	Description
enable	bool	false	0- Disable SNMP, 1- Indicates SNMP is enabled
enableType	int	1	1- indicates enable snmpV1, 2- indicates enable snmpV2, 3- indicates enable snmpV3
serverPort	int	161	snmp message receiving port, default 161, port range 1-65535
readCommunity	string	"public"	Read community, up to 31 bytes, letter, number and underline only, default "public"
writeCommunity	string	"private"	Write community, up to 31 bytes, letter, number and underline only, default "private"
trapHostIP	string		IP only supports IPV4
trapHostPort	int		trap host port, port range 1-65535, default value 162
trapName	string		trap host name, maximum length 31 bytes, letter, number and underline only
roUser	object		Read-only users
username	string	"admin"	User name, maximum length 31 bytes, read and write security name cannot be the same
secLevel	int	1	Security level: 1- not verified, 2- Not authorized verification, 3- authorized verification, the code is written dead as 3
authType	int	1	Authentication type: 0- MD5, 1- SHA, 2- None
privType	int	1	Encryption type: 0 - DES, 1 - AES, 2 - None
authpass	string	"12345678"	Authentication password, for AES128 encryption type, encrypted data for base64 encoding, length range 8-15 bytes
privpass	string	"12345678"	Encryption password, for AES128 encryption type, encrypted data for base64 encoding, length range 8-15 bytes
encryptionType	string	"aes128-1"	Encryption type. Get at the encryption capability set
iv	string	"abcde1234567890"	Offset ,16in characters, for AES encryption type
datetime	string		Date time. Format: "YYYY-MM-DDThh:mm:ss"

Encryption instructions

The transmission of sensitive information such as user name and password is encrypted by aes128 CBC algorithm, encryption padding zeropadding, encryption key and offset iv. The encryption key is

generated by the client and the server by their own calculation of the agreed key Type, and is not passed. The offset iv and time are passed to the server in the parameters when the content is requested.

7.3.12、Wifi Hotspot parameters configuration attribute

Brief description:

- Wifi hotspot parameters configuration attribute, you need to get the encryption capability set first, and then get the parameters

*Request URL:

- /api/network/wifiAPInfo

Request Method:

- POST

Get Example Get Parameter Request:

```
{
  "action": "get",
  "data": NULL
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enableAP": 1,
    "enableDhcp": 1,
    "enableAuthentication": 1,
    "authenticationMethod": 2,
    "wirelessNetworkMode": 0,
    "wirelessWorkChannel": 0,
    "IPRange": 100,
    "devIP": "192.168.1.100",
    "devSubNetMask": "255.255.255.0",
    "devGateway": "192.168.1.1",
    "firstIP": "192.168.1.101",
    "dhcpDns": "192.168.1.1",
    "dhcpGateway": "192.168.1.1",
    "APssid": "Wifi-free",
    "APPASSWORD": ""
  }
}
```

Setting example Setting parameter request:

```
{
  "code": 0,
  "data": {
    "enableAP": 1,
    "enableDhcp": 1,
    "enableAuthentication": 1,
    "authenticationMethod": 2,
    "wirelessNetworkMode": 0,
    "wirelessWorkChannel": 0,
    "IPRange": 100,
    "devIP": "192.168.1.100",
    "devSubNetMask": "255.255.255.0",
    "devGateway": "192.168.1.1",
    "firstIP": "192.168.1.101",
    "dhcpDns": "192.168.1.1",
    "dhcpGateway": "192.168.1.1",
    "APssid": "Wifi-free",
    "APPassword": "12345678",
    "encryptionType": "aes128-1",
    "datetime": "2021-03-06T10:47:05",
    "iv": "abcde1234567890"
  }
}
```

Set parameters to return

```
{
  "code": 0,
  "data": {}
}
```

Parameters:

Parameter Name	Type	Example values	Description
loginName	string	admin	Login User Name
enableAP	int	1	0- Turn off wifi AP, 1- Turn on wifi AP
enableDhcp	int	1	0- Turn off Dhcp, 1- Turn on Dhcp
enableAuthentication	int	1	0- Turn off authentication, 1- Turn on authentication
authenticationMethod	int	2	wifi hotspot authentication method, 2: WEP 8: WPA 32: WPA2 default 2
wirelessNetworkMode	int	0	wifi network standard, default 0, 802.11b
wirelessWorkChannel	int	0	wifi operating band, default 0, band range 0-14
IPRange	int	100	wifi hotspot assign address range, default 100
devIP	string		IP Address
devSubNetMask	string		Mask
devGateway	string		Gateway

Parameter Name	Type	Example values	Description
firstIP	string		Start Address
dhcpDns	string		Specific DNS
dhcpGateway	string		Specific Gateway
APSSid	string		wifi Hotspot SSID
APPassword	string		Encrypted password, for AES128 encryption type, encrypted data for base64 encoding, length range 8-15 bytes
encryptionType	string	"aes128-1"	Encryption type.
iv	string	"abcde1234567890"	Offset ,16 characters, for AES encryption type
datetime	string		Date time. Format: "YYYY-MM-DDThh:mm:ss"

Encryption instructions

The transmission of sensitive information such as user name and password is encrypted by aes128 CBC algorithm, encryption padding zeropadding, encryption key and offset iv. The encryption key is generated by the client and the server by their own calculation of the agreed key Type, and is not passed. The offset iv and time are passed to the server in the parameters when the content is requested.

7.4、Audio and video parameters configuration

7.4.1、Code range attribute configuration

Brief description:

- Code range attribute configuration

Request URL:

- /api/video/video-encode-scope

Request Method:

- POST

Sending example

```
{
  "action": "get",
  "data": null
}
```

Return Example

```
{
  "code": 0,
  "data": [
    [
      {
        "encodeScope": 7,
        "h264Attr": {
          "resolutionCnt": 6,
          "encodeType": 0,
          "resolutionList": [
            {"encodeType": 0, "width": 2592, "height": 1944, "suitableFrameRate": 20, "suitableBitRate": 6144, "maxFrameRate": 20},
            {"encodeType": 0, "width": 2560, "height": 1920, "suitableFrameRate": 20, "suitableBitRate": 6144, "maxFrameRate": 20},
            {"encodeType": 0, "width": 2592, "height": 1536, "suitableFrameRate": 25, "suitableBitRate": 6144, "maxFrameRate": 25},
            {"encodeType": 0, "width": 2560, "height": 1440, "suitableFrameRate": 25, "suitableBitRate": 6144, "maxFrameRate": 25},
            {"encodeType": 0, "width": 1920, "height": 1080, "suitableFrameRate": 25, "suitableBitRate": 4096, "maxFrameRate": 25},
            {"encodeType": 0, "width": 1280, "height": 720, "suitableFrameRate": 25, "suitableBitRate": 2048, "maxFrameRate": 25},
          ]
        },
        "h265Attr": {
          "resolutionCnt": "6",
          "encodingType": "0",
          "resolutionList": [
            {"encodeType": 0, "width": 2592, "height": 1944, "suitableFrameRate": 20, "suitableBitRate": 6144, "maxFrameRate": 20},
            {"encodeType": 0, "width": 2560, "height": 1920, "suitableFrameRate": 20, "suitableBitRate": 6144, "maxFrameRate": 20},
            {"encodeType": 0, "width": 2592, "height": 1536, "suitableFrameRate": 25, "suitableBitRate": 6144, "maxFrameRate": 25},
            {"encodeType": 0, "width": 2560, "height": 1440, "suitableFrameRate": 25, "suitableBitRate": 6144, "maxFrameRate": 25},
            {"encodeType": 0, "width": 1920, "height": 1080, "suitableFrameRate": 25, "suitableBitRate": 4096, "maxFrameRate": 25},
            {"encodeType": 0, "width": 1280, "height": 720, "suitableFrameRate": 25, "suitableBitRate": 2048, "maxFrameRate": 25},
          ]
        },
        "mjpegAttr": {
          "resolutionCnt": 1,
          "encodeType": 0,
          "resolutionList": [
            {"encodingType": 0, "width": 1920, "height": 1080, "suitableFrameRate": 15, "suitableBitRate": 10240, "maxFrameRate": 15},
          ]
        }
      },
      {
        "encodeScope": 7,
        "encodeBiterateMin": 64
        "encodeBiterateMax": 12000
        "h264Attr": {
          "resolutionCnt": 1,
          "encodeType": 0,
        }
      }
    ]
  ]
}
```



```

        "resolutionList": [
            {"encodingType":0,"width":720, "height":576,"suitableFrameRate":20, "suitableBitRate":1024,"maxFrameRate":20},
        ]
    },
    "h265Attr": {
        "resolutionCnt": 1,
        "encodeType": 0,
        "resolutionList": [
            {"encodingType":0,"width":720, "height":576,"suitableFrameRate":20, "suitableBitRate":512,"maxFrameRate":20},
        ]
    },
    "mjpegAttr": {
        "resolutionCnt": 1,
        "encodeType": 0,
        "resolutionList": [
            {"encodingType":0,"width":720, "height":576,"suitableFrameRate":15, "suitableBitRate":2560,"maxFrameRate":15},
        ]
    }
},
{
    "encodeScope": 7,
    "h264Attr": {
        "resolutionCnt": 1,
        "encodeType": 0,
        "resolutionList": [
            {"encodingType":0,"width":352, "height":288,"suitableFrameRate":15, "suitableBitRate":512,"maxFrameRate":15},
        ]
    },
    "h265Attr": {
        "resolutionCnt": 1,
        "encodeType": 0,
        "resolutionList": [
            {"encodingType":0,"width":352, "height":288,"suitableFrameRate":15, "suitableBitRate":128,"maxFrameRate":20},
        ]
    },
    "mjpegAttr": {
        "resolutionCnt": 1,
        "encodeType": 0,
        "resolutionList": [
            {"encodingType":0,"width":352, "height":288,"suitableFrameRate":15, "suitableBitRate":1024,"maxFrameRate":15},
        ]
    }
}
]
}

```

Return parameter description

Parameter Name	Type	Example values	Description
encodeScope	int		Video encoding supported type, fill in ARG_ENCODING_SCOPE_TYPE

Parameter Name	Type	Example values	Description
resolutionCnt	int		Number of resolution
encodeType	int		Video encoding Fill in the video part of (ARG_ENCODING_TYPE)
width	int	1920	width
height	int	1080	Height
suitableFrameRate	int		Optimal frame rate
suitableBitRate	int	1024	Optimal Bit Rat
maxFrameRate	int		Maximum frame rate
encodeBiterateMin	int		Bit rate range minimum.
encodeBiterateMax	int		Bit rate range maximum

7.4.2、Video encoding parameters configuration attribute

Brief description:

- Video encoding parameters configuration attribute

Request URL:

- /api/video/video-encode

Request Method:

- POST

Acquisition Example:

Get parameter request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "streamEncode": [{
      "encodeType": 1,
      "quality": 0,
      "streamType": 0,
      "profile": 1,
      "width": 2560,
      "height": 1440,
    }
  ]
}
```

```

        "frameRate": 25,
        "bitRate": 2048,
        "keyFrameInterval": 75,
        "bitrateType": 0,
        "videoType": 1,
        "enableSmartEnc": 1,
        "isSupport": 0
    },
    {
        "encodeType": 1,
        "quality": 0,
        "streamType": 0,
        "profile": 1,
        "width": 1280,
        "height": 720,
        "frameRate": 25,
        "bitRate": 512,
        "keyFrameInterval": 75,
        "bitrateType": 0,
        "videoType": 1,
        "enableSmartEnc": 1,
        "isSupport": 0
    },
    {
        "encodeType": 1,
        "quality": 0,
        "streamType": 0,
        "profile": 1,
        "width": 1280,
        "height": 720,
        "frameRate": 25,
        "bitRate": 1024,
        "keyFrameInterval": 75,
        "bitrateType": 0,
        "videoType": 1,
        "enableSmartEnc": 1,
        "isSupport": 0
    }
}

```

Example of setting:

Set the parameter request:

```

{
  "action": "set",
  "data": {
    "streamEncode": [{
      "encodeType": 1,
      "quality": 0,
      "streamType": 0,
      "profile": 1,
      "width": 2560,
      "height": 1440,
      "frameRate": 25,
      "bitRate": 2048,
      "keyFrameInterval": 75,
      "bitrateType": 0,
    }
  ]
}

```

```

        "videoType": 1,
        "enableSmartEnc": 1,
        "isSupport": 0
    },
    {
        "encodeType": 1,
        "quality": 0,
        "streamType": 0,
        "profile": 1,
        "width": 1280,
        "height": 720,
        "frameRate": 25,
        "bitRate": 512,
        "keyFrameInterval": 75,
        "bitrateType": 0,
        "videoType": 1,
        "enableSmartEnc": 1,
        "isSupport": 0
    },
    {
        "encodeType": 1,
        "quality": 0,
        "streamType": 0,
        "profile": 1,
        "width": 1280,
        "height": 720,
        "frameRate": 25,
        "bitRate": 1024,
        "keyFrameInterval": 75,
        "bitrateType": 0,
        "videoType": 1,
        "enableSmartEnc": 1,
        "isSupport": 0
    }
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter Description

Parameter Name	Type	Example values	Description
encodeType	int		Video encoding Fill in the video part of (ARG_ENCODING_TYPE)
quality	int		Image quality 0 - lowest 1 - lower 2 - low 3 - medium 4 - higher 5 - highest
streamType	int		Stream type: 0:main stream,1:sub stream 2:third stream
profile	int		Coding complexity 0-low 1-medium 3-high
width	int	720	Width

Parameter Name	Type	Example values	Description
height	int	576	Height
frameRate	int	15	Frame Rate
bitRate	int	256	Bit rate in k byte (Uppercase B)
keyFrameInterval	int	15	I-frame interval
bitrateType	int		Bit rate type 0:fixed bit rate, 1:variable bit rate
videoType	int		Video Type: 0:Video Stream 1:Composite Stream
enableSmartEnc	bool		Smart+ encoding enable: false off true:enable
isSupport	int		Encoding stream support: 0 :Support 1 :Not support

7.4.3. Audio parameters configuration attribute

Brief description:

- Audio parameter configuration attribute interface

Request URL:

- /api/video/audio

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": true,
    "outputVolume": 50,
    "inputVolume": 50,
    "inputModel": 0,
    "outputModel": 0
  }
}
```

Setting Example: Set the parameter request:

```
{
```

```
"action": "set",
"data": {
  "enable": false,
  "inputModel": 0,
  "outputModel": 0,
  "inputVolume": 50,
  "outputVolume": 50
}
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter description:

Parameter Name	Type	Example values	Description
enable	bool	true	Enablde, false: off true: enabled
outputVolume	int	50	Output volume 0-100 100 max
inputVolume	int	50	Input volume 0-100 100 max
inputModel	int		Input mode 0:mic input 1:active input (Line)
outputModel	int		Output mode 0:built-in 1:external
code	int		Return error code

7.4.4、Audio encoding parameters configuration attribute

Brief description:

- Audio encoding parameters configuration attribute interface

Request URL:

- /api/video/audio-encode

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```

{
  "code":0
  "data":[
    {
      "property":{
        "encodeType":16,
        "sampleRate":0,
        "bitWidth":0,
        "soundMode":0
      }
    }
  ],
}

```

Setting Example: Set the parameter request:

```

{
  "action":"set",
  "data":[
    {
      "property":{
        "encodeType":16,
        "sampleRate":0,
        "bitWidth":0,
        "soundMode":0
      }
    }
  ]
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter description:

Parameter Name	Type	Example values	Description
code	int		Return error code
encodeType	int		Encoding format: Fill in the audio part of (ARG_ENCODING_TYPE)
sampleRate	int		Sampling rate
bitWidth	int		Sampling width
soundMode	int		Mode: 0 mono(momo) 1 stereo(steror)

7.4.5、Watermark configuration attribute

Brief description:

- Watermark configuration attribute

Request URL:

- /api/video/watermark

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": [
    {
      "channel": 0,
      "enable": true,
      "markText": ""
    },
    {
      "channel": 1,
      "enable": true,
      "markText": ""
    },
    {
      "channel": 2,
      "enable": true,
      "markText": ""
    },
    {
      "channel": 3,
      "enable": true,
      "markText": ""
    }
  ]
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": [
    {
      "channel": 0,
      "enable": false,
      "markText": ""
    }
  ]
}
```



```
    },
    {
      "channel": 1,
      "enable": false,
      "markText": ""
    },
    {
      "channel": 2,
      "enable": false,
      "markText": ""
    },
    {
      "channel": 3,
      "enable": false,
      "markText": ""
    }
  ]
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter Description

Parameter Name	Type	Example values	Description
channel	int		Channel number
enable	bool		Enabled false off true on
markText	string		Marker Text

7.4.6、Audio alarm configuration interface

Brief description:

- Audio alarm configuration interface

Request URL:

- /api/video/audio-alarm

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
}
```

```
"data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": true,
    "audioLanguage": 0
    "selAudioType": 0,
    "playDelayTime": 5
  }
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "enable": false,
    "audioLanguage": 0
    "playListenTest": 0,
    "selAudioType": 0,
    "playDelayTime": 5,
    "audioFileLength":0
    "audioFileData":
  }
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter Description:

Parameter Name	Type	Example values	Description
enable	bool	true	Enabled, false: off true: enabled
audioLanguage	int	0	Sound language 0: Chinese 1: English
playListenTest	int	0	Sound testing 0: Non-listening testing 1: listening testing
selAudioType	int	0	Sound Type (0: Private Area 1: Alert Area 2: monitoring Area 3: Welcome 4: Deep Water Warning 5: Dangerous Area 6: Valuables 7: Highland Warning 8: Custom)
playDelayTime	int	5	Delay time 5,10,20 30 (sec)
audioFileLength	int	5	Sound file length (in bytes)
audioFileData	string		Sound file data base64 encoding

Parameter Name	Type	Example values	Description
code	int		Return error code

7.5、Image parameters configuration

7.5.1、Image parameters configuration attribute

Brief description:

- Image parameter configuration interface

Request URL:

- /api/image/image

Request Method:

- POST

Get parameter request:

```
{
  "action": "get",
  "data": null
}
```

Get Return Example

```
{
  "code": 0,
  "data": {
    "streamType": 1,
    "brightness": 128,
    "contrast": 128,
    "saturation": 128,
    "hue": 128,
    "enableGainCtrl": 1,
    "gainCtrl": 128,
    "gainRCtrl": 128,
    "gainGCtrl": 128,
    "gainBCtrl": 128,
    "whiteBalance": 0,
    "enableBackLight": 0,
    "backLightStrength": 0,
    "exposureCtrl": 0,
    "enableWideDynamic": 0,
    "wideDynamicLevel": 0,
    "enableFlickerCtrl": 0,
    "flickerCtrlModel": 0,
    "dnrModel": 0,
    "dnr2DStrength": 32,
  }
}
```

```
"dnr3DStrength": 32,  
"enableDefog": 0,  
"defogStrength": 0,  
"enableStrongLightInh": 0,  
"strongLightInhibitionStreangth": 0,  
"darkAreaAscension": 0,  
"enableHdr": 0,  
"sceneMode": 0,  
"enableElectronicAntiShake": 0,  
"enableLdc": 1,  
"ldcStrength": 171  
}  
}
```

Set the parameter request:

```
{  
  "action": "set",  
  "data": {  
    "brightness": 128,  
    "contrast": 128,  
    "saturation": 128,  
    "hue": 128,  
    "exposureCtrl": 1,  
    "enableGainCtrl": 112,  
    "gainCtrl": 112,  
    "enableBackLight": 0,  
    "backLightStrength": 0,  
    "enableStrongLightInh": 0,  
    "strongLightInhibitionStreangth": 0,  
    "darkAreaAscension": 0,  
    "whiteBalance": 0,  
    "gainRCtrl": 128,  
    "gainGCtrl": 128,  
    "gainBCtrl": 128,  
    "dnrModel": 0,  
    "dnr2DStrength": 32,  
    "dnr3DStrength": 32,  
    "flickerCtrlModel": 0,  
    "enableHdr": 0,  
    "enableElectronicAntiShake": 0,  
    "enableDefog": 0,  
    "defogStrength": 0,  
    "enableLdc": 1,  
    "ldcStrength": 171,  
    "channel": 0,  
    "enableFlickerCtrl": 0,  
    "sceneMode": 0,  
    "streamType": 1,  
    "enableWideDynamic": 0,  
    "wideDynamicLevel": 0  
  }  
}
```

Set Return Example

```
{  
  "code": 0,  
}
```

```
"data": {}
}
```

Return parameter description

Parameter Name	Type	Example values	Description
streamType	int		Stream type: Fill in (ARG_STREAM_TYPE) video part
brightness	int	128	Brightness
contrast	int	128	Contrast Ratio
saturation	int	128	Saturation
hue	int	128	Color Tones
enableGainCtrl	int		Gain adjustment enable, 0: Manual 1: Auto
gainCtrl	int	128	Gain adjustment 0-255
gainRCtrl	int	128	Red gain adjustment 0-255
gainGCtrl	int	128	Green gain adjustment 0-255
gainBCtrl	int	128	Blue gain adjustment 0-255
whiteBalance	int		White balance: 0 auto 1 manual
enableBackLight	int		Backlight compensation: 0 off 1 on
backLightStrength	int		Backlight intensity: 0 low 1 medium 2 high. Backlight activation is effective
exposureCtrl	int		Auto Exposure Control: 0:On 1:Off
enableWideDynamic	int		WDR 0:off 1:on
wideDynamicLevel	int		WDR Level 0:Auto 1:Weak 2:Medium 3:Strong 4:Super Strong
enableFlickerCtrl	int		Flicker control 0:off 1 on
flickerCtrlModel	int		Flicker control mode: 0:Indoor NTSC 1:Indoor PAL 2 Outdoor
dnrModel	int		DNR mode Fill in the ARG_DNRMODE_TYPE section
dnr2DStrength	int		DNR 2D intensity: 0-255
dnr3DStrength	int		DNR 3D intensity: 0-255
defogEnable	int		Defog switch: 0:off 1:enable 2:auto
defogStrength	int		Defogging intensity: 0-255
enableStrongLightInh	int		HLC switch: 0:off 1:enable
strongLightInhibitionStreangth	int		HLC Intensity: 0-255
darkAreaAscension	int		Dark area boost intensity: 0-255
enableHdr	int		Dark area boost intensity: 0-255
sceneMode	int		Scene Mode: 0:Indoor 1:Outdoor
enableElectronicAntiShake	int		Electronic anti-shake: 0:off 1:enable
enableLdc	int	LDC enable: 0:off 1:enable	
ldcStrength	int		LDC intensity: 0-255

Parameter Name	Type	Example values	Description
code	int		Return error code

7.5.2、OSD configuration attribute

Brief description:

- OSD configuration attribute

Request URL:

- /api/image/osd

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "mirrorMode": 0,
    "electronicShutter": 0,
    "enableIrLed": 0,
    "ircutMode": 0,
    "ircutSwitchTime": 3,
    "ircutSensitivity": 0,
    "visiable": 0,
    "postion": 0,
    "dateVisiable": 1,
    "dateFormat": 2,
    "nameVisiable": 0,
    "text": "",
    "rotateAngle": 0
  }
}
```

设置示例: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "dateVisiable": 1,
    "nameVisiable": 0,
  }
}
```

```

    "weekVisiable": 0,
    "weekLanguageFormat": 0,
    "dateFormat": 2,
    "postion": 0,
    "text": "",
    "mirrorMode": 0,
    "rotateAngle": 0,
    "electronicShutter": 0,
    "enableIrLed": 0,
    "ircutMode": 0,
    "ircutSwitchTime": 3,
    "ircutSensitivity": 0,
    "visiable": 0
  }
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter description:

Parameter Name	Type	Example values	Description
mirrorMode	int	Whether mirroring: 0 no mirroring, 1 horizontal mirroring 2 vertical mirroring 3 horizontal and vertical mirroring	
electronicShutter	int	Electronic shutter: Fill in ARG_SHUTTERSPEED_TYPE	
enableIrLed	int	0:IRLED off 1:IRLED on	
ircutMode	int	0:LDR Auto 1:Video Auto 2:Color Mode 3:B&W Mode	
ircutSwitchTime	int	ircut switching time, 3-120 seconds	
ircutSensitivity	int	Sensitivity 0-7	
visiable	int	0 Do not show osd, 1 show osd	
postion	int	Fill in ARG_POSITON_TYPE, valid when osd is displayed	
dateVisiable	int	Display time 0 Display 1 No display	
dateFormat	int	Date format 0:y/m/d 1:m/d/y 2:d/m/y	
weekVisiable	int	Display week 0 display 1 no display	
weekLanguageFormat	int	Week language format 0 English 1 Chinese	
nameVisiable	int	Show text 0 show ipc name 1 don't show ipc name	
text	string	osd text	
rotateAngle	int	Rotation angle: Fill in ARG_ROTATE_TYPE	

7.5.3、Fill light configuration attribute

Brief description:

- Fill light parameters configuration attribute

Request URL:

- /api/image/fill-light

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "dayNightMode": 0,
    "fillMode": 0,
    "keepFullColor": 0
    "nightVal": 3,
    "openDelay": 3,
    "dawnTime": 390,
    "darkTime": 1080,
    "lightMode": 0,
    "lightVal": 100,
    "realTimeVal": 11,
    "dayVal": 21,
    "closeDelay": 3
  }
}
```

Set the parameter request:

```
{
  "action": "set",
  "data": {
    "dayNightMode": 0,
    "fillMode": 0,
    "keepFullColor": 0
    "nightVal": 3,
    "openDelay": 3,
    "dawnTime": 390,
    "darkTime": 1080,
    "lightMode": 0,
    "lightVal": 100,
    "realTimeVal": 11,
    "dayVal": 21,
    "closeDelay": 3
  }
}
```



```
}  
}
```

Set the parameters to return:

```
{  
  "code": 0,  
  "data": {}  
}
```

Return Example

Parameter description:

Parameter Name	Type	Example values	Description
fillMode	int		Fill light mode 0:Auto 1:Daytime 2:Night 3:Timing
dayNightMode	int		Day and night mode 0:smart infrared 1:smart alert 2:White light full color
keepFullColor	bool		Keep full color false: off, true: on
nightVal	int		Sensitivity: Range (0-7)
openDelay	int		Filtration time: (3-10)
closeDelay	int		Turn off light delay (sec)
realTimeVal	int		Real-time brightness (2 times/sec) Refresh
dayVal	int		Daylight Brightness
lightMode	int		Light mode 1:Auto 0:Manual
lightVal	int		Light brightness: range (0-100)
dawnTime	int		Dawn time (min)
darkTime	int		Darkness Time (min)

7.5.4、Zoom focus configuration attribute

Brief description:

- Zoom and focus configuration attribute

Request URL:

- /api/image/zoom-focus

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "step": 4,
    "apertureMode": 0,
    "apertureVal": 0,
    "focusModel": 0,
    "enableDayNightFilter": 0
  }
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "step": 4,
    "apertureMode": 0,
    "apertureVal": 0,
    "focusModel": 0,
    "enableDayNightFilter": 0
  }
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter description:

Parameter Name	Type	Example values	Description
code	int	Return error code	
step	unsigned char	Speed, range 1-8, 8 control effect is the most obvious, movement speed	
apertureMode	unsigned char	Aperture mode 0 auto 1 control	
apertureVal	unsigned char	Aperture opening value 0-100 (valid when aperture mode is control)	
focusModel	unsigned char	0 Auto-focus 1 Manual Focus	
enableDayNightFilter	unsigned char	Day and night switch to enable auto-focus 0: off, 1: on	

7.5.5、PTZ configuration attribute

Brief description:

- PTZ configuration attribute.

Request URL:

- /api/image/ptz-config

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "serial": {
      "protocol": 0,
      "addr": 1,
      "addrMin": 0,
      "addrMax": 255,
      "databit": 8,
      "databitList": [6, 7, 8],
      "stopbit": 1,
      "stopbitList": [1, 2],
      "checkType": 0,
      "checkTypeList": [0, 1, 2],
      "baudrate": 2400,
      "baudrateList": [110, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 921600],
    }
  }
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "serial": {
      "protocol": 1,
      "addr": 1,
      "baudrate": 9600,
    }
  }
}
```

```

        "databit":8,
        "stopbit":1,
        "checkType":0,
    }
}
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter Description

Parameter Name	Type	Example values	Description
protocol	int		protocol
protocollist	array	[0, 1]	Protocol Protocol Type Rang
addr	int	1	Address
addrMin	0	0	Address b minimum
addrMax	0	255	Address b maximum
baudrate	int	2400	Bitrate
baudratelist	array	[110, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 921600]	Bitrate parameter range
databit	int	8	Data bits
databitlist	array	[6, 7, 8]	Data bits
stopbit	int	1	Stop bit
stopbitlist	array	[1, 2]	Stop bit
checkType	int		Check Typ
checkTypelist	array	[0, 1, 2]	check Typ

7.5.6、PTZ control

Brief description:

- PTZ Control

Request URL:

- /api/image/ptz-ctrl

Request Method:

-
- POST
-

Get Example: Get Parameter Request Example:

-

```
{  
  "action": "get",  
  "data": null  
}
```

Get parameter return:

-

```
{  
  "code": 0,  
  "data": {  
    "auto3DStatus": 0, "presetNum":  
    255, "presetMin": 1, "presetMax":  
    255, "presetList": [  
      {  
        "presetIdx": 0,  
        "presetName": "preset1",  
        "enablePreset": 0,  
        "readOnly": 0,  
      },  
      {  
        "presetIdx": 0,  
        "presetName": "preset1",  
        "enablePreset": 0,  
        "readOnly": 0,  
      },  
      .....  
      {  
        "presetIdx": 0,  
        "presetName": "preset1",  
        "enablePreset": 0,  
        "readOnly": 0,  
      }  
    ]  
  }  
}
```

Setting Example: Set the parameter request:

```

{
  "action": "set",
  "data": {
    "optType": 4,
    "speed":4,
    "preset"
    {
      "presetIdx":1,
      "enablePreset":0,
      "presetName":"preset1",
    },
    "cruise"
    {
      "cruiseIdx":0,
    }
    "position3D"{
      "x1" : 0,
      "y1" : 0,
      "x2" : 0,
      "y2" : 0,
      "width" : 1920,
      "height" : 1920,
    }
  }
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter description:

Parameter Name	Type	Example values	Description
optType	int	Operation type: View General Enumeration List - Cloud Control Enumeration Values	
speed	int	PTZ control speed, range (1-10)	
preset	object	PTZ preset point parameter, only preset point setting, recall, delete command displayed in this parameter, others named as this field is empty	
auto3DStatus	int	Auto 3D Status, 0 Off, 1 On	
presetNum	int	Number of preset points	
presetMin	int	Preset Point Min.	
presetMax	int	Preset point maximum	
presetList	array	[]	List of preset points
presetIdx	int	PTZ preset point index number, range (1-255)	
enablePreset	int	PTZ enabled and disabled,0 for disabled,1 for enabled	
readOnly	int	Whether PTZ is read-only,0 is modifiable, 1 is read-only	

Parameter Name	Type	Example values	Description
		preset point	
presetName	string	Preset point names, maximum supports 20 bytes	
cruise	object	PTZ cruise route parameter, only enable cruise route, stop cruise route, set cruise route command displayed in this parameter, others named as this field is empty	
cruiseIdx	int	PTZ Cruise Index number	
position3D	object	3D positioning parameter, only enable 3D positioning, close 3D positioning displayed in this parameter, others named as this field is empty	
x1	int	x-coordinate of the upper left, relative to the display area (upper left is (0,0), increasing to the left and down)	
y1	int	y-coordinate of the upper left, relative to the display area (top-left is (0,0), increasing to the left and down)	
x2	int	x-coordinate of the lower right, relative to the display area (upper left is (0,0), increasing to the left and down)	
y2	int	y-coordinate of the lower right, relative to the display area (upper left is (0,0), increasing to the left and down)	
width	int	Display area width	
height	int	Display area high	

7.5.7、PTZ cruise route

Brief description:

- PTZ Cruise

Request URL:

- /api/image/ptz-cruise

Request Method:

•
POST

•
Get example: Get PTZ cruise route request:

•

```
{
  "action": "get",
  "data": {
    "cruiseIdx": 4,
  }
}
```

•

Get parameter return:

```
{
  "code":0,
  "msg":null,
  "data":{
    "enableCruise":0,
    "cruiseMin":0, "cruiseMax":
    8, "cruisePresetMin":0,
    "cruisePresetMax":32,
    "cruisePresetNum":32,
    "cruisePresetList":[
      {
        "presetIdx":0,
        "speed":0,
        "presetDweelTime":0,
      },
      {
        "presetIdx":1,
        "speed":0,
        "presetDweelTime":0,
      },
      .....
      {
        "presetIdx":31,
        "speed":0,
        "presetDweelTime":0,
      }
    ]
  }
}
```

Set PTZ cruise route request:

```
{
  "action": "set",
  "data":{
    "channel": 0,
    "cruiseIdx":0,
    "enableCruise":0,
    "cruisePresetList":[
      {
        "presetIdx":0,
        "speed":0,
        "presetDweelTime":0,
      },
      {
```



```

        "presetIdx":1,
        "speed":0,
        "presetDweelTime":0,
    },
    .....
    {
        "presetIdx":31,
        "speed":0,
        "presetDweelTime":0,
    }
]
}
}
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter description:

Parameter Name	Type	Example values	Description
cruiseIdx	int	Cruise route index	
cruiseMin	int	Cruise route range minimum	
cruiseMax	int	Cruise route range maximum	
enableCruise	bool	Cruise route enable 1 is true; 0 is false;	
cruisePresetNum	int	Cruise route contains preset points, the maximum 32 preset points	
cruisePresetMin	int	Cruise route contains preset point minimum	
cruisePresetMax	int	Cruise route contains preset point maximum	
cruisePresetList	array	[]	Cruise route preset point list
presetIdx	int	Preset point serial number	
speed	int	Preset Point Cruise Speed	
presetDweelTime	int	Preset point dwell time (5-3600S)	

7.6、Storage parameters configuration

7.6.1、Recording parameters configuration attribute

Brief description:

- Recording configuration parameters interface

Request URL:

- /api/storage/regular-record-property

Get Example:

- POST

Sending parameter description Get parameter request

```
{
  "action": "get",
  "data": null
}
```

Get parameter return

```
{
  "code": 0,
  "data": {
    "videoRecord": {
      "enable": 1,
      "storageType": 0,
      "videoStreamType": 0,
      "audioStreamType": 0,
      "validWeek": 0,
      "duration": 0
    },
    "weekScop": {
      "scopeModel": 0,
      "dayScope": [
        {
          "socpeMode": 3,
          "hourScope": [
            [], [], [], [], [], []
          ]
        },
        {
          "socpeMode": 3,
          "hourScope": [
            [], [], [], [], [], []
          ]
        },
        {
          "socpeMode": 3,
          "hourScope": [
            [], [], [], [], [], []
          ]
        },
        {
          "socpeMode": 3,
          "hourScope": [
            [], [], [], [], [], []
          ]
        }
      ]
    }
  }
}
```


Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter Description

Parameter Name	Type	Example values	Description
enable	bool		Whether to enable video recording: false: off, true: on
validWeek	int	Valid weeks: 0 permanently valid, 1-255 specific weeks	
duration	int	Duration	
storageType	int		Storage mode: Fill HS_STORAGE_TYPE_E
videoStreamType	int		Stream type, 0 main stream, 1 sub stream, 2 triple stream
scopeModel	int		Time mode, fill in HS_TIME_SCOPE_MODE
hourScope	int		Time period, there are up to 6 periods in a day, ["start time", "end time", "time type"] unit S, time type fill HS_RECORD_TIME_TYPE_E, for example [0,1440,0]
code	int		Return error code

7.6.2. Recording storage parameters configuration attribute

Brief description:

- Recording storage configuration parameters interface

Request URL:

- /api/storage/storage-property

Request Method:

- POST

Get example: Get parameter request:

```
{
  "ation": "get",
  "data": {
    "storageType": 4,
    "id": 255
  }
}
```

```
}  
}
```

Get parameter return:

```
{  
  "code": 0,  
  "data": {  
    "storageType": 2,  
    "packingModel": 0,  
    "lackSpaceModel": 1,  
    "preRecordTime": 1,  
    "recordDelay": 1,  
    "videoStoragePercentage": 81,  
    "packingValue": {  
      "packingTime": 60  
    }  
  }  
}
```

Setting Example: Set the parameter request:

```
{  
  "ation": "set",  
  "data": {  
    "lackSpaceModel": 1,  
    "preRecordTime": 1,  
    "recordDelay": 1,  
    "videoStoragePercentage": 90  
  }  
}
```

Set the parameters to return:

```
{  
  "code": 0,  
  "data": {}  
}
```

Parameter description:

参数名	类型	示例值	说明
code	int		Return error code
storageType	int		Storage mode: Fill HS_STORAGE_TYPE_E
packingModel	int		Packing mode 0 by time 1 by size 2 none
lackSpaceModel	int		Recording mode, 0 no override 1 loop override
preRecordTime	int		Pre-record time: Fill HS_STORAGE_PRERECORDTIME_TYPE_E
recordDelay	int		Recording delay: Fill HS_STORAGE_RECORDEDELAY_TYPE_E
packingTime	int		Packing time 0-255 unit minutes


```

0], [0, 0, 0]]
    ]
  }
}
}

```

Setting Example: Set the parameter request:

```

{
  "action": "set",
  "data": {
    "quality": 0,
    "eventCapture": {"enable": true,"pictureCodecType": 3,"pictureRes
olution": 1,"captureInterval": 10,"captureNumber": 1},
    "timerCapture": {"enable": true,"captureIntervalSec": 10},
    "weekScop": {
      "scopeModel": 0,
      "dayScope":[
        [[0, 1440, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0,
0], [0, 0, 0]],
        [[0, 1440, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0,
0], [0, 0, 0]],
        [[0, 1440, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0,
0], [0, 0, 0]],
        [[0, 1440, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0,
0], [0, 0, 0]],
        [[0, 1440, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0,
0], [0, 0, 0]],
        [[0, 1440, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0,
0], [0, 0, 0]],
        [[0, 1440, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0,
0], [0, 0, 0]]
      ]
    }
  }
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter Description:

Parameter Name	Type	Example values	Description
eventCapture	object		Event Trigger
enable	bool		Event Trigger Enable Switch
pictureCodecType	int		Image format Fixed 3 corresponds to MJPEG
pictureResolution	int		Image resolution Fill 1~3 Corresponding to 1080p, 720p, D1 Default 1080p
captureInterval	int		Capture time interval, fill in 1~5

Parameter Name	Type	Example values	Description
			corresponding to 10s, 30s, 1min, 5min, 10min default 1
captureNumber	int		Number of captures, (0-120) Default 1
quality	int		(New added for 2021-01) Image quality (ARG_SNAPSHOT_IMAGE_QUALITY), 0/1/2 corresponds to high/medium/low, if the device does not support setting image quality, then this field is not sent
eventCapture.captureIntervalSec	int		(New added for 2021-01) Event interval (in seconds), replaces capture Interval enumeration
timerCapture.enable	bool		(New added for 2021-01) Enable timed capture function
timerCapture.captureIntervalSec	int		(New added for 2021-01) Timed capture interval (seconds)
weekScope	object		(New added for 2021-01) Weekly scheduling, same as recording scheduling, total 7 days, there are 6 periods a day, including 7 a day Scope
dayScope	array		array, 7 hour Scope
socpeMode	int		HS_TIME_SCOPE_MODE, fixed to 3
hourScope	array		Daily scheduling array containing 6 periods, each with a start time (minutes), an end time (minutes), and a type (timed, motion detection, etc.)
code	int		Return error code

7.6.4、SD card information configuration attribute

Brief description:

- Get/Format SD card information configuration attribute

Request URL:

- /api/storage/storage-info

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
  "data": {
    "storageType": 1,

```



```

    "id": 255
  }
}

```

Get parameter return:

```

{
  "code": 0,
  "data": {
    "storageType": 1,
    "status": 1,
    "percent": 0,
    "capacity ": 0,
    "free": 0,
    "id": 0,
    "videoTotalSpace":4096,
    "videoFreeSpace":1023,
    "snapshotTotalSpace":512,
    "snapshotFreeSpace":34
  }
}

```

Setting Example: Set the parameter request:

```

{
  "action": "set",
  "data": {
    "storageType": 1
  }
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter Description:

Parameter Name	Type	Example values	Description
storageType	int		Storage mode: Fill ARG_STORAGE_TYPE
optType	int		Operation type: Fill in STORAGE_OPTION_TYPE
status	int		Status 0 available 1 not available 2 formatting in progress 3 not formatted 4 read/write exception
percent	int		Formatting progress 0-100
capacity	int		Total capacity: Unit megabyte (M)
free	int		Available space: in megabytes (M)
id	int		id: Fill in STORAGE_ID_TYPE
videoTotalSpace	int		Total video capacity (MB)

Parameter Name	Type	Example values	Description
videoFreeSpace	int		Video space remaining (MB)
snapshotTotalSpace	int		Total capture capacity (MB)
snapshotFreeSpace	int		Remaining space for capture (MB)

7.6.5、SD card formatting configuration attribute

Brief description:

- Formatting SD Card Information Configuration attribute

Request URL:

- /api/storage/storage-option

Request Method:

- POST

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "storageType": 1,
    "id": 0,
    "optType": 0
  }
}
```

Set the parameters to return:

```
"code": 0,
"data": {}
```

Parameter Description

Parameter Name	Type	Example values	Description
storageType	int		Storage mode: Fill ARG_STORAGE_TYPE
id	int		id: Fill in STORAGE_ID_TYPE
optType	int		Operation type: Fill in STORAGE_OPTION_TYPE

7.6.6、Cloud parameters configuration attribute

Brief description:

- Cloud Parameters Configuration attribute

Request URL:

- /api/storage/cloud

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": {
    "bindType": 17
  }
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "bindType": 0,
    "website": "",
    "verificationCode": ""
  }
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "bindType": 0,
    "website": "",
    "verificationCode": ""
  }
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter Description

Parameter Name	Type	Example values	Description
----------------	------	----------------	-------------

Parameter Name	Type	Example values	Description
bindType	int		Binding Type
website	string		Website
verificationCode	string		Verification Code

7.6.7、Cloud status acquire

Brief descriptionJ:

- Cloud Status acquire

Request URL:

- /api/storage/cloud-status

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "bindType": 0,
    "bindStatus": 0,
    "username": "",
    "totalSize": "",
    "usedSize": ""
  }
}
```

Parameter Description

Parameter Name	Type	Example values	Description
bindStatus	int		Binding status:1 for success,0 for failure
bindType	int		Binding Type
username	string		User Name
totalSize	string		Total size
usedSize	string		Use Size

7.6.8. Ipeye parameters configuration attribute

Brief description:

- Ipeye parameters configuration attribute

Request URL:

- /api/storage/ipeye

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": false,
    "url": "http://172.18.193.180:8282"
  }
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "enable": false,
    "url": "http://172.18.193.180:8282"
  }
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter Description

- Alarm arming time parameter interface

Request URL:

- /api/event/alarm-schedule

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
  "data": {
    "alarmType": 1
  }
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": true,
    "alarmType": 1,
    "alarmLinkScope": 23,
    "personFilterStatus": 1,
    "ioStatus": 0,
    "schedule": [
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]]
    ],
    "trigger": {
      "enableAlarmSntp": "0",
      "enableAlarmFtp": "0",
      "enableAlarmCloud": "0",
      "enableAlarmRecord": "0",
      "enableAlarmIo": "0",
      "enableAlarmFlash": "0",
      "enableAlarmSound": "0",
      "enableAlarmFlashAll": "0"
    }
  }
}
```

Setting Example: Set the parameter request:


```

{
  "action": "set",
  "data": {
    "enable": true
    "alarmLinkScope": 15,
    "personFilterStatus": 1,
    "alarmType": 1,
    "ioStatus": 1
    "process": {
      "enableAlarmSntp": 0,
      "enableAlarmFtp": 0,
      "enableAlarmCloud": 0,
      "enableAlarmRecord": 0,
      "enableAlarmFlash": 0,
      "enableAlarmSound": 0,
      "enableAlarmIo": 0
    },
    "schedule": [
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]]
    ],
  },
}

```

Set parameters to return

```

{
  "code": 0,
  "data": {}
}

```

Return parameter description

Parameter Name	Type	Example values	Description
enable	bool		IO alarm enable: fill in false: off, true: on
alarmType	int		Alarm type: Fill in HS_ALARM_TYPE_E
alarmLinkScope	int		Alarm linkage type, fill in HS_ALARM_LINK_TYPE_E
schedule	int	[0,1440]	Alarm time, seven days, eight time periods per day, in S, ["start time", "end time"], e.g. [0,1440]
trigger	array		Alarm processing, fill in HS_ALARM_PROCESS
ioStatus	int		IO alarm type: Fill 0, 1 corresponding to normally open, normally closed
personFilterStatus	int		Humanoid filter enable: Fill 1, 0 corresponds to on, off
code	int		Return error code

7.7.3、Masking alarm parameters interface

Brief description:

- Masking alarm parameter interface

Request URL:

- /api/event/video-cover

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": NULL
}
```

Example of getting parameters returned:

```
{
  "code": 0,
  "data": {
    "enable": 1,
    "alarmLinkScope": 3,
    "sensitivity": 0,
    "area": [
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0},
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0},
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0},
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0}
    ],
    "schedule": [
      [[0, 1440], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0]],
      [[0, 1440], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0]],
      [[0, 1440], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0]],
      [[0, 1440], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0]],
      [[0, 1440], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0]],
      [[0, 1440], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0], [0, 0]]
    ],
    "trigger": {
      "enableAlarmSntp": 0,
      "enableAlarmFtp": 0,
      "enableAlarmCloud": 0,
      "enableAlarmRecord": 0,
      "enableAlarmIo": 0,
    }
  }
}
```

```

        "enableAlarmFlash": 0,
        "enableAlarmSound": 0,
        "enableAlarmFlashAll": 0
    }
}

```

Setting Example: Set the parameter request:

```

{
  "action": "set",
  "data": {
    "enable": 1,
    "alarmLinkScope": 3,
    "sensitivity": 0,
    "area": [
      {"x1": 0,"y1": 0,"x2": 0,"y2": 0},
      {"x1": 0,"y1": 0,"x2": 0,"y2": 0},
      {"x1": 0,"y1": 0,"x2": 0,"y2": 0},
      {"x1": 0,"y1": 0,"x2": 0,"y2": 0}
    ],
    "schedule": [
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],
0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],
0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],
0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],
0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],
0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],
0]]
    ],
    "trigger": {
      "enableAlarmSntp": 0,
      "enableAlarmFtp": 0,
      "enableAlarmCloud": 0,
      "enableAlarmRecord": 0,
      "enableAlarmIo": 0,
      "enableAlarmFlash": 0,
      "enableAlarmSound": 0,
      "enableAlarmFlashAll": 0
    }
  }
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter Description

Parameter Name	Type	Example values	Description
enable	bool		Area intrusion detection enable 0:off 1:on
alarmLinkScope	int		Alarm linkage type
sensitivity	int		4 Group sensitivity sub Sensitivity [0, 100]
time	int		4 Group Time Threshold sub Time [0, 10]
area	array		4 groups of 10,000 points ratio area x1 y1 x2 y2
schedule	array		Schedule: 8 periods per day for 7 days unit in S, ["start time", "end time"], e.g. [0,1440]

Remarks

- For more return error codes, please see the error code description on the home page

7.7.4、Alarm output parameters configuration attribute

Short description:

- Alarm output parameter configuration attribute interface

Request URL:

- /api/event/alarm-out

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
  "data": NULL
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "delayTime": 2,
    "enableManualAlarm": 0,
    "schedule": [
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],
0]],

```

```

[[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
[[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
[[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
[[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
[[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
[[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]]
]
}
}

```

Setting Example: Set the parameter request:

```

{
  "action": "set",
  "data": {
    "delayTime": 2,
    "enableManualAlarm": 0,
    "schedule": [
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]],
      [[0, 1440],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0],[0, 0]]
    ]
  }
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter description:

Parameter Name	Type	Example values	Description
delayTime	int		Delay time: Fill 1~7, corresponding to 5s, 10s, 30s, 1min, 2min, 5min, 10min Default 2

Parameter Name	Type	Example values	Description
enableManualAlarm	int		Manual alarm enable, fill in 0 or 1
schedule	array	[0,1440]	Alarm time, seven days, eight periods per day, unit in S, ["start time", "end time"], e.g. [0,1440]
code	int		Return error code

7.7.5、Abnormal event configuration attribute

Brief description:

- Abnormal Event Configuration attribute

Request URL:

- /api/event/exception

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
  "data": {
    "abnormalType": 1
  }
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "abnormalType": 1,
    "alarmLinkScope": 16,
    "trigger": {
      "enableAlarmSntp": 0,
      "enableAlarmFtp": 0,
      "enableAlarmCloud": 0,
      "enableAlarmRecord": 0,
      "enableAlarmIo": 0,
      "enableAlarmFlash": 0,
      "enableAlarmSound": 0,
      "enableAlarmFlashAll": 0
    }
  }
}
```

Setting Example: Set the parameter request:

```

{
  "action": "set",
  "data": {
    "trigger": {
      "enableAlarmSmtplib": 0,
      "enableAlarmFtplib": 0,
      "enableAlarmCloud": 0,
      "enableAlarmRecord": 0,
      "enableAlarmIo": 0,
      "enableAlarmFlash": 0,
      "enableAlarmSound": 0,
      "enableAlarmFlashAll": 0
    },
    "abnormalType": 1,
    "alarmLinkScope": 16
  }
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter description:

Parameter Name	Type	Example values	Description
abnormalType	int		Exception type, fill in ARG_ABNORMAL_TYPE
alarmLinkScope	int		Alarm linkage type, fill in ARG_ALARM_LINK_TYPE
enableAlarmSmtplib	bool		SMTP linkage enable
enableAlarmFtplib	bool		FTP linkage enable
enableAlarmCloud	bool		Upload Cloud Enable
enableAlarmRecord	bool		Recording Enabling
enableAlarmIo	bool		IO linkage enable
enableAlarmFlash	bool		Flashing light linkage enable
enableAlarmSound	bool		Sound Linkage Enable
trigger	int		Alarm linkage processing

7.7.6、Alarm range parameters configuration attribute

Brief description:

- Alarm range parameters configuration attribute

Request URL:

- /api/event/alarm-scope

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": null
}
```

Return Example

```
{
  "code": 0,
  "data": {
    "alarmTypeScope": 1,
    "enableStreamRecord": 1,
    "streamRecordStorageScope": 25,
    "enableSnapshot": 1,
    "snapshotScope": 56,
    "enableMessage": 1,
    "messageScope": 16
  }
}
```

Return parameter description

Parameter Name	Type	Example values	Description
alarmTypeScope	int		Supported alarm types: ARG_ALARM_TYPE Read-only
enableStreamRecord	int		Whether to support record: 0 not supported 1 supported
streamRecordStorageScope	int		Supported storage types for video recordings: ARG_STORAGE_TYPE Read-only
enableSnapshot	int		Whether to support capture: 0 not supported 1 supported 持
snapshotScope	int		Supported storage types for capture: ARG_STORAGE_TYPE Read-only
enableMessage	int		Whether to support message: 0 not supported 1 supported
messageScope	int		Supported storage types for message: ARG_STORAGE_TYPE Read-only

7.7.7、ROI configuration attribute

Brief description:

- ROI Configuration attribute

Request URL:

- /api/event/roi

Request Method:

- POST

Get example: Get parameter request:

```
{
  "action": "get",
  "data": null
}
```

Get parameter return:

```
{
  "code":0,
  "data":{
    "enable":0,
    "bgFrameRate":30,
    "region":[
      {
        "qpType":1,
        "qpVal":0,
        "enable":0,
        "area":{
          "x1":0,
          "y1":0,
          "x2":0,
          "y2":0
        }
      },
      {
        "qpType":1,
        "qpVal":0,
        "enable":0,
        "area":{
          "x1":0,
          "y1":0,
          "x2":0,
          "y2":0
        }
      },
      {
        "qpType":1,
        "qpVal":0,
        "enable":0,
        "area":{
          "x1":0,
          "y1":0,
          "x2":0,
          "y2":0
        }
      }
    ]
  }
}
```

```

    ]
  }
}

```

Setting Example: Set the parameter request:

```

{
  "action": "set",
  "data": {
    "bgFrameRate": 30,
    "region": [
      {
        "qpType": 1,
        "qpVal": 0,
        "area": {
          "x1": 13,
          "y1": 16,
          "x2": 30,
          "y2": 38
        }
      },
      {
        "qpType": 1,
        "qpVal": 0,
        "area": {
          "x1": 50,
          "y1": 17,
          "x2": 78,
          "y2": 33
        }
      },
      {
        "qpType": 1,
        "qpVal": 0,
        "area": {
          "x1": 22,
          "y1": 50,
          "x2": 81,
          "y2": 81
        }
      }
    ]
  }
}

```

Parameter description:

Parameter Name	Type	Example values	Description
enable	bool		false disable ROI, true enable ROI
bgFrameRate	int		Background frame rate: 0 disable background frame rate, 1-255 values are the background frame rate. and enable the background frame rate
qpType	int		QP type: Fill ROI_QP_TYPE
qpVal	int		QP value: [-51, 51] in relative mode [0, 51] in absolute mode

Parameter Name	Type	Example values	Description
area	object	{"x1": 0, "y1": 0, "x2": 0, "y2": 0}	Percentage area
x1	int	Upper left x coordinate	
y1	int	Upper left y-coordinate	
x2	int	Lower right x coordinate	
y2	int	Lower right y-coordinate	
code	int		Return error code

7.8、Smart event parameters configuration

7.8.1、Face recognition attribute

Brief description:

- Get face recognition parameters interface

Request URL:

- /api/event/face-detect

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": NULL
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": 1,
    "startPointX1": 1,
    "startPointY1": 1,
    "endPointX2": 10000,
    "endPointY2": 10000,
    "enableRegionDetect": 1,
  }
}
```


Parameter Name	Type	Example values	Description
			ratio Y coordinate
enableRegionDetect	bool		Area identification switch 0:off 1:on
alarmLinkScope	int		Alarm linkage type, face linkage currently only supports IO alarm
alarmTriggerMode	int		Trigger mode: 0:off; 1:white list; 2:blacklist, associated with process
captureMode	int		Capture mode
enableFaceAlarm	bool		Face alarm switch 0:off 1:on
minFaceSizes	int		Minimum pixels for face recognition [30, 300]
enableFaceTrackingBox	bool		Face tracking box display enable 0:off 1:on
enableOsdRegion	bool		enable osd overlay 0:off 1:on
faceScore	int		Face comparison recognition threshold [0, 100]
modelVersion	string		Model version number string
sensitivity	int		Sensitivity Retention
blurThreshold	int		Capture Blur Threshold
faceScope	int		smart event support type Reserved
smartType	int		smart event type 0:Face recognition 1:Face recognition + Smart event 2:Face recognition + Smart event Reserved
schedule	array		Schedule: 8 periods per day in 7 days, unit in S, ["start time", "end time"], e.g. [0,1440]

Remarks

- For more return error codes, please see the error code description on the home page

7.8.2、Area intrusion detection configuration

Brief description:

- Interface to obtain area intrusion detection parameters

Request URL:

- /api/event/intrusion

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": NULL
}
```

Example of getting parameters returned:

```
{
  "code": 0,
  "data": {
    "enable": true,
    "alarmLinkScope": 3,
    "sensitivity": [
      {"subSensitivity": 50},
      {"subSensitivity": 50},
      {"subSensitivity": 50},
      {"subSensitivity": 50}
    ],
    "time": [
      {"subTime": 50},
      {"subTime": 50},
      {"subTime": 50},
      {"subTime": 50}
    ],
    "area": [
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0, "x3": 0, "y3": 0, "x4": 0,
"y4": 0},
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0, "x3": 0, "y3": 0, "x4": 0,
"y4": 0},
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0, "x3": 0, "y3": 0, "x4": 0,
"y4": 0},
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0, "x3": 0, "y3": 0, "x4": 0,
"y4": 0}
    ],
    "schedule": [
      [[0,1440], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0]],
      [[0,1440], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0]],
      [[0,1440], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0]],
      [[0,1440], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0]],
      [[0,1440], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0]],
      [[0,1440], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0]],
      [[0,1440], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0], [0,0]]
    ],
    "trigger": {
      "enableAlarmSntp": 0,
      "enableAlarmFtp": 0,
      "enableAlarmCloud": 0,
      "enableAlarmRecord": 0,
      "enableAlarmIo": 0,
      "enableAlarmFlash": 0,
      "enableAlarmSound": 0,
      "enableAlarmFlashAll": 0
    }
  }
}
```

Setting Example: Set the parameter request:

```

{
  "action": "set",
  "data": {
    "enable": true,
    "alarmLinkScope": 3,
    "sensitivity": [
      {"subSensitivity": 50},
      {"subSensitivity": 50},
      {"subSensitivity": 50},
      {"subSensitivity": 50}
    ],
    "time": [
      {"subTime": 50},
      {"subTime": 50},
      {"subTime": 50},
      {"subTime": 50}
    ],
    "area": [
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0, "x3": 0, "y3": 0, "x4": 0,
"y4": 0},
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0, "x3": 0, "y3": 0, "x4": 0,
"y4": 0},
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0, "x3": 0, "y3": 0, "x4": 0,
"y4": 0},
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0, "x3": 0, "y3": 0, "x4": 0,
"y4": 0}
    ],
    "schedule": [
      [[0,1440],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0]],
      [[0,1440],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0]],
      [[0,1440],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0]],
      [[0,1440],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0]],
      [[0,1440],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0]],
      [[0,1440],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0]],
      [[0,1440],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0]]
    ],
    "trigger": {
      "enableAlarmSntp": 0,
      "enableAlarmFtp": 0,
      "enableAlarmCloud": 0,
      "enableAlarmRecord": 0,
      "enableAlarmIo": 0,
      "enableAlarmFlash": 0,
      "enableAlarmSound": 0,
      "enableAlarmFlashAll": 0
    }
  }
}

```

Set the parameters to return:

```

{
  "code": 0,
  "data": {}
}

```

Parameter Description

Parameter Name	Type	Example values	Description
enable	bool		Area intrusion detection enable 0:off 1:on
alarmLinkScope	int		Alarm linkage type
sensitivity	int		4 Group sensitivity sub Sensitivity [0, 100]
time	int		4 Group Time Threshold sub Time [0, 10]
area	array		4 Group ten thousand points ratio area x1 y1 x2 y2 x3 y3 x4 y4
schedule	array		Schedule:8 periods per day in 7 days, unit in S, ["start time", "end time"], e.g. [0,1440]

Remarks

- For more return error codes, please see the error code description on the home page

7.8.3、Line crossing detection configuration

Brief description:

- Interface for line-crossing detection parameters

Request URL:

- /api/event/line-crossing

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": NULL
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": true,
    "alarmLinkScope": 3,
    "sensitivity": [
      {"subSensitivity": 50}, {"subSensitivity": 50},
      {"subSensitivity": 50}, {"subSensitivity": 50}
    ],
    "direction": [
```

```

        {"subDirection": 50}, {"subDirection": 50},
        {"subDirection": 50}, {"subDirection": 50}
    ],
    "area": [
        {"startPointx1": 0, "startPointy1": 0, "endPointx2": 0, "
endPointy2": 0},
        {"startPointx1": 0, "startPointy1": 0, "endPointx2": 0, "
endPointy2": 0},
        {"startPointx1": 0, "startPointy1": 0, "endPointx2": 0, "
endPointy2": 0},
        {"startPointx1": 0, "startPointy1": 0, "endPointx2": 0, "
endPointy2": 0},
    ],
    "schedule": [
        [0,1440], [0,0], [0,0], [0,0],
        [0,0], [0,0], [0,0], [0,0]],
        [0,1440], [0,0], [0,0], [0,0],
        [0,0], [0,0], [0,0], [0,0]],
        [0,1440], [0,0], [0,0], [0,0],
        [0,0], [0,0], [0,0], [0,0]],
        [0,1440], [0,0], [0,0], [0,0],
        [0,0], [0,0], [0,0], [0,0]],
        [0,1440], [0,0], [0,0], [0,0],
        [0,0], [0,0], [0,0], [0,0]],
        [0,1440], [0,0], [0,0], [0,0],
        [0,0], [0,0], [0,0], [0,0]],
        [0,1440], [0,0], [0,0], [0,0],
        [0,0], [0,0], [0,0], [0,0]],
        [0,1440], [0,0], [0,0], [0,0],
        [0,0], [0,0], [0,0], [0,0]],
    ],
    "trigger": {
        "enableAlarmSntp": 0,
        "enableAlarmFtp": 0,
        "enableAlarmCloud": 0,
        "enableAlarmRecord": 0,
        "enableAlarmIo": 0,
        "enableAlarmFlash": 0,
        "enableAlarmSound": 0,
        "enableAlarmFlashAll": 0
    }
}
}
}

```

Setting Example: Set the parameter request:

```

{
    "action": "set",
    "data": {
        "enable": true,
        "alarmLinkScope": 3,
        "sensitivity": [
            {"subSensitivity": 50}, {"subSensitivity": 50},
            {"subSensitivity": 50}, {"subSensitivity": 50}
        ],
        "direction": [
            {"subDirection": 50}, {"subDirection": 50},
            {"subDirection": 50}, {"subDirection": 50}
        ],
        "area": [
            {"startPointx1": 0, "startPointy1": 0, "endPointx2": 0, "

```

```

endPointy2": 0},
    {"startPointx1": 0,"startPointy1": 0,"endPointx2": 0,"
endPointy2": 0},
    {"startPointx1": 0,"startPointy1": 0,"endPointx2": 0,"
endPointy2": 0},
    {"startPointx1": 0,"startPointy1": 0,"endPointx2": 0,"
endPointy2": 0},
    ],
    "schedule": [
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]]
    ],
    "trigger": {
        "enableAlarmSntp": 0,
        "enableAlarmFtp": 0,
        "enableAlarmCloud": 0,
        "enableAlarmRecord": 0,
        "enableAlarmIo": 0,
        "enableAlarmFlash": 0,
        "enableAlarmSound": 0,
        "enableAlarmFlashAll": 0
    }
}
}

```

Set the parameters to return:

```

{
    "code": 0,
    "data": {}
}

```

Parameter Description

Parameter Name	Type	Example values	Description
enable	bool		Line crossing detection enable 0:off 1:on
alarmLinkScope	int		Alarm linkage type
sensitivity	array		4 Group sensitivity sub Sensibility [0, 100]
direction	array		4 Group Direction sub Direction 0:A<->B 1:A->B 2:B->A
area	array		4 group ten thousand points ratio area startPointx1 startPointy1 endPointx1 endPointy1
schedule	array		Schedule:8 periods per day in 7 days, unit in S, ["start

Parameter Name	Type	Example values	Description
			time", "end time"], e.g. [0,1440]

Remarks

- For more return error codes, please see the error code description on the home page

7.8.4、Wandering detection configuration

Brief description:

- Wandering detection parameter interface

Request URL:

- /api/event/loitering

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": NULL
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": true,
    "alarmLinkScope": 3,
    "sensitivity": [
      {"subSensitivity": 50}, {"subSensitivity": 50},
      {"subSensitivity": 50}, {"subSensitivity": 50}
    ],
    "direction": [
      {"subDirection": 50}, {"subDirection": 50},
      {"subDirection": 50}, {"subDirection": 50}
    ],
    "area": [
      {"startPointx1": 0, "startPointy1": 0, "endPointx2": 0, "endPointy2": 0},
      {"startPointx1": 0, "startPointy1": 0, "endPointx2": 0, "endPointy2": 0},
      {"startPointx1": 0, "startPointy1": 0, "endPointx2": 0, "endPointy2": 0}
    ]
  }
}
```

```

endPointy2": 0},
    {"startPointx1": 0,"startPointy1": 0,"endPointx2": 0,"
endPointy2": 0},
    ],
    "schedule": [
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]]],
    ],
    "trigger": {
        "enableAlarmSntp": 0,
        "enableAlarmFtp": 0,
        "enableAlarmCloud": 0,
        "enableAlarmRecord": 0,
        "enableAlarmIo": 0,
        "enableAlarmFlash": 0,
        "enableAlarmSound": 0,
        "enableAlarmFlashAll": 0
    }
}
}

```

Setting Example: Set the parameter request:

```

{
    "action": "set",
    "data": {
        "enable": true,
        "alarmLinkScope": 3,
        "sensitivity": [
            {"subSensitivity": 50}, {"subSensitivity": 50},
            {"subSensitivity": 50}, {"subSensitivity": 50}
        ],
        "direction": [
            {"subDirection": 50}, {"subDirection": 50},
            {"subDirection": 50}, {"subDirection": 50}
        ],
        "area": [
            {"startPointx1": 0,"startPointy1": 0,"endPointx2": 0,"
endPointy2": 0},
            {"startPointx1": 0,"startPointy1": 0,"endPointx2": 0,"
endPointy2": 0},
            {"startPointx1": 0,"startPointy1": 0,"endPointx2": 0,"
endPointy2": 0},
            {"startPointx1": 0,"startPointy1": 0,"endPointx2": 0,"
endPointy2": 0},
        ],
        "schedule": [

```

```

        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
        ],
    "trigger": {
        "enableAlarmSntp": 0,
        "enableAlarmFtp": 0,
        "enableAlarmCloud": 0,
        "enableAlarmRecord": 0,
        "enableAlarmIo": 0,
        "enableAlarmFlash": 0,
        "enableAlarmSound": 0,
        "enableAlarmFlashAll": 0
    }
}
}

```

Set the settings to return:

```

{
    "code": 0,
    "data": {}
}

```

Parameter Description

Parameter Name	Type	Example values	Description
enable	bool		Wandering detection enable 0:off 1:on
alarmLinkScope	int		Alarm linkage type
sensitivity	array		4 Group sensitivity sub Sensitivity [0, 100]
time	array		4 Group Time Threshold sub Time [0, 10]
area	array		4 Group ten thousand points ratio area x1 y1 x2 y2 x3 y3 x4 y4
schedule	array		Schedule: 8 periods per day in 7 days, unit in S, ["start time", "end time"], e.g. [0,1440]

Remarks

- For more return error codes, please see the error code description on the home page

7.8.5、People gathering detection configuration

Brief description:

- People gathering detection parameter interface

Request URL:

- /api/event/people-gathering

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": NULL
}
```

Get parameter return:

```
{
  "code": 0,
  "data": {
    "enable": true,
    "alarmLinkScope": 3,
    "sensitivity": [
      {"subSensitivity": 50}, {"subSensitivity": 50},
      {"subSensitivity": 50}, {"subSensitivity": 50}
    ],
    "time": [
      {"subTime": 50}, {"subTime": 50},
      {"subTime": 50}, {"subTime": 50}
    ],
    "area": [
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0, "x3": 0, "y3": 0, "x4":
0, "y4": 0},
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0, "x3": 0, "y3": 0, "x4":
0, "y4": 0},
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0, "x3": 0, "y3": 0, "x4":
0, "y4": 0},
      {"x1": 0, "y1": 0, "x2": 0, "y2": 0, "x3": 0, "y3": 0, "x4":
0, "y4": 0},
    ],
    "schedule": [
      [0,1440], [0,0], [0,0], [0,0],
      [0,0], [0,0], [0,0], [0,0]],
      [0,1440], [0,0], [0,0], [0,0],
      [0,0], [0,0], [0,0], [0,0]],
      [0,1440], [0,0], [0,0], [0,0],
      [0,0], [0,0], [0,0], [0,0]],
      [0,1440], [0,0], [0,0], [0,0],
      [0,0], [0,0], [0,0], [0,0]]
    ]
  }
}
```



```
        [[0,1440],[0,0],[0,0],[0,0],
        [0,0],[0,0],[0,0],[0,0]],
    ],
    "trigger": {
        "enableAlarmSmtp": 0,
        "enableAlarmFtp": 0,
        "enableAlarmCloud": 0,
        "enableAlarmRecord": 0,
        "enableAlarmIo": 0,
        "enableAlarmFlash": 0,
        "enableAlarmSound": 0,
        "enableAlarmFlashAll": 0
    }
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter Description

Parameter Name	Type	Example values	Description
enable	bool		People gathering detection enable 0:off 1:on
alarmLinkScope	int		Alarm linkage type
persionNum	array		Number of 4 groups of people 0-100
schedule	array		schedule:8 periods per day in 7 days, unit in S, ["start time", "end time"], e.g. [0,1440]
area	array		4 Group ten thousand points ratio area x1 y1 x2 y2 x3 y3 x4 y4

Remarks

- For more return error codes, please see the error code description on the home page

7.8.6、Smart detection mode

Short description:

- Smart detection mode

Request URL:

- /api/event/smart-mode

Request Method:

- POST

Get Example: Get Parameter Request:

```
{
  "action": "get",
  "data": NULL
}
```

Get parameter return:

```
{
  "code": "0",
  "data": {
    "smartMode": 1,
    "face2FtpEn": 1,
    "audioAlarmEn": 1
  },
}
```

Setting Example: Set the parameter request:

```
{
  "action": "set",
  "data": {
    "smartMode": 1,
    "face2FtpEn": 1,
    "audioAlarmEn": 1
  },
}
```

Set the parameters to return:

```
{
  "code": 0,
  "data": {}
}
```

Parameter Description:

Parameter Name	Type	Example values	Description
smartMode	int		smart detection mode 0:off, 1:face, 2 humanoid, 3 humanoid + face + car model
face2FtpEn	int		Upload face to ftp 0:off, 1:on
audioAlarmEn	int		1: Turn on the audio alarm output 0: Turn off the audio alarm output

Remarks

- For more return error codes, please see the error code description on the home page

ivcan.com