

HTTP API Protocol For IP Media Device

CONTENTS

HTTP API Protocol For IP Media Device	1
1 References	8
2 OVERVIEW.....	8
2.1 Transmission Mechanism.....	8
2.2 JSON.....	8
2.3 URL.....	8
2.4 HTTP Header Field	9
2.5 Response Messages Body	10
2.6 Command Categories.....	12
2.7 Error Code.....	12
2.8 Abbreviations	13
2.9 Common error_code.....	13
3 SYSTEM	16
3.1 General	17
3.2 Date&Time	18
3.3 NTP.....	20
3.4 DST.....	21
3.5 Output Configuration.....	23
3.6 User.....	24
3.7 System Information.....	30
3.8 Channel Information.....	31
3.9 Record Information	32
3.10 Network State	33
4 NETWORK	36
4.1 Network Configuration	37
4.1.1 Network Base Information.....	37
4.1.2 WLANScan.....	43
4.2 DDNS.....	45
4.2.1 DDNS Parameters.....	45
4.3 Email	47
4.3.1 Email Parameters	47
4.3.2 Email Test	49
4.4 FTP	50
4.4.1 FTP Parameters	50
4.5 IP Filter Parameters.....	53
4.6 Hhttps Configure.....	55
4.7 Voice Assistant	56
4.8 ipv6	57
4.9 snmp	58
4.10 Rtsp.....	59
4.11 Onvif.....	60
4.12 Tuya.....	60

5 CHANNEL.....	61
5.1 Channel Configuration	61
5.1.1 Broadcast ipc.....	61
5.1.2 Digital channel	62
5.1.3 Custom protocol management	68
5.1.4 Analog channel	69
5.1.5 Wireless channel.....	70
5.1.6 Digital wireless channel configuration	72
5.2 OSD	74
5.3 Image Control	78
5.4 Video Color	86
5.5 Video Cover.....	88
5.6 PTZ	90
5.7 ROI	91
5.8 POE Power	102
5.9 Video Crop	103
6 STREAM.....	105
6.1 Encode	105
6.2 Capture	110
6.3 Rtsp Url	111
7 ALARM	112
7.1 IO Alarm.....	113
7.2 Motion Alarm.....	115
7.3 PTZ Linkage	119
7.4 Exception	122
7.5 PIR.....	123
7.6 Flood-light.....	127
7.7 Sound Detection	130
7.8 Occlusion Detection.....	133
7.9 Perimeter Intrusion Detection.....	135
7.10Line Crossing Detection	141
7.11Stationary Object Detection	146
7.12Pedestrian Detection	152
7.13Face Detection.....	157
7.14Cross Counting.....	162
7.15IntelligentAnalysis.....	165
7.16 Voice Prompt	168
7.16.1 Description.....	168
7.16.2 Syntax.....	168
7.16.3 Example.....	172
7.17 Combination Alarm.....	174
7.18 Linkage Schedule	180
7.19 Serial Transport.....	183
7.20 IO-AlarmStatus.....	184

7.21 Disarming	185
8 STORAGE	190
8.1 Disk	190
8.1.1 Disk Configuration	190
8.1.2 Disk Control	192
8.1.3 Disk Format	193
8.2 Disk Group	195
8.3 Cloud	197
8.4 Audio	200
8.5 RAID	202
8.6 Cloud(new)	139
9 SCHEDULES	212
9.1 Schedules	212
10 RECORD	214
10.1 Record Configuration	214
10.2 Search Record	215
10.3 Playback rtsp url	219
10.6 Month Search	222
10.7 Pic Playback	222
10.7.1 Syntax	223
10.7.2 Parameters	223
10.8 Playback Page	226
10.9 Record Tag	228
11 MAINTENANCE	231
11.1 Log	231
11.2 Load Default Parameter	239
11.3 Auto Reboot	240
11.4 Ftp Upgrade	240
11.5 Device Reboot	244
11.6 Import/Export Parameter	245
11.6.1 Description	245
11.6.2 Syntax	245
11.6.3 Parameters	245
11.7 IPC Maintenance	246
11.7.1 Description	246
11.7.2 Syntax	246
11.7.3 Parameters	248
11.8 System Upgrade	253
11.9 Developer Mode	255
11.10 Device Shutdown	257
11.11 Http Upgrade	257
11.12 Defogging Fan	258
12 LOGIN	258
12.1 Login	259

12.1.1 Description.....	259
12.1.2 Syntax.....	261
12.1.3 LOGIN.....	271
12.1.4 Recover Password.....	273
12.2 Get Device Page.....	279
12.2.1 Description.....	279
12.2.2 Syntax.....	280
12.2.3 Example.....	281
12.3 Preview Control.....	285
12.3.1 Description.....	285
12.3.2 Syntax.....	286
12.3.3 Example.....	294
12.3.3 Example.....	294
12.4 LogOut.....	300
12.5 Download root certificate File(A01).....	301
12.5.1 Description.....	301
12.5.2 Syntax.....	301
12.5.3 Example.....	302
12.6 Account Rules.....	303
12.6.1 Description.....	303
12.6.2 Syntax.....	303
12.6.3 Example.....	304
12.7 Request pubkey or randbyte.....	305
12.7.1 Description.....	305
12.7.2 Syntax.....	306
12.7.3 Example.....	307
12.7.4 Instrucion.....	308
13 MutexParam.....	310
13.1 MutexParam.....	310
14 Function.....	310
14.1 Snapshot.....	310
14.2 Request I Frame.....	313
14.3 ANR.....	313
14.4 ETR.....	314
15 Event.....	315
15.1 event check.....	315
15.1 event check Description and example.....	321
15.2 Http listening.....	323
15.2 Http listening description.....	324
15.2.1 Push alarm through Get mode.....	324
15.2.2 Push alarm through POST mode.....	325
15.2.3 Http listening keep alive.....	330
16 Custom.....	331
16.1 Push server.....	331

16.2 Shared Cloud(A01 customized).....	332
16.3 Write Log(A01 customized)	332
16.4 Cloud Upgrade(A01 customized)	333
16.5 Develop Mode (A01 customized)	334
16.6 Cert Manage (A01 customized)	336
16.7 Zero Channel(B09 customized).....	336
16.8 Zero Channel State(B09 customized).....	338
16.9 Serail(B09 customized)	338
16.10 Force I Frame	339
17 AI.....	340
17.1 Setup.....	340
17.1.1 Face Detection	341
17.1.2 Human & Vehicle Detection.....	346
17.1.3 Perimeter Intrusion Detection.....	351
17.1.4 Line Crossing Detection	357
17.1.5 Cross Counting	362
17.1.6 Heat Map	366
17.1.7 Stationary Object Detection	369
17.1.8 Crowd Density Detection	375
17.1.9 Queue Lenght Detection.....	379
17.1.10 License Plate Detection.....	383
17.1.11 Rare Sound Detection	388
17.1.12 AI Func Schedule.....	392
17.2 Recognition.....	396
17.2.1 Model Configuratuon.....	396
17.2.2Face Group.....	398
17.2.3 Face.....	410
17.2.4Add Compare Face Image	415
17.2.5 Database face information query.....	420
17.2.6 Additional Face Image.....	427
17.2.7Snaped Faces Search and Match	430
17.2.8 Snaped Objects Search	438
17.2.9 License Plate Group	443
17.2.10 License Plate	454
17.2.11 Database license plate information query	460
17.2.12 Snaped License Plates Search and Match	466
17.2.13 Snaped Faces and Objects Count Get (VHD).....	467
17.3 Face Attendance(NVR only).....	468
17.4 Statastics (NVR only).....	472
17.4.1 Face Search	472
17.4.2 Hman & Vehicle Search.....	476

17.4.3 Cross Counting Statistics	478
17.4.4 Heat Map Statistics	480
17.5 Snaped face or object (pedestrian, car) real time alarm (NVR、IPC only, A01 no)	483
17.6Attribute Detection.....	487
17.7Repeat Customer	490
17.8 Cross Counting Scenario	495
17.8.1 RealTime Info.....	496
17.8.2 Map.....	499
17.8.3 Statistics	501
17.8.4 Config.....	505
17.8.5 Image Manage	509
17.x AI Operation Result Type(NVR only)	511
18 Extended Functionality	512
18.1 IPCVoice Prompt	512
18.1.1 Description.....	512
18.1.2 Syntax.....	513
18.1.3 Example.....	514
19 Push.....	516
19.1.1 Description.....	516
19.1.2 Syntax.....	517
19.1.3 Example.....	521
Appendixes	541
A.1Additional Error Codes(A01).....	541
A.2 Secondary certification development guide.....	546
A.3 Set Password Development Guide	548
FQA	550

1 References

- [1] RFC2616 Hypertext Transfer Protocol-HTTP/1.1
- [2] RFC7159 The JavaScript Object Notation (JSON) Data Interchange Format
- [3] RFC 2396 Uniform Resource Identifiers (URI): Generic Syntax and Semantics
- [4] RFC 2617 HTTP Authentication:Basic and Digest Access Authentication
- [5]<http://json-schema.org/>

2 OVERVIEW

2.1 Transmission Mechanism

The HTTP API transaction starts from a request from a client application. The web server on the IP media devices processes the request and sends the response back to the client application. The HTTP request is taken in POST or Get method form as described in the following paragraphs. If the request is successful, the IP media video device will return a HTTP header contains 200 OK. The HTTP Body will contain actual result or error message by JSON format if an error occurs.

2.2 JSON

A device must support the syntax defined by RFC7159 and UTF-8 character set . All Json files must adopt UTF-8 encoding according to RFC3629

2.3 URL

The URL scheme is used to locate device resources via a specific protocol in the network. This section defines the syntax and semantics for http(s) URLs.

<protocol>://<host>[:port][/*fixed string*]/<cmd name>

protocol: URL scheme for the particular request. The HTTP protocol is allowed in this specification.

host: The host field refers to the hostname, IP address, or the Fully Qualified Domain Name of an IP device.

port: The port field refers to the port number of that host on which the identified resource is located at the IP device listening for TCP connections. If the port is empty or not given, the defaultport is assumed. For HTTP, the default port is 80. For HTTPS the default port is 443.

fixed string: Fixed string, usually use “API”.

cmd name:The specific command to an IP device.

2.4 HTTP Header Field

Requests from the video management system or the client application are packed in HTTP messages. A request message composed of three parts: the connection header field, the authorization header field, and the entity body field. HTTP/1.1 is implemented and utilized according to RFC 2616 in the IP devices. For a video management system or client application that uses persistent connection for multiple transactions, it is required to implement “Connection: Keep-Alive” HTTP header field, while also adopt the “Connection: close” HTTP header field for the last transaction of the persistent connection.

When a video management system or client application sends any request to the device, it must be authenticated by means of Basic Access or Digest Access Authorization according to RFC 2617, and thus all the devices are required to support Basic Access or Digest Access. Authorization header field is sent along with each request, and if a user is authenticated, the request will follow the normal execution flow. If client HTTP request is with no authentication credentials, unauthorized HTTP response (401) will be returned with WWW-Authenticate header field.

The Content-Type entity-header field indicates the media type of the entity body. The Content-Type may be designated as “application/json; charset='UTF-8'”, “application/octet-stream”, etc. For configuration information, the Content-Type is usually “application/json; charset='UTF-8'”.

If there is a JSON block for the HTTP request or response, the Content-Type and Content-Length will be set in the headers of the HTTP message.

The Content-Type is usually "application / json; character set = "UTF-8"". The request uses the POST method, the body is in JSON format, and contains at least one version number field.

For example:

HTTP Request with message body:

Except for the few APIs before login, other APIs need to be authenticated with the cookie and X-csrfToken fields returned by login in the http header. See chapter 12.1 for login. The format of the API request is as follows, where the requested data and the returned data are in the data field, and version is the version number. Because it is a reserved field and is not really used, the value "1.0" is temporarily fixed. When the server returns data, it will return the error code and error reason if there is an error.

```
POST/API/SystemInfo/Base/Get HTTP/1.1
Content-Type: application/json
X-csrfToken: 14559384d58d55d9d80bf4baf048684f366c77905d665d99ba1bde2cdcf81b08
Cookie: session=cc3ec99c6f1295489c86f8842e1dd719c194c637af5b965a23932c41188a3acc
Content-Length: 1234,
{
  "version": "x.x"
  "data": {
    }
}
```

If only a small amount of API support is needed, it can be accessed through summary authentication. The format is as follows:

```
POST/API/SystemInfo/Base/Get HTTP/1.1
Authorization: Digest username="admin",realm="RS realm",nonce="d7357432-26b6-49d7-bc31-
```

```
cd2c690fd115",uri="/action/getConfig/Camera",cnonce="dedb934132796d7066c7cb59068e3487",nc=00000001,response="de
8918839a8fa301bf6badf2967965c8",qop="auth"
Content-Type: application/json
Content-Length: 1234,
{
  "version": "x.x"
  "data": {
    }
  }
}
```

2.5 Response Messages Body

The response message from the IP device is a standard HTTP response, information can be included in the entity body field in JSON format. This information includes the result to a request message, or the detailed parameters that required by a request message.

A successful response that don't includes any parameters is as follows:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 40
{
  "result": "success",
  "data": {
    }
  }
}
```

In case of failure, the return format is as follows, where reason is the more detailed error reason, error_code is an error code. Please refer to Chapter 2.9 for the specific error code. The client can make corresponding prompt according to the error code. If there is still data to return in case of error, the returned data is stored in the data field.

```
HTTP/1.1 400 OK
Content-Type: application/json
Content-Length: 40
{
  "result": "failed",
  "reason": "xxxxxxx"
  "error_code": "user_blocked"
  "data": {
    }
  }
}
```

A part of channel failed response that is as follows:

```
HTTP/1.1 400 OK
Content-Type: application/json
```

Content-Length: 40

```
{
  "version": "x.x",
  "error_code": "part_failed",
  "reason": "xxxxxxx"
  "ch_error_code": [{"error_code": "no_support", "channel": "IP_CH1"}],
  "data": {
  }
}
```

cloud_video_upload_chn_limit Return information attached to the error code

```
{
  "result": "failed",
  "reason": "Save parameter failed",
  "error_code": "cloud_video_upload_chn_limit",
  "data": {
    "cloud_video_is_used": [
      "CH1",
      "CH3"
    ],
    "max_cloud_video_upload_num": 2
  }
}
```

Smart mutex return example:

```
{
  "result": "failed",
  "channel_max": 1,
  "error_code": "part_failed",
  "ch_error_code": [
    {
      "channel": "CH1",
      "error_code": "illegal_operation",
      "reason": "Illegal Operation, against the intelligent mutual exclusion !",
      "mutual_array": [
        "PID",
        "LCD"
      ]
    }
  ],
  "data": {
  }
}
```

```
}
```

Server enable mutex return example:

```
{  
  "result": "failed",  
  "error_code": "server_mutually",  
  "reason": "The server is mutually exclusive !",  
  "mutual_array": [  
    "DROPBOX",  
    "Google Drive"  
  ],  
  "data": {}  
}
```

2.6 Command Categories

This specification is divided into different command categories. The following command is defined:

Command	Description
System	Configure and operate the general system functions.
Network	Configure network interfaces and parameters
Channel	Configure channel parameters
Stream	Configure and control the streaming media content
Alarm	Configure and control alarm functions and parameter
Storage	Configure device storage
Schedules	Configure schedules
Record	Configure and control record functions and parameter
Maintenance	Configure maintenance information

2.7 Error Code

As with any other protocol, errors may occur during communications, protocol or message processing. Errors may contain header value or be received in a not expected or experience a socket timeout. To indicate and interpret protocol error, HTTP protocol has defined a set of standard status codes. According to this specification, the IP

devices will use appropriate HTTP protocol defined status codes for error reporting and when received handle accordingly.

Error Code	Description
200	The request has succeeded.
400	The request was badly formed. This is commonly used for creating or updating a resource, but the data was incomplete or incorrect.
401	The request requires user authentication to access this resource. If the request contains invalid authentication data, this code is sent.
403	The request is not allowed because the server is refusing to fill the request.
404	The requested resource does not exist.
500	An internal server error has occurred.
501	The requested is not implemented.

2.8 Abbreviations

For the purposes of the present document, the following abbreviations apply:

IPC	IP Camera
NVR	Network Video Recorder
CH	Analog channel
IP_CH	Digital channel
WIFI_CH	Wireless channel

2.9 Common error_code

Common error_code include channel Table-2.9.1

General	param_error	Request parameter error
	no_permission	No permission
	first_login	First login, mandatory password change
	part_failed	Channel save part failed
	no_support	No support
	frequent_operation	Frequent operation
	passwd_expired_login	Login with expired password and force password change
	default_failed	Failed to restore default value
	token_generation_failed	Failed to generate token
	token_invalid	token invalid
	device_busy	The device is processing the maximum number of

		APIs
	user_expired_login	User expired login
	data_saving_busy	The device is busy saving
	user_locked_login	User locked login
	function_busy	Function busy
	network_port_conflict	Network port conflict
	group_name_error	Group name error
Session	no_login	No login
	expired	Login expired
	one_IE	Only one user can log in to a browser
	logout	Logout
	login_at_other	Account login in other places
	device_reboot	Device reboot
	passwd_expired	Password expired
	param_changed	User parameter changed
	network_changed	Network port changed
	ssl_error	Certificate re-signing
	netip_limited	After the logged in IP is added to the blacklist
	forced_offline	Forced offline
	user_expired	User expired
	no_heartbeat	Heartbeat timeout failure
	disk_changed	Hardware status or parameters change
	ipc_state_changed	Upgrade IPC online, IPC state changed
	have_login	The current number of logins has exceeded the maximum number of logins set by the user
Secondary certification	current_pwd_error_ntime	If the authentication fails for more than a certain number of times, the data will be attached with "retain" _ lo...cked_ Time field, representing the remaining lock time
Save parameters	save_failed	Save failed
	pwd_weak	Modified IPC password has low complexity (IPC)
	modify_failed	Modify failed
	modify_failed_pwd_err	Error in modifying the IP user name or password of the IPC
	modify_failed_syntax_err	Modify the IP syntax configuration error of IPC (mask or gateway)
search	search_failed	Search failed
playback	device_play_locked	The board end is playing back and rejecting the web request
operation	operation_failed	Operation failed
User name, password and permission	session_invalid	
	illegal_request	Illegal request
	overreach	Sub-users can only set their own information

modification	short_modify_time	Password change interval is too short
	current_pwd_error	Current password error
	username_empty	User name empty
	username_repeat	User name repeat
	username_invalid	Only letters, numbers and underscores are allowed for user name
	unmatched_pwd	The two inputs are different
	pwd_empty	Password empty
	pwd_length_err	Password length error
	pwd_equal_name	Password cannot be the same as user name or user name reverse
	pwd_equal_old	The new password cannot be the same as the previous N passwords. The number field will be attached to the data, representing N
	pwd_repeated	The new password and the old password must have two different characters
	pwd_weak_rule	The password must contain at least two combinations of the following characters: lowercase letters, uppercase letters, numbers, special characters and spaces
	pwd_risk	Password belongs to weak password dictionary
	pwd_invalid	Password invalid
Mail test	user_auth_failed	Authentication failed
	data_error	Sending data error
	net_unreachable_or_dns_wrong	Network failure or DNS server error
	connect_server_err	Failed to connect to the server
	check_smtp_port	Connection failed. Check whether the port is correct
	tls_ssl_handshake_err	TLS/SSL handshake error
	email_connect_err	Connection error, please check the recipient account
upgrade	in_user_interface	User interface
	updating	The board end is being upgraded
	lack_memory	insufficient memory
	file_error	File error
	no_need_upgrade	The current version is the latest and does not need to be updated
http protocol	not_modified	The page is not modified (304)
	http_redirect_https	http jump https (307)
	not_found	Page not found (404)
	method_not_allowed	Request method is not allowed (405)
	payload_too_large	Payload too large (413)
	uri_too_long	uri too long (414)

	internal_server_error	Server internal error (500)
	service_unavailable_error	The service is unavailable (503, exceeding the maximum number of tcp connections)
Configure IPC parameters		See details A.1
DDNS test	connect_error	Network connection error
	send_error	Send error
	receive_error	Receive error
	network_info_error	Network info error
	unknown_error	Unknown error
	net_unreachable_or_dns_wrong	Network failure or DNS server filling error!
	connect_ip_server_timeout	Connection to IP server timed out!
	connect_ddns_server_timeout	Connection to DDNS server timed out!
	send_request_to_server_timeout	Sending request to DDNS server timed out!
	ddns_server_response_timeout	DDNS server response timeout!
	invalid_username_or_password	Invalid user name or password!
	hostname_not_exist	Hostname not exist !
	hostname_or_username_not_exist	The host name or user name does not exist !
	hostname_blocked	Hostname exists but is locked !
	username_or_password_is_empty	User name or password cannot be empty !
	not_donor_stop_update	Not donor, update stopped !
	not_fully_qualified_domain_name	The domain name is not fully compliant!
	host_under_different_account	The host exists but under different accounts !
	too_many_or_too_few_hosts_found	Too many or too few servers found
	dns_service_error	DNS server error, please try again in an hour!
requested_ip_address_failed	Failed to request IP address	
Cloud video	cloud_video_upload_chn_limit	The video cloud upload opening channel exceeds the limit
Auto capture	auto_capture_chn_limit	The automatic capture opening channel exceeds the limit
FTP test	illegal_param	Illegal parameter
	connect_failed	Failed to connect to Ftp
	login_failed	User login failed
	write_file_failed	fail to write to file
	created_dir_failed	Failed to create folder
	ftp_close_failed	Failed to close Ftp service
Add IPC	lack_channel	There is no free channel for adding new IPC

3 SYSTEM

3.1 General

GET	
URL	POST /API/SystemConfig/General/Get
Description	Get System General Information
Request Body	None
Successful Response	System general information JSON (show as follow Table-3.1)

SET	
URL	POST /API/SystemConfig/General/Set
Description	Set System General Information
Request Body	System general information JSON (show as follow Table-3.1)
Successful Response	The successful result response that described in 2.5

Table-3.1 (System general information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
device_name	Max length: 31byte	string	Device name
menu_timeouts	0,30,60,120,300,600	int	Menu auto lock Time. Unit:Second
FisheyeOrAI	"Fisheye" "AI"	string	AI fisheye mutual exclusion (NVR only)
cert_warn_time	0-90	Int	Certificate alert time (A01 only)
root_cert_warn		bool	Enable alert for device root certificate(A01 only)
session_timeout	5-1440	int	Web session timeout (minutes)
preview_session_timeout		bool	Preview/playback session timeout

Example:

Request message:

POST /API/SystemConfig/General/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {

  }
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "result": "success",
  "data": {
    "device_name": "N5208EN",
    "menu_timeouts": 30,
    "session_timeout": 5,
    "preview_session_timeout": true
  }
}
```

```

    }
}

```

3.2 Date&Time

GET	
URL	POST /API/SystemConfig/DateTime/Get
Description	Get System Date & Time Information
Request Body	none
Successful Response	Date& Time information JSON (show as follow Table-3.2)

SET	
URL	POST /API/SystemConfig/DateTime/Set
Description	Set System Date & TimeInformation
Request Body	Date& Time information JSON (show as follow Table-3.2)
Successful Response	The successful result response that described in 2.5

Table-3.2 (Date& Time information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
date		string	System date Format is: yyyy/mm/dd
time			System time Format is:hh:mm:ss.
synchronize_computer_time		Bool	Synchronize computer time (IPC only)
date_format	"MM/DD/YYYY", "YYYY-MM-DD", "DD/MM/YYYY"	string	Date format
time_format	12,24	int	Time format, Unit: hour
time_zone	"GMT-12:00", "GMT-11:00", "GMT-10:00", "GMT-9:00", "GMT-8:00", "GMT-7:00", "GMT-6:00", "GMT-5:00", "GMT-4:30", "GMT-4:00", "GMT-3:30", "GMT-3:00",	string	Time zone

	"GMT-2:00", "GMT-1:00", "GMT", "GMT+1:00", "GMT+2:00", "GMT+3:00", "GMT+3:30", "GMT+4:00", "GMT+4:30", "GMT+5:00", "GMT+5:30", "GMT+5:45", "GMT+6:00", "GMT+6:30", "GMT+7:00", "GMT+8:00", "GMT+9:00", "GMT+9:30", "GMT+10:00", "GMT+11:00", "GMT+12:00", "GMT+13:00"		
--	--	--	--

Example:

Request message:

POST /API/SystemConfig/DateTime/Get HTTP/1.1

```
{
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
```

"result": "success",

"version": "1.0",

"data": {

"date": "05/29/2019",

"time": "15:35:39",

"date_format": "MM/DD/YYYY",

"time_format": 24,

"time_zone": "GMT+8:00"

```
}
```

```
}
```

3.3 NTP

GET	
URL	POST /API/SystemConfig/NTP/Get
Description	Get System NTP Information
Request Body	none
Successful Response	NTP information JSON (show as follow Table-3.3)

SET	
URL	POST /API/SystemConfig/NTP/Set
Description	Set System NTP Information
Request Body	NTP information JSON (show as follow Table-3.3)
Successful Response	The successful result response that described in 2.5

Table-3.3 (NTP information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ntp_enable		bool	NTP enable flag
server	"time.windows.com", "time.nist.gov", "pool.ntp.org", "UserDefined"	string	Network server address
custom_server	Max length: 63byte	string	Customserveraddress Note: Use this variable when the Server value is "User-Defined".
sync_period	0-65535	Int	Synchronization period

Example:

Request message:

```
POST /API/SystemConfig/NTP/GetHTTP/1.1
{
}
```

Response message:

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "result": "success",
  "data": {
    "ntp_enable": "Enable",
```

```

"server": "UserDefined",
  "custom_server": "ntp.ntsc.ac.cn"
}
}

```

3.4 DST

GET	
URL	POST /API/SystemConfig/DST/Get
Description	Get System DST Information
Request Body	none
Successful Response	DST information JSON (show as follow Table-3.4)

SET	
URL	POST /API/SystemConfig/DST/Set
Description	Get System DST Information
Request Body	DST information JSON (show as follow Table-3.4)
Successful Response	The successful result response that described in 2.5

Table-3.4 (DST information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
dst_enable		bool	DST function switch
support_crossyear		bool	Whether it supports cross-year
time_offset	1,2	int	DST offset value.Unit: hour
dst_mode	"Week", "Date"	string	DST mode
start_date		string	DST startdate. The date format is MM/DD/YYYY
end_date		string	DST end date. The date format is MM/DD/YYYY
start_hour		string	DST start time. The time format is hh:mm:ss
end_hour		string	DST end time. For example : "00:20:00" The time format is hh:mm:ss
start_month	"Jan", "Feb", "Mar", "Apr", "May", "Jun",	string	Start month
end_month	"Jul", "Aug",	string	End month

	"Sep", "Oct", "Nov", "Dec"		
start_week	"1st", "2nd", "3rd",	string	The week of the month
end_week	"4th", "Last"	string	The week of the month
start_weekday	"Sun", "Mon", "Tue", "Wed",	string	Which day start from.
end_weekday	"Thu", "Fri", "Sat"	string	Which day end of.

Example:

Request message:

POST /API/SystemConfig/DST/Get HTTP/1.1

```
{
"version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
"version": "1.0",
"result": "success",
"data": {
"dst_enable": false,
"time_offset": 1,
"dst_mode": "Week",
"start_month": "Mar",
"end_month": "Nov",
"start_week": "The 2nd",
"end_week": "The 1st",
"start_weekday": "Sun",
"end_weekday": "Sun",
"start_date": "01/01/2010",
"end_date": "01/01/2010",
"start_hour": "02:00:00",
"end_hour": "02:00:00"
}
```

```

}
}

```

3.5 Output Configuration

(NVR only)

GET	
URL	POST /API/SystemConfig/Output/Get
Description	Get System Output Information
Request Body	None
Successful Response	Output information JSON (show as follow Table-3.5.1)

SET	
URL	POST/API/SystemConfig/Output/Set
Description	SetSystem OutputInformation
Request Body	Output information JSON (show as follow Table-3.5.1)
Successful Response	The successful result response that described in 2.5

Table-3.5.1 (Output information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
output		Json Object	Output information JSON(show as follow Table-3.5.2)

Table-3.5.2 (OutputInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
LIVE-OUT		Json Object	Output information JSON(show as follow Table-3.5.3)

Table-3.5.3 (SingleOutput information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
output_resolution,	"1024x768", "1280x1024", "1440x900", "720P(1280x720)", "1080P(1920x1080)", "1600x1200", "1920x1200", "2K(2560x1440)", "4K(3840x2160)"	string	The VGA resolution of the current output display, the maximum range is determined according to the highest resolution supported by the actual.

3.6 User

GET	
URL	POST /API/SystemConfig/User/Get
Description	Get SystemUser Information
Request Body	User information JSON (show as follow Table-3.6.x)
Successful Response	User information JSON (show as follow Table-3.6.1)

SET	
URL	POST /API/SystemConfig/User/Set
Description	Set SystemUser Information
Request Body	User information JSON (show as follow Table-3.6.1)
Successful Response	The successful result response that described in 2.5

Note: The api is divided into the acquisition and setting of permissions and the acquisition and setting of user passwords. It is reflected in the type field.

Administrators can use user_Level obtains the authority and user password information of ordinary users, and can modify the user name, password, authority and other parameters of all users.

Ordinary users can only obtain their own user password information.

Table-3.6.x (User information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
user_level	"ADMIN" "USER1" "USER2" ... "USER6"	string	It is only required when the administrator gets user permissions, but not for other scenarios
type	"SavePassword" "SavePermission" "SaveTime" (A01) "GetPermission", "LockUser" (A01)	string	Save the user name and password, save the permission, and pass the parameter when obtaining the permission. Not required for other scenarios
support_user_filter		bool	User name search filtering function, used when the number of system channels is greater than or equal to 256

Table-3.6.1 (User information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
type	"SavePassword" "SavePermission" "SaveTime"	string	

	“GetPermission”, “LockUser”		
default_user	“None”, “Admin”, “User2”, “User3” ...	string	Automatic login user, "None" means disable automatic. All enabled users can be set as automatic login user by username. (A01 no)
pwd_unequal_to_ol d_number	"1","2"	string	The new password is different from the previous N historical passwords (A01 dedicated)
min_password_leng th	5-16 /8-15	int	Minimum password length. nvr/dvr: 5-16 IPC: 8-15 (A01 only)
min_password_expi res	0-997	int	The minimum password usage period, 0 means that the password can be changed immediately (A01 only)
advice_password_e xpires	0-998	int	The recommended password usage period is 0, which means no prompt for password change. If it is not 0, it must be longer than min_password_ Expires large (Special for A01)
max_password_expi res	0-999	int	The maximum service life of the password, 0 which means that the password can never be changed. When it is not 0, it must be higher than advice_password_ Expires large (A01 special)
user_info		Json Object	Singleuserinformation JSON(show as follow Table-3.6.2)
secondary_authentic ation	Max length: 16byte	string	Current user password Setuser_info Required, not for get
base_secondary_aut hentication		Json Object	Encryption auth, Table-12.7.2.4

Table-3.6.2(Singleuser user JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ADMIN		Json Object	JSON show as follow Table-3.6.3
USER1		Json Object	
USER2		Json Object	
...		Json Object	
USER6		Json Object	

Table-3.6.3 (Singleuserinformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
user_state	"None" "Normal" "NotYetStart"	string	User state "None": no

	"Expires" "ExpiresClearInfo"		"Normal": normal "NotYetStart": Expiry date is not reached "Expires": Expired, user information will be cleared three days after the end time "ExpiresClearInfo": Expired, user information has been cleared
user_enable		bool	Enable user
start_date		string	Start date of validity. The date format is MM/DD/YYYY (A01 only)
start_time		string	Start time of validity. The time format is hh:mm:ss (A01 only)
end_date		string	Enddate of validity. The date format is MM/DD/YYYY (A01 only)
end_time		string	Endtime of validity. The time format is hh:mm:ss (A01 only)
username	Max length: 16 byte	string	User name
password_enable		bool	Enable user password (A01. IPC is not used, and password must be set)
password_empty		bool	Is password empty Required for Get, not for Set
password	Max length: 16 byte	string	User password
confirm_password	Max length: 16 byte	string	User confirm password Required for Set, not for Get
modify_password_time		string	Date and time of last password modification Required for Get, not for Set (A01 only)
user_locked		bool	Whether the user is locked (A01 only)
permission		Json Object	Userpermissioninformation JSON(show as follow Table-3.6.4) [Note: [Get]For general users only] [Note: [Set]For admin only] Pass parameters when obtaining and setting permissions.
login_num	1-5	int	Maximum number of logins per user
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

Table-3.6.4 (Userpermissioninformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYP E	

log_search		bool	Log permission
parameter		bool	Parameter setting permission
auto_reboot		bool	AutoRebootpermission
manual_record		bool	ManualRecordpermission (A01 no)
disk		bool	Disk permission (A01、IPC no)
remote_login		bool	RemoteLoginpermission (IPC no)
seq_control		bool	SEQControlpermission (A01、IPC no)
manual_capture		bool	ManualCapturepermission (A01 no)
audio		bool	Audio permission (A01 no)
ip_camera_maintain		bool	Ip Camera Maintain permission(A01 only)
ipc_channel_manage		bool	Ipc Channel Manage permission(A01 only)
alarm_config		bool	Alarm Config permission(A01 only)
record_config		bool	Record Config permission(A01 only)
channel_config		bool	Channel Config permission(A01 only)
face_search		bool	Face search permission(A01 only)
backup_enable		bool	Backup permission: (NVR only)
live_enable		bool	Preview permission
playback_enable		bool	Video playback permission
ptz_enable		bool	Ptz Control permission
backup_channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x”	array	Backup Channel (NVR only) Each array bit represents a channel with a string.
live_channel	“WIFI_CH1”...” WIFI_CH1x”	array	Preview Channel Each array bit represents a channel with a string.
playback_channel	The number of channels depends on the capabilities of the device.	array	Video playback Channel Each array bit represents a channel with a string.
ptz_channel		array	Ptz Control Channel Each array bit represents a channel with a string.
rtsp_right		Bool	Rtsp permission (IPC only)
license_plate		bool	License Plate permission

Example:

Scenario 1: admin obtains user parameters

Request message:

POST/API/SystemConfig/User/Get HTTP/1.1

```
{
"version": "1.0",
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
```

```
"result":"success",
"data":{
  "user_info":{
    "ADMIN":{
      "user_enable":true,
      "username":"admin",
      "password_empty":false
    },
    "USER1":{
      "user_enable":true,
      "username":"user1",
      "password_empty":false
    },
    "USER2":{
      "user_enable":false,
      "username":"user2",
      "password_empty":true
    },
    "USER3":{
      "user_enable":false,
      "username":"user3",
      "password_empty":true
    },
    "USER4":{
      "user_enable":false,
      "username":"user4"
    },
    "USER5":{
      "user_enable":false,
      "username":"user5",
      "password_empty":true
    },
    "USER6":{
      "user_enable":true,
      "username":"liuyong",
      "password_empty":false
    }
  }
}
```

Scenario 2: admin turns on user 2

Request message:

POST/API/SystemConfig/User/Set HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "user_info": {
      "USER2": {
        "user_enable": true,
        "username": "user2",
        "password_empty": true,
        "password": "111111",
        "confirm_password": "111111"
      }
    },
    "type": "SavePassword",
    "secondary_authentication": "admin123.."
  }
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{"result":"success","data":{}}
```

Scenario 3: admin sets the permissions of user 2

Request message:

POST/API/SystemConfig/User/Set HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "user_info": {
      "USER2": {
        "username": "user2",
        "permission": {
          "log_search": true,
          "parameter": true,
          "auto_reboot": true,
          "manual_record": true,
          "disk": true,
          "remote_login": true,
          "seq_control": false,
          "manual_capture": false,
          "audio": true,
          "face_search": true,
          "backup_enable": true,
          "backup_channel": [
```

```

    ],
    "live_enable": true,
    "live_channel": [
        "CH1",
        "CH2",
        "CH3",
        "CH4",
        "CH5",
        "CH6",
        "CH7",
        "CH8"
    ],
    "playback_enable": false,
    "playback_channel": [

    ],
    "ptz_enable": true,
    "ptz_channel": [
        "CH1",
        "CH2",
        "CH3",
        "CH4",
        "CH5",
        "CH6",
        "CH7",
        "CH8"
    ]
    }
}
},
"secondary_authentication": "admin123..",
"type": "SavePermission"
}
}
}

```

Response message:

```

HTTP/1.1 200 OK
Content-Type: application/json
{"result":"success","data":{}}

```

3.7 System Information

GET

URL	POST/API/SystemInfo/Base/Get
Description	Get System Base Information, Including software version date, network, video format, hard disk capacity, etc
Request Body	None
Successful Response	System Information JSON(show as follow Table-3.7)

Table-3.7 (System Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
device_id		string	DeviceID (A01 no)
device_name		string	DeviceName
device_type		string	DeviceType
hardware_version		string	HardwareVersion
software_version		string	SoftwareVersion
build_time		string	Publication Time
ie_client_version		string	IE Client Version (A01 no)
video_format	“PAL” “NTSC”	string	Video Format
hdd_volume		string	HDD Volume
ip_address		string	IP Address
ipv6_address		string	IPv6 Address
web		string	Web Internal port ,Web external port
client		string	Client Internal port , Clientexternal port
mac_address		string	MAC Address
wireless_mac		string	Wireless MAC (Routing module address)
p2p_id		string	P2P ID (A01 no)
p2p_switch		bool	(A01no)
network_state	“Connected”, “Unconnected”	string	Network connection state
public_cloud_state	"Disabled","Connecte d","Unconnected"	string	Public cloud state (A01 only)
channel_num		int	Channel number (A01 only)
serialNum		string	Serial Number(B18)
language		string	Current language of the device

Please note that the following products will integrate the web and client ports. The client port will not be opened and this item will be removed from the page in the future. If the customer does development docking, please do not consider this port again

3.8 Channel Information

GET

URL	POST/API/SystemInfo/Channel/Get
Description	Get System Channel Information
Request Body	None
Successful Response	Channel Information JSON(show as follow Table-3.8.1)

Table-3.8.1 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		Json Object	Single Channel Information JSON show as follow Table-3.8.2
channel_max		int	Max channel

Table-3.8.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-3.8.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-3.8.3(Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel		string	Channel number
alias		string	IP Camera alias.
state	"Offline", "Online"	string	IP Camera connection status.
mainstream	"Resolution, FrameRate, Bitrate"	string	IP Camera's mainstream information.
substream		string	IP Camera's substream information.
mobile_stream		string	IP Camera's mobile stream information.
motion_detection	"Support", "Nonsupport"	string	Motion Detection.
privacy_zone	"Support", "Nonsupport"	string	Privacy Zone.

3.9 Record Information

GET	
URL	POST /API/SystemInfo/Record/Get
Description	Get System Channel Record Information
Request Body	None
Successful Response	Channel Information JSON(show as follow Table-3.9.1)

Table-3.9.1 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		Json Object	Single Channel Information JSON show as follow Table-3.9.2
channel_max		int	Max channel

Table-3.9.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-3.9.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-3.9.3(Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel		string	Channel Number
record_state	"On", "Off"	string	Record State
record_switch		bool	Record Switch
stream_type	"Mainstream", "Substream", " "DualStream "	string	Video stream type
resolution		string	Resolution: Format: "Main Stream Resolution Sub Stream Resolution"
fps		string	IP Camera's mainstream frame rate.(/Fps) Format:"Main Stream FPS Sub Stream FPS"
bitrate		string	IP Camera's mainstream bitrate rate.(/Kbps) Format:"Main Stream Bitrate Sub Stream Bitrate"

3.10 Network State

GET	
URL	POST/API/SystemInfo/Network/Get
Description	Get System Network State Information
Request Body	None

Successful Response	Network state Information JSON(show as follow Table-3.10.1)
---------------------	---

Table-3.10.1(Network state Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
wan		JSON object	Single JSON(show as follow Table-3.10.2)
lan		JSON object	Single JSON(show as follow Table-3.10.3)
pppoe	"Enable", "Disable"	string	PPPoE switch
3g	"Enable", "Disable"	string	3G network switch
wifi	"Enable", "Disable"	string	WIFI switch
port		JSON object	Single JSON(show as follow Table-3.10.4)
bandwidth		JSON object	Single JSON(show as follow Table-3.10.5)
toe	"Enable", "Disable"	string	TOE Acceleration sign
dns1		string	Preferred DNS server, for example: 8.8.8.8
dns2		string	Alternate DNS server, for example: 8.8.8.8
lan1		JSON Object	Single JSON(show as follow Table-3.10.6) (used in the new network card mode)
lan2		JSON Object	Single JSON(show as follow Table-3.10.7) (used in the new network card mode)
poe		JSON Object	Single JSON(show as follow Table-3.10.8) (used in the new network card mode)

Table-3.10.2 (Wan info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
dhcp	"Enable", "Disable"	string	DHCP swith
ip_address		string	IP address, such as:192.168.1.24
subnet_mask		string	Subnet mask, such as:255.255.255.0
gateway		string	Default gateway, such as:192.168.1.1
mac_address		string	MAC address, such as:88-60-50-4E-87-29
ipv6_address	Max length: 46 byte	string	IPv6 address
ipv6_gateway	Max length: 46 byte	string	IPv6 gateway

Table-3.10.3 (Lan info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ip_address		string	Switch IP address
subnet_mask		string	Switch subnet mask

Table-3.10.4 (Port info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
web	"Port, Export, ExportState, Upnp"	string	format:

client		string	"Internal port number, external port number, port UPnP function status, port UPnP function enabled" {ExportState:" Active"," Inactive"} {Upnp:" Enable"," Disable"}
rtsp		string	
https		string	

Table-3.10.5 (Bandwidth info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
total_bandwidth		string	Total bandwidth
used_bandwidth		string	Used bandwidth

Table-3.10.6 (Lan1 info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
dhcp	"Enable", "Disable"	string	IPv4 DHCP switch
ip_address		string	IP address, such as:192.168.1.24
subnet_mask		string	Subnet mask, such as:255.255.255.0
gateway		string	Default gateway, such as:192.168.1.1
dhcpv6	"Enable", "Disable"	string	IPv6 DHCP switch
ipv6_address		string	IPv6 address
ipv6_prefixlen		int	
ipv6_gateway		string	IPv6 gateway
mac_address		string	MAC address, such as:88-60-50-4E-87-29

Table-3.10.7 (Lan2 info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
dhcp	"Enable", "Disable"	string	IPv4 DHCP switch
ip_address		string	IP address, such as:192.168.1.24
subnet_mask		string	Subnet mask, such as:255.255.255.0
gateway		string	Default gateway, such as:192.168.1.1
dhcpv6	"Enable", "Disable"	string	IPv6 DHCP switch
ipv6_address		string	IPv6 address
ipv6_prefixlen		int	
ipv6_gateway		string	IPv6 gateway
mac_address		string	MAC address, such as:88-60-50-4E-87-29

Table-3.10.8 (Poe info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
poe_dhcp	"Enable", "Disable"	string	Poe DHCP switch
poe_ip_address		string	POE IP address, such as:192.168.1.24

poe_subnet_mask		string	POE subnet mask, such as:255.255.255.0
-----------------	--	--------	--

Please note that the following products will integrate the web and client ports. The client port will not be opened and this item will be removed from the page in the future. If the customer does development docking, please do not consider this port again

Example:

Request message:

POST /API/SystemInfo/Network/Get HTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

```
{
  "result": "success",
  "data": {
    "wan": {
      "dhcp": "Disable",
      "ip_address": "172.16.10.159",
      "subnet_mask": "255.255.0.0",
      "gateway": "172.16.8.1",
      "mac_address": "22-11-33-41-15-66",
      "ipv6_address": "2000:0:0:0:0:0:1 / 128",
      "ipv6_gateway": "2000:0:0:0:0:0:0:1",
      "dns1": "172.18.1.222",
      "dns2": "8.8.8.8"
    },
    "port": {
      "web": "80,80,Inactive,Disable",
      "client": "9000,9000,Inactive,Disable",
      "https": "443,443,Inactive,Disable"
    },
    "bandwidth": {
      "total_bandwidth": "172032Kbps",
      "used_bandwidth": "41856Kbps"
    }
  }
}
```

4 NETWORK

4.1 Network Configuration

4.1.1 Network Base Information

GET	
URL	POST/API/NetworkConfig/NetBase/Get
Description	It is used to get the device network interfaces
Request Body	none
Successful Response	Network base info JSON(show as follow Table-4.1.1.1)

SET	
URL	POST/API/NetworkConfig/NetBase/Set
Description	It is used to set the device network interfaces
Request Body	Network base info JSON(show as follow Table-4.1.1.1)
Successful Response	The successful result response that described in 2.5

Table-4.1.1.1 (Network base info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
wan		JSON object	Single JSON(show as follow Table-4.1.1.2)
lan		JSON object	Single JSON(show as follow Table-4.1.1.3) (NVR only)
pppoe		JSON object	Single JSON(show as follow Table-4.1.1.4)
3g		JSON object	Single JSON(show as follow Table-4.1.1.5) (NVR only)
wifi		JSON object	Single JSON(show as follow Table-4.1.1.6)
port		JSON array	Single JSON(show as follow Table-4.1.1.7)
extern_ip		string	Internet IP, valid when the Internet is enabled (Get only)
p2p_switch		bool	P2P switch
ipeye_switch		bool	IPEYE switch (DVR、NVR only)
device_instruction	“nvr” “xvr” “dvr” “ipc”	string	Device type description mark
web_compatibility_mode		bool	Web compatibility mode
toe		bool	Switch mode (NVR only)
video_encrypt_transfer	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on	array	Video Encrypt TransferChannel Each array bit represents a channel with a string.

	the capabilities of the device.		
lan1		JSON Object	Single JSON(show as follow Table-4.1.1.8) (used in the new network card mode)
lan2		JSON Object	Single JSON(show as follow Table-4.1.1.9) (used in the new network card mode)
poe		JSON Object	Single JSON(show as follow Table-4.1.1.10) (used in the new network card mode)
net_card_mode	"Single Address Mode" ,"Double Address Mode"	string array	Net card mode
default_route	"WAN" "LAN1", "LAN2"	string array	Default route (used in the new network card mode)
net_car_select	"WAN" "LAN1", "LAN2"	string array	Net card selection (used in the new network card mode)
pppoe_net_card	"LAN1", "LAN2"	string array	PPPOE network card selection (used in new network card mode)
dns1		string	Preferred DNS server, for example: 8.8.8.8 (dedicated for new network card mode)
dns2		string	Standby DNS server, for example: 8.8.8.8 (dedicated for new network card mode)

Table-4.1.1.2 (Wan info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
dhcp		bool	DHCP switch
ip_address		string	IP address, such as:192.168.1.24
subnet_mask		string	Subnet mask, such as:255.255.255.0
gateway		string	Default gateway, such as:192.168.1.1
ipv6_address		string	IPv6 address
ipv6_prefixlen		int	
ipv6_gateway		string	IPv6 gateway
dns1		string	Preferred DNS server, for example: 8.8.8.8
dns2		string	Alternate DNS server, for example: 8.8.8.8
main_multicast_enable		bool	Main stream multicast switch (IPC dedicated)
main_multicast_address	(224.0.0.0-239.255.255.255)	string	Main stream multicast switch (IPC dedicated)
main_multicast_port	1024-65535	int	Main stream multicast switch (IPC dedicated)
sub_multicast_enable		bool	Sub-stream multicast switch (IPC dedicated)
sub_multicast_addr	(224.0.0.0-	string	Sub-stream multicast switch (IPC dedicated)

ess	239.255.255.255)		
dhcp_enable		bool	Dhcp enable switch, DHCP switch is not enabled after PPPOE is turned on (NVR only)
rtsp_url		string	The url format of the rtsp server (for NVR and DVR only)
encryption		bool	Video encryption

Table-4.1.1.3 (Lan info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ip_address		string	Switch IP address
subnet_mask		string	Switch subnet mask

Table-4.1.1.4 (PPPoE info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
enable		bool	PPPoE switch
username	Max length: 35byte	string	PPPoE user name
password	Max length: 35byte	string	PPPoE password
password_empty		bool	Password empty
ip_address		string	IP address, such as:192.168.1.24
subnet_mask		string	Subnet mask, such as:255.255.255.0
gateway		string	Default gateway, such as:192.168.1.1
dns1		string	Preferred DNS server, for example: 8.8.8.8
dns2		string	Alternate DNS server, for example: 8.8.8.8
pppoenewstruct		bool	Whether to use two sets of range control
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

Table-4.1.1.5 (3G info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
enable		bool	3G network switch
apn	Max length: 35 byte	string	3G apn
dial_code	Max length: 35 byte	string	3G DialCode
username	Max length: 35 byte	string	3G dial-up user name
password	Max length: 35 byte	string	3G dial-up password
ip_address		string	IP address, such as:192.168.1.24
subnet_mask		string	Subnet mask, such as:255.255.255.0
gateway		string	Default gateway, such as:192.168.1.1
dns1		string	Preferred DNS server, for example: 8.8.8.8
dns2		string	Alternate DNS server, for example: 8.8.8.8
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

Table-4.1.1.6 (WIFI info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
enable		bool	WIFI switch
ssid	Max length: 35 byte	string	WIFISSID
password	Max length: 35 byte	string	WIFI password
ip_address		string	IP address obtained by connecting to WIFI (Get only)
status	"Unconnected", "Connected"	string	WIFI connection status (Get only)
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

Table-4.1.1.7 (Portinfo JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
service	"Web", "Client", "Rtsp", "Https",	string	Port type (A01 no)
protocol	"TCP"	string	Protocol type (Get only)
internal_port	[1~65535]	int	Internal port
external_port	[1~65535]	int	External port
upnp_status	"Inactive", "Active"	string	Port UPnP function status (Get only)
upnp		bool	Port enables UPnP function
mapping_strategy	"Manual""Auto"	string	Mapping method

Table-4.1.1.8 (Lan1 JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
dhcp		bool	IPv4 DHCP switch
ip_address		string	IP address, such as:192.168.1.24
subnet_mask		string	Subnet mask, such as:255.255.255.0
gateway		string	Default gateway, such as:192.168.1.1
dhcpv6		bool	IPv6 DHCP switch
ipv6_address		string	IPv6 address
ipv6_prefixlen		int	
ipv6_gateway		string	IPv6 gateway
dhcp_enable		bool	Dhcp Enable switch, DHCP switch is not enabled after PPPOE is turned on (NVR only)
mac_address		string	MAC address, such as:88-60-50-4E-87-29

Table-4.1.1.9 (Lan2 JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
dhcp		bool	IPv4 DHCP switch
ip_address		string	IP address, such as:192.168.1.24
subnet_mask		string	Subnet mask, such as:255.255.255.0
gateway		string	Default gateway, such as:192.168.1.1
dhcpv6		bool	IPv6 DHCP switch
ipv6_address		string	IPv6 address
ipv6_prefixlen		int	
ipv6_gateway		string	IPv6 gateway
mac_address		string	MAC address, such as:88-60-50-4E-87-29

Table-4.1.1.10 (Poe JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
poe_dhcp		bool	Poe DHCP switch
poe_ip_address		string	POE IP address, such as:192.168.1.24
poe_subnet_mask		string	POE subnet mask, such as:255.255.255.0

Example:

Request message:

POST /API/NetworkConfig/NetBase/Get HTTP/1.1

```
{
"version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

```
{
"version": "1.0",
"result": "success",
"data": {
"wan": {
"dhcp": false,
"ip_address": "192.168.1.108",
"subnet_mask": "255.255.000.000",
"gateway": "172.018.000.001",
"dns1": "192.168.000.001",
"dns2": "008.008.008.008",
"dhcp_enable": false
},
"lan": {
"ip_address": "010.010.025.100",
"subnet_mask": "255.255.000.000"
},
}
```

```
"pppoe": {
  "enable": false,
  "user": "",
  "password": ""
},
"3g": {
  "enable ": false,
  "apn": "3gnet",
  "dial_code": "*99#",
  "user": "card",
  "password": "card"
},
"wifi": {
  "enable": false,
  "ssid": "",
  "password": ""
},
"port": [
  {
    "service": "Web",
    "protocol": "TCP",
    "internal_port": 80,
    "external_port": 80,
    "upnp_status": "Inactive",
    "upnp": false
  },
  {
    "service": "Client",
    "protocol": "TCP",
    "internal_port": 9000,
    "external_port": 9000,
    "upnp_status": "Inactive",
    "upnp": false
  },
  {
    "service": "Rtsp",
    "protocol": "TCP",
    "internal_port": 554,
    "external_port": 554,
    "upnp_status": "Inactive",
    "upnp": false
  },
  {
    "service": "Https",
```

```

"protocol": "TCP",
"internal_port": 443,
"external_port": 443,
"upnp_status": "Inactive",
"upnp": false
    }
],
"p2p_switch": true,
"toe": false
}
}

```

4.1.2 WLANScan

GET	
URL	POST/API/NetworkConfig/ScanWlan/Scan
Description	It is used to get the Wifilist.
Request Body	none
Successful Response	Wlan Scan JSON(show as follow Table-4.1.2.1)

SET	
URL	POST/API/NetworkConfig/ScanWlan/Join
Description	It is used to addwifi.
Request Body	WIFI JSON(show as follow Table-4.1.2.2)
Successful Response	The successful result response that described in 2.5

SET	
URL	POST/API/APNetworkCfg/WifiStaParam/Set
Description	Wifi AP Network configuration
Request Body	STA Parameter info JSON(show as follow Table-4.1.2.3)
Successful Response	The successful result response that described in 2.5 Note: This API interface can only be registered and enabled in AP mode

Table-4.1.2.1(Wlan ScanJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
wifi_info		JSON array	JSON(show as follow Table-4.1.2.2)

Table-4.1.2.2(WIFIJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

ssid	Max length: 35 byte	string	WIFI SSID
signal_strength	0 - 4	int	Wifi Signal Strength Note: only use to ScanWlan
security	Max length: 128 byte	string	WIFI encryption type Note: only use to ScanWlan
password	Max length: 35 byte	string	WIFI password Note: only use to JoinWlan
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

Table-4.1.2.3(STA Parameter info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ssid		string	Route ssid
base_enc_password		Json Object	Route wifi password Encrypted Password, Table4.1.2.4
encryptionType		string	NONE/WEP/WPA-PSK/WPA2-PSK/WPA-NONE
cloudhost		string	
cloudport		int	

Table-4.1.2.4(STA Parameter info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
seq		int	
peer_key		string	
cipher		string	

Example:

Request message:

POST/API/NetworkConfig/ScanWlan/Scan HTTP/1.1

```
{
"version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
"result": "success",
"data": {
"wifi_info": [
{
"ssid": "xxxxxxxxxxxxxx",
"signal_strength": 4,

```

```

        "security": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
        },
        .....
    ]
}
}

```

Request message:

POST/API/NetworkConfig/ScanWlan/Join HTTP/1.1

```

{
  "version": "1.0",
  "data": {
    "ssid": "xxxxxxxxxxxx",
    "password": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
  }
}

```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 250

Connection: keep-alive

```

{
  "version": "1.0",
  "result": "success"
}

```

4.2 DDNS

4.2.1 DDNS Parameters

GET	
URL	POST/API/NetworkConfig/DDNS/Get
Description	It is used to get the DDNS parameters
Request Body	none
Successful Response	DDNS info JSON(show as follow Table-4.2.1)

SET	
URL	POST/API/NetworkConfig/DDNS/Set
Description	It is used to set the DDNS parameters

Request Body	DDNS info JSON(show as follow Table-4.2.1)
Successful Response	The successful result response that described in 2.5

SET	
URL	POST/API/NetworkConfig/DDNS/Test
Description	It is used to set the DDNS parameters
Request Body	DDNS info JSON(show as follow Table-4.2.1)
Successful Response	The successful result response that described in 2.5 and 2.9

Table-4.2.1(DDNS info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ddns_enable		bool	Host name of this device
server	"DDNS_3322", "DYNDNS", "NO_IP", "CHANGEIP", "DNSEXIT"	string	DDNS server IP address or name.
domain	Max length: 35 byte	string	Host name of this device
domain_suffix		string	Domain suffix
username	Max length: 35 byte	string	DDNS user name
password	Max length: 35 byte	string	DDNS user password
password_empty		bool	Whether the password is empty
service_id		string	B03 only
test_default_save		bool	Whether to test before saving, the default is FALSE
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

Example:

Request message:

POST/API/NetworkConfig/DDNS/Get HTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 123

Connection: keep-alive

```
{  
"version": "1.0",  
"result": "success",  
"data": {  
  
    "ddns_enable": false,  
    "server": "NO_IP",  
    "domain": "",  
    "user": "",  
    "password": ""  
    }  
}
```

4.3 Email

4.3.1 Email Parameters

GET	
URL	POST/API/NetworkConfig/Email/Get
Description	It is used to get the Email parameters
Request Body	none
Successful Response	Emailinfo JSON(show as follow Table-4.3.1.1)

SET	
URL	POST/API/NetworkConfig/Email/Set
Description	It is used to set the Email parameters
Request Body	Emailinfo JSON(show as follow Table-4.3.1 .1)
Successful Response	The successful result response that described in 2.5

Table-4.3.1.1(Emailinfo JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
email_enable		bool	Enable email
encryption	"Disable", "SSL", "TLS", "Auto"	string	Encryption type
smtp_port	[1~65535]	int	SMTP port
smtp_server	Max length: 127byte	string	SMTP server

username			User name for mail
password			Mail password
sender			Sender's address
recvemail			recvemail JSON(show as follow Table-4.3.1 .2)
password_empty		bool	Password if empty
interval_time	1,3,5,10	int	Sending interval time. Unit: minute
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

Table-4.3.1.2(recvemail JSON)

recvemail_1	Max length: 127 byte	String	sender1 address
recvemail_2	Max length: 127 byte	String	sender2 address
recvemail_3	Max length: 127 byte	string	Sender3 address

Example:

Request message:

POST /API/NetworkConfig/Email/Get HTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 165

Connection: keep-alive

```
{
  "version": "1.0",
  "result": "success",
  "data": {
    "email_enable": false,
    "encryption": "AUTO",
    "smtp_port": 25,
    "smtp_server": "",
    "username": "",
    "password": "",
    "sender": "",
    "recvemail_1": "",
    "recvemail_2": "",
    "recvemail_3": "",
    "interval_time": 3
  }
}
```


4.3.2 Email Test

GET	
URL	POST/API/NetworkConfig/Email/Test
Description	It is used to test email
Request Body	(show as follow Table-4.3.2.1)
Successful Response	Emailinfo JSON(show as follow Table-4.3.2.2)

Table-4.3.2.1(Emailinfo JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
email_enable		bool	Mail enable
encryption	"Disable", "SSL", "TLS", "AUTO"	string	Encryption type
smtp_port	[1~65535]	int	SMTP port
smtp_server	Max length: 127 byte	string	SMTP server
username			Mail user name
password			Mail password
password_empty		bool	Password if empty
sender			Sender address
recvemail			JSON(show as follow Table-4.3.2.3)
interval_time	1,3,5,10	int	Sending interval time. Unit: minute
test_id	1-65535	int	test id
email_test_flag	"Start"、"Stop"、 "Query"	string	Mail test instructions (start test, stop test, get test results)
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

Table-4.3.2.2(Emailinfo JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
test_state	"Testing" (testing) "Ok" (test finish)	string	Mail test status (if test fails, an error will be returned directly)

Table-4.3.2.3(Emailinfo JSON)

recvemail_1			Email addresses of 3 receivers
recvemail_2			
recvemail_3			

Example:

Request message:

POST /API/NetworkConfig/Email/Test HTTP/1.1

```
{
  "data": {
    "email_enable": false,
    "encryption": "AUTO",
    "smtp_port": 25,
    "smtp_server": "smtp163.com",
    "username": "123456@qq.com ",
    "password": "321",
    "sender": "123456@qq.com",
    "recvemail_1": "654321@qq.com ",
    "recvemail_2": "",
    "recvemail_3": "",
    "interval_time": 3,
    "test_id": 3,
    "email_test_Flag": "start"
  }
}
```

Response message:

```
{
  "result": "success",
  "data": {}
}
```

4.4 FTP

4.4.1 FTP Parameters

GET	
URL	POST/API/NetworkConfig/Ftp/Get
Description	It is used to get the FTP parameters
Request Body	none
Successful Response	FTPinfo JSON(show as follow Table-4.4.1)

SET	
URL	POST/API/NetworkConfig/Ftp/Set
Description	It is used to set the FTP parameters
Request Body	FTP info JSON(show as follow Table-4.4.1)
Successful Response	The successful result response that described in 2.5

TEST	
URL	POST/API/NetworkConfig/Ftp/Test
Description	It is used to Test the FTP parameters
Request Body	FTP info JSON(show as follow Table-4.4.1)
Successful Response	The successful result response that described in 2.5

Table-4.4.1(FTPInfo JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ftp_enable		bool	FTP switch
ftp_test		bool	If support FTP test
server_ip	Max length: 64 byte Note: DVR maximum input characters 15	string	FTP IP address
port	[1~65535]	int	FTP Server Port
username	Max length: 64 byte Note: DVR maximum input characters 15	string	Login user name
password	Max length: 64 byte Note: DVR maximum input characters 15	string	Login user password
password_empty		bool	Password if empty
upgrade_picture		bool	Upgrade picture switch
picture_resolution	"1920x1080", "1280x720", "1024x768", "640x480", "320x240", "176x144" or "176x120"	string	Picture resolution (for NVR only)
picture_quality	"Highest", "Higher", "Medium", "Low", "Lower", "Lowest"	string	Picture quality (for NVR only)
video_stream_type	"Mainstream","Substream"	string	Video stream type
video_type	"RF","AVI","MP4"	string	Video file type
max_package_interval	10,20,30,45,60	int	Maximum subcontracting interval . Unit: minute
directory_name	Max length: 95byte	string	Upload path

upload_normal_video		JSON array	Upload video information JSON(show as follow Table-4.4.2)
enc_mode	"FTP", "SFTP", "FTPS",	string	FTP type
certpem		string	certificate
certpwd		string	secret key
privatekey		string	Private certificate
privatekeypwd		string	Private secret key
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

Table-4.4.2(Upload video information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
enable		bool	Upload conventional video switch
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... "WIFI_CH1x" The number of channels depends on the capabilities of the device.	array	Channel number of regular video to be uploaded Each array bit represents a channel with a string.

Example:

Request message:

POST /API/NetworkConfig/Ftp/GetHTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 165

Connection: keep-alive

```
{
  "version": "1.0",
  "result": "success",
  "data": {
```

```

"ftp_enable": false,
"server_ip": "192.168.1.100",
"port": 21,
"username": "",
"password_empty": "",
"picture_resolution": "1280x720",
"picture_quality": "Higher",
"video_stream_type": "SubStream",
"max_package_interval": 30,
"directory_name": "",
"upload_normal_video": {
  "enable": true,
  "channel": [
    "IP_CH2",
    "IP_CH3",
    "IP_CH4"
  ]
}
}
}

```

4.5 IP Filter Parameters

GET	
URL	POST/API/NetworkConfig/IPFilter/Get
Description	It is used to get the IP Filter Parameters
Request Body	none
Successful Response	IP FilterJSON(show as follow Table-4.5.1)

SET	
URL	POST/API/NetworkConfig/IPFilter/Set
Description	It is used to set the IP Filter Parameters
Request Body	IP Filter JSON(show as follow Table-4.5.1)
Successful Response	The successful result response that described in 2.5

Table-4.5.1(IP Filter JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
enable		bool	Control the IP Filter function .
choose	"Whitelist","Blacklist"	string	Enable restriction mode
restricted_type	"Whitelist","Blacklist"	string	Type of black-and-white list
whitelist		JSON array	Single JSON(show as follow Table-4.5.2)

blacklist		JSON array	Single JSON(show as follow Table-4.5.2)
-----------	--	------------	--

Table-4.5.2(Single JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
start_address	Max length: 16byte、64byte(NVR/DVR usage)	string	Start filtering IP address
end_address	Max length: 16byte、64byte(NVR/DVR usage)	string	End filtering IP address
ip_type	"Ipv4","Ipv6"	string	IP type (IPC only supports: "Ipv4")

Table-4.5.3(Error Code)

error_code	commet
ip_filter_list_empty	The blacklist/whitelist list is empty

Example:

Request message:

POST /API/NetworkConfig/IPFilter/GetHTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 250

Connection: keep-alive

```
{
"version": "1.0",
"result": "success",
"data": {
  "enable": true,
  "choose": "Whitelist",
  "restricted_type": "Blacklist",
  "whitelist": [
    {
      "start_address": "192.168.1.100",
      "end_address": "192.168.1.100",
      "ip_type": "Ipv4"
    }
  ]
}
```

```

    ],
    "blacklist": [
        {
            "start_address": "192.168.1.106",
            "end_address": "192.168.1.106",
            "ip_type": "Ipv4"
        }
    ]
}
}

```

4.6 Https Configure

GET	
URL	POST/API/NetworkConfig/https/Get
Description	It is used to get thehttpsParameters
Request Body	none
Successful Response	JSON(show as follow Table-4.6.1)

SET	
URL	POST/API/NetworkConfig/https/Set
Description	It is used to set the https Parameters
Request Body	JSON(show as follow Table-4.6.1)
Successful Response	The successful result response that described in 2.5

Table-4.6.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
https_enable	true false	bool	Don't use
operate	"Install""Uninstall" "Switch"		
file_type	"Default", "Custom"	string	Default: default install Custom: custom install Transfer on switching
file_exist		int	0: does not exist, 1: exists (only get used, set not, Custom required)
ca_file	0-10240	string	(only set used, get not)
key_file	0-10240	string	(only set used, get not)
key_password	0-128	string	(only set used, get not)
root_ca_file	0-10240	string	(only set used, get not)
subject		string	Issued to (required by Custom)
issuer		string	Issued to (required by Custom)

not_before		string	Start time (required by Custom)
not_after		string	End time (required by Custom)

error_code

install_failed	Install fail
uninstall_failed	Uninstall fail
uninstall_not_allow	In use, uninstallation is not allowed
switch_failed	Switch fail
upload_failed	Upload fail
upload_success	Upload success
length_too_long	Length too long
unsafe_siganature	Unsafe signature
cert_key_not_match	certificate key not match
invalid_cert_time	Invalid certificate time
invalid_private_key	Invalid private key
invalid_cert	Invalid certificate
invalid_key_usage	Invalid key usage
invalid_cert_chain	Invalid certificate chain
invalid_cacert	Invalid root certificate
invalid_cacert_time	Contains an invalid root certificate

4.7 Voice Assistant

GET	
URL	POST/API/NetworkConfig/VoiceAssistant/Get (API is not implemented yet)
Description	It is used to get theVoiceAssistantParameters
Request Body	JSON(show as follow Table-4.7.1)
Successful Response	JSON(show as follow Table-4.7.2)

Control	
URL	POST/API/NetworkConfig/VoiceAssistant/Control ((API is not implemented yet)
Description	It is used to control voice assistant
Request Body	JSON(show as follow Table-4.7.3)
Successful Response	The successful result response that described in 4.7.4

Table-4.7.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
type	“Amazon”“Google”	string	

Table-4.7.2

KEY	VALUE		COMMENT
	RANGE	TYPE	
user		string	User name
state	“Activated”“Unactivated” "Unconnected"	string	Active state
screen_stream	“Mainstream”, “Substream”	string	Screen cast code stream

Table-4.7.3

KEY	VALUE		COMMENT
	RANGE	TYPE	
type	“Amazon”“Google”	string	
command	"Bind""Unbind" "Apply"	string	Active state
user		string	
screen_stream	“Mainstream”, “Substream”	string	Screen cast code stream

Table-4.7.4

KEY	VALUE		COMMENT
	RANGE	TYPE	
state		int	0: fail 1: success 2: other error, try it later -1: Network failure
user		string	
screen_stream	“Mainstream”, “Substream”	string	Screen cast code stream

4.8 ipv6

GET	
URL	POST/API/NetworkConfig/Ipv6/Get (DVR/NVR don't support)
Description	It is used to get the ipv6 Parameters
Request Body	none
Successful Response	JSON(show as follow Table-4.8.1)

Control	
URL	POST/API/NetworkConfig/ipv6/Set (DVR/NVR don't support)
Description	It is used to set the ipv6 parameters
Request Body	JSON(show as follow Table-4.8.1)
Successful Response	The successful result response that described in 2.5

Table-4.8.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
prefixlen		int	
local_ipv6_addr	Max length: 40byte		
global_ipv6_addr	Max length: 40byte		

4.9 snmp

GET	
URL	POST/API/NetworkConfig/Snmp/Get
Description	It is used to get thesnmpParameters
Request Body	none
Successful Response	JSON(show as follow Table-4.9.1)

Control	
URL	POST/API/NetworkConfig/snmp/Set
Description	It is used to setthe snmp parameters
Request Body	JSON(show as follow Table-4.9.1)
Successful Response	The successful result response that described in 2.5

Table-4.9.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
snmp_port		int	
snmp_enable		bool	false:disable true:enable
trap_port		int	
snmp_versions	"V1", "V2", "V1,V2" "V3"	String	0:snmp v1 1:snmp v2 2:snmp v1 v2 3:snmp v3
trap_ipaddr	Max length: 32byte	string	Management server address
read_community	Max length: 16byte	string	Set the IPC read community name, server to use the common name and only read the snmp information on the IPC
write_community	Max length: 16byte	string	Set the IPC read-write community name, server to use the community name to read and write the snmp information on the IPC
authentication		object json	Shown as 4.9.2
base_enc_authentication_password		Json Object	Encrypted Password, Table-12.7.2.3

base_encrypted_password		Json Object	Encrypted Password, Table-12.7.2.3
-------------------------	--	-------------	------------------------------------

Table-4.9.2

readonly_user		object	Set to read user Shown as 4.9.3
readwrite_user		object	Set to read and write user Shown as 4.9.3

Table-4.9.3

KEY	VALUE		COMMENT
	RANGE	TYPE	
authentication_type	“MD5”, “SHA”	string	MD5 authentication mode SHA authentication mode
encrypted_type	“CBC-DES”	string	For encryption algorithm expansion
username	Max length: 16byte	string	
authentication_password	Max length: 16byte	string	
authentication_password_empty		bool	Password if empty
encrypted_password	Max length: 16byte	string	
encrypted_password_empty		bool	Password if empty

4.10 Rtsp

GET	
URL	POST/API/NetworkConfig/Rtsp/Get (DVR/NVR don't suport)
Description	It is used to get the rtsp Parameters
Request Body	none
Successful Response	JSON(show as follow Table-4.10.1)

Control	
URL	POST/API/NetworkConfig/Rtsp/Set (DVR/NVR don't support)
Description	It is used to setthe rtsp parameters
Request Body	JSON(show as follow Table-4.10.1)
Successful Response	The successful result response that described in 2.5

Table-4.10.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
rtsp_enable		bool	
rtsp_check_flag		bool	Checkflag
anonymous_login		bool	No username or password required
ipeye_enable		bool	Ipeye enable

4.11 Onvif

GET	
URL	POST/API/NetworkConfig/Onvif/Get
Description	It is used to get the Onvif Parameters
Request Body	none
Successful Response	IP FilterJSON(show as follow Table-4.11.1)

SET	
URL	POST/API/NetworkConfig/ Onvif /Set
Description	It is used to set the Onvif Parameters
Request Body	IP Filter JSON(show as follow Table-4.11.1)
Successful Response	The successful result response that described in 2.5

Table-4.11.1(Onvif JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
enable		bool	Enable
authentication	"Digest_sha256", " Digest ", " Digest /WSSE", "WSSE", "None"	string	Encryption method
protocol	"HTTP/HTTPS", "HTTPS", "HTTP "	string	protocol
username		string	User name
password		string	password
password_empty		bool	Password if empty
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

4.12 Tuya

GET	
URL	POST/API/NetworkConfig/Tuya/Get
Description	It is used to get the Tuya Parameters
Request Body	none
Successful Response	JSON(show as follow Table-4.12.1)

SET	
-----	--

URL	POST/API/NetworkConfig/Tuya/Set
Description	It is used to set the Tuya Parameters
Request Body	JSON(show as follow Table-4.12.1)
Successful Response	The successful result response that described in 2.5

Table-4.12.1(Onvif JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
enable		bool	enable

5 CHANNEL

5.1 Channel Configuration

This item includes broadcast, digital channel configuration, analog channel configuration, protocol management configuration, multiple APIs, and wireless channel configuration.

The digital channel configuration includes multiple scenarios such as automatically adding ipc, manually adding, deleting, and modifying ipc.

5.1.1 Broadcast ipc

GET	
URL	POST/API/ChannelConfig/RemoteDev/Search
Description	Search ipc information(Broadcast search ipc)
Request Body	None
Successful Response	Device Information JSON (show as follow Table-5.1.2)

GET	
URL	POST/API/ChannelConfig/RemoteDev/Set
Description	Set remote ipc parameters (broadcast to modify the ip of ipc)
Request Body	Channel Request JSON(show as follow Table-5.1.2)
Successful Response	The successful result response that described in 2.5

5.1.2 Digital channel

GET	
URL	POST/API/ChannelConfig/IPChannel/Get
Description	Get the ipc channel parameters
Request Body	ChannelRequest JSON(show as follow Table-5.1.1)
Successful Response	Channel Information JSON (show as follow Table-5.1.3)

SET	
URL	POST/API/ChannelConfig/IPChannel/Set
Description	Set the ipc channel parameters(including adding, deleting and modifying ipc)
Request Body	Channel Information JSON(show as follow Table-5.1.3)
Successful Response	show as follow Table-5.1.3.2

SET	
URL	POST /API/ChannelConfig/AutoAddIPC/Set
Description	Automatically add IPC on board
Request Body	None
Successful Response	The successful result response that described in 2.5

Table-5.1.1 (Channel Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“IP_CH1”...” IP_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-5.1.2 (DeviceInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
device_info		JSON array	Single Channel Information JSON show as follow Table-5.1.4

Table-5.1.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
operation_type	"AddOrEditChannel", "EditIPCParam", "EditIPCpwd"	string	Operation type: add or modify channel parameters, edit IPC parameters of connection, and edit IPC password of connection
remove_ipc	“IP_CH1”...” IP_CH1x”	array	Each array bit represents a channel with a string.

	The number of channels depends on the capabilities of the device.		
channel_info		JSON object	Channel InformationJSON show as follow Table-5.1.3.1

Table-5.1.3.1 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-5.1.4
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-5.1.4 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_num		int	Number of channels supported by each device
state	"NotConfigured", "Offline", "Online"	string	Prompt when the channel is not configured
switch_mode	"ManualMode", "AutoMode"	string	Only POE channels have this variable. " POE channel mode, "ManualMode": manual mode "AutoMode": automatic mode
ip_address	Max length: 63byte	string	IP address/domain name
subnet_mask	Max length: 15byte	string	Subnet mask
gateway	Max length: 15byte	string	gateway
dns1	Max length: 15byte	string	Preferred DNS server, for example: 8.8.8.8
dns2	Max length: 15byte	string	Alternate DNS server, for example: 8.8.8.8
port	[1~65535]	int	Media port
web_port	[1~65535]	int	
protocol	Max length: 15byte	string	IPC access protocol
username	Max length: 31byte	string	User name
password	Max length: 31byte	string	password
password_empty		bool	Password if empty
camera_mode	"Auto", "Normal", "Fisheye"	string	Camera mode

manufacturer	Max length: 35byte	string	maufacturer
device_type	Max length: 35byte	string	Device type
device_type_flag	Max length: 32byte	string	The manufacturer's flag indicates that the third-party IPC can broadcast the protocol search and modify the IP, but the ONVIF protocol is still required to go online (for NVR only)
mac_address	Max length: 35byte	string	MAC address
software_version	Max length: 40byte	string	Firmware version
version_flag	[0~255]	int	Version flag
security	"Risk", "Weak", "Medium", "Strength"	string	Password security
can_modify_pwd		bool	Whether password modification is supported
new_password	Max length: 31byte	string	Change new password
modify_all_chn_pwd		bool	Whether to modify all channel passwords
network_mode	"Dhcp", "Static"	string	Network mode
can_set_netmode		bool	Can I choose network mode
main_url	Max length: 120byte	string	Main stream URL
sub_url	Max length: 120byte	string	Sub stream URL
connect_method	"General", "Security"	string	When it is the ONVIF protocol, it is used to indicate whether to use the normal mode or the security mode to connect the front-end device
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

Table-5.1.3.2 (protocolRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
modify_ipc_pwd_success_num		int	Number of successful IPC password modification (A01)

Example:

Scene 1: Broadcast search ipc

Request message:

POST /API/ChannelConfig/RemoteDev/Search?2020-10-28@13:28:23 HTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

```
{
  "result": "success",
  "data": {
```



```

"device_info": [
  {
    "network_mode": "Dhcp",
    "ip_address": "3.1.141.230",
    "subnet_mask": "000.000.000.000",
    "gateway": "000.000.000.000",
    "dns1": "000.000.000.000",
    "dns2": "000.000.000.000",
    "port": 80,
    "channel_num": 1,
    "protocol": "Onvif",
    "manufacturer": "HIKVISION DS-2DC2D20FW-D3",
    "device_type": "",
    "device_type_flag": "0",
    "mac_address": "94-E1-AC-DA-E3-72",
    "software_version": "",
    "version_flag": 0
  },
  {
    "network_mode": "Dhcp",
    "ip_address": "172.16.10.3",
    "subnet_mask": "255.255.000.000",
    "gateway": "172.016.008.001",
    "dns1": "172.018.001.222",
    "dns2": "008.008.008.008",
    "port": 9000,
    "web_port": 80,
    "channel_num": 32,
    "protocol": "Private",
    "manufacturer": "",
    "device_type": "N5332N",
    "device_type_flag": "5932088458482681345",
    "mac_address": "88-2E-42-90-41-18",
    "software_version": "V8.1.0-20201015",
    "version_flag": 1
  },
  {
    "network_mode": "Dhcp",
    "ip_address": "172.16.10.4",
    "subnet_mask": "255.255.252.000",
    "gateway": "172.016.008.001",
    "dns1": "172.018.001.222",
    "dns2": "008.008.008.008",
    "port": 9000,

```

```

        "web_port": 80,
        "channel_num": 8,
        "protocol": "Private",
        "manufacturer": "",
        "device_type": "N5208EN",
        "device_type_flag": "5932088458482680320",
        "mac_address": "4A-A4-43-BD-26-6D",
        "software_version": "V8.1.0-20201018",
        "version_flag": 1
    },
    {
        "network_mode": "Dhcp",
        "ip_address": "172.16.10.6",
        "subnet_mask": "255.255.252.000",
        "gateway": "172.016.008.001",
        "dns1": "172.018.001.222",
        "dns2": "008.008.008.008",
        "port": 9000,
        "web_port": 80,
        "channel_num": 20,
        "protocol": "Private",
        "manufacturer": "",
        "device_type": "#Y-DVR",
        "device_type_flag": "5932091774199202304",
        "mac_address": "C2-E5-88-F9-49-10",
        "software_version": "V8.1.0-20201010",
        "version_flag": 1
    }
]
}
}

```

Scenario 2: broadcast and modify the ip of ipc

Request message:

POST /API/ChannelConfig/RemoteDev/Set?2020-10-28@13:30:33 HTTP/1.1

```

{
  "version": "1.0",
  "data": {
    "device_info": [
      {
        "network_mode": "Static",
        "ip_address": "172.16.10.25",
        "subnet_mask": "255.255.252.000",
        "gateway": "172.16.8.1",

```

```

        "dns1": "255.255.255.255",
        "dns2": "255.255.255.255",
        "port": 80,
        "channel_num": 0,
        "protocol": "Private",
        "manufacturer": "",
        "device_type": "",
        "device_type_flag": "0",
        "mac_address": "88:22:29:8C:45:70",
        "software_version": "8.1.0.B036",
        "version_flag": 1,
        "web_port": 80,
        "old_ip_address": "172.16.10.24",
        "username": "11",
        "password": "111"
    }
}
]
}
}

```

Response message:

HTTP/1.1 200 OK

```

{
  "version": "1.0",
  "result": "success",
  "data": {
  }
}

```

Scene3: add ipc

Request message:

POST /API/ChannelConfig/IPChannel/Set?2020-10-28@14:09:37 HTTP/1.1

```

{
  "version": "1.0",
  "data": {
    "channel_info": {
      "CH6": {
        "ip_address": "172.16.10.87",
        "port": 9000,
        "protocol": "Private",
        "username": "admin",
        "password": "11111"
      }
    }
  },
  "operation_type": "AddOrEditChannel"
}

```

```
}  
}
```

Response message:

HTTP/1.1 200 OK

```
{  
  "version": "1.0",  
  "result": "success",  
  "data": {  
  }  
}
```

Scene 4: delete ipc

Request message:

POST /API/ChannelConfig/IPChannel/Set?2020-10-28@14:09:37 HTTP/1.1

```
{  
  "version": "1.0",  
  "data": {  
    "remove_ipc": ["CH6"]  
  }  
}
```

Response message:

HTTP/1.1 200 OK

```
{  
  "version": "1.0",  
  "result": "success",  
  "data": {  
  }  
}
```

5.1.3 Custom protocol management

GET	
URL	POST/API/ChannelConfig/ProtocolManage/Get
Description	Get the protocol parameters
Request Body	None
Successful Response	Channel Information JSON (show as follow Table-5.1.6)

SET	
URL	POST/API/ChannelConfig/ProtocolManage/Set
Description	Set the protocol parameters
Request Body	Channel Information JSON (show as follow Table-5.1.6)
Successful Response	The successful result response that described in 2.5

Table-5.1.6 (protocolInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
protocol_info		Json Object	ProtocolInformation JSON show as follow Table-5.1.6.1

Table-5.1.6.1 (protocol Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
protocol1		Json Object	JSON show as follow Table-5.1.7
...			
protocol16		Json Object	

Table-5.1.7 (Single protocolInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
protocol_name	Max length: 15byte	string	
custom_stream		Json array	Please refer to Table-5.1.8

Table-5.1.8 (Single protocolInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
stream_name	“Mainstream”“Substream”	string	
enable	true, false	bool	Enable bitstream; available for Substream only.
type	“Rtsp”	string	
port	1-65535	int	
source_path	Max length: 119byte	string	

5.1.4 Analog channel

GET	
URL	POST/API/ChannelConfig/AnalogChannel/Get
Description	Get theanalogconfig
Request Body	ChannelRequest JSON(show as follow Table-5.1.1)
Successful Response	Channel Information JSON (show as follow Table-5.1.9)

SET	
-----	--

URL	POST/API/ChannelConfig/AnalogChannel/Set
Description	Set the analog config
Request Body	Channel Information JSON (show as follow Table-5.1.9)
Successful Response	The successful result response that described in 2.5

Table-5.1.9 (analog channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		Json Object	SingleInformation JSON show as follow Table-5.1.9.1

Table-5.1.9.1 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-5.1.10
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-5.1.10 (Single Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
state	“Disable”“Enable”	string	Analog channel status
channel_name	Max length: 31byte	string	Channel name
switch		bool	true: enable false: disable

Table-5.1.11(Error Code)

error_code	commet
Channel_limit	Exceeded the max number of channels in the system

5.1.5 Wireless Channel

GET	
URL	POST/API/ChannelConfig/WirelessCamera/Get
Description	Get the wireless camera config
Request Body	ChannelRequest JSON(show as follow Table-5.1.1)

Successful Response	Channel Information JSON (show as follow Table-5.1.11)
---------------------	--

SET	
URL	POST/API/ChannelConfig/WirelessCamera/Set
Description	Set the wireless camera config
Request Body	Channel Information JSON (show as follow Table-5.1.13)
Successful Response	The successful result response that described in 2.5

Table-5.1.11 (analog channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		Json Object	SingleInformation JSON show as follow Table-5.1.11.1

Table-5.1.11.1 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-5.1.10
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-5.1.12 (Single Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_name	Max length: 31byte	string	Channel name
software_version	Max length: 40byte	string	Wireless channel version,read-only
switch		bool	true: enable false: disable this switch only disable。 (from true to false)

Table-5.1.13 (analog channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		Json Object	SingleInformation JSON show as follow Table-5.1.13.1
page_type	"ChannelConfig"	string	

Table-5.1.13.1 (Channel Information JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-5.1.14
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-5.1.14 (Single Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_name	Max length: 31byte	string	Channel name
software_version	Max length: 40byte	string	Wireless channel version,read-only
switch		bool	true: enable false: disable this switch only disable。 (from true to false)
chn_index	"CH1"... "CHx" "IP_CH1"... "IP_CHx" "WIFI_CH1"... "WIFI_CHx"	string	
page	"chn_wireChn"	string	
battery	0-100	int	Power value
post_recording	"Continuous" "10" "20" "30"	string	Recording delay time
pair_state	-1 - 7	int	Matching status of wireless Ipc

5.1.6 Digital wireless channel configuration

GET	
URL	POST/API/ChannelConfig/ChannelConfig/Get
Description	Get the channel config
Request Body	ChannelRequest JSON(show as follow Table-5.1.6.1)
Successful Response	Channel Information JSON (show as follow Table-5.1.6.2)

SET	
URL	POST/API/ChannelConfig/ChannelConfig/Set

Description	Set the channel config
Request Body	Channel Information JSON (show as follow Table-5.1.6.2)
Successful Response	The successful result response that described in 2.5

Table-5.1.6.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x”	string array	Each array bit represents a channel with a string.

Table-5.1.6.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_max		int	Max number of channels
limit_wireless_max		int	Max number of wireless channel
channel_info		Json Object	JSON show as follow Table-5.1.6.x

Table-5.1.6.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Objec	JSON show as follow Table-5.1.6.3
...		Json Objec	
CHx		Json Objec	

Table-5.1.6.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
camera_type	("Digital") ("Wireless")	string	Mode of current channel

Example:

Request message:

POST /API/ChannelConfig/ChannelConfig/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "channel": ["CH1"]
  }
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
```

```

    "CH1": {
      "camera_type": "Digital"
    }
  }
}

```

5.2 OSD

GET	
URL	POST/API/ChannelConfig/OSD/Get
Description	Get the channel osd parameters
Request Body	ChannelRequest JSON(show as follow Table-5.2.1)
Successful Response	Channel Information JSON (show as follow Table-5.2.2)

SET	
URL	POST/API/ChannelConfig/OSD/Set
Description	Set the channel osd parameters
Request Body	Channel Information JSON (show as follow Table-5.2.2)
Successful Response	The successful result response that described in 2.5

Table-5.2.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... "WIFI_CH1x" The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-5.2.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		Json Object	JSON show as follow Table-5.2.x
channel_max		int	Max number of channel
support_copy		bool	Whether support to copy (NVR、DVR)

Table-5.2.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

CHI		Json Object	JSON show as follow Table-5.2.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-5.2.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline", "Online"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
name		JSON Object	Name JSON show as follow Table-5.2.4
datetime		JSON Object	DateTime JSON show as follow Table-5.2.5
alarm		JSON Object	Alarm_Pos JSON show as follow Table-5.2.6
covert		bool	Preview switch (NVR/DVR only)
refresh_rate	"50Hz", "60Hz"	string	Refresh frequency (A01 not used) Note: Analog channel is not supported
alpha	1-128	int	OSD transparency Note: Analog channel does not support
camera_type	"AUTO""AHD""TVI""C VI"	string	Image type (dvr only)
eq_level	"AUTO""1(0~50m)""2(5 0~150m)""3(150~250m)"" 4(250~350m)""5(350~4 50m)""	string	EQ level (dvr only)
opy_ch	"digit""analog""wifi"	string	Flag supporting channel replication (for NVR and DVR only)
channel_enable		bool	Judge whether the current channel parameters can be configured
osd_invert		bool	OSD invert switch

Table-5.2.4 (Name JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
show		bool	OSD show channel name or not
text	Max length: 31byte	string	OSD text(DVR/NVR/IPC)
pos		JSON object	Pos JSON show as follow Table-5.2.7

Table-5.2.5 (DateTime JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
show		bool	Whether the video window displays the date
date_format	"MM/DD/YYYY", "YYYY-MM-DD", "DD/MM/YYYY"	string	Date format Note: Analog channel is not supported
time_format	24,12	int	Time mode, unit: hour Note: Analog channel is not supported
time		int	Current device time (A01 not used)
show_week		bool	Whether the OSD displays the week (A01 only)
pos		JSON object	Pos JSON show as follow Table-5.2.7

Table-5.2.6 (Alram_Pos JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
show		bool	Alarm_Pos whether support motion
text	Max length:32/40byte	string	Alarm OSD text, IPC max 32byte
pos		JSON object	Pos JSON show as follow Table-5.2.7

Table-5.2.7 (Pos JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
x	[0~704]	int	X oordinate
y	[0~576]	int	y oordinate

Example:

Request message:

POST /API/ChannelConfig/OSD/Get?2020-10-28@14:13:58 HTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "CH1": {
        "name": {
          "show": true,
          "text": "Camera",
          "pos": {
            "x": 199,
```

```

        "y": 293
    }
},
"datetime": {
    "show": true,
    "date_format": "YYYY-MM-DD",
    "time_format": 24,
    "pos": {
        "x": 338,
        "y": 258
    }
},
"refresh_rate": "60Hz",
"covert": false,
"alpha": 64
},
"CH2": {
    "status": "Offline"
},
"CH3": {
    "name": {
        "show": true,
        "text": "fisheye",
        "pos": {
            "x": 282,
            "y": 278
        }
    },
    "datetime": {
        "show": true,
        "date_format": "YYYY-MM-DD",
        "time_format": 24,
        "pos": {
            "x": 279,
            "y": 242
        }
    },
    "refresh_rate": "50Hz",
    "covert": false,
    "alpha": 58
},
"CH4": {
    "status": "Offline"
},

```

```

"CH5": {
  "name": {
    "show": true,
    "text": "Channel 2",
    "pos": {
      "x": 150,
      "y": 0
    }
  },
  "datetime": {
    "show": true,
    "date_format": "MM/DD/YYYY",
    "time_format": 24,
    "pos": {
      "x": 309,
      "y": 0
    }
  },
  "refresh_rate": "50Hz",
  "covert": false,
  "alpha": 64
},
"CH6": {
  "reason": "Not configured"
},
"CH7": {
  "reason": "Not configured"
},
"CH8": {
  "reason": "Not configured"
}
}
}
}

```

5.3 Image Control

GET	
URL	POST/API/ChannelConfig/ImageControl/Get
Description	Get the channel image control parameters
Request Body	ChannelRequest JSON(show as follow Table-5.3.1)
Successful Response	Channel Information JSON (show as follow Table-5.3.2)

SET

URL	POST/API/ChannelConfig/ImageControl/Set
Description	Set the channel image control parameters
Request Body	Channel Information JSON (show as follow Table-5.3.2)
Successful Response	The successful result response that described in 2.5

DEFAULT	
URL	POST/API/ChannelConfig/ImageControl/Default
Description	Get the channel image control default parameters
Request Body	ChannelRequest JSON(show as follow Table-5.3.1)
Successful Response	Channel Default Information JSON (show as follow Table-5.3.2)

Table-5.3.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-5.3.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON Object	Channel Information JSON show as follow Table-5.3.x
channel_max		Int	Max number of channel

Table-5.3.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-5.3.3 (neutral) JSON show as follow Table-5.3.5 (A01 only)
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-5.3.3 (camera modeInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

status	"Offline","Online"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
support_default		bool	Whether restoring default values is supported
camera_param_mode	"Daylight""Night"	string	Only supports the existence of day and night
image_setting	"FullColorMode", "DayNightMode", "Schedule"	string	Mode selection under the new full-color logic
FullColorMode		Json Object	Full-color mode parameter structure, including parameters as shown in Table 5.3.3.1
DayNightMode		Json Object	Day and night mode parameter structure, including parameters as shown in Table 5.3.3.2
image_setting_schedule		JSON array	Schedule mode, refer to 9.1.3 for structure
Daylight		Json Object	Day (or ordinary) parameter structure, including parameters as shown in Table 5.3.4
Night		Json Object	The parameters included in the night parameter structure (only exists when the day and night parameter mode is supported) are shown in Table 5.3.4
ir_cut_mode	"AutoMode" "ColorMode", "BlackWhiteMode", "VideoMode" "TimeSchedule"	string	IR-CUT mode (using when don't support full color)
ir_cut_mode	"Auto" "Day", "Night", "Image" "Schedule"	string	IR-CUT mode (using when support full color)
start_time		string	Start time
end_time		string	End time
ir_cut_delay	1-36	int	Ir-cut delay
ir_led	"Off", "On", "Auto", "Manual",	string	Ir led

low_beam_light	1-100	int	Low beam light (Ir lamp takes effect in manual mode)
high_beam_light	1-100	int	High beam light (Ir lamp takes effect in manual mode)
mirror_mode	"Close", "VerticalMirroring", "HorizontalMirroring", "All"	string	Mirror model
corridor_mode	"Close","Open"	string	Corridor mode
angle_rotation	"0","90","180","270"	string	Angle rotation

Table-5.3.3.1 (Full Color Mode Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
white_light	"Auto", "Manual", "Schedule", "OFF"	string	Mode of fill light
light_distance	0-100	Int	The brightness of the fill light is only valid when (white_light=="Manual")
whitelight_schedule		JSON array	Refer to 9.1.3 for the structure of fill lamp schedule mode

Table-5.3.3.2 (Day Night tMode Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ir_cut_mode	"Auto" "Day", "Night", "Image" "Schedule"	string	IR-CUT mode
ircut_schedule		JSON array	Refer to 9.1.3 for the structure of schedule mode
ir_cut_delay	1-36	int	Ir cut delay
ir_led	"Manual", "SmartIR", "OFF",	string	Ir led
low_beam_light	0-100	int	Low beam light (Ir lamp takes effect in manual mode)
high_beam_light	0-100	int	High beam light (Ir lamp takes effect in manual mode)

Table-5.3.4 (camera mode infoJSON)

exposure_mode	"Auto","Manual","ShutterFirst",	string	Exposure mode
---------------	---------------------------------	--------	---------------

	"IrisFirst"		
iris	"f1.4","f1.6","f2.0","f2.4","f2.8", "f3.4","f4.0","f4.8","f5.6","f6.8", "f9.6","f11","f14","f16","f19", "f22"	string	Iris
iris_max	0-100	Int	Iris maximum
iris_min	0-100	Int	Iris minimum (less than aperture maximum)
shutter_min	"1/5","1/8","1/15","1/25","1/30", "1/50","1/60","1/100","1/120", "1/150","1/180","1/200","1/240", "1/250","1/300","1/360","1/480", "1/500","1/600","1/700","1/1000", "1/1500","1/2000","1/2500", "1/5000","1/10000","1/12000","1/20000", "Flickerless",	string	Minimum shutter limit (less than maximum shutter limit)
shutter_max	"1/5","1/8","1/15","1/25","1/30", "1/50","1/60","1/100","1/120", "1/150","1/180","1/200","1/240", "1/250","1/300","1/360","1/480", "1/500","1/600","1/700","1/1000", "1/1500","1/2000","1/2500", "1/5000","1/10000","1/12000","1/20000", "Flickerless",	string	Maximum shutter limit
gain	0-100 (A01 only) 、 1-128 (new) / "Disable", "Low", "Middle", "High"(old)	int/string (old)	Gain: IPC ranges from 1-128; A01 ranges from 0-100
angle_rotation	"0","90","180","270"	string	Angle rotation
shutter_limit	"1/5","1/8","1/15","1/25","1/30", "1/50","1/60","1/100","1/120", "1/150","1/180","1/200","1/240", "1/250","1/300","1/360","1/480", "1/500","1/600","1/700","1/1000", "1/1500","1/2000","1/2500", "1/5000","1/10000","1/12000","1/20000", "Flickerless",	string	Shutter limit(s)
back_light	"WDR", "HLC", "DWDR", "BacklightCompensation", "Close"	string	Back light
wdr_hide_ai_area		bool	Whether WDR and dash box display are mutually exclusive
wdr_coefficient	0-100、 1-255	int	WDR coefficient

dwdr_coefficient	0-100、1-255	int	DWDR coefficient
hlc_strength	1-255	int	HLT strength coefficient
bhc_level	1-15	Int	Backlight compensation intensity
back_light_area	"Top","Left","Down","Right","Center"	string	Backlight compensation area
white_balance	"Auto","Manual", "AutoTracking","OnePush", "Indoor","Outdoor", "FixedSodiumLamp", "FluorescentLamp1", "FluorescentLamp2", "IncandescentLamp", "Sunlight"	string	White balance Ipc data "Auto" "Manual" , "Indoor"
red_tuning	0-100、1-255	int	Red tuning
blue_tuning	0-100、1-255	int	Blue tuning
green_tuning	1-255	Int	Green tuning
defog_mode	"Disable","Auto","Manual"	string	Defog mode
defogging_level	0-100/(1-255)	int	Defogging level,(IPC:1-255)
denoising	"Close", "CommonMode", "ExpertMode" or "Disable", "Auto", "Manual"	string	Denoising switch
denoising_level	0-100/(1-255)	int	Denosing level,(IPC: 1-255)
distort_correct	"Close","Open"	string	distort_correct
distort_correct_mode	"Auto","Manual"	string	distort_correct_mode
distort_correct_level	0-100	int	distort_correct_level
<p>Be careful!!!</p> <ol style="list-style-type: none"> 1. The lower limit of color temperature cannot be greater than the upper limit of color temperature 2. Manual exposure and wide dynamic, strong light suppression, background frequency, backlight compensation are mutually exclusive 3. In automatic exposure mode: wide dynamic and strong light suppression are mutually exclusive, slow shutter and wide dynamic, strong light suppression are mutually exclusive 			

Table-5.3.5 (camera mode infoJSON)

status	"Offline","Online"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
mirror_mode	"Close", "VerticalMirroring", "HorizontalMirroring", "All"	string	Mirror mode
corridor_mode	"Close","Open"	string	Corridor mode

ir_cut_mode	"AutoMode" "ColorMode", "BlackWhiteMode", "VideoMode"/"ImageMode" "TimeSchedule"	string	IR-CUT mode (non A01 dedicated) see Table-5.3.4.1), note: no photosensitive IPC changes VideoMode to ImageMode.
ir-cut_mode	"AutoMode" "ColorMode", "BlackWhiteMode",	string	IR-CUT mode (A01 only)
ir_cut_delay	1-36	int	Ir cut delay
ir_led	"Off", "On", "Auto", "Manual",	string	Ir led (Manual see Table-5.3.4.2)
image_sensitivity	"Low", "Middle", "High"	string	No photosensitive control sensitivity, supporting the use of photosensitive devices.
day/night_mode	"SignalTriggered", "ImageTriggered"	string	Day/night mode (A01 only)
sensitivity	"Low","Middle","High"	string	Sensitivity (A01 only)
exposure_mode	"Auto","Manual","ShutterFirst", "IrisFirst"	string	Exposure mode
shutter_speed	"1/3","1/4","1/8","1/12","1/15", "1/25","1/30","1/50", "1/60",,, "1/90","1/100","1/120", "1/125","1/180","1/250","1/350" ,"1/500","1/725","1/1000","1/1500", "1/2000","1/3000", "1/4000","1/6000","1/10000","1/30000", "1/100000"	string	Shutter data(A01 only)
iris	"f1.4","f1.6","f2.0","f2.4","f2.8", "f3.4","f4.0","f4.8","f5.6","f6.8", "f9.6","f11","f14","f16","f19", "f22"	string	Iris
iris_max	0-100	Int	Iris max
iris_min	0-100	Int	Iris min (less than maximum iris limit)
gain	0-100 (A01 only) 、 1-128 (new) / "Off", "Low", "Middle", "High"(old)	int/string (old)	Gain, IPC: 1-128; A01:0-100
angle_rotation	"0","90","180","270"	string	Angle rotation
coefficient	0-100	int	Exposure compensation

			coefficient (A01 only)
background_frequency	"Close","50Hz","60Hz"	string	Background frequency (A01 only)
auto_slow_shutter	"Close","Open"	string	Slow shutter switch (A01 only)
shutter_limit(s)	"1/5","1/8","1/15","1/25","1/30", "1/50","1/60","1/100","1/120", "1/150","1/180","1/200","1/240", "1/250","1/300","1/360","1/480", "1/500","1/600","1/700","1/1000", "1/1500","1/2000","1/2500", "1/5000","1/10000","1/12000","1/20000", "Flickerless",	string	Shutter limit
back_light	"WDR", "HLC", "BacklightCompensation", "Close"	string	Backlight
wdr_coefficient	0-100、 1-255	int	WDR coefficient
hlc_coefficient	"Low", "Middle", "High" (A01 specific) / 1-255	string /int	HLT coefficient (A01 only) , IPC:1-255
hlc_strength	1-255	int	HLT STRENGTH coefficient (non A01 only)
blc_level	1-15	Int	BLC strength
back_light_area	"Top","Left","Down","Right","Center"	string	BLC area (non A01 only)
blc_area	"Center Area""Custom"	string	(A01 only)
white_balance	"Auto","Manual", "AutoTracking","OnePush", "Indoor","Outdoor", "FixedSodiumLamp", "FluorescentLamp1", "FluorescentLamp2", "IncandescentLamp", "Sunlight"	string	White balance Ipc : "Auto" "Manual" , "Indoor"
red_fine-tuning	0-100	int	Red fine tuning(A01 only)
blue_fine-tuning	0-100	int	Blue fine tuning (A01 only)
red_tuning	0-100、 1-255	int	Red tuning
blue_tuning	0-100、 1-255	int	Blue tuning
green_tuning	1-255	Int	Green tuning
ct_upper_limit	1500-15000	int	Color temperature upper limit (A01only)
ct_lower_limit	1500-15000	int	Color temperature lower limit (lower limit must be less than upper limit) (A01 only)
defog_mode	"Disable","Auto","Manual"	string	Defog mode

defogging_level	0-100/(1-255)	int	Defogging level,(IPC:1-255)
elect-img-stabilization	"Close","Open"	string	elect-img-stabilization (A01 only)
denoising	"Close","CommonMode", "ExpertMode" or "Disable" "Auto" , "Manual"	string	Denoising switch
denoising_level	0-100/(1-255)	int	Denoising level,(IPC: 1-255)
spatial_denoising_level	0-100	int	Spatial denosing level(A01 only)
Temporal_denoising_level	0-100	int	Temporal denosing level(A01 only)
distort_correct	"Close","Open"	string	Distort correct
distort_correct_mode	"Auto","Manual"	string	Distort correct mode
distort_correct_level	0-100	int	Distort correct level
support_default		bool	Whether support load default
<p>Be careful:!!!</p> <ol style="list-style-type: none"> 1. The lower limit of color temperature cannot be greater than the upper limit of color temperature 2. Manual exposure and wide dynamic, strong light suppression, background frequency, backlight compensation are mutually exclusive 3. In automatic exposure mode: wide dynamic and strong light suppression are mutually exclusive, slow shutter and wide dynamic, strong light suppression are mutually exclusive 			

Table-5.3.4.1 (ir_cut_mode TimeSchedule)

start_time		string	Start time
end_time		string	End time

Table-5.3.4.2 (ir_led Manual)

low_beam_light	1-100	int	Low beam light
high_beam_light	1-100	int	High beam light

5.4 Video Color

GET	
URL	POST/API/ChannelConfig/Color/Get
Description	Get the channel video color parameters
Request Body	ChannelRequest JSON(show as follow Table-5.4.1)
Successful Response	Channel Information JSON (show as follow Table-5.4.2)

SET	
URL	POST/API/ChannelConfig/Color/Set
Description	Set the channel video color parameters
Request Body	Channel Information JSON (show as follow Table-5.4.2)
Successful Response	The successful result response that described in 2.5

DEFAULT	
URL	POST/API/ChannelConfig/Color/Default
Description	Get the channel video color default parameters
Request Body	ChannelRequest JSON(show as follow Table-5.4.1)
Successful Response	Channel Default Information JSON (show as follow Table-5.4.2)

Table-5.4.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-5.4.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as follow Table-5.4.x

Table-5.4.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-5.4.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-5.4.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
hue	0-255	int	hue
bright	0-255	int	bright

contrast	0-255	int	contrast
saturation	0-255	int	saturation
sharpness	0-255	int	sharpness
support_default		bool	Whether support load default

5.5 Video Cover

GET	
URL	POST/API/ChannelConfig/VideoCover/Get
Description	Get the channel video cover parameters
Request Body	ChannelRequest JSON(show as follow Table-5.5.1)
Successful Response	Channel Information JSON (show as follow Table-5.5.2)

SET	
URL	POST/API/ChannelConfig/VideoCover/Set
Description	Set the channel video cover parameters
Request Body	Channel Information JSON (show as follow Table-5.5.2)
Successful Response	The successful result response that described in 2.5

Table-5.5.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-5.5.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON Object	Channel Information JSON show as follow Table-5.5.x
channel_max		Int	Max channel
support_copy		bool	Whether support copy (NVR、DVR only)

Table-5.5.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-5.5.3

...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-5.5.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
privacy_zone_enable		Bool	Privacy zone switch
zone_info_p		JSON array	Zone JSON show as follow Table-5.5.5 Note: The A01 SDK protocol supports up to 4 occlusion regions (A01 only)
zone_info		JSON array	Zone JSON show as follow Table-5.5.6 The onvif protocol supports up to 4 occlusion regions
copy_ch	"digit""analog""wifi"	string	Flag supporting channel replication (NVR and DVR only)

Table-5.5.5 (Zone JSON-A01 SDK)

KEY	VALUE		COMMENT
	RANGE	TYPE	
zone_no	1-8	int	Occlusion area number
maxnum	8	int	Support the maximum number of occluded regions
color		int	Rgb color value
point		JSON array	Zone coordinates JSON show as follow Table-5.5.7

Table-5.5.6 (Zone JSON-onvif)

KEY	VALUE		COMMENT
	RANGE	TYPE	
zone_no	1-4	int	Occlusion area number
zone_enable		bool	Occlusion area switch
rect		JSON object	Zone coordinates JSON show as follow Table-5.5.7

Table-5.5.7 (Zone coordinates JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
left	0-704(A01 SDK)	int	X coordinate
top	0-576(A01 SDK)	int	Y coordinate
width	0-704	int	width
height	0-576	int	height

5.6 PTZ

GET	
URL	POST/API/ChannelConfig/PTZ/Get
Description	Get the channel PTZparameters
Request Body	ChannelRequest JSON(show as follow Table-5.6.1)
Successful Response	Channel Information JSON (show as follow Table-5.6.2)

SET	
URL	POST/API/ChannelConfig/PTZ/Set
Description	Get the channel PTZparameters
Request Body	Channel Information JSON (show as follow Table-5.6.2)
Successful Response	The successful result response that described in 2.5

Table-5.6.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-5.6.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_max		int	Max channel nummber
support_copy		bool	Whether support copy (NVR、DVR only)
channel_info		JSON object	Channel Information JSON show as follow Table-5.6.x

Table-5.6.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-5.6.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-5.6.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
signal_type	"Analog" "Digital"	string	Signal type
protocol	"PelcoD","PelcoP" "COAX1", "COAX2"	string	PTZ protocol
baudrate	"1200","2400","4800", "9600"	string	Baud rate
databit	"8","7","6","5"	string	Data bits
stopbit	"1","2"	string	Stop bit
parity	"None","Odd","Even", "Mark","Space"	string	Parity bit
address	1-255.	int	Address code
copy_ch	"digit""analog""wifi"	string	Flag supporting channel replication (for NVR and DVR only)

5.7 ROI

GET	
URL	POST/API/ChannelConfig/ROI/Get
Description	Get the channel ROIparameters
Request Body	ChannelRequest JSON(show as follow Table-5.7.1)
Successful Response	Channel Information JSON (show as follow Table-5.7.2)

SET	
URL	POST/API/ChannelConfig/ROI/Set
Description	Get the channel ROIparameters
Request Body	Channel Information JSON (show as follow Table-5.7.1)
Successful Response	The successful result response that described in 2.5

Table-5.7.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...”IP_CH1x” “WIFI_CH1”...”WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-5.7.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as follow Table-5.7.x IPC ONLY USE CH1

Table-5.7.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-5.7.2
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-5.7.2 ((Stream Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
main_stream_info		Json Object	Stream Information JSON show as follow Table-5.7.3
sub_stream_info		Json Object	Stream Information JSON show as follow Table-5.7.3
mobile_stream_info		Json Object	Stream Information JSON show as follow Table-5.7.3

Table-5.7.3 (Single Stream Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
region_id_1	1-8	Json object	Region ID, single ROI Information JSON as

			follow Table
...		Json Object	
region_id_8		Json Object	

Table-5.7.4 ((ROI mainstream Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
roi_switch		bool	ROI Enable
roi_level	"Lowest", "Lower" "Low", "Medium" "Higher", "Highest"	string	ROI Level
main_non_roi_fps		string	Non ROI Zone Fps
rect		Json Object	ROI Zone Rection, Rection Information JSON as follow Table-5.7.7

Table-5.7.5 ((ROI substream Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
roi_switch		bool	ROI Enable
roi_level	"Lowest", "Lower" "Low", "Medium" "Higher", "Highest"	string	ROI Level
sub_non_roi_fps		string	Non ROI Zone Fps
rect		Json Object	ROI Zone Rection, Rection Information JSON as follow Table-5.7.7

Table-5.7.6 ((ROI mobilestream Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
roi_switch		bool	ROI Enable
roi_level	"Lowest", "Lower" "Low", "Medium" "Higher", "Highest"	string	ROI Level
mobile_non_roi_fps		string	Non ROI Zone Fps
rect		Json Object	ROI Zone Rection, Rection Information JSON as follow Table-5.7.7

Table-5.6.7 (Rection Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
left	0-704	int	Left

top	0-576	int	Top
width	0-704	int	Width
height	0-576	int	Height

Example:

Request message:

POST /API/ChannelConfig/ROI/Get HTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 207

Connection: keep-alive

```
{
  "result": "success",
  "data": {
    "roi_info": {
      "main_stream_info": {
        "roi_id": [
          {
            "roi_zone_no": 1,
            "roi_index": 0,
            "roi_switch": true,
            "is_abs_qp": 0,
            "roi_level": 1,
            "non_roi_fps": 2,
            "rect": {
              "left": 0,
              "top": 180,
              "width": 360,
              "height": 180
            }
          },
          {
            "roi_zone_no": 2,
            "roi_index": 1,
            "roi_switch": false,
            "is_abs_qp": 0,

```

```
"roi_level": 1,
"non_roi_fps": 30,
"rect": {
  "left": 0,
  "top": 0,
  "width": 0,
  "height": 0
}
},
{
  "roi_zone_no": 3,
  "roi_index": 2,
  "roi_switch": false,
  "is_abs_qp": 0,
  "roi_level": 1,
  "non_roi_fps": 30,
  "rect": {
    "left": 0,
    "top": 0,
    "width": 0,
    "height": 0
  }
},
{
  "roi_zone_no": 4,
  "roi_index": 3,
  "roi_switch": false,
  "is_abs_qp": 0,
  "roi_level": 1,
  "non_roi_fps": 30,
  "rect": {
    "left": 0,
    "top": 0,
    "width": 0,
    "height": 0
  }
},
{
  "roi_zone_no": 5,
  "roi_index": 4,
  "roi_switch": false,
  "is_abs_qp": 0,
  "roi_level": 1,
  "non_roi_fps": 30,
```

```
"rect": {
  "left": 0,
  "top": 0,
  "width": 0,
  "height": 0
}
},
{
  "roi_zone_no": 6,
  "roi_index": 5,
  "roi_switch": false,
  "is_abs_qp": 0,
  "roi_level": 1,
  "non_roi_fps": 30,
  "rect": {
    "left": 0,
    "top": 0,
    "width": 0,
    "height": 0
  }
},
{
  "roi_zone_no": 7,
  "roi_index": 6,
  "roi_switch": false,
  "is_abs_qp": 0,
  "roi_level": 1,
  "non_roi_fps": 30,
  "rect": {
    "left": 0,
    "top": 0,
    "width": 0,
    "height": 0
  }
},
{
  "roi_zone_no": 8,
  "roi_index": 7,
  "roi_switch": false,
  "is_abs_qp": 0,
  "roi_level": 1,
  "non_roi_fps": 30,
  "rect": {
    "left": 0,
```



```

        "top": 0,
        "width": 0,
        "height": 0
    }
}
]
},
"sub_stream_info": {
    "roi_id": [
        {
            "roi_zone_no": 1,
            "roi_index": 0,
            "roi_switch": false,
            "is_abs_qp": 0,
            "roi_level": 1,
            "non_roi_fps": 10,
            "rect": {
                "left": 0,
                "top": 0,
                "width": 0,
                "height": 0
            }
        },
        {
            "roi_zone_no": 2,
            "roi_index": 1,
            "roi_switch": false,
            "is_abs_qp": 0,
            "roi_level": 1,
            "non_roi_fps": 10,
            "rect": {
                "left": 0,
                "top": 0,
                "width": 0,
                "height": 0
            }
        },
        {
            "roi_zone_no": 3,
            "roi_index": 2,
            "roi_switch": false,
            "is_abs_qp": 0,
            "roi_level": 1,
            "non_roi_fps": 10,

```

```
"rect": {
  "left": 0,
  "top": 0,
  "width": 0,
  "height": 0
}
},
{
  "roi_zone_no": 4,
  "roi_index": 3,
  "roi_switch": false,
  "is_abs_qp": 0,
  "roi_level": 1,
  "non_roi_fps": 10,
  "rect": {
    "left": 0,
    "top": 0,
    "width": 0,
    "height": 0
  }
},
{
  "roi_zone_no": 5,
  "roi_index": 4,
  "roi_switch": false,
  "is_abs_qp": 0,
  "roi_level": 1,
  "non_roi_fps": 10,
  "rect": {
    "left": 0,
    "top": 0,
    "width": 0,
    "height": 0
  }
},
{
  "roi_zone_no": 6,
  "roi_index": 5,
  "roi_switch": false,
  "is_abs_qp": 0,
  "roi_level": 1,
  "non_roi_fps": 10,
  "rect": {
    "left": 0,
```

```

        "top": 0,
        "width": 0,
        "height": 0
    }
},
{
    "roi_zone_no": 7,
    "roi_index": 6,
    "roi_switch": false,
    "is_abs_qp": 0,
    "roi_level": 1,
    "non_roi_fps": 10,
    "rect": {
        "left": 0,
        "top": 0,
        "width": 0,
        "height": 0
    }
},
{
    "roi_zone_no": 8,
    "roi_index": 7,
    "roi_switch": false,
    "is_abs_qp": 0,
    "roi_level": 1,
    "non_roi_fps": 10,
    "rect": {
        "left": 0,
        "top": 0,
        "width": 0,
        "height": 0
    }
}
]
},
"mobile_stream_info": {
    "roi_id": [
        {
            "roi_zone_no": 1,
            "roi_index": 0,
            "roi_switch": false,
            "is_abs_qp": 0,
            "roi_level": 1,
            "non_roi_fps": 10,

```

```
"rect": {
  "left": 0,
  "top": 0,
  "width": 0,
  "height": 0
}
},
{
  "roi_zone_no": 2,
  "roi_index": 1,
  "roi_switch": false,
  "is_abs_qp": 0,
  "roi_level": 1,
  "non_roi_fps": 10,
  "rect": {
    "left": 0,
    "top": 0,
    "width": 0,
    "height": 0
  }
},
{
  "roi_zone_no": 3,
  "roi_index": 2,
  "roi_switch": false,
  "is_abs_qp": 0,
  "roi_level": 1,
  "non_roi_fps": 10,
  "rect": {
    "left": 0,
    "top": 0,
    "width": 0,
    "height": 0
  }
},
{
  "roi_zone_no": 4,
  "roi_index": 3,
  "roi_switch": false,
  "is_abs_qp": 0,
  "roi_level": 1,
  "non_roi_fps": 10,
  "rect": {
    "left": 0,
```

```
        "top": 0,
        "width": 0,
        "height": 0
    }
},
{
    "roi_zone_no": 5,
    "roi_index": 4,
    "roi_switch": false,
    "is_abs_qp": 0,
    "roi_level": 1,
    "non_roi_fps": 10,
    "rect": {
        "left": 0,
        "top": 0,
        "width": 0,
        "height": 0
    }
},
{
    "roi_zone_no": 6,
    "roi_index": 5,
    "roi_switch": false,
    "is_abs_qp": 0,
    "roi_level": 1,
    "non_roi_fps": 10,
    "rect": {
        "left": 0,
        "top": 0,
        "width": 0,
        "height": 0
    }
},
{
    "roi_zone_no": 7,
    "roi_index": 6,
    "roi_switch": false,
    "is_abs_qp": 0,
    "roi_level": 1,
    "non_roi_fps": 10,
    "rect": {
        "left": 0,
        "top": 0,
        "width": 0,
```

```

        "height": 0
      }
    },
    {
      "roi_zone_no": 8,
      "roi_index": 7,
      "roi_switch": false,
      "is_abs_qp": 0,
      "roi_level": 1,
      "non_roi_fps": 10,
      "rect": {
        "left": 0,
        "top": 0,
        "width": 0,
        "height": 0
      }
    }
  ]
}

```

5.8 POE Power

GET	
URL	POST/API/ChannelConfig/PoePower/Get
Description	Get the wireless camera config
Request Body	ChannelRequest JSON(show as follow Table-5.8.1)
Successful Response	Channel Information JSON (show as follow Table-5.8.2)

Table-5.8.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-5.8.2 (analog channel Information JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
channel_info		Json Object	SingleInformation JSON show as follow Table-5.8.2.1
actual_power		string	Used power
surplus_power		string	Residual power
total_power		string	Total power
channel_min_power		string	Minimum power per channel
channel_max_power		string	Maximum power per channel
channel_max		int	Total number of channels

Table-5.8.2.1 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-5.8.2.1.1
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-5.8.2.1.1 (Single Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_current_power	Max length: 7byte	string	Power for current channel

5.9 Video Crop

GET	
URL	POST/API/ChannelConfig/VideoCrop/Get
Description	Get the channel video crop parameters
Request Body	ChannelRequest JSON(show as follow Table-5.9.1)
Successful Response	Channel Information JSON (show as follow Table-5.9.2)

SET	
URL	POST/API/ChannelConfig/VideoCrop/Set
Description	Set the channel video crop parameters
Request Body	Channel Information JSON (show as follow Table-5.9.2)
Successful Response	The successful result response that described in 2.5

Table-5.9.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... "WIFI_CH1x" The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-5.9.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON Object	Channel Information JSON show as follow Table-5.9.x
channel_max		Int	Max channel number

Table-5.9.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-5.9.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-5.9.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
enable		bool	Enable switch
fps	1-10	int	Crop Fps
zone_info		JSON array	Zone JSON show as follow Table-5.9.4 Note: only support 2 zone

Table-5.9.4(Zone JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
zone_no	1-2	int	Zone number
rect		JSON object	Zone coordinates JSON show as follow Table-5.9.5

Table-5.9.5 (Zone coordinates JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
left	0-704	int	X coordinate
top	0-576	int	Y coordinate
width	0-704	int	width
height	0-576	int	height

5.10 Remote Pair (Wireless specific)

SET	
URL	POST/API/Login/ChannelPairing/Set
Description	Setting up remote pairing
Request Body	ChannelRequest JSON(show as follow Table-5.10.1)
Successful Response	Channel Information JSON (show as follow Table-5.10.2)

Table-5.10.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel		int	Channel number of remote pairing
pair_status		int	Remote pairing operation command, 1: pair, 0: unpair, 1: pair, 0: unpair

Table-5.10.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
state	Max_length:20	string	Status of remote pairing. Pairing: pairing Device is busy: Device is busy

6 STREAM

6.1 Encode

GET	
URL	POST/API/StreamConfig/MainStream/Get
	POST/API/StreamConfig/SubStream/Get
	POST/API/StreamConfig/MobileStream/Get
	POST/API/StreamConfig/EventStream/Get
Description	Get video channel stream information
Request Body	ChannelRequest JSON(show as follow Table-6.1.1)
Successful Response	Channel Information JSON (show as follow Table-6.1.2)

SET	
URL	POST/API/StreamConfig/MainStream/Set
	POST/API/StreamConfig/SubStream/Set
	POST/API/StreamConfig/MobileStream/Set
	POST/API/StreamConfig/EventStream/Set
Description	Set video channel stream information
Request Body	Channel Information JSON (show as follow Table-6.1.2)
Successful Response	The successful result response that described in 2.5

Table-6.1.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... "WIFI_CH1x" The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-6.1.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON Object	Channel Information JSON show as follow Table-6.1.x
channel_max		int	Total channels of current equipment
support_copy		bool	Whether the page supports copy (NVR and DVR only)
analog_max		int	Maximum number of device analog channels
bandwidth		object	Information shows as follow Table-6.1.5

Table-6.1.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-6.1.3

...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-6.1.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline", "Online"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
stream_type	"Normal","Alarm"	string	Stream type (A01 only)
max_frame_rate	20、25、30、50、60	int	Full frame rate (A01only)
resolution	"320 x 240", "640 x 480", "1280x720", "1280x960", "1920x1080", "2304x1296", "2048x1520", "2592x1520", "2592 x 1944", "3072 x 1728", "3840 x 2160"	string	resolution The resolution, frame rate and code rate are linked. If one of them transmits a value, the other two must follow. This linkage rule can be broken by passing the device capacity value breakStreamLink.
fps		int	Frame rate
video_encode_type	"H.264","H.265", "H.264+","H.265+", "MJPEG"	string	Video encoding type
bitrate_control	"CBR","VBR"	string	Bitrate type
video_quality	"Lowest","Lower","Low", "Medium","Higher","Highest"	string	Image quality Note: This parameter is only supported when Type is VBR option
bitrate_mode	"Predefined","UserDefined"	string	Bitrate mode
bitrate	8,16,24,32,48,64,80,96,128,160, 192,224,256,320,384,448,512,6 40,768, 896,1024,1280,1536,1792,2048, 3072,4096,5120,6144,7168,819 2,10240,12288,16384,100000	int	Preset bitrate value size
bitrate_default		JSON Object	JSON show as follow Table-6.1.4

custom_bitrate		int	Custom stream size
audio		bool	Audio switch.
i_frame_interval		int	I frame interval
i_frame_interval_rate		int	I frame interval is changed to x times the frame rate
etr		bool	Dynamic recording switch (NVR only)
etr_resolution	"320 x 240", "640 x 480", "1280x720", "1280x960", "1920x1080", "2304x1296", "2048x1520", "2592x1520", "2592 x 1944", "3072 x 1728", "3840 x 2160"	string	Dynamic recording resolution (NVR only)
etr_fps		int	Frame rate of dynamic video recording (NVR only)
etr_bitrate	8,16,24,32,48,64,80,96,128,160, 192,224,256,320,384,448,512,640,768, 896,1024,1280,1536,1792,2048, 3072,4096,5120,6144,7168,8192,10240,12288,16384,100000	int	Preset stream size of dynamic video (NVR only)
etr_custom_bitrate		int	Custom stream size for dynamic video recording (NVR only)
etr_stream_type	"Alarm"	string	Stream type of dynamic video recording (NVR only)
etr_video_encode_type	"H.264","H.265", "H.264+","H.265+", "MJPEG"	string	Video encoding type of dynamic video recording (NVR only)
etr_bitrate_control	"CBR","VBR"	string	Stream type of dynamic video (NVR only)
etr_video_quality	"Lowest","Lower","Low", "Medium","Higher","Highest"	string	Image quality of dynamic video recording (NVR only) Note: This parameter is only supported when Type is VBR option
etr_bitrate_mode	"Predefined","UserDefined"	string	Dynamic video stream mode (NVR only)
etr_audio		bool	Audio switch. (NVR only)
etr_i_frame_interval		int	frame rate interval of dynamic video recording (NVR only)
copy_ch	"digit""analog""wifi"	string	Flag supporting channel replication (NVR、DVR only)

enable_stream		bool	Mobilestream or the fourth stream - switch
video_encode_level	“Baseline” “MainProfile” “HighProfile”	string	Video coding level (IPC only) Note: H265 only has MainProfileMainProfile
resolution_mode	“SUPPORT_D1”, “SUPPORT_960H”, “SUPPORT_720P”, “SUPPORT_1080P”, “SUPPORT_3W”, “SUPPORT_720PCIF”, “SUPPORT_4W”, “SUPPORT_3W_TVI”, “SUPPORT_5W”, “SUPPORT_8W”, “SUPPORT_3W_HALF”, “SUPPORT_4W_HALF”, “SUPPORT_5W_HALF”, “SUPPORT_8WV2”, “SUPPORT_8WCVI”,	string	Simulate gun mode. The copy can only be made when the mode is consistent. Digital channel not supported
rtsp_enable		bool	Disable the IPC parameter settings when using the RTSP protocol.

Table-6.1.4 (ResolutionInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
“1280 * 720”		JSON Object	Resolution rate value, corresponding JSON show as follow Table-6.1.4.1
...		JSON Object	Resolution rate value, corresponding JSON show as follow Table-6.1.4.1

Table-6.1.4.1 (FpsInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
“1”		JSON Object	Frame rate value, corresponding toJSON show as follow Table-6.1.4.2
...		JSON Object	Frame rate value, corresponding toJSON show as follow Table-6.1.4.2

Table-6.1.4.2 (Bitrate DefaultInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
default_value	256,320,384,448,512,640,768, 896,1024,1280,1536,1792,2048,	int	Each fps corresponds to one bitrate default value.

	3072,4096,5120,6144,8192		
--	--------------------------	--	--

Table-6.1.5 (Bandwidth info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
total_bandwidth		int	Total bandwidth(unit: kpbs)
remaining_bandwidth		int	Remaining bandwidth(unit: kpbs)

6.2 Capture

GET	
URL	POST/API/StreamConfig/Capture/Get
Description	Get the channel capture parameters
Request Body	ChannelRequest JSON(show as follow Table-6.2.1)
Successful Response	Channel Information JSON (show as follow Table-6.2.2)

SET	
URL	POST/API/StreamConfig/Capture/Set
Description	Set the channel capture parameters
Request Body	Channel Information JSON (show as follow Table-6.2.2)
Successful Response	The successful result response that described in 2.5
Error_Code Response	Error Information JSON (show as follow Table-6.2.4) (auto_capture_chn_limit Return information attached to the error code)

Table-6.2.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-6.2.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as follow Table-6.2.x

Table-6.2.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-6.2.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-6.2.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
auto_capture	false:off, true:on	bool	Auto capture switch
stream_type	"Mainstream", "Substream"	string	Stream type
normal_interval	5,10,30,60,600,1800,3600	int	Normal type snapshot interval.. Unit: Second
alarm_interval		int	Alarm type snapshot interval. Unit: Second
alarm_resolution		string	Image resolution
alarm_quality	"Highest", "Higher", "Medium", "Low", "Lower", "Lowest"	string	Image quality
copy_ch	"digit""analog""wifi"	string	Flag supporting channel replication (NVR 、DVRonly)

Table-6.2.4(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
auto_capture_used	0~MAX_PARA_CH N_NUM	array	The channel number for which the automatic capture function is enabled
max_auto_capture_num	16	int	The maximum number of channels that can be opened for automatic capture

6.3 Rtsp Url

GET	
URL	POST/Preview/StreamUrl/Get
Description	Gets the url used to access the device RTSP Real-time streaming
Request Body	ChannelRequest JSON(show as follow Table-6.3.1)
Successful Response	Channel Information JSON (show as follow Table-6.3.2)

Table-6.3.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... "WIFI_CH1x" The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-6.3.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as follow Table-6.3.3

Table-6.3.3 single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... "WIFI_CH1x"	string	Channel number
mainstream_url	Max length: 63 byte	string	Mainstream URL
substream_url	Max length: 63 byte	string	Substream URL
mobilestream_url	Max length: 63 byte	string	Mobile stream URL

Example:

Request message:

POST/API/Preview/StreamUrl/Get HTTP/1.1

```
{
  "data": {
    "version": "1.0",
    "channel": ["CH1", "IP_CH1"]
  }
}
```

Response message:

HTTP/1.1 200 OK

```
{
  "result": "success",
  "data": {
    "channel_info": [
      {
        "channel": "CH1",
```



```

    "mainstream_url": "rtsp://172.16.10.169:80/rtsp/streaming?channel=1&subtype=0",
    "substream_url": "rtsp://172.16.10.169:80/rtsp/streaming?channel=1&subtype=1"
  },
  {
    "channel": "IP_CH1",
    "mainstream_url": "rtsp://172.16.10.169:80/rtsp/streaming?channel=17&subtype=0",
    "substream_url": "rtsp://172.16.10.169:80/rtsp/streaming?channel=17&subtype=1",
    "mobile_stream_url": "rtsp://172.16.10.169:80/rtsp/streaming?channel=17&subtype=2"
  }
]
}
}
}

```

7 ALARM

7.1 IO Alarm

GET	
URL	POST /API/AlarmConfig/IO/Get
Description	It is used to get the IO alarm parameters
Request Body	ChannelRequest JSON(show as follow Table-7.1.1)
Successful Response	Channel Information JSON (show as follow Table-7.1.2)

SET	
URL	POST /API/AlarmConfig/IO/Set
Description	It is used to set the IO alarm parameters
Request Body	Channel Information JSON (show as follow Table-7.1.2)
Successful Response	The successful result response that described in 2.5
Error_Code Response	Error Information JSON (show as follow Table-7.1.4) (cloud_video_upload_chn_limit Return information attached to the error code)

Table-7.1.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
alarm_in	“Local<-1” “IP_CH1<-1” The number of channels depends on the capabilities of the device.	String array	Each array bit represents a channel with a string.

Table-7.1.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON Object	Information JSON show as follow Table-7.1.x
channel_max		int	Total channels of current device

Table-7.1.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Local<-1		Json Object	JSON show as follow Table-7.1.3
...		Json Object	
IP_CH1<-1		Json Object	
...		Json Object	

Table-7.1.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
alarm_type	"NormallyOpen", "NormallyClose", "Off"	string	I/O alarm type
latch_time	"10", "20", "40", "60" or "5", "10", "20", "30"	string	Alarm out latch time
buzzer	"0","10","20","40", "60"	string	Buzzer time (NVR/DVR only)
alarm_out	“Local->1”... ”Local->x” “IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2” The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents aalarm output channel with a string.
post_recording	"30", "60", "120", "300" or "0", "5", "10", "20", "30"	string	Post recording time
send_email		bool	Send Email switch.
ftp_picture_upload		bool	Channel video FTP upload switch
picture_to_cloud		bool	Picture upload switch
video_to_cloud		bool	Video cloud upload switch (NVR/DVR

			only)
ftp_video_upload		bool	Channel video FTP upload switch (NVR/DVR only)
snap_enable		bool	Snap switch (A01 only)
show_message		bool	Show Message switch (NVR/DVR only)
full_screen		bool	FullScreen switch (NVR/DVR only)
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x”	array	Alarm channel switch Channel alarm linkage switch, each value represents a channel switch.
detect_interval	1-256	int	Alarm detection interval (front-end IO, A01 only)
voice_prompts_index	0~4294967295	int	File index (0~4294967295, 0 is None, that is, no audio file is selected). Each file is named as "index_file name" (1_i will try). When displayed on the page, "index_" should be hidden, such as "1_i will try", and only "i will try" should be displayed
voice_prompts_select		array	Playback channel, calculated by bit (bit0 is local, bit1 corresponds to front-end channel 1, bit2 corresponds to channel 2...)
voice_prompts_time		array	Time period, 12 time periods cannot have time conflicts

Table-7.1.4(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CHN_NUM	array	Channel number of cloud video upload function enabled
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	The maximum number of channels supported for cloud video upload

7.2 Motion Alarm

GET	
URL	POST /API/AlarmConfig/Motion/Get
Description	It is used to get the motion alarm parameters
Request Body	ChannelRequest JSON(show as follow Table-7.2.1)
Successful Response	Channel Information JSON (show as follow Table-7.2.2)

SET	
URL	POST /API/AlarmConfig/Motion/Set
Description	It is used to set the motion alarm parameters
Request Body	Channel Information JSON (show as follow Table-7.2.2)
Successful Response	The successful result response that described in 2.5
Error_Code Response	Error Information JSON (show as follow Table-7.2.4) (cloud_video_upload_chn_limit Return information attached to the error code)

Table-7.2.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.
page_type	“ChannelConfig”, “AlarmConfig” “AllConfig”	string	Data used to distinguish between the channel configuration page and the alarm configuration page. "AllConfig" is used when nvr gets and sets ipc parameters, which is more light than ChannelConfig_ Linkage field

Table-7.2.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON Object	Channel Information JSON show as follow Table-7.2.x
page_type	“ChannelConfig”, “AarmConfig”	string	It is used to distinguish the data of channel configuration page or alarm configuration page
channel_max		int	Total channels of current device
support_copy		bool	Whether support copy (NVR、DVR only)

Table-7.2.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.2.3
...		Json Object	
IP_CH1		Json Object	

...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.2.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline", "Online"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
switch		bool	Motion Switch.
mbcol	1 - 44(onvif)	int	Motion area is divided into row * col blocks Decided by camera, Set does not need
mbrow	1 - 30(onvif)	int	Motion area is divided into row * col blocks Decided by camera, Set does not need
copy_ch	"digit""analog""wifi"	string	Flag supporting channel replication (for NVR and DVR only)
maxrect	4 (A01 SDK)	int	Motion max rect num add (4) (A01 only)
curid	(A01 SDK)	int	Number of regions currently added
rect	(A01 SDK)	JSON array	Rect JSON show as follow Table-7.2.4
sensitivity	1 - 5、 1-8	int	Motion detection sensitivity
intervals	1-256	int	Motion alarm detection interval (s)
smart_moiton_detection		bool	
buzzer	"0","10","20","40", "60"	string	Buzzer time (NVR/DVR only)
alarm_out	"Local->1"... "Local->x" "IP_CH1->1"... "IP_CH1->2"... " IP_CHx->1" " IP_CHx->2" The number of channels depends on the capabilities of the device.	array	Alarm out channel Each array bit represents aalarm output channel with a string.
latch_time	"10","20","40", "60"	string	Alarm outut time
record_enable		bool	Record channel switch

record_channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x”	array	Alarm output channel Channel alarm linkage switch.
post_recording	"30","60","120","300"	string	Post recording time
region_setting		array	Currently, region is divided into 36 lines and 44 blocks/line. A bit describes a block in the line. Bit = 1: motion in this block is monitored.. Example: Region[0] = (FF-FF-FF-FF-FF-F0-00-00-00-00-00):: motion in channel 0 line 0's 44 blocks is monitored. Line 1's 44 blocks not monitored.
snap_enable		bool	Snap switch (A01 only)
show_message		bool	Show Message switch (NVR/DVR only)
send_email		bool	Send Email switch
full_screen		bool	FullScreen switch (NVR/DVR only)
ftp_picture_upload		bool	Channel snapshot FTP upload switch
ftp_video_upload		bool	Channel recording upload FTP switch (NVR/DVR only)
picture_to_cloud		bool	Picture upload cloud switch
video_to_cloud		bool	Video cloud upload switch (NVR/DVR only)
light_linkage		bool	The white light linkage switch, when the motion is triggered, triggers the white light linkage
multiple_switch	"Disable" "Motion" "PIR_and_Motion" "Person" "PIR_and_Person"	string	Motion switch with multiple options (A01 not used)
voice_prompts_index	0~4294967295	int	File index (0~4294967295, 0 is None, that is, no audio file is selected). Each file is named as "index_file name" (1_i will try). When displayed on the page, "index_" should be hidden, such as "1_i will try", and only "i will

			try" should be displayed
voice_prompts_select		array	Playback channel, calculated by bit (bit0 is local, bit1 corresponds to front-end channel 1, bit2 corresponds to channel 2...)
voice_prompts_time		array	Time period, 12 time periods cannot have time conflicts

Table-7.2.4(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CHN_NUM	array	Channel number of cloud video upload function enabled
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	The maximum number of channels supported for cloud video upload

7.3 PTZ Linkage

GET	
URL	POST /API/AlarmConfig/PTZLinkage/Get
Description	It is used to get the PTZ Linkage alarm parameters
Request Body	ChannelRequest JSON(show as follow Table-7.3.1)
Successful Response	Channel Information JSON (show as follow Table-7.3.2)

SET	
URL	POST /API/AlarmConfig/PTZLinkage/Set
Description	It is used to set the PTZ Linkage alarm parameters
Request Body	Channel Information JSON (show as follow Table-7.3.2)
Successful Response	The successful result response that described in 2.5

Table-7.3.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"...CH1x" "IP_CH1"...IP_CH1x" "WIFI_CH1"...WIFI_CH1x" The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-7.3.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_max		int	Max channel number
support_copy		bool	Whether support copy (NVR、DVR only)
channel_info		JSON Object	Channel Information JSON show as follow Table-7.3.x

Table-7.3.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.3.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.3.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
switch		bool	Channel enable switch
motion		bool	Motion alarm trigger switch
pir		bool	PIR linkage switch
io		array	IO alarm input switch If the value obtained by channel 0(1) is 1, it means that Local<-1(Local<-2) is checked on the interface, otherwise it is not checked.
ptz_info		array	Single Channel Information JSON show as follow Table-7.3.4
all_alarm		Json Object	Alarm type switch Table-7.3.5
copy_ch	"digit""analog" "wifi"	string	Flag supporting channel replication(NVR、DVR only)

Table-7.3.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ptz_chn	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x”	string	Channel alarm linkage The channel that the alarm is linked to.

linkage_ptz_point_index	0,1,2,3.....255	int	The pointing point corresponding to the channel of the alarm linkage. Note: The client needs to be compatible with two methods. The old method of type string has many redundant fields, which will cause a crash. Version 8.2.3 changed to int to solve the problem. The hexadecimal string in the structure indicates whether the option exists by bit, a total of 256bits. The small end storage, bit0 represents the None option, 1-255bit represents the 1-255 digital option, bit 1 represents the existence, and bit 0 represents the nonexistence (example:ptz_info.items.items.items[{"linkage_ptz_point_index": "1","linkage_ptz_point_index": "2"...}] Change to ptz_info.item.item: "0000...000")
ptz_switch		bool	Alarm linkage group number switch. (4group)

Table-7.3.5 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ptz_motion		bool	Motion alarm trigger switch
pir		bool	PIR linkage switch
io		array	I O alarm output switch If the value obtained by channel 0(1) is 1, it means that Local<-1(Local<-2) is checked on the interface, otherwise it is not checked.
linkage_lcd		bool	LCD linkage switch
linkage_pid		bool	PID linkage switch
linkage_sod		bool	SOD linkage switch

linkage_pd		bool	PD linkage switch
linkage_pdvd		bool	PDVD linkage switch
linkage_fd		bool	FD linkage switch
linkage_cc		bool	CC linkage switch
linkage_sound		bool	Sound linkage switch
linkage_vt		bool	Vt linkage switch
linkage_ad		bool	AD linkage switch
linkage_cd		bool	CD linkage switch
linkage_qd		bool	QD linkage switch
linkage_lpd		bool	LPD linkage switch
linkage_rsd		bool	RSD linkage switch
linkage_lpr		bool	LPR linkage switch
linkage_fr		bool	FR linkage switch
linkage_ai_pid		bool	AI PID linkage switch
linkage_ai_lcd		bool	AI LCD linkage switch
linkage_ai_pdvd		bool	AI PDVD linkage switch

7.4 Exception

GET	
URL	POST /API/AlarmConfig/Exception/Get
Description	It is used to get theException parameters
Request Body	None
Successful Response	Exception Information JSON(show as follow Table-7.4.1)

SET	
URL	POST /API/AlarmConfig/Exception/Set
Description	It is used to set the Exception parameters
Request Body	Exception Information JSON(show as follow Table-7.4.1)
Successful Response	The successful result response that described in 2.5

Table-7.4.1 (ExceptionInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
exception_info		Json Object	JSON show as follow Table-7.4.2

Table-7.4.2 (ExceptionInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
video_loss		Json Object	JSON show as follow Table-7.4.3

disk_error		Json Object	JSON show as follow Table-7.4.3
no_space_on_disk		Json Object	JSON show as follow Table-7.4.3
fan_abnormal		Json Object	JSON show as follow Table-7.4.3

Table-7.4.3(Exception Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
switch		bool	Enable switch. (A01 only)
buzzer	"0","10","20", "40","60"	string	Buzzer chirping time.
alarm_out	“Local->1”...” Local->x” “IP_CH1->1”...” IP_CH1->x” “IP_CHx->1”...” IP_CHx->1”	array	alarm output channel switch Each array bit represents aalarm output channel with a string.
latch_time	"10","20","40","60"	string	Alarm output latch time Alarm output, output time:
show_message		bool	Show Message switch (A01 only)
send_email		bool	Send Email switch
voice_prompts_index	0~4294967295	int	File index (0~4294967295, 0 is None, that is, no audio file is selected). Each file is named as "index_file name" (1_i will try). When displayed on the page, "index_" should be hidden, such as "1_i will try", and only "i will try" should be displayed
voice_prompts_select		array	Playback channel, calculated by bit (bit0 is local, bit1 corresponds to front-end channel 1, bit2 corresponds to channel 2...)
voice_prompts_time		array	Time period, 12 time periods cannot have time conflicts

7.5 PIR

GET	
URL	POST /API/AlarmConfig/PIR/Get
Description	It is used to get the pir config parameters
Request Body	See Table -7.5.1
Successful Response	Channel Information JSON (show as follow Table-7.5.2)

SET

URL	POST /API/AlarmConfig /PIR/Set
Description	It is used to set the pir parameters
Request Body	Channel Information JSON (show as follow Table-7.5.2)
Successful Response	Like example
Error_Code Response	Error Information JSON (show as follow Table-7.5.4) (cloud_video_upload_chn_limit Return information attached to the error code) Examples show as follow 2.5

Table-7.5.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig” “AllConfig”	string	Data used to distinguish between channel configuration page and alarm configuration page "AllConfig" is used when nvr gets and sets ipc parameters, which is lighter than ChannelConfig_Linkage field

Table-7.5.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON array	Information JSON show as follow Table-7.5.x
channel_max		int	Total channels of current equipment
support_copy		bool	Whether the page supports copy (NVR and DVR only)
page_type	“ChannelConfig”, “AarmConfig”	string	It is used to distinguish the data of channel configuration page or alarm configuration page

Table-7.5.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.5.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	

WIFI_CH1		Json Object	
...		Json Object	

Table-7.5.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
switch		bool	pir switch, false: close true: open
sensitivity	1 - 8	int	Pirdetection sensitivity
mbcol	1 - 44	int	Pir area is divided into row * col blocks Decided by camera, Set does not need Keep consistent with motion
mbrow	1 - 30	int	Pir area is divided into row * col blocks Decided by camera, Set does not need Keep consistent with motion
region_setting	200*8	int array	The Pir area is divided into row * col blocks, and each bit describes the value of a block (8 bits per byte) Bit = 1: motion in this block is monitored. Example: region_setting [0] = (FF-FF-FF-FF-FF-F0-00- 00-00-00-00): motion in channel 0 line 0's 44 blocks is monitored. Line 1's 44 blocks not monitored.
detect_area	8	json array	detect_area JSON show as follow Table-7.5.5 Note: The polygon area of the simulation channel is used
copy_ch	"digit""analog""wifi"	string	Flag supporting channel replication (NVR、 DVR only)
light_linkage		bool	White light linkage switch, when PIR is triggered, white light is triggered
buzzer	"0","10","20","40", "60"	string	Buzzer beeping time
alarm_out	"Local->1"... "Local->x" "IP_CH1->1"... "IP_CH1->2"... "IP_CHx->1" "IP_CHx->2"	array	Alarm output time Each array bit represents aalarm output channel with a string.

	The number of channels depends on the capabilities of the device.		
latch_time	"10","20","40","60"	string	Alarm output latch time, IPC value 5 10 20 30
record_enable		bool	Record channel switch
record_channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x”	array	Alarm output channel Channel alarm linkage switch.
post_recording	"30","60","120", "300" or “5” “10” “20” “30”	string	Post recording time
show_message		bool	Show Message switch
send_email		bool	Send Email switch
full_screen		bool	FullScreen switch
ftp_picture_upload		bool	Channel snapshot FTP upload switch
ftp_video_upload		bool	Channel video FTP upload switch
picture_to_cloud		bool	Picture upload switch
video_to_cloud		bool	Video cloud upload switch
voice_prompts_index	0~4294967295	int	File index (0~4294967295, 0 is None, that is, no audio file is selected). Each file is named as "index_file name" (1_i will try). When displayed on the page, "index_" should be hidden, such as "1_i will try", and only "i will try" should be displayed
voice_prompts_select		array	Playback channel, calculated by bit (bit0 is local, bit1 corresponds to front-end channel 1, bit2 corresponds to channel 2...)
voice_prompts_time		array	Time period, 12 time periods cannot have time conflicts

Table-7.5.4(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_use d	0~MAX_PARA_CH N_NUM	array	Channel number of cloud video upload function enabled
max_cloud_video_u pload_num	MAX_CLOUD_VID EO_RECORD_NUM	int	The maximum number of channels supported for cloud video upload

Table-7.5.5(detect_area JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
x	0~704	int	Abscissa
y	0~576	int	Ordinate

7.6 Flood-light

GET	
URL	POST /API/AlarmConfig/Deterrence/Get
Description	It is used to get the flood light parameters
Request Body	ChannelRequest JSON(show as follow Table-7.6.1)
Successful Response	Channel Information JSON (show as follow Table-7.6.2)

SET	
URL	POST /API/AlarmConfig/Deterrence/Set
Description	It is used to set the flood light parameters
Request Body	Channel Information JSON (show as follow Table-7.6.2)
Successful Response	Like example

Default	
URL	POST /API/AlarmConfig/Deterrence/Default
Description	It is used to set the flood light parameters
Request Body	Channel Information JSON (show as follow Table-7.6.2)
Successful Response	Like example

Table-7.6.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. (NVR only)

Table-7.6.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as follow

			Table-7.6.x
--	--	--	-------------

Table-7.6.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.6.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.6.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
flood_light_switch	true false	bool	White light switch, false: close true: open
flood_light_mode	"Warninglight" "Strobe"	String	White light mode, Warninglight: always on, Strobe: flashing And determinence_ The mode is different, flood_light_Mode indicates the way the white light lights up when the white light is triggered_ Mod indicates the picture color of white light in night vision mode
flood_light_value	1 - 100	int	White brightness
bright_time	5 - 180	int	White light duration
strobe_frequency	"Low","Middle", "High"	String	Enabled in flashing mode, 0: low; 1:middle; 2:high
sensitivity	1 - 8	int	8 grades: 8-high, 7-high 1-Low Sensitivity of area detection triggering white light
siren_switch	true false	bool	Horn switch false: close true: open
siren_value	1 - 10	int	Horn volume
siren_time	5 - 180	int	Horn duration
color_image_ctrl		bool	0: Off, 1: On When it is turned on, when white light is triggered, IRCUT will be forced to operate in daytime mode. When it is not turned on, IRCUT will not respond to the effect of photosensitivity.

region_setting	200*8	array	The region is divided into row * col blocks, and each bit describes the value of a block The row and col of white light are subject to motion. Region: white light is triggered only when motion is in this regionBit = 1: motion in this block is monitored. Example: region_setting [0] = (FF-FF-FF-FF-FF-F0-00-00-00-00-00):: motion in channel 0 line 0's 44 blocks is monitored. Line 1's 44 blocks not monitored.
mbcol	44	Int	The area is divided into row * col block
mbrow	30	Int	The area is divided into row * col block
dualtalk_volume	1 - 10	int	Intercom volume control.
enforcer_light_switch	true false	bool	Red and blue light switch false: close true: open
enforcer_bright_time	5 - 180	int	Duration of red and blue lights
time_schedule		JSON array	Time_schedule JSON show as follow Table-7.6.4
deterrence_mode	" Normal "," Full Color "," Alarm "	string	Use in the white light night vision mode of the wireless battery program:, 1:Normal; 2:Full Color;3: Alarm. Has difference with flood_light_mode. flood_light_Mode indicates the way the white light lights up when the white light is triggered_ Mod indicates the picture color of white light in night vision mode

Table-7.6.4(Time_schedule JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
schedule_type	"Deterrence"	string	White led schedule
week		JSON array	WeekJSON show as follow Table-7.6.5

Table-7.6.5(WeekJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
day	Sun,Mon,Tue,Wed, Thu,Fri,Sat	string	Identify the day of the week
time	0: Close time period 1: Open this time period	array	Each array bit (int) identifies half an hour.

7.7 Sound Detection

GET	
URL	POST /API/AlarmConfig/Intelligent/SoundDetection/Get
Description	It is used to get the SoundDetectionconfig parameters
Request Body	See Table -7.7.1
Successful Response	Channel Information JSON (show as follow Table-7.7.2)

SET	
URL	POST /API/AlarmConfig/Intelligent/SoundDetection /Set
Description	It is used to set theSoundDetection parameters
Request Body	Channel Information JSON (show as follow Table-7.7.2)
Successful Response	Like example
Error_Code Response	Error Information JSON (shown in Table-7.7.6) (return information accompanied by the cloud_video_upload_chn_limit error code)

Table-7.7.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	Data used to distinguish between channel configuration page and alarm configuration page

Table-7.7.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_max		int	Maximum number of channels
support_copy		bool	Whether the page supports copy (NVR and DVR only)
channel_info		JSON object	Channel Information JSON show as follow Table-7.7.x
page_type	“ChannelConfig”, “AarmConfig”	string	It is used to distinguish the data of channel configuration page or alarm configuration page

Table-7.7.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.7.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.7.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", "Nonsupport"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
switch	true, false	bool	Sound Detection switch, false: close true: open
rise_switch	true, false	bool	Sound increases detection switch
rise_sensitivity	1 - 100	int	Sound detection threshold sensitivity (only for sound increase detection)
sound_intensity	1 - 100	int	Sound increases detection sensitivity
decline_switch	true, false	bool	Sound reduction detection switch
decline_sensitivity	1 - 100	int	Sound reduces sensitivity
copy_ch	"digit""analog""wifi"	string	Flag supporting channel replication (NVR and DVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer beep time (NVR only)
alarm_out	"Local->1"... "Local->x" "IP_CH1->1"... "IP_CH1->2"... " IP_CHx->1" " IP_CHx->2" The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents aalarm output channel with a string.
latch_time	"10","20","40", "60"	string	Alarm output latch time
record_enable	true, false	bool	Record channel switch
record_channel	"CH1"... "CH1x"	array	Alarm output channel

	“IP_CH1”...” IP_CH1x” “WIFI_CH1”...”		Channel alarm linkage switch.
post_recording	"30","60","120", "300"	string	Post recording time
show_message		bool	Show Message switch (NVR only)
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Channel snapshot FTP upload switch (NVR only)
ftp_video_upload	true, false	bool	Channel video FTP upload switch (NVR only)
picture_to_cloud	true, false	bool	Picture upload switch (NVR only)
video_to_cloud	true, false	bool	Video cloud upload switch (NVR only)
full_screen		bool	Full Screen switch (NVR only)
time_schedule		JSON array	Time_schedule JSON show as follow Table-7.7.4
voice_prompts_index	0~4294967295	int	File index (0~4294967295, 0 is None, that is, no audio file is selected). Each file is named as "index_file name" (1_i will try). When displayed on the page, "index_" should be hidden, such as "1_i will try", and only "i will try" should be displayed
voice_prompts_select		array	Playback channel, calculated by bit (bit0 is local, bit1 corresponds to front-end channel 1, bit2 corresponds to channel 2...)
voice_prompts_time		array	Time period, 12 time periods cannot have time conflicts

Table-7.7.4(Time_schedule JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
schedule_type	“SD”	string	Sound alarm time schedule
week		JSON array	WeekJSON show as follow Table-7.7.5

Table-7.7.5(WeekJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
day	Sun,Mon,Tue,Wed, Thu,Fri,Sat	string	Identify the day of the week
time	0: Close time period 1: Open this time period	array	Each array bit (int) identifies half an hour.

Table-7.7.6(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CHN_NUM	array	Channel number of cloud video upload function enabled
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	The maximum number of channels supported for cloud video upload

7.8 Occlusion Detection

GET	
URL	POST /API/AlarmConfig/Intelligent/OcclusionDetection/Get
Description	It is used to get the OcclusionDetectionconfig parameters
Request Body	See Table -1.2.1
Successful Response	Channel Information JSON (show as follow Table-7.8.2)

SET	
URL	POST /API/AlarmConfig/Intelligent/OcclusionDetection/Set
Description	It is used to set theOcclusionDetection parameters
Request Body	Channel Information JSON (show as follow Table-7.8.2)
Successful Response	Like example
Error_Code Response	Error Information JSON (show as follow Table-7.8.4) (cloud_video_upload_chn_limit Return information attached to the error code)

Table-7.8.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	String array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	Data used to distinguish between channel configuration page and alarm configuration page

Table-7.8.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

channel_max		int	Maximum number of channels
support_copy		bool	Whether the page supports copy (NVR and DVR only)
channel_info		JSON Object	Channel Information JSON show as follow Table-7.8.x
page_type	“ChannelConfig”, “AarmConfig”	string	It is used to distinguish the data of channel configuration page or alarm configuration page

Table-7.8.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.8.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.8.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", “Nonsupport”	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
switch	true, false	bool	Occlusion Detection switch, false: close true: open
sensitivity	1-6	int	Lens occlusion alarm detection sensitivity
copy_ch	"digit""analog""wifi"	string	Flag supporting channel replication (NVR and DVR only)
alarm_out	“Local->1”... ”Local->x” “IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2” The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents a alarm output channel with a string.
latch_time	"10","20","40", "60"	string	Alarm latch time

record_enable	true, false	bool	Record channel switch (NVR only)
record_channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...”	array	Alarm output channel (NVR only) Channel alarm linkage switch.
post_recording	"30","60","120", "300"	string	Post recording latch time (NVR only)
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Channel snapshot FTP upload switch (NVR only)
ftp_video_upload	true, false	bool	Channel video FTP upload switch (NVR only)
picture_to_cloud	true, false	bool	Picture upload switch (NVR only)
video_to_cloud	true, false	bool	Video cloud upload switch (NVR only)
full_screen		bool	Full Screen switch (NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer time (NVR only)
show_message		bool	Show Message switch (NVR only)
voice_prompts_index	0~4294967295	int	File index (0~4294967295, 0 is None, that is, no audio file is selected). Each file is named as "index_file name" (1_i will try). When displayed on the page, "index_" should be hidden, such as "1_i will try", and only "i will try" should be displayed
voice_prompts_select		array	Playback channel, calculated by bit (bit0 is local, bit1 corresponds to front-end channel 1, bit2 corresponds to channel 2...)
voice_prompts_time		array	Time period, 12 time periods cannot have time conflicts

Table-7.8.4(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_use d	0~MAX_PARA_CH N_NUM	array	Channel number of cloud video upload function enabled
max_cloud_video_u pload_num	MAX_CLOUD_VID EO_RECORD_NUM	int	The maximum number of channels supported for cloud video upload

7.9 Perimeter Intrusion Detection

GET	
URL	POST /API/AlarmConfig/Intelligent/PID/Get
Description	It is used to get the PID config parameters

Request Body	See Table -7.9.1
Successful Response	Channel Information JSON (show as follow Table-7.9.2)

SET	
URL	POST /API/AlarmConfig/Intelligent/PID/Set
Description	It is used to set thePID parameters
Request Body	Channel Information JSON (show as follow Table-7.9.2)
Successful Response	Like example
Error_Code Response	Error Information JSON (show as follow Table-7.9.7) (Cloud_video_upload_chn_limit Return information attached to the error code)

Table-7.9.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	String array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	Data used to distinguish between channel configuration page and alarm configuration page

Table-7.9.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as follow Table-7.9.x
page_type	“ChannelConfig”, “AarmConfig”	string	It is used to distinguish the data of channel configuration page or alarm configuration page
channel_max		int	Maximum number of channels
support_copy		bool	Whether the page supports copy (NVR、DVR only)

Table-7.9.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.9.3
...		Json Object	

IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.9.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online" , "Nonsupport"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
alarm_out	"Local->1"... "Local->x" "IP_CH1->1"... "IP_CH1->2"... " IP_CHx->1" " IP_CHx->2" The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents aalarm output channel with a string. Camera: Local->1: On, Null: Off
latch_time	"10","20","40", "60"	string	Alarm output time
record_enable	true, false	bool	Record channel switch
record_channel	"CH1"... "CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"...	array	Alarm output channel Channel alarm linkage switch.
post_recording	"30","60","120", "300"	string	Recording delay time
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Channel snapshot FTP upload switch (NVR only)
ftp_video_upload	true, false	bool	Channel video FTP upload switch (NVR only)
picture_to_cloud	true, false	bool	Picture upload switch (for NVR)
video_to_cloud	true, false	bool	Video cloud upload switch (for NVR)
full_screen		bool	FullScreen switch (NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer beep time (NVR only)

show_message		bool	Show Message switch (NVR only)
switch	true, false	bool	PID enable switch, false: close true: open
detection_type	"Off" "Pedestrian" "Vehicle" "Pedestrian &Vehicle"	string	Detection type, used by intelligent second-generation IPC of humanoid and vehicular)
sensitivity	1-4	int	detection sensitivity
iva_lines	0 - No line drawn 1 - Line drawn	bool	Whether there will be dashes in preview and playback (used by intelligent second-generation IPC)
scene	"Indoor", "Outdoor"	string	/*Scenario 0 indoor 1 outdoor */(used by intelligent generation IPC)
rule_info		JSON object	Single Channel Information JSON show as follow Table-7.9.4
copy_ch	"digit""analog""wifi"	string	Flag supporting channel replication (for NVR and DVR only)
voice_prompts_index	0~4294967295	int	File index (0~4294967295, 0 is None, that is, no audio file is selected). Each file is named as "index_file name" (1_i will try). When displayed on the page, "index_" should be hidden, such as "1_i will try", and only "i will try" should be displayed
voice_prompts_select		array	Playback channel, calculated by bit (bit0 is local, bit1 corresponds to front-end channel 1, bit2 corresponds to channel 2...)
voice_prompts_time		array	Time period, 12 time periods cannot have time conflicts

Table-7.9.4 (rule_info Information JSON)

rule_number1		Object	Rule No. 1, Information JSON show as follow Table-7.9.5
rule_number2		Object	Rule No.2, Information JSON show as follow Table-7.9.5
rule_number3		Object	Rule No.3, Information JSON show as follow Table-7.9.5
rule_number4		Object	Rule No.4, Information JSON show as follow Table-7.9.5

Table-7.9.5 (rule_number Information JSON)

rule_switch		bool	Rule switch
rule_type	"A->B", "B->A",	string	Rule type

	"A<->B"		
rule_rect		Object	rectangle, Information JSON show as follow Table-7.9.6

Table-7.9.6 (Channel Information JSON)

x1	0-704	short	x1 Coordinate point
y1	0-576	short	y1 Coordinate point
x2	0-704	short	x2 Coordinate point
y2	0-576	short	y2 Coordinate point
x3	0-704	short	x3 Coordinate point
y3	0-576	short	y3 Coordinate point
x4	0-704	short	x4 Coordinate point
y4	0-576	short	y4 Coordinate point

Table-7.9.7(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CHN_NUM	array	Channel number of cloud video upload function enabled
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	The maximum number of channels supported for cloud video upload
analog_channel_is_used	0~MAX_ANALOG_CHN	array	Analog channel number of smart switch turned on
max_intelligent_analog_channel_num	1	int	The maximum number of analog channels supported by the intelligent switch

Example:

Request message:

POST /API/AlarmConfig/Intelligent/PID/Get HTTP/1.1

```
{
  "version": "1.0"
  "channel":["IP_CH2"]
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "result": "success",
```

```

"data": {
  "channel_info": {
    "CH1": {
      "status": "Online",
      "switch": false,
      "sensitivity": 2,
      "detection_type": "Off",
      "latch_time": "5",
      "post_recording": "5",
      "alarm_out": [],
      "send_email": false,
      "record_enable": false,
      "rule_info": {
        "rule_number1": {
          "rule_switch": false,
          "rule_type": "A->B",
          "rule_rect": {
            "x1": 0,
            "y1": 0,
            "x2": 0,
            "y2": 0,
            "x3": 0,
            "y3": 0,
            "x4": 0,
            "y4": 0
          }
        },
        "rule_number2": {
          "rule_switch": false,
          "rule_type": "A->B",
          "rule_rect": {
            "x1": 0,
            "y1": 0,
            "x2": 0,
            "y2": 0,
            "x3": 0,
            "y3": 0,
            "x4": 0,
            "y4": 0
          }
        },
        "rule_number3": {
          "rule_switch": false,
          "rule_type": "A->B",

```


Error_Code Response	Error Information JSON (show as follow Table-7.10.7) (cloud_video_upload_chn_limit (response info of error code.))
---------------------	---

Table-7.10.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	the data to identify for Channel Config or Alarm Config

Table-7.10.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as follow Table-7.10.x
page_type	“ChannelConfig”, “AarmConfig”	string	In Set, the data to identify for Channel Config or Alarm Config
channel_max		int	Maximum channel number
support_copy		bool	The page supports copy or not(NVR、DVR only)

Table-7.10.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.10.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.10.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", “Nonsupport”	string	Channel online status, only for IP channel. Note: when channel is online, no such string.

alarm_out	<p>“Local->1”... ”Local->x” “IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2”</p> <p>The number of channels depends on the capabilities of the device.</p>	array	<p>Alarm output channel</p> <p>Each array bit represents a alarm output channel with a string.</p> <p>Camera: Local->1:enable, no value: disable</p>
latch_time	"10","20","40", "60"	string	Alarm latching time
record_enable	true, false	bool	Record channel switch
record_channel	<p>“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...”</p>	array	<p>Alarm output channel</p> <p>Channel alarm linkage switch.</p>
post_recording	"30","60","120","300"	string	Post recording time
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Chapture picture in channel. FTP upload switch, (NVR only)
ftp_video_upload	true, false	bool	Channel recording FTP upload switch, (NVR only)
picture_to_cloud	true, false	bool	Picture upload switch (NVR only)
video_to_cloud	true, false	bool	Video upload to cloud switch, (NVR only)
full_screen		bool	FullScreen switch, (NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer time (NVR only)
show_message		bool	Show Message switch, (NVR only)
switch	true, false	bool	swtich, false: close true: open
sensitivity	1-4	int	Sensitivity of detection
detection_type	<p>"Off" "Pedestrian" "Vehicle" "Pedestrian &Vehicle"</p>	string	Detection type, Pedestrian or Vehicle.
iva_lines	0 : no lines 1: show lines	int	In liveview and playback shows lines or not.
rule_info		JSON object	Single Channel Information JSON show as follow Table-7.10.4

copy_ch	"digit""analog""wifi"	string	Symbol of supporting channel copy.(NVR、DVR only)
voice_prompts_index	0~4294967295	int	File index(0~4294967295, 0 is None, no chosen voice file.)。Each file is named“index_file name” (1_i will try) , when show in the page, it needs to hide the “index_” .For example, “1_i will try” , it will show “i will try” only.
voice_prompts_select		array	Play channel, (bit0: local, bit1: camera ch1, bit2: camera ch2...)
voice_prompts_time		array	Voice prompts time, The 12 time ranges cannot conflict with each other.

Table-7.10.4 (rule_info Information JSON)

rule_number1		Object	Rule no. 1, Information JSON show as follow Table-7.9.5
rule_number2		Object	Rule no. 2, Information JSON show as follow Table-7.9.5
rule_number3		Object	Rule no. 3, Information JSON show as follow Table-7.9.5
rule_number4		Object	Rule no. 4, Information JSON show as follow Table-7.9.5

Table-7.10.5 (rule_number Information JSON)

rule_switch		bool	Rule switch
rule_type	"A->B", "B->A", "A<->B"	string	Rule type
rule_line		Object	Each channel has a maximum of 64 point coordinates, that is, 32 lines. If it is companion line, only the first line is used. If it is a perimeter, the 8 lines are connected in turn to form a closed figure Information JSON show as follow Table-7.10.6

Table-7.10.6 (rule_line Information JSON)

x1	0-704	short	x1 coordinate point
y1	0-576	short	y1 coordinate point
x2	0-704	short	x2 coordinate point
y2	0-576	short	y2 coordinate point

Table-7.10.7(Error Information JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARAM_NUM	array	Channel number that enable the cloud video upload.
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	supporting maximum channel number of cloud video upload.
analog_channel_is_used	0~MAX_ANALOG_CHN	array	analog channel is enable the switch
max_intelligent_analog_channel_num	1	int	Maximum number of enabled analog channels supported by the smart switch

Example:

Request message:

POST /API/AlarmConfig/Intelligent/LCD/Get HTTP/1.1

```
{
  "version": "1.0"
  "channel":["IP_CH2"]
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "IP_CH2": {
        "status": "Online",
        "switch": false,
        "sensitivity": 2,
        "latch_time": "5",
        "post_recording": "10",
        "send_email": false,
        "record_enable": false,
        "alarm_out": [],
        "rule_info": {
          "rule_number1": {
            "rule_switch": false,
            "rule_type": "A->B",
            "rule_line": {
              "x1": 0,
              "y1": 0,
              "x2": 0,
              "y2": 0
            }
          }
        }
      }
    }
  }
}
```

```

    },
    "rule_number2": {
      "rule_switch": false,
      "rule_type": "A->B",
      "rule_line": {
        "x1": 0,
        "y1": 0,
        "x2": 0,
        "y2": 0
      }
    },
    "rule_number3": {
      "rule_switch": false,
      "rule_type": "A->B",
      "rule_line": {
        "x1": 0,
        "y1": 0,
        "x2": 0,
        "y2": 0
      }
    },
    "rule_number4": {
      "rule_switch": false,
      "rule_type": "A->B",
      "rule_line": {
        "x1": 0,
        "y1": 0,
        "x2": 0,
        "y2": 0
      }
    }
  }
}

```

7.11 Stationary Object Detection

GET	
URL	POST /API/AlarmConfig/Intelligent/SOD/Get
Description	It is used to get the SOD config parameters
Request Body	See Table -7.11.1

Successful Response	Channel Information JSON (show as follow Table-7.11.2)
---------------------	--

SET	
URL	POST /API/AlarmConfig/Intelligent/SOD/Set
Description	It is used to set theSOD parameters
Request Body	Channel Information JSON (show as follow Table-7.11.2)
Successful Response	Like example
Error_Code Response	Error Information JSON (show as follow Table-7.11.7) (cloud_video_upload_chn_limit r (esponse info of error code)

Table-7.11.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC no need CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	the data to identify for Channel Config or Alarm Config

Table-7.11.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as follow Table-7.11.x
page_type	“ChannelConfig”, “AarmConfig”	string	In Set, the data to identify for Channel Config or Alarm Config
channel_max		int	Maximum channel number
support_copy		bool	The page supports copy or not(NVR、DVR only)

Table-7.11.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.11.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.11.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", "Nonsupport"	string	Channel online status, only for IP channel. Note: when channel is online, no such string.
alarm_out	"Local->1"... "Local->x" "IP_CH1->1"... "IP_CH1->2"... "IP_CHx->1" "IP_CHx->2" The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents a alarm output channel with a string. Camera: Local->1:enable, no value: disable
latch_time	"10","20","40", "60"	string	Alarm latching time
record_enable	true, false	bool	Record channel switch
record_channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"...	array	Alarm output channel Channel alarm linkage switch.
post_recording	"30","60","120", "300"	string	Post recording time
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Chapture picture in channel. FTP upload switch, (NVR only)
ftp_video_upload	true, false	bool	Channel recording FTP upload switch, (NVR only)
picture_to_cloud	true, false	bool	Picture upload switch (NVR only)
video_to_cloud	true, false	bool	Video upload to cloud switch, (NVR only)
full_screen		bool	FullScreen switch, (NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer time (NVR only)
show_message		bool	Show Message switch, (NVR only)
switch	true, false	bool	swtich, false: close true: open
sensitivity	1-4、0-100	int	Sensitivity of detection
snap_mode	"Default" "RealTimeMode"	string	IntervalMode JSON show as follow Table-7.12.4

	"IntervalMode"		
min_pixel	64-1080	int	Minimum pixel value
max_pixel	320-1080	int	Maximum pixel value
detection_type	"Pedestrian" "Vehicle" "Pedestrian &Vehicle"	string	Detection type, Pedestrian or Vehicle.
detection_mode	"StaticMode" "MotionMode"	string	Detection mode
detection_range	"FullScreen" "Customize"	string	Detection range
rule_info		JSON array	Single Channel Information JSON show as follow Table-7.11.4
iva_lines	0 : no lines 1: show lines	int	In liveview and playback shows lines or not.
scene	"Indoor","Outdoor"	string	/* scene 0: indoor 1:outdoor*/(IPC first generation of AI)
copy_ch	"digit""analog""wifi"	string	Symbol of supporting channel copy.(NVR、DVR only)
voice_prompts_index	0~4294967295	int	File index(0~4294967295, 0 is None, no chosen voice file.)。 Each file is named“index_file name” (1_i will try) , when show in the page, it needs to hide the “index_” . For example, “1_i will try” , it will show “i will try” only.
voice_prompts_select		array	Play channel, (bit0: local, bit1: camera ch1, bit2: camera ch2...)
voice_prompts_time		array	Voice prompts time, The 12 time ranges cannot conflict with each other.

Table-7.11.4 (rule_info Information JSON)

rule_number1		Object	Rule no. 1, Information JSON show as follow Table-7.11.5
rule_number2		Object	Rule no. 2, Information JSON show as follow Table-7.11.5
rule_number3		Object	Rule no. 3, Information JSON show as follow Table-7.11.5
rule_number4		Object	Rule no. 4, Information JSON show as follow Table-7.11.5

Table-7.11.5 (rule_number Information JSON)

rule_rect		Object	Each channel has a maximum of 64 point coordinates, that is, 32 lines. If it is companion line, only the first line is used. If it is a perimeter, the 8 lines are connected in turn to form a closed figure Information JSON show as follow Table-7.11.6
rule_switch		bool	Rule switch
rule_type	“Legacy” “Lost” “Lost & Legacy”	string	Rule type

Table-7.11.6 (rule_line Information JSON)

x1	0-704	short	x1 coordinate point
y1	0-576	short	y1 coordinate point
x2	0-704	short	x2 coordinate point
y2	0-576	short	y2 coordinate point
x3	0-704	short	x3 coordinate point
y3	0-576	short	y3 coordinate point
x4	0-704	short	x4 coordinate point
y4	0-576	short	y4 coordinate point

Table-7.11.7(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CHN_NUM	array	Channel number that enable the cloud video upload.
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	supporting maximum channel number of cloud video upload.
analog_channel_is_used	0~MAX_ANALOG_CHN	array	analog channel is enable the switch
max_intelligent_analog_channel_num	1	int	Maximum number of enabled analog channels supported by the smart switch

Example:

Request message:

POST /API/AlarmConfig/Intelligent/SOD/Get HTTP/1.1

```
{
  "version": "1.0"
  "channel":["IP_CH2"]
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "CH1": {
        "status": "Online",
        "switch": false,
        "sensitivity": 2,
        "latch_time": "5",
        "post_recording": "10",
        "send_email": false,
        "record_enable": false,
        "alarm_out": [],
        "rule_info": {
          "rule_number1": {
            "rule_switch": false,
            "rule_type": "Legacy",
            "rule_rect": {
              "x1": 0,
              "y1": 0,
              "x2": 0,
              "y2": 0,
              "x3": 0,
              "y3": 0,
              "x4": 0,
              "y4": 0
            }
          },
          "rule_number2": {
            "rule_switch": false,
            "rule_type": "Legacy",
            "rule_rect": {
              "x1": 0,
              "y1": 0,
              "x2": 0,
              "y2": 0,
              "x3": 0,
              "y3": 0,
              "x4": 0,
              "y4": 0
            }
          }
        }
      }
    }
  }
}
```


Description	It is used to set the PD parameters
Request Body	Channel Information JSON (show as follow Table-7.12.2)
Successful Response	Like example
Error_Code Response	Error Information JSON (show as follow Table-7.12.8) (cloud_video_upload_chn_limit response info of error code.)

Table-7.12.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	the data to identify for Channel Config or Alarm Config

Table-7.12.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Single Channel Information JSON show as follow Table-7.12.x
page_type	“ChannelConfig”, “AarmConfig”	string	In Set, the data to identify for Channel Config or Alarm Config
channel_max		int	Maximum channel number

Table-7.12.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.12.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.12.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

status	"Offline","Online" , "Nonsupport"	string	Channel online status, only for IP channel. Note: when channel is online, no such string.
alarm_out	"Local->1"... "Local->x" "IP_CH1->1"... "IP_CH1->2"... "IP_CHx->1" "IP_CHx->2" The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents a alarm output channel with a string. Camera: Local->1:enable, no value: disable
latch_time	"10","20","40", "60"	string	Alarm latching time, IPC: 5 10 20 30
record_enable	true, false	bool	Record channel switch
record_channel	"CH1"... "CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"...	array	Alarm output channel Channel alarm linkage switch.
post_recording	"30","60","120", "300"	string	Post recording time, IPC: 0 5 10 20 30
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Chapture picture in channel. FTP upload switch, (NVR only)
ftp_video_upload	true, false	bool	Channel recording FTP upload switch, (NVR only)
picture_to_cloud	true, false	bool	Picture upload switch (NVR only)
video_to_cloud	true, false	bool	Video upload to cloud switch, (NVR only)
full_screen		bool	FullScreen switch, (NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer time (NVR only)
show_message		bool	Show Message switch, (NVR only)
switch	true, false	bool	swtich, false: close true: open
is_ai_param	true, false	bool	Identify the normal Human detection or Intelligent Human detection (NVR only)
level	"Low" "Middle" "High"	string	Sensitivity of detection
sensitivity	0-100	int	Sensitivity of detection
snap_mode	"Default"	string	Snapshot mode,

	"RealTimeMode" "IntervalMode"		IntervalMode JSON show as follow Table-7.12.4
min_pixel	64-1080	int	Minimum pixel value
max_pixel	320-1080	int	Maximum pixel value
detection_type	"Pedestrian" "Vehicle" "Pedestrian &Vehicle"	string	Detection type, Pedestrian or Vehicle.
detection_mode	"StaticMode" "MotionMode"	string	Detection mode
detection_range	"FullScreen" "Customize"	string	Detection range
iva_lines	0 - No line drawn 1 - Line drawn	int	In liveview and playback shows lines or not.
rule_info		Object	Information JSON show as follow Table-7.12.5
voice_prompts_index	0~4294967295	int	File index(0~4294967295, 0 is None, no chosen voice file.). Each file is named“index_file name” (1_i will try) , when show in the page, it needs to hide the “index_” .For example, “1_i will try” , it will show “i will try” only.
voice_prompts_select		array	Play channel, (bit0: local, bit1: camera ch1, bit2: camera ch2...)
voice_prompts_time		array	Voice prompts time, The 12 time ranges cannot conflict with each other.
copy_ch	"digit""analog""wifi"	string	Symbol of supporting channel copy (NVR、DVR only)

Table-7.12.4 (snap interval mode Information JSON)

snap_num	"1" "2" "3" "Unlimited"	string	Snapshot number
snap_frequency		int	Snapshot frequency, unit: s/pic

Table-7.12.5 (rule_info Information JSON)

rule_number1		Object	Rule no. 1, Information JSON show as follow Table-7.12.6
--------------	--	--------	--

Table-7.12.6 (rule_number Information JSON)

rule_switch		bool	Rule switch
-------------	--	------	-------------

rule_type	"Normal"	string	Rule type
rule_rect		Object	rectangle, Information JSON show as follow Table-7.12.7

Table-7.12.7 (Channel Information JSON)

x1	0-704	short	x1 coordinate point
y1	0-576	short	y1 coordinate point
x2	0-704	short	x2 coordinate point
y2	0-576	short	y2 coordinate point
x3	0-704	short	x3 coordinate point
y3	0-576	short	y3 coordinate point
x4	0-704	short	x4 coordinate point
y4	0-576	short	y4 coordinate point

Table-7.12.8(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CH N_NUM	array	Channel number that enable the cloud video upload.
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	supporting maximum channel number of cloud video upload.

Example:

Request message:

POST /API/AlarmConfig/Intelligent/PD/Get HTTP/1.1

```
{
  "version": "1.0"
  "channel":["IP_CH2"]
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "IP_CH2": {
        "status": "Online",
        "switch": false,
        "sensitivity": "High",
        "alarm_out": [],
        "rule_info": {
          "rule_number1": {
```


	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	the data to identify for Channel Config or Alarm Config

Table-7.13.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as follow Table-7.13.x
page_type	“ChannelConfig”, “AarmConfig”	string	In Set, the data to identify for Channel Config or Alarm Config
channel_max		int	Maximum channel number

Table-7.13.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.13.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.13.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", “Nonsupport”	string	Channel online status, only for IP channel. Note: when channel is online, no such string.
alarm_out	“Local->1”... ”Local->x” “IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2”	array	Alarm output channel Each array bit represents aalarm output channel with a string. Camera: Local->1:enable, no value: disable

	The number of channels depends on the capabilities of the device.		
latch_time	"10","20","40", "60"	string	Alarm latching time Ipc value 5 10 20 30
record_enable	true, false	bool	Record channel switch
record_channel	"CH1"... "CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"..."	array	Alarm output channel Channel alarm linkage switch.
post_recording	"30","60","120", "300"	string	Post recording time Ipc value 0 5 10 20 30
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Chapture picture in channel. FTP upload switch, (NVR only)
ftp_video_upload	true, false	bool	Channel recording FTP upload switch, (NVR only)
picture_to_cloud	true, false	bool	Picture upload switch (NVR only)
video_to_cloud	true, false	bool	Video upload to cloud switch, (NVR only)
full_screen		bool	FullScreen switch, (NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer time (NVR only)
show_message		bool	Show Message switch, (NVR only)
switch	true, false	bool	swtich, false: close true: open
is_ai_param	true, false	bool	Identify normal AI and deep learning AI (NVR only)
face_enhance		bool	face_enhance switch
detection_mode	"StaticMode" "MotionMode"	string	Detection mode
snap_mode	" RealTimeMode " "OptimalMode" "IntervalMode"	string	Snapshot mode IntervalMode JSON show as follow Table-7.13.4
apply_mode	"FrontalView" "MultiAngle" "Customize"	string	Customize JSON show as follow Table-7.13.5
min_pixel	32-1080	int	Minimum pixel value
max_pixel	320-1080	int	Maximum pixel value

iva_lines	0 - No line drawn 1 - Line drawn	int	In liveview and playback shows lines or not.
rule_info		Object	Information JSON show as follow Table-7.13.6
voice_prompts_index	0~4294967295	int	File index(0~4294967295, 0 is None, no chosen voice file.). Each file is named“index_file name” (1_i will try) , when show in the page, it needs to hide the “index_” . For example, “1_i will try” , it will show “i will try” only.
voice_prompts_select		array	Play channel, (bit0: local, bit1: camera ch1, bit2: camera ch2...)
voice_prompts_time		array	Voice prompts time, The 12 time ranges cannot conflict with each other.
copy_ch	"digit""analog""wifi"	string	Symbol of supporting channel copy.(NVR、DVR only)

Table-7.13.4 (snap interval mode Information JSON)

snap_num	"1" "2" "3" "Unlimited"	string	Snapshot numbers
snap_frequency		int	Snapshot frequency, unit/pic

Table-7.13.5 (apply_mode Information JSON)

roll_range	0-180	int	
pitch_range	0-180	int	
yaw_range	0-180	int	
picture_quality	0-100	int	

Table-7.13.6 (rule_info Information JSON)

rule_number1		Object	Rule no. 1, Information JSON show as follow Table-7.13.7
--------------	--	--------	--

Table-7.13.7 (rule_number Information JSON)

rule_switch		bool	Rule switch
rule_type	"Normal" "A->B" "B->A"	string	Rule type
rule_kind	"Rect" "Line"	string	Rule type
detection_range	"FullScreen" "Customize"	string	Detection range

rule_rect		Object	Rectangle, Information JSON show as follow Table-7.13.8
rule_line		Object	Line, Information JSON show as follow Table-7.13.9

Table-7.13.8 (Channel Information JSON)

x1	0-704	short	x1 coordinate point
y1	0-576	short	y1 coordinate point
x2	0-704	short	x2 coordinate point
y2	0-576	short	y2 coordinate point
x3	0-704	short	x3 coordinate point
y3	0-576	short	y3 coordinate point
x4	0-704	short	x4 coordinate point
y4	0-576	short	y4 coordinate point

Table-7.13.9 (rule_line Information JSON)

x1	0-704	short	x1 coordinate point
y1	0-576	short	y1 coordinate point
x2	0-704	short	x2 coordinate point
y2	0-576	short	y2 coordinate point

Table-7.13.10(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CHN_NUM	array	Channel number that enable the cloud video upload.
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	supporting maximum channel number of cloud video upload.

Example:

Request message:

POST /API/AlarmConfig/Intelligent/FD/Get HTTP/1.1

```
{
  "version": "1.0"
  "channel":["IP_CH2"]
}
```

Response message:

```
{
  "result": "success",
```

```

"data": {
  "channel_info": {
    "CH1": {
      "status": "Online",
      "switch": false,
      "alarm_out": [],
      "post_recording": "0",
      "send_email": false,
      "record_enable": false,
      "rule_info": {
        "rule_number1": {
          "rule_switch": true,
          "rule_type": "Normal",
          "rule_rect": {
            "x1": 30,
            "y1": 30,
            "x2": 30,
            "y2": 545,
            "x3": 675,
            "y3": 545,
            "x4": 675,
            "y4": 30
          }
        }
      }
    }
  }
}

```

7.14 Cross Counting

GET	
URL	POST /API/AlarmConfig/Intelligent/CC/Get
Description	It is used to get the CC config parameters
Request Body	See Table -7.14.1
Successful Response	Channel Information JSON (show as follow Table-7.14.2)

SET	
URL	POST /API/AlarmConfig/Intelligent/CC/Set
Description	It is used to set the CC parameters
Request Body	Channel Information JSON (show as follow Table-7.14.2)

Successful Response	Like example
Error_Code Response	Error Information JSON (show as follow Table-7.14.7) (cloud_video_upload_chn_limit response info of error code.)

Table-7.14.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	the data to identify for Channel Config or Alarm Config

Table-7.14.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON array	Single Channel Information JSON show as follow Table-7.14.x
page_type	“ChannelConfig”, “AarmConfig”	string	In Set, the data to identify for Channel Config or Alarm Config
channel_max		int	Maximum channel number

Table-7.14.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.14.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.14.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline", "Online", “Nonsupport”	string	Channel online status, only for IP channel. Note: when channel is online, no such string.

alarm_out	<p>“Local->1”... ”Local->x” “IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2”</p> <p>The number of channels depends on the capabilities of the device.</p>	array	<p>Alarm output channel</p> <p>Each array bit represents a alarm output channel with a string.</p> <p>Camera: Local->1:enable, no value: disable</p>
latch_time	"10","20","40", "60"	string	Alarm latching time Ipc value 5 10 20 30
record_enable	true, false	bool	Record channel switch
record_channel	<p>“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...”</p>	array	<p>Alarm output channel</p> <p>Channel alarm linkage switch.</p>
post_recording	"30","60","120", "300"	string	Post recording time, Ipc value 0 5 10 20 30
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Chapture picture in channel. FTP upload switch, (NVR only)
ftp_video_upload	true, false	bool	Channel recording FTP upload switch, (NVR only)
picture_to_cloud	true, false	bool	Picture upload switch (NVR only)
video_to_cloud	true, false	bool	Video upload to cloud switch, (NVR only)
full_screen		bool	FullScreen switch, (NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer time (NVR only)
show_message		bool	Show Message switch (NVR only)
switch	true, false	bool	swtich, false: close true: open
sensitivity	1-4	int	Sensitivity of detection
iva_lines	0 - No line drawn 1 - Line drawn	int	In liveview and playback shows lines or not.
voice_prompts		int	Voice prompts (NVR only)
rule_info		Object	Information JSON show as follow Table-7.14.4
voice_prompts_index	0~4294967295	int	File index(0~4294967295, 0 is None, no chosen voice file.) Each file is named “index_file name” (1_i will try) , when show in the page, it needs to hide the “index_” .For example, “1_i will

			try” , it will show “i will try” only.
voice_prompts_select		array	Play channel, (bit0: local, bit1: camera ch1, bit2: camera ch2...)
voice_prompts_time		array	Voice prompts time, The 12 time ranges cannot conflict with each other.
copy_ch	"digit""analog""wifi"	string	Symbol of supporting channel copy.(NVR、DVR only)

Table-7.14.4 (rule_info Information JSON)

rule_number1		Object	Rule no. 1, Information JSON show as follow Table-7.14.5
--------------	--	--------	--

Table-7.14.5 (rule_number Information JSON)

rule_switch		bool	Rule switch
rule_type	"A->B", "B->A",	string	Rule type
rule_line		Object	rectangle,Information JSON show as follow Table-7.14.6

Table-7.14.6 (Channel Information JSON)

x1		short	x1 coordinate point
y1		short	y1 coordinate point
x2		short	x2 coordinate point
y2		short	y2 coordinate point

Table-7.14.7(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CH N_NUM	array	Channel number that enable the cloud video upload.
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	supporting maximum channel number of cloud video upload.

7.15 Intelligent Analysis

GET	
URL	POST /API/Intelligent/IntelligentAnalysis/Get
Description	getIntelligent Analysis information
Request Body	ChannelRequest JSON(show as follow Table-7.15.1)
Successful Response	Analysis Information JSON (show as follow Table-7.15.2)

SET	
URL	POST /API/Intelligent/IntelligentAnalysis/Set

Description	search and get Intelligent Analysis' data
Request Body	Channel Information JSON (show as follow Table-7.15.2)
Successful Response	Analysis data JSON (show as follow Table-7.15.5)

Table-7.15.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... "WIFI_CH1x" The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC not need

Table-7.15.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON Object	Channel Information JSON show as follow Table-7.15.x
channel_max		int	Current maximum channels

Table-7.15.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.15.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.15.3 (alarmtype request Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Perimeter_Intrusion_Detection		Json object	PID alarm type JSON show as follow Table-7.15.4 Only used by IPC
Line_Crossing_Detection		Json object	LCD alarm type JSON show as follow Table-7.15.4 Only used by IPC
Stationary_Object_Detection		Json object	SOD alarm type JSON show as follow Table-7.15.4 Only used by IPC

Pedestrian_Detection		Json object	PD alarm type JSON show as follow Table-7.15.4 Only used by IPC
Cross_Counting		Json object	CC alarm type JSON show as follow Table-7.15.4 Only used by IPC
report_type	"Daily report", "weekly report", "Monthly report", "Annual report",	string	Report type
cross_type	"Number of in" "Number of out"	string	Cross type,DVR/NVR need
search_date	10	string	Search date
ai_cross_count		bool	Identify 1 st , 2 nd generation of CC

Table-7.15.4 (singal type Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
statistical_type	PID & LCD: Number of A -> B Number of B -> A Number of A <-> B SOD Number of legacy Number of lost Number of legacy & lost PD: Number CC: Number of in Number of out Number of in & out	string	Statistical type. Only used by IPC

Table-7.15.5 (analysis data JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON Object	Channel Information JSON show as follow Table-7.15.6

Table-7.15.6 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.15.7
...		Json Object	
IP_CH1		Json Object	
...		Json Object	

WIFI_CH1		Json Object	
...		Json Object	

Table-7.15.7 (analysis data JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
report		unsigned int array	Statistical report. The array length is divided into daily report 24, weekly report 7, monthly report 0 to 31, and annual report 12. The value of each digit (0-65535) indicates the number of statistics collected in the period.

7.16 Voice Prompt

7.16.1 Description

GET	
URL	POST /API/AlarmConfig/VoicePrompts/Get
Description	It is used to get the voice prompt config parameters
Request Body	Request JSON(show as follow Table-7.16.2.1)
Successful Response	Response Information JSON (show as follow Table-7.16.2.2)

SET	
URL	POST /API/AlarmConfig/VoicePrompts/Set
Description	It is used to operate the voice prompt function
Request Body	Request JSON(show as follow Table-7.16.2.4)
Successful Response	Response Information JSON (show as follow Table-7.16.2.2)

7.16.2 Syntax

Table-7.16.2.1 (Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
control_type	"Normal", "AiFace" "AiPlate"	string	Control type Normal: non Alface and non AI plate AiFace: AI face

			AiPlate: AI plate
download_mode	"g711a" "aac" "wav" "mp4" "mp3"	string	Remote audio file format to download when listening
license_plate		string	When the control_type is AiPlate, it must use this string. Otherwise no need to use this string. In menu it's "License Plate"

Table-7.16.2.2 (Response JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
info		json object	JSON show as follow Table-7.16.2.3
audio_list		json array	Audio file list

Table-7.16.2.3 (analysis data JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
language		string	Current system language, used to cut into the network server when importing text default language
ipc_channel_info		json array	The IPC supports a list of channels for language output
url		string	The url for IPC to audition the audio file. For example http://IP:Port/VoicePromptsTransform/api/FileName , IPC needs to replace the IP、Port、FileName. FileName is the request file+file format of audio_lis. The file format is released when request download_mode (http://127.0.0.1:80/VoicePromptsTransform/api/test.mp3)

Table-7.16.2.1 (Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

command	"Upload" "Remove" "Transform" "Exit" "Play"	string	Control command Upload: uploads an audio file or text Remov: Deletes an audio file Transform: Transform the audio file (If the remote end does not support G711A decoding during the listening trial, run this command to ask the board to convert the audio file to the required format. Note that if this command has been run, when exiting the audio broadcast page, run the Exit command to let the board release resources) Exit: tells the board to release resources occupied by the Transform command (if the Transform command is not delivered, this command is not required when exiting the audio broadcast page) Play: Play an audio locally on the board
control_type	" Normal " " AiFace " "AiPlate"	string	Control type Normal: non AI face and AI plate AiFace: AI face AiPlate: AI plate During the commend of "Upload" "Remove" "Transform" "Play"to place the tring
convert_mode	"File" "LocalText" "NetworkText"	string	Convert mode File: convert to audio file LocalText: local convert text, NetworkText: network convert text In "Upload" commend to place
packet_index	0~65535	int	command : "Upload" and convert_mode: "File" to release at the same time, means the index of current file packet.
file_count	0~65535	int	command : "Upload" and convert_mode: "File" to release at the same time, means the total file packet.

file_name	1~64	string	command : "Upload" and convert_mode: "File" to release at the same time, means the upload audio file name.
file_data	control_type: "Normal" - 1~5M control_type: "AiFace"- 1~500K	string	command : "Upload" and convert_mode: "File" to release at the same time, means the update audio file content. control_type: " Normal "file size is 1~5M。 control_type: " AiFace " the whole file should be 1~500K (each single file should not over 1.5M)
text	1~1024	string	command : "Upload" and convert_mode: "File" to release at the same time, means the text
index	1~11	int	command : "Upload" and convert_mode: "File" to release at the same time, means to export in that time. command : " Remove "、 control_type: " Normal " is released, means to delete the file index of the delete audio list. command : " Remove "、 control_type: " AiFace " released, means the audio to delete. command : " Transform "、 control_type: " Normal " released, means the index of the file. command : " Transform "、 control_type: " AiFace "released, means switch to which time period. command : " Play " control_type: " Normal " released, means to play the index of that file. command : " Play " control_type: " AiFace " released, means to play the audio of that time period.

language		string	command : "Upload"、 convert_mode: "Text"released, means the text is this language.
download_mode	"g711a" "aac" "wav" "mp4"	string	command : " Transform " release, means to convert the index relative audio format.
ipc_channel_info		json array	command : " Play " release, means to play with which channel. 255 means locally to the responding IPC channel. Start from 0.
license_plate		string	When control_type is AiPlate, must use this string, for the others no need to use. In the interface it's "License Plate"

7.16.3 Example

7.16.3.1 RequestGet Audio Files List

Request message:

POST/API/AlarmConfig/VoicePrompts/Get

```
{
  "data": {
    "command": "GetAudioFilesList ",
  }
}
```

Response message:

HTTP/1.1 200 OK

```
{
  "result": "success",
  "data": {
    "audio_list": [
      "5_13227.mp3", // Each item in the list is divided into two parts by underscores (_). The first part is
a unique serial number assigned by the system to the audio file, and the second part is an audio file name.
"6_9528.mp3",
      "7_AlienBoi.mp3"
    ]
  }
}
```

7.16.3.2 Request Upload Audio Files

Request message:

POST/API/AlarmConfig/VoicePrompts/Set

```
{
  "data": {
    "command": "UploadAudioFile",
    "file_name": "9528.mp3",
    "file_data": [0,73,1,68,2,51,3,3,4,0,5,0,6,0,7,0,8,34,9,69,10
,84,11,73,12,84,13,51,14,0,15,0,16,0,17,24,18,0,19,0,20,...]
  }
}
```

Response message:

HTTP/1.1 200 OK

```
{
  "result": "success",
  "data": { }
}
```

7.16.3.3 Request Remove Audio Files

Request message:

POST/API/AlarmConfig/VoicePrompts/Set

```
{
  "data": {
    "command": "RemoveAudioFile",
    "audio_index": 5
  }
}
```

Response message:

HTTP/1.1 200 OK

```
{
  "result": "success",
  "data": {
    "audio_list": [
      "6_9528.mp3",
      "7_AlienBoi.mp3"
    ]
  }
}
```

7.17 Combination Alarm

GET	
URL	POST /API/AlarmConfig/Combination/Get
Description	It is used to get the combination alarm config parameters
Request Body	See Table -7.17.1
Successful Response	Channel Information JSON (show as follow Table-7.17.2)

SET	
URL	POST /API/AlarmConfig/Combination/Set
Description	It is used to set the combination alarm parameters
Request Body	Channel Information JSON (show as follow Table-7.17.2)
Successful Response	The successful result response that described in 2.5

Table -7.17.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC not need

Table -7.17.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Single Channel Information JSON show as follow Table-7.17.3

Table-7.17.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.17.4
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.17.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", "Nonsupport"	string	Channel online status, only for digital channels. Note: This field does not exist when the channel is online
enable_alarm	"Disable" "Enable"	string	Combination alarm switch
combination_configure		Json array	Combination alarm configure JSON combination show as follow Table-7.17.5
alarm_out	"Local->1"... "Local->x" "IP_CH1->1"... "IP_CH1->2"... "IP_CHx->1" "IP_CHx->2" The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents a alarm output channel with a string. Camera: Local->1:enable, no value: disable
latch_time	"10","20","40", "60"	string	Alarm output channel
record_enable	true, false	bool	Record channel switch
record_channel	"CH1"... "CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"...	array	Alarm output channel Channel alarm linkage switch.
post_recording	"30","60","120", "300"	string	Post recording time
send_email	true, false	bool	Send Email switch
full_screen		bool	FullScreen switch (NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer time (NVR only)
show_message		bool	Show Message switch, (NVR only)
ftp_picture_upload		bool	Switch for snapshot upload to FTP
ftp_video_upload		bool	Switch for uploading recording to FTP (NVR/DVR only)
picture_to_cloud		bool	Switch for loading picture
video_to_cloud		bool	Switch for loading video (NVR/DVR only)
voice_prompts_index		array	File index(0~4294967295, 0 is None, no chosen voice file.) Each file is named "index_file name" (1_i will try), when show in the page, it needs to hide the "index_". For example, "1_i will try", it will show "i will try" only.

voice_prompts_select		array	Play channel, (bit0: local, bit1: camera ch1, bit2: camera ch2...)
voice_prompts_time		Json array	Voice prompts time, The 12 time ranges cannot conflict with each other. JSON voice prompts time show as follow Table-7.17.6
copy_ch	"digit""analog""wif i"	string	Symbol of supporting channel copy.(NVR、DVR only)

Table -7.17.5 (Combination Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
alarm_type	"AT_MOTION" "AT_PIR" "AT_IO" "AT_PID" "AT_LCD" "AT_SOD" "AT_PD&VD" "AT_FaceAttr" "AT_FD" "AT_CC" "AT_CD" "AT_QD" "AT_LPD" "AT_RSD" "AT_Sound" "AT_VT"	string	Combination alarm type
alarm_source	"IP Camera" "Analog Channels " " Local->1"... " Local->x"	string	Alarm source
support_ipc_io	true false	bool	IPC channel supports IO or not

Table -7.17.6 (Voice prompts time Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
start_hour	0~23	int	Start time h
start_minute	0~59	int	Start time m
start_second	0~59	int	Start time s
end_hour	0~23	int	End time h
end_minute	0~59	int	End time m
end_second	0~59	int	End time s

Example:

Request message:

POST /API/AlarmConfig/Combination/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "page_type": "AlarmConfig",
```



```
    "channel":["CH1"]
  }
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "CH1": {
        "enable_alarm": "Enable",
        "combination_configure": [{
          "alarm_type": "AT_MOTION",
          "alarm_source": "IP Camera",
          "support_ipc_io": true
        }],
        {
          "alarm_type": "AT_PIR",
          "alarm_source": "IP Camera",
          "support_ipc_io": true
        }
      ],
      "buzzer": "0",
      "alarm_out": [],
      "latch_time": "10",
      "record_enable": true,
      "record_channel": ["CH1"],
      "post_recording": "30",
      "show_message": true,
      "send_email": true,
      "full_screen": false,
      "ftp_picture_upload": true,
      "ftp_video_upload": false,
      "picture_to_cloud": true,
      "video_to_cloud": false,
      "voice_prompts_index": [
        155,
        0,
        0,
        0,
        0,
        0,
        0,
        0,
        0,
        0,
        0
      ]
    }
  }
}
```

```
    0,
    0,
    0,
    0
  ],
  "voice_prompts_select": [
    1,
    0,
    0,
    0
  ],
  "voice_prompts_time": [{
    "start_hour": 0,
    "start_minute": 0,
    "start_second": 0,
    "end_hour": 23,
    "end_minute": 59,
    "end_second": 59
  },
  {
    "start_hour": 0,
    "start_minute": 0,
    "start_second": 0,
    "end_hour": 23,
    "end_minute": 59,
    "end_second": 59
  },
  {
    "start_hour": 0,
    "start_minute": 0,
    "start_second": 0,
    "end_hour": 23,
    "end_minute": 59,
    "end_second": 59
  },
  {
    "start_hour": 0,
    "start_minute": 0,
    "start_second": 0,
    "end_hour": 23,
    "end_minute": 59,
    "end_second": 59
  }
]
```


	“IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.		string.
page_type	“FloodLight”, “Siren”	string	Schedule for identify combination alarm type

Table-7.18.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON Object	Channel Information JSON show as follow Table-7.18.x
page_type	“FloodLight”, “Siren”	string	In Set, Schedule for identify combination alarm type

Table-7.18.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.18.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.18.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
schedule		JSON array	JSON show as follow Table-7.18.4

Table-7.18.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
schedule_type	“Motion”, “IO” “PIR”, “FD” “PVD”, “PID” “LCD”, “SOD”	string	Alarm schedule type


```
    },
    {
        "day": "Mon",
        "time":
[1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1]
    },
    {
        "day": "Tue",
        "time":
[1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1]
    },
    {
        "day": "Wed",
        "time":
[1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1]
    },
    {
        "day": "Thu",
        "time":
[1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1]
    },
    {
        "day": "Fri",
        "time":
[1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1]
    },
    {
        "day": "Sat",
        "time":
[1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1]
    },
    ]
}
]
}
}
}
```

7.19 Serial Transport

SET

URL	POST/API/ AlarmConfig/SerialTransport/Set
Description	Trigger the alarm of external alarm device
Request Body	Serial Transport JSON (show as follow Table-7.19.1)
Successful Response	none

Table-7.19.1(Serial Transport JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
serial_info		Json Object	SetialInfo JSON show as follow Table-7.19.2

Table-7.19.2(SerialInfoJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
data_type		int	Data type. Use to identify the operation of 232/485 alarm device
data_size		int	Data size
data_buf		string	get the commend

7.20 IO-AlarmStatus

GET	
URL	POST/API/ AlarmConfig/IOAlarmStatus/Get
Description	Get the local alarm output IO status
Request Body	none
Successful Response	IO-AlarmStatus JSON(show as follow Table-7.20.1)

SET	
URL	POST/API/ AlarmConfig/IOAlarmStatus/Set
Description	Set the local alarm output IO status
Request Body	IO-AlarmStatus JSON(show as follow Table-7.20.3)
Successful Response	none

Table-7.20.1(IO-AlarmStatus JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ioalarm_states		JSON array	IoAlarm_States JSON show as follow Table-7.20.2

Table-7.20.2(IoAlarm_States JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ioalarm_states		int	Statue value of alarm IO port

Table-7.20.3(IO-AlarmStatus JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ioalarm_info		Json Object	IoAlarm_Info JSON show as follow Table-7.20.4

Table-7.20.4(IoAlarm_Info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ioalarm_id		int	IO alarm channel
ioalarm_state		int	IO alarm status value

7.21 Disarming

GET	
URL	POST/API/AlarmConfig/Disarming/Get
Description	Get disarming status of DVR/NVR
Request Body	none
Successful Response	Disarming JSON(show as follow Table-7.21.1)

SET	
URL	POST/API/ AlarmConfig/Disarming/Set
Description	Set disarming status of DVR/NVR
Request Body	Disarming JSON(show as follow Table-7.21.1)
Successful Response	none

Table-7.21.1(Disarming JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
disarming		bool	disarming switch
action		Json Object	Action JSON show as follow Table-7.21.2
disarming_channel	“CH1”...x” “IP_CH1”...x” “WIFI_CH1”...x”	array	disarming channel
channel_info		Json Object	Single Channel Information JSON show as follow Table-7.21.3

Table-7.21.2(action JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
buzzer		bool	buzzer switch

alarm_out		bool	alarm_out switch
show_message		bool	show_message switch
send_email		bool	send_email switch
full_screen		bool	full_screen switch
voice_prompts		bool	voice_prompts switch

Table-7.21.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-7.21.4
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-7.21.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
disarming_schedule		JSON array	Time_schedule JSON show as follow Table-7.21.5

Table-7.21.5 (Time_schedule JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
schedule_type	“Disarming”	string	disarming schedule
week		JSON array	WeekJSON show as follow Table-7.21.6

Table-7.21.6(WeekJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
day	Sun,Mon,Tue,Wed, Thu,Fri,Sat	string	Mark the date
time	0: close this time 1: enable this time	array	Each array bit (int) identifies half an hour.

Example:

Request message:

POST /API/AlarmConfig/Disarming/Get HTTP/1.1

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

Response message:

Success:

```
{
  "result": "success",
  "data": {
    "disarming": false,
    "action": {
      "buzzer": false,
      "alarm_out": false,
      "show_message": false,
      "send_email": false,
      "full_screen": false,
      "voice_prompts": false
    },
    "disarming_channel": ["CH1", "CH2"],
    "channel_info": {
      "CH1": {
        "disarming_schedule": [{
          "schedule_type": "Disarming",
          "week": [{
            "day": "Sun",
            "time": [1, 1, 1, ..., 1, 1, 1]
          }, {
            "day": "Mon",
            "time": [1, 1, 1, ..., 1, 1, 1]
          }, {
            "day": "Tue",
            "time": [1, 1, 1, ..., 1, 1, 1]
          }, {
            "day": "Wed",
            "time": [1, 1, 1, ..., 1, 1, 1]
          }, {
            "day": "Thu",
            "time": [1, 1, 1, ..., 1, 1, 1]
          }, {
            "day": "Fri",
            "time": [1, 1, 1, ..., 1, 1, 1]
          }, {
            "day": "Sat",
            "time": [1, 1, 1, ..., 1, 1, 1]
          }
        ]
      }
    },
    "CH2": {...},
    "CH3": {...},
  }
}
```


	RANGE	TYPE	
siren_switch		bool	Siren function switch
siren_time		int	Siren latching time
siren_value		int	Siren alarm voice volume
siren_type	“Alarm1” “Alarm2” “User-defined1” “User-defined2” “User-defined3”	string array	Siren alarm audio type
siren_file_name	0-127	string	Siren file name
siren_file	0-256k	string	show the import audio plug-in file name or not
siren_file_type	0-127	string	Siren file type
siren_support_form at	".wav" ".pcm"	string array	Siren supports file form

Table-7.22.2(VoiceAlarm JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
siren_switch		bool	Siren switch
siren_time	5-180	int	Siren latching time
siren_value	1-10	int	Siren alarm voice volume
siren_type_content		Json Object	siren_type_content JSON show as follow Table-7.22.3
siren_type	“Alarm1” “Alarm2” “User-defined1” “User-defined2” “User-defined3”	string array	Siren alarm audio type
siren_file_name	0-127	string	Siren file name

Table-7.22.3(siren_type_content JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
delete_button		bool	delete/import audio button
siren_file_name	0-127	string	Siren file name
siren_file	0-256k	string	show the import audio plug-in file name or not

Table-7.22.6(VoiceAlarm JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
siren_type	“Alarm1” “Alarm2” “User-defined1”	string array	Siren audio type (“Alarm1”, “Alarm2” Are system audio)

	“User-defined2” “User-defined3”		
file_name	0-127	string	Alarm file name
file_data	0-256k	string	Audio file data
siren_file_type	0-127	string	Audio file type

Table-7.22.7(VoiceAlarm JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
siren_type	“Alarm1” “Alarm2” “User-defined1” “User-defined2” “User-defined3”	string array	Siren audio type (“Alarm1”, “Alarm2” Are system audio can’t be deleted)

8 STORAGE

8.1 Disk

8.1.1DiskConfiguration

GET	
URL	POST/API/StorageConfig/Disk/Get
Description	It is used to get the Disk parameters
Request Body	none
Successful Response	Storage JSON(show as follow Table-8.1.1)

SET	
URL	POST/API/StorageConfig/Disk/Set
Description	It is used to set the Disk parameters
Request Body	Storage JSON(show as follow Table-8.1.1)
Successful Response	The successful result response that described in 2.5

Table-8.1.1.1(StorageJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
disk_info		JSON array	Disk InformationJSON(show as follow Table-

			8.1.2)
over_write	"Off", "Auto", "1Day", "3Days", "7Days", "14Days", "30Days", "90Days"	string	Overwrite type
esata_record		bool	E-SATA record function switch
hdd_format_type	"AllHddData" "OnlyHddRecord", "OnlyHddData",	string	HDD format type /*format whole HDD*/ /*only format the record field*/ /*only format the general data field*/

Table-8.1.1.2(Disk Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
id		int	HDD no.
display_id		int	HDD no.(NVR、DVR only)
model	Max length: 40byte	string	HDD model
serial_no	Max length: 20byte	string	HDD serial no.
firmware	Max length: 8byte	string	HDD firmware model
device_type	"None", "Normal", "Esata", "Sd", "Usb", "Network", "Raid" "All"	string	HDD type
active	"Inactive","Active"	string	HDD working status
status	"NoHdd", "Unformat", "Normal", "Full", "ReadOnly", "HddError", "Connecting", "Offline", "Occupied", "Oversized", "Broken",	string	HDD status

	"Degrade", "Rebuilding", "Backup", "RaidHdd"		
total_size	Unit: Megabytes	int	Total size
free_size	Unit: Megabytes	int	Free size
total_time	Unit: Megabytes	int	Total time
free_time	Unit: Megabytes	int	Free time
format_enable		bool	If it can be format or not
delete_enable		bool	If it can be deleted or not
disk_type	"ReadAndWriteDisk", "RedundantDisk", "ReadOnlyDisk"	string	Disk type
disk_group_id		int	disk group no.
NasMaxCount		int	Max count NAS HDD
nas_info		Json Object	JSON show as follow Table-8.1.1.2

Table-8.1.1.2 (NetHdd Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Enable		int	Switch
Index		int	NetHDD no.
Type	"NFS", "SMB/CIFS"	string	Protocol type (NFS SMB/CIFS)
Username		string	User name
Password		string	Password
password_empty		bool	If there is password or not
Ip		string	IP
Dir		string	Direction route
Size		Int	HDD size (GB)
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

8.1.2 Disk Control

Control	
URL	POST/API/StorageConfig/Disk/Control
Description	It is used to control the Disk parameters
Request Body	Request JSON (show as follow Table-8.1.2.1)
Successful Response	The successful result response that described in Table-8.1.2.3

Table-8.1.2.1 (NetHdd Information JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
Type		string	NetHdd、 RaidHdd、 Hdd
Info		Json Object	JSON show as follow Table-8.1.2.2(NetHdd) JSON show as follow Table-8.1.2.3(Hdd)

Table-8.1.2.2 (NetHdd Control JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Mode		string	Add、 Test、 Del
Enable		int	switch
Index		int	NetHDD no.(in Add of Test it can't transform.)
Type	"NFS", "SMB/CIFS"	string	Protocol type(NFS SMB/CIFS)
Username		string	Username
Password		string	Password
Ip		string	IP
Dir		string	Direction route
Size		Int	HDD size (GB)
id		int	HDD no.
disk_type	"ReadAndWriteDisk", "RedundantDisk", "ReadOnlyDisk"	string	Disk group type
disk_group_id		int	Disk group no.
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3

Table-8.1.2.3 (Hdd Control JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
id		int	Disk group no.
disk_type	"ReadAndWriteDisk", "RedundantDisk", "ReadOnlyDisk"	string	Disk group type
disk_group_id		int	Disk group no.

Table-8.1.2.4 (dualtalk_param JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
result		string	success failed

8.1.3Disk Format

SET

URL	POST/API/StorageConfig/Disk/Format
Description	It is used to format disk
Request Body	Request JSON(show as follow Table-8.1.3.1)
Successful Response	The successful result response that described in 2.5

GET	
URL	POST/API/StorageConfig/Disk/Format/Progress
Description	It is used to get format percent
Request Body	None
Successful Response	Disk Format Percent JSON(show as follow Table-8.1.3.2)

Table-8.1.3.1(Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
base_secondary_authentication		Json Object	Encryption auth, Table-12.7.2.4
hdd_id		int array	HDD no.
hdd_format_type	"AllHddData", "OnlyHddRecord", "OnlyHddData",	string	HDD format type /*format whole HDD*/ /*only format the record field*/ /*only format the general data field*/

Table-8.1.3.2(Disk Format Percent JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
hdd_id		int array	HDD no
hdd_format_state	"Formatting","Ok"	string	HDD format state
hdd_format_percent	0~100	int	HDD format percent
hdd_format_info		json array	Table-8.1.2.3

Table-8.1.3.3(Disk Format Percent JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
hdd_id		Int	HDD no
hdd_format_result	"Idle", "Ok", "InvalidParam", "NotExist", "HwError", "Failed", "Oversized"	string	HDD format result /*unformatted*/ /*OK*/ /*InvalidParam*/ "NotExist", "HwError", "Failed", "Oversized"

Example:

Request message:

POST/API/StorageConfig/Disk/FormatHTTP/1.1

```
{  
  "version": "1.0",  
  "data": {  
    "id": 1  
  }  
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 250

Connection: keep-alive

```
{  
  "version": "1.0",  
  "result": "success"  
}
```

8.2 Disk Group

GET	
URL	POST/API/StorageConfig/DiskGroup/Get
Description	It is used to get the Disk Group parameters
Request Body	none
Successful Response	Disk Group JSON(show as follow Table-8.2.1)

SET	
URL	POST/API/StorageConfig/DiskGroup/Set
Description	It is used to set the DiskGroup parameters
Request Body	Disk Group JSON(show as follow Table-8.2.1)
Successful Response	The successful result response that described in 2.5

Table-8.2.1(Disk Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
disk_group_info		JSON array	Single Group JSON(show as follow Table-8.2.2)
limit_channel_max		int	The maximum channel for the disk group

Table-8.2.2(Single Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

disk_group_type	"None", "RecordDiskGroup", "RedundanceDiskGroup"	string	Disk group type
group_num	1~MAX_HDD_GROUP_NUM	int	record disk group no./ Redundant disk group no.
channel	"CH1"...CH1x" "IP_CH1"...IP_CH1x" "WIFI_CH1"...WIFI_CH1x"	string	record disk group channel note: one channel can be only added to one disk group.

Example:

Request message:

POST/API/StorageConfig/DiskGroup/Get HTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 250

Connection: keep-alive

```
{
  "version": "1.0",
  "result": "success",
  "data": {
    "disk_group_info": [
      {
        "disk_group_type": "Record Disk Group",
        "group_array": [
          {
            "group_num": "Record Disk Group 1",
            "channel": [
              "CH1",
              "IP_CH1",
              "IP_CH2"
            ]
          }
        ],
      },
      {
        "group_num": "Record Disk Group 2",
        "channel": [
          "CH2",

```


	RANGE	TYPE	
cloud_storage		bool	Cloud storage function switch
cloud_type	"DROPBOX","Google Drive"	string	Cloud storage type
cloud_status	"Activated","CloudFull","Unactivated", "NetworkBlocked","Disabled"	string	Cloud storage connection status (DVR/NVR only) Note: only support to obtain, not support to set.
total_size	Unit: byte	unsigned long long	Total size (DVR/NVR only) Note: only support to obtain, not support to set.
used_size	Unit: Bytes	unsigned long long	Used size (DVR/NVR only) Note: only support to obtain, not support to set.
cloud_over_write	"Off","Auto","1Day","3Days","7Days", "14Days","30Days","90Days"	string	Cloud overwrite time (DVR/NVR only)
video_type	"RF","AVI","MP4"	string	Video file type (DVR/NVR only)
channel_info		JSON array	DriveName JSON(show as follow Table-8.3.x)
channel_max		int	Total channels

Table-8.3.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-8.3.2
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-8.3.2(DriveName JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	0 ~ Maximum channel number	string	Channel no.
folder_name	Max length: 63 byte	string	Each channel creates a folder on cloud storage Note: The folder name must comply with the file name specification. Invalid characters are not allowed

Table-8.3.3(Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_type	"DROPBOX","Google Drive"	string	Cloud storage type

Table-8.3.4(Url JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
url		string	Request connect URL
code		string	Verify code (for Google Drive cloud storage)

Example:

Request message:

POST/API/StorageConfig/Cloud/Get HTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 250

Connection: keep-alive

```
{
  "version":"1.0",
  "result": "success",
  "data":{
    "cloud_storage":true,
    "cloud_type":"DROPBOX",
    "cloud_status":"Unactivated",
    "total_size":0,
    "used_size":0,
    "cloud_over_write":"Auto",
    "video_type":"MP4",
    "drive_names":[
      {
        "channel":"CH1",
        "folder_name":"CH1"
      },
      {
        "channel":"CH2",
        "folder_name":"CH2"
      }
    ]
  }
}
```

```

        {
        "channel": "CH3",
        "folder_name": "CH3"
        },
        .....
    ]
}
}

```

8.4 Audio

GET	
URL	POST/API/DeviceConfig/Audio/Get
Description	It is used to get the audio Parameters
Request Body	ChannelRequest JSON(show as follow Table-8.4.1)
Successful Response	JSON(show as follow Table-8.4.2)

Control	
URL	POST/API/DeviceConfig/Audio/Set
Description	It is used to set the audio parameters
Request Body	JSON(show as follow Table-8.4.2)
Successful Response	The successful result response that described in 2.5

Table-8.4.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-8.4.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON Object	Channel Information JSON show as follow Table-8.4.x
channel_max		int	total channel in current device
support_copy		bool	Current page supports copy or not (NVR、DVR only)

Table-8.4.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-8.4.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-8.4.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
audio_enable		bool	
audio_type	"G711A","G711U", "ADPCM", "G726_16K", "G726_24K", "G726_32K", "G726_40K" "ACC"	string	Audio type
in_volume	0-10	int	Channel Device in volume
out_volume	0-10	int	Channel Device out volume

Example:

Request message:

POST/API/DeviceConfig/Audio/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "channel": ["IP_CH1", "IP_CH2"]
  }
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: 244

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "IP_CH1": {
        "copy_ch": "digit",
```


Table-8.5.1.1 (disk_info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
No	1~16	int32	show the HDD no.
enable		bool	Whether the hard drive is in the configurable state, that is, it can be selected to serve as a hot spare drive or rebuild a RAID array
check		bool	Select the current hard drive to be used as a hot spare drive or rebuild a RAID array
id	0~16	int32	HDD id code
disk_model	max_length:40	string	HDD model
serial_no	max_length:20	string	HDD serial no.
total_size	2147483647	int32	HDD size, unit: GB
array_name	max_length:20	string	HDD belongs to which RAID disk
disk_type	"Normal Disk", "HotDisk", "Raid Disk"	string	It indicates that the hard drive is a common hard drive, a hot spare drive, and a RAID drive
button_type	"" , "Add HotDisk", "Remove HotDisk"	string	The button for setting or removing a hot spare drive is displayed. If the value is blank, the operation is not supported and the button is not displayed
slot_no	max_length:8	string	Slot number of the hard disk

Table-8.5.1.2 (raid_infoJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
No		int32	show the HDD no.
check		bool	Select the current hard drive for rebuilding or deleting the current RAID array
raid_id	0~16	int32	show the RAID disk no.
raid_name	max_length:32	string	show the RAID disk no.
raid_type	"RAID0", "RAID1", "RAID4", "RAID5", "RAID0", "RAID10"	string	RAID type
total_size	0~ 2147483647	int32	RAID size
raid_status	"Normal", "Degraded", "Offline"	string	RAID status "Normal", "Degraded", "Offline"
hot_disk	0~16	int32	RAID hot disk number
sub_disk_list		array	Set up RAID HDD id
rebuild_button	"" , "Rebuild RAID"	int32	The display operation button for rebuilding the current RAID drive. If the string is empty, it does not need to be displayed

task		object	Information Json Show as follow Table-8.5.1.2.1.
------	--	--------	--

Table-8.5.1.2.1 (task JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"None", "Rebuild", "Sync"	string	RAID disk operation progress, none, RAID reconstruction (which takes several hours, 3 days or more), and data synchronization
progress	0~100	int32	Percentage of the operation progress

Table-8.5.1.3 (about_raid_info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
max_raid_num	0~16	int32	Maximum RAID disk number
raid_type		int32	RAID type list
hotdisk_type	"Public HotDisk"	"string"	Supports only one hot spare disk type
support_rebuild	"No", "Yes"	int	Supports rebuild or not

Table-8.5.1.4 (create_raid JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
item	"OneKeyCreate", "CreateRaid", "DeleteRaid", "AddHotDisk", "RemoveHotDisk", "Rebuild"	string	Supportive operation
raid_type	"RAID0", "RAID1", "RAID4", "RAID5", "RAID6", "RAID10"	string	RAID type list
disk_info		array	build RAID HDD, Json Show as follow Table-8.5.1.4.1.

Table-8.5.1.4.1 (ctrl's disk_info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
id	0~16	int32	Hdd id
serial_no	max_length:20	string	Hdd serial no.
check		bool	Build RAID Hdd

Table-8.5.1.5 (ctrl JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
item	"OneKeyCreate", "CreateRaid", "DeleteRaid", "AddHotDisk", "RemoveHotDisk", "Rebuild"	string	Build RAID name

Table-8.5.2 (JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ctrl	"OneKeyCreate", "CreateRaid", "DeleteRaid", "AddHotDisk", "RemoveHotDisk", "Rebuild"	string	The command of operation (corresponding to the JSON when the parameter is get, fill and modify the parameter and send back)

Example:

Request message:

POST/API/DeviceConfig/Audio/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
  }
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: 244

```
{
  "result": "success",
  "data": { {
    "disk_info": [{
      "No": 0,
      "id": 0,
      "enable": false,
      "check": false,
      "slot_no": "HDD1",
      "disk_model": "TOSHIBA DT02ABA400VH",
      "serial_no": "81P3S3TDSSQH",

```

```

    "total_size": 3726,
    "array_name": "RAID5_0",
    "disk_type": "Raid Disk",
    "button_type": ""
  }, {
    "No": 1,
    "id": 3,
    "enable": false,
    "check": false,
    "slot_no": "HDD4",
    "disk_model": "TOSHIBA DT02ABA400VH",
    "serial_no": "81P3S3SRSSQH",
    "total_size": 3726,
    "array_name": "RAID5_0",
    "disk_type": "Raid Disk",
    "button_type": ""
  }, {
    "No": 2,
    "id": 5,
    "enable": false,
    "check": false,
    "slot_no": "HDD6",
    "disk_model": "TOSHIBA DT02ABA400VH",
    "serial_no": "81P3S3SDSSQH",
    "total_size": 3726,
    "array_name": "RAID5_0",
    "disk_type": "Raid Disk",
    "button_type": ""
  }, {
    "No": 3,
    "id": 6,
    "enable": true,
    "check": false,
    "slot_no": "HDD7",
    "disk_model": "TOSHIBA DT01ACA300",
    "serial_no": "48HNBW7AS",
    "total_size": 2794,
    "array_name": "-",
    "disk_type": "Normal Disk",
    "button_type": "Add HotDisk"
  }, {
    "No": 4,
    "id": 7,
    "enable": true,

```

```

        "check": false,
        "slot_no": "HDD8",
        "disk_model": "TOSHIBA DT01ACA300",
        "serial_no": "48HNEM8AS",
        "total_size": 2794,
        "array_name": "-",
        "disk_type": "Normal Disk",
        "button_type": "Add HotDisk"
    }
],
"raid_info": [{
    "raid_name": "RAID5_0",
    "No": 0,
    "raid_id": 0,
    "check": false,
    "raid_type": "RAID5",
    "total_size": 7630637,
    "raid_status": "Degrade",
    "sub_disk_list": [0, 3, 5],
    "hot_disk": "0",
    "task": {
        "status": "Rebuild",
        "progress": 19
    },
    "rebuild_button": ""
}
],
"about_raid_info": {
    "max_raid_num": 16,
    "raid_type": [0, 1, 4, 5, 6, 10],
    "hotdisk_type": "Public HotDisk",
    "support_rebuild": "Yes"
},
"create_raid": {
    "raid_name": "",
    "raid_type": "RAID0",
    "disk_info": [{
        "id": 6,
        "serial_no": "48HNBW7AS",
        "check": false
    }, {
        "id": 7,
        "serial_no": "48HNEM8AS",
        "check": false
    }
}

```


accesstoken	1-128	string	Dropbox activation token
-------------	-------	--------	--------------------------

Table-8.6.1(CloudJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
enc_mode	"FTP""SFTP""AWS S3""DROPBOX"Google Drive"	string	Server mode (DVR/NVR only)
video_stream_type	"Mainstream""Substream"	string	Video stream type (DVR/NVR only)
video_type	"MP4""RF"	string	Video type (DVR/NVR only)
max_package_interval	10 30 60 180 300 600	Int	Max package interval. second (DVR/NVR only)
cloud_status	"Activated","CloudFull","Unactivated","NetworkBlocked","Disabled"	string	Cloud storage connection status(DVR/NVR only) Note: Only access is allowed, and setting is not supported
total_size	Unit: Bytes	unsigned long long	Total capacity (DVR/NVR dedicated) Note: Only access is allowed, and setting is not supported
used_size	Unit: Bytes	unsigned long long	Used capacity (DVR/NVR dedicated) Note: Only access is allowed, and setting is not supported
cloud_over_write	"Off","Auto","1Day","3Days","7Days","14Days","30Days","90Days"	string	Cloud storage data overwrite time (DVR/NVR only)
progress	[0~100]	int	Percentage of capacity usage of cloud storage (DVR/NVR dedicated) Note: Only access is allowed, and setting is not supported
enc_agreement	"ftp:/" "sftp:/" "https:/"	string	IP front protocol Total capacity (DVR/NVR dedicated) Note: Only access is allowed, and setting is not supported
server_ip	Max length: 255 byte	string	IP address (DVR/NVR only)
port	[1~65535]	int	Server Port
username	Max length: 255byte	string	Login user name
password	Max length: 255byte	string	Login user password
password_empty		bool	Whether the password is empty

region	[0~31]	string	Region (DVR/NVR only)
enable		bool	Enable the server (DVR/NVR dedicated)
test		bool	Support TEST (DVR/NVR only)
activate_cloud_btn		bool	Whether the activation button is supported (for DVR/NVR only)
show_server_config	"enable" "test" "port" "password" "password_empty" "video_stream_type" "server_ip" "video_type" "max_package_interval" "enc_agreement" "region" "activate_cloud_btn" "cloud_status" "total_size" "used_size" "progress" "cloud_over_write"	string	Tell the remote corresponding mode which controls should be displayed

Table-8.6.3(Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_type	"DROPBOX","Google Drive"	string	Cloud type

Table-8.6.4(Uri JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
url		string	Request connection URL
code		string	Verification code (for Google Drive cloud storage)

Example:

Request message:

POST/API/StorageConfig/Server/Get HTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 250

Connection: keep-alive

```
{
  {
    "result": "success",
    "data": {
      "enc_mode": "FTP",
      "enc_agreement": "ftp://",
      "video_stream_type": "Substream",
      "video_type": "RF",
      "max_package_interval": 10,
      "cloud_over_write": "Auto",
      "cloud_status": "Unactivated",
      "total_size": "0.00B",
      "used_size": "0.00B",
      "progress": 0,
      "ftp_info": {
        "enable": true,
        "server_ip": "172.16.8.124",
        "port": 21,
        "username": "hwy",
        "password": "",
        "password_empty": false
      },
      "sftp_info": {
        "enable": true,
        "server_ip": "172.16.8.12",
        "port": 21,
        "username": "8888",
        "password": "",
        "password_empty": false
      },
      "aws_info": {
        "enable": true,
        "server_ip": "rsvediockt.s3.ap-southeast-1.amazonaws.com",
        "region": "ap-southeast-1",
        "username": "AKIK4RT23BIPRM2",
        "password": "",
        "password_empty": false
      },
      "dropbox_info": {"enable": true},
      "google_drive_info": {"enable": false}
    }
  }
}
```

9 SCHEDULES

9.1 Schedules

GET	
URL	POST/API/Schedules/Record/Get
	POST/API/Schedules/Capture/Get
	POST/API/Schedules/Ftp/Get
	POST/API/Schedules/Email/Get
	POST/API/Schedules/Intelligent/Get
	POST/API/Schedules/Alarm/Get
	POST /API/Schedules/AI/Get
Description	It is used to get the schedules
Request Body	ChannelRequest JSON(show as follow Table-9.1.1)
Successful Response	Channel Information JSON (show as follow Table-9.1.2)

SET	
URL	POST/API/Schedules/Record/Set
	POST/API/Schedules/Capture/Set
	POST/API/Schedules/Ftp/Set
	POST/API/Schedules/Email/Set
	POST/API/Schedules/Intelligent/Set
	POST/API/Schedules/Alarm/Set
	POST /API/Schedules/AI/Set
Description	It is used to set the schedules
Request Body	Channel Information JSON (show as follow Table-9.1.2)
Successful Response	The successful result response that described in 2.5

Table-9.1.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-9.1.2 (Channel Information JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as follow Table-9.1.x

Table-9.1.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-9.1.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-9.1.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
category		JSON array	CategoryJSON show as follow Table-9.1.4
copy_ch	"digit""analog""wifi"	string	Flags that support channel duplication (for NVR and DVR only)

Table-9.1.4(CategoryJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
schedule_type	"Normal", "Motion", "Io", "Pir" "IntelligentAnalysis", "Exception", "Buzzer" "Push" "PIR and Motion", "Pir_Motion_Person" "pid""lcd""sod""pvd""fd""cc""sound""od""ad" "cd""qd" "lpd""rsd""face""ai_pvd""ai_pid""ai_lcd""lpr"	string	Schedule Types Allowed to be Configured Recording schedule contains type { Normal, Motion, IO, PIR } The capture schedule contains types { Normal, Motion, IO, PIR } Ftp schedule contains type { Normal, Motion, IO, IntelligentAnalysis } Email schedule contains types { Motion, IO, Exception, In-Analysis, PIR } Alarm schedule contains type { IO, Buzzer } Intelligent does not have this field Server plan contains type { "Normal" "Motion" "Io" "PIR and Motion", "Pir_Motion_Person" "Motion" "pid" "lcd" "sod" "pvd" "fd" "cc" "sound" "od" "ad" "cd" "qd" "lpd" "rsd" "face" "ai_pvd" "ai_pid" "ai_lcd" "lpr" } Note: Depending on device capabilities, actual schedule types may be less than allowable configurations
week		JSON array	WeekJSON show as follow Table-9.1.5

show_list		Bool	Show logic with new schedule table
-----------	--	------	------------------------------------

Table-9.1.5(WeekJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
day	Sun,Mon,Tue,Wed, Thu,Fri,Sat	string	identify the day of the week
time	0: close the time period 1: Open the time period	array	Each array bit (int) identifies half an hour.

10 RECORD

10.1 Record Configuration

GET	
URL	POST/API/RecordConfig/Get
Description	Get the channel record parameters
Request Body	ChannelRequest JSON(show as follow Table-10.1.1)
Successful Response	Channel Information JSON (show as follow Table-10.1.2)

SET	
URL	POST/API/RecordConfig/Set
Description	Set the channel record parameters
Request Body	Channel Information JSON (show as follow Table-10.1.2)
Successful Response	The successful result response that described in 2.5

Table-10.1.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... WIFI_CH1x" The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-10.1.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

channel_info		JSON Object	Channel Information JSON show as follow Table-10.1.x
--------------	--	-------------	--

Table-10.1.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-10.1.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-10.1.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
record_switch		bool	Video function switch
stream_mode	"Mainstream", "DualStream"	string	Stream type, only supports dual stream devices, (for NVR/DVR only)
prerecord		bool	Pre-recording switch
manual_record		bool	Indicates the status of manual recording
net_break_record		bool	Indicates the status of manual recording
copy_ch	"digit""analog""wifi"	string	Flags that support channel duplication (for NVR and DVR only)

10.2 Search Record

GET	
URL	POST/API/Playback/SearchRecord/Search
Description	Search record information.
Request Body	SearchRequest JSON(show as follow Table-10.2.1)
Successful Response	Record Information JSON (show as follow Table-10.2.2)

Table-10.2.1 (SearchRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... "WIFI_CH1x" The number of channels	string array	Each array bit represents a channel with a string.

	depends on the capabilities of the device.		
start_date		string	Search startdate. The date format is MM/DD/YYYY
end_date		string	Search end date. The date format is MM/DD/YYYY
start_time		string	Search start time. The time format is hh:mm:ss
end_time		string	Search end time. The time format is hh:mm:ss
record_type		int	Video type
stream_mode	"Mainstream","Substream"	string	Stream type (IPC only supports main stream)
smart_region	18	Int array	Smart playback filter area, the size is 15*22

Table-10.2.2 (RecordInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
record		JSON array	Single Channel Record Information JSON show as follow Table-10.2.3

Table-10.2.3 (Single Record Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel		string	channel number
stream_mode	"Mainstream","Substream"	string	Stream type (IPC only supports main stream)
record_type	NormalRecord: 0x1, AlarmRecord:0x1c06, IORecord: 0x4, PIRRecord: 0x400, SoundRecord: 0x1000, NetbreakRecord: 0x800, MotionRecord: 0x2, MotionAndIo: 0x6, AllIntelliRec: 0x1c8, PEARRecord: 0x40, PEALineRecord: 0x8, HDRecord: 0x80, FDRecord: 0x100, PCCRecord: 0x4000,	int	Recording type (IPC only)

	SmartRecord: 0x80000,//2018.05.30 OcclusionRecord: 0x100000,//occlusion detection INE_ALL_RECORD: (0x10 0x20 0x40 0x200 0x1000 0x2000 0x4000), AlarmAssemble: 0x7FFFFFFF,//5.0 Use both AllRecord: 0xFFFFFFFF, PicAllRecord: 0x7FFFF		
record_type	NormalRecord: 0x1, AlarmRecord: 0x2, MotionRecord: 0x4, IORecord: 0x8, PEARRecord: 0x10, PEALineRecord: 0x10, AVDRecord: 0x20, OSRecord: 0x40, AllIntelliRec: 0x80, SensorRecord: 0x100, PEAAreaRecord: 0x200, OCCRecord: 0x400,//private zone NetbreakRecord: 0x800,//netbreak HDRecord: 0x1000, 4096 FDRecord: 0x2000, 8192 PCCRecord: 0x4000, 16384 MothionAndIo: 0x8000, PIRRecord: 0x10000, SoundRecord: 0x20000, ManualRecord: 0x40000, SmartRecord: 0x80000,//2018.05.30 OcclusionRecord: 0x100000,//occlusion detection PersonRecord: 0x200000 FaceAttribute:0x400000 INE_ALL_RECORD: (0x10 0x20 0x40 0x200 0x1000 0x2000 0x4000), AlarmAssemble: 0x7FFFFFFF,//5.0 Use both AllRecord: 0xFFFFFFFF, PicAllRecord: 0x7FFFF	int	Recording type (for NVR only)
start_date		string	Record file startdate. The date format is MM/DD/YYYY
end_date		string	Record file end date. The date format is MM/DD/YYYY
start_time		string	Record file start time.

			The time format is hh:mm:ss
end_time		string	Record file end time. The time format is hh:mm:ss
Size		int	file size, unit byte
record_id	0 - 0xFFFFFFFF	unsigned int	
disk_event_id	0 - 0xFFFFFFFF	unsigned int	

Example:

Request message:

POST /API/ Playback/SearchRecord/SearchHTTP/1.1

```
{
  "version": "1.0",
  "stream_mode": "Substream",
  "channel": ["IP_CH1", "IP_CH2"],
  "start_date": "01/03/2020",
  "start_time": "00:00:00",
  "end_date": "01/03/2020",
  "end_time": "23:59:59",
  "record_type": 4294967295
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 207

Connection: keep-alive

```
{
  "version": "1.0",
  "record": [
    [
      {
        "channel": "IP_CH1",
        "stream_mode": "Substream",
        "record_type": 1,
        "start_date": "06/24/2019",
        "start_time": "00:00:01",
        "end_date": "06/24/2019",
        "end_time": "00:38:17",
        "time": 3211264
      }
    ]
  ]
}
```

```

.....
{
  "channel": "IP_CH1",
  "stream_mode": "Substream",
  "record_type": 3,
  "start_date": "06/24/2019",
  "start_time": "15:09:31",
  "end_date": "06/24/2019",
  "end_time": "15:15:55",
  "time": 2416640
}
]
}

```

10.3 Playback rtsp url

The video playback adopts the Rtsp protocol, and the url format is as follows:

Back-end equipment (NVR/DVR):

rtsp://ip:port/rtsp/playback?channel=1&subtype=0&starttime=2021-03-24T01:30:00Z&endtime=2021-03-24T07:30:59Z&localtime=true

Front-end equipment (IPC):

[327DE platform]: rtsp://ip:port/rtsp/playback?channel=1&subtype=0&starttime=2021-03-24T01:30:00Z&endtime=2021-03-24T07:30:59Z

[Other platforms]: rtsp://ip:port/cam/playback?channel=1&starttime=2021-03-24T01:30:00Z&endtime=2021-03-24T07:30:59Z

Table-10.3.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel		int	channel, starting from 1
subtype	0: main stream, 1: sub stream	int	stream type
starttime		string	Play start time
endtime		string	playback end time
localtime	true: use local time false: use UTC time	bool	Whether to use local time (this field can be omitted, UTC time is used) (note: currently only the backend supports this field, and the frontend IPC does not support it yet)

Note: When the time entered in the request is UTC time, the response time is also UTC time.

GET	
URL	POST/API/Playback/PlaybacRtspkUrl/Get (only for A01. Not supported for non-A01)
Description	Get playback Rtsp URL
Request Body	SearchRequest JSON(show as follow Table-10.3.1.1)
Successful Response	Record Information JSON (show as follow Table-10.3.2)

Table-10.3.1.1 (SearchRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... "WIFI_CH1x" The number of channels depends on the capabilities of the device.	string	represents a channel with a string.
transport_type		string	"tcp"、 "udp"、 "http" "https"
start_date		string	Search startdate. (The passed date is calculated according to the 0 time zone) The date format is YYYY-MM-DD
end_date		string	Search end date. The date format is YYYY-MM-DD
start_time		string	Search start time. The time format is hh:mm:ss
end_time		string	Search end time. The time format is hh:mm:ss
stream_mode	"Mainstream", "Substream"	string	Stream type (IPC only supports main stream)
record_type	1<<0 (scheduled recording) 1<<2 (motion detection recording) 1<<3 (IO alarm recording) 1<<4 (Cross-line detection alarm recording) 1<<7 (smart alarm recording) 1<<9 (intrusion detection alarm recording) 1<<17 (audio alarm recording) 1<<18 (manual recording) 1<<20 (occlusion alarm recording) 0x7FFFFFFE (all alarm recordings)	int	

	0xFFFFFFFF (all recordings)		
--	-----------------------------	--	--

Table-10.3.2 (Single Channel Playback Rtsp URL JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
playback_url	Max length: 127 bytes Reference: "rtsp://192.168.1.100:554/rtsp/playback?channel=2&sutbype=0&recordtype=4294967295&starttime=2020-07-03T00:29:37Z&endtime=2020-07-03T15:33:30Z&token=123"	string	The playback rtsp URL of the request channel (if the request fails, this field will not be returned, and the corresponding error reason will be returned)

Example:

Request message:

POST /API/ Playback/SearchRecord/SearchHTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "stream_mode": "Substream",
    "channel": "CH1",
    "start_date": "2020-07-09",
    "start_time": "00:00:00",
    "end_date": "2020-07-09",
    "end_time": "23:59:59",
    "transport_type": "tcp",
    "record_type": 4294967295
  }
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 207

Connection: keep-alive

```
{
```

```

"version": "1.0",
"data":{
    "playback_url ": "rtsp://192.168.1.100:554/rtsp/playback?channel=2&subtype=0&starttime=2020-07-03T00:29:37Z&endtime=2020-07-03T15:33:30Z"
}
}

```

10.6 Month Search

GET	
URL	POST/API/Playback/SearchMonth/Get
Description	Search record information.
Request Body	SearchRequest JSON(show as follow Table-10.6.1)
Successful Response	Record Information JSON (show as follow Table-10.6.2)

Table-10.6.1 (SearchRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... "WIFI_CH1x" The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. (An empty array means searching all channels)
search_type	"Record", "Picture", "FD", "PVD", "PidLcd", "Repeat", "FaceAttendance"	string	month search type
start_date		string	Search startdate. The date format is MM/DD/YYYY
stream_type	"Mainstream", "Substream"	string	stream type

Table-10.6.2 (Record Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
is_has_rec		int array	
record_type		int array	

10.7 Pic Playback

10.7.1 Syntax

GET	
URL	POST/API/Playback/Picture/Get
Description	Search picture information.
Request Body	SearchRequest JSON(show as follow Table-10.7.1)
Successful Response	Record Information JSON (show as follow Table-10.7.2)

10.7.2 Parameters

Table-10.7.1 (SearchRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. Pass the parameter when searching for the first time. When IE already has a picture information list, this field does not need to be passed.
start_date		string	Search start date. The date format is MM/DD/YYYY Each array bit represents a channel with a string. Pass the parameter when searching for the first time. When IE already has a picture information list, this field does not need to be passed.
end_date		string	Search end date. The date format is MM/DD/YYYY Each array bit represents a channel with a string. Pass the parameter when searching for the first time. When IE already has a picture information list, this field does not need to be passed.
start_time		string	Search start time. The time format is hh:mm:ss Each array bit represents a channel with a string. Pass the parameter when searching for the first time. When IE already has a picture information list, this field does not need to be passed.
end_time		string	Search end time. The time format is hh:mm:ss

			Each array bit represents a channel with a string. Pass the parameter when searching for the first time. When IE already has a picture information list, this field does not need to be passed.
pic_info		string	When searching for the first time, IE does not have any pic_info information, so this field is not passed. The board will do a search and return all the information to IE. And return the data of the first image to IE
pic_width		int	
pic_height		int	
pic_sort		int	0: order 1: reverse order
record_type	NormalRecord: 0x1, MotionRecord: 0x4, IORecord: 0x8, PIRRecord: 0x10000, ManualRecord: 0x40000	unsigned int	Can be a single search type or a mix of types
msg_type		string	When this parameter is requested, the content of the field is returned as is.

Table-10.7.2 (Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
result		bool	
overload		bool	More than 5000 pictures
msg_type		string	When this parameter is requested, the content of the field is returned unchanged
picture		json	Return a picture information and base64 picture content, see 10.7.3
all_pic_info		string array	When IE does not have a pic_info field, the board will search and return this field
all_pic_num	0-500	int	When IE does not have a pic_info field, the board will search and return this field

Table-10.7.3 (Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

channel		string	
time		string	“yyyy/MM/dd hh:mm:ss”
image		string	picture content

Example:

When searching for pictures for the first time, the request includes information such as start and end time, channel, etc., and the server returns a list of the searched data. In subsequent requests, the client needs to take the pic_info information to the server to fetch pictures one by one.

An example of the first request is as follows:

Request message:

POST /API/Playback/Picture/Get?2020-10-28@14:19:11 HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "start_date": "10/28/2020",
    "start_time": "00:00:00",
    "end_date": "10/28/2020",
    "end_time": "23:59:59",
    "record_type": 524287,
    "channel": ["CH5"],
    "pic_sort": 0
  }
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: 207

Connection: keep-alive

```
{
  "version": "1.0",
  "result": "success",
  "data": {
    "overload": false,
    "all_pic_num": 4,
    "all_pic_info": [{
      "pic_info":
      "AQAAAwAAAAAGAAAACAAAAAQAAAAfQAAAAcTDhwKFAAA0IoBAZYAABAAAA
      AAAAA"
    }, {
      "pic_info":
      "AQAAAgEAAAAGAAAACAAAAAQAAAAAATAAAAcTDhwKFAABZosBAZYAABEAAA
      AAAAA"
    }, {
      "pic_info":

```

```

"AQAAAQIAAAAGAAAACAAAAAQAAAAAAAAAAAAcTDhwKFAAC/IsBAZYAABIAAAA
AAAAA"
    }, {
      "pic_info":
"AQAAAAMAAAAGAAAACAAAAAQAAAAAAAAAAAAcTDhwKFAADkowBAZYAABMAA
AAAAAAA"
    }
  }
}

```

The client requests a single image in sequence according to the reply image information, the example is as follows:

Request message:

POST /API/Playback/Picture/Get?2020-10-28@14:19:11 HTTP/1.1

```

{
  "version": "1.0",
  "data": {
    "pic_info":
"AQAAAwAAAAAGAAAACAAAAAQAAAAAfQAAAAcTDhwKFAAA0IoBAZYAABAAAAAAAAAAAA"
  }
}

```

Response message:

HTTP/1.1 200 OK

```

{"version":"1.0","result":"success","data":{"picture":{"channel":"CH5","time":"2020/10/28
14:19:07","image":"/9j/4AAQSkZJRgABAQAAQABAAD/2wDFAAgGBgcGBQgHBwcJCQgKDBQNDAsLD
BkSEw8UHRofHh0aHBwgJC4nICsIxwcKdcpLDAxNDQ0Hyc5PTgyPC4zNDIBCQkJDAsMGA0NGDIhHCEy
MjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMgIJCQkMCwwYDQ0YMIecITy
MjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIyMjIy/8AAEQgC0AUAAwEiAAIRAQ
MRAv/EAaIAAAEFAQEBAQEBAAAAAAAAAAAABAgMEBQYHCAkKCxAAAgEDAwIEAwUFBAQAAAF9
AQIDAAQRBRIhMUEGE1FhByJxFDKBkaEII0KxwRVS0fAkM2JyggkKFhcYGRolJicoKSo0NTY3OD "}}}

```

10.8 Playback Page

GET	
URL	POST /API/Playback/PlaybackPage/Range
Description	Get the channel record parameters
Request Body	none
Successful Response	Channel Information JSON (show as follow Table-10.8.x)

Table-10.8.x (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Normal		Json object	(show as follow Table-10.8.1)

Picture		Json object	(show as follow Table-10.8.1)
HumanVehicle		Json object	(show as follow Table-10.8.1)
PidLcd		Json object	(show as follow Table-10.8.1)
supportFaceAttr		bool	Whether the AI playback page has a control for face attributes
param_limit		Json object	(show as follow Table-10.8.1)

Table-10.8.1 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
display_mode	“Treelike”, “Checkbox”	string	UI display type: tree, checkbox
default_sort_of_results		int	Used to search information in default order/reverse order
items		Json object	(show as follow Table-10.8.1.1)
similarity_face		int	Control the range and default value of the similarity input box of IE playback Face Detection
similarity_custom er		int	Control the range and default value of the Fault-tolerant input box of the IE playback License Plate
similarity_license		int	Control the range and default value of the similarity input box of IE playback Repeat customer

Table-10.8.1.1 (Single items Information JSON)

record_type		Json object	Playback type (show as follow Table-10.8.2)
stream_mode	“Mainstream”, “Substream”	string	Main substream type (this field is only available for Normal)

Table-10.8.2 (Single Channel Information JSON)

normal		bool	Normal
manual		bool	Manual
alarm		Json object	Alarm (show as follow Table-10.8.3) (for IPC only)
smart		Json object	Smart (show as follow Table-10.8.4) (for IPC only)
Human		bool	AI human (for NVR only)
Vehicle		bool	AI vehicle (for NVR only)
IO		bool	Io alarm recording type (for NVR only)
motion		bool	Motion alarm recording type (for NVR only)
PIR		bool	Pir (for NVR only)

Sound		bool	Sound alarm (for NVR only)
occlusion		bool	Lens blocking (for NVR only)
Intelligent		bool	Intelligent analysis (for NVR only)
PID		bool	AI Perimeter Invasion (for NVR only)
LCD		bool	AI cross-line detection (for NVR only)
faceAttr		bool	Face attributes (for NVR only)

Table-10.8.3 (Single Channel Information JSON)

IO		Json object	Io alarm recording type
motion		Json object	Motion alarm recording type
PIR		Json object	Pir
Sound		Json object	sound alarm
Netbreak		Json object	disconnected video

Table-10.8.4 (Single Channel Information JSON)

PID		Json object	
LCD		Json object	
SOD		Json object	
PD		Json object	
FD		Json object	
CC		Json object	

10.9 Record Tag

GET	
URL	POST /API/Playback/Tag/Get
Description	Search Tag information
Request Body	Record Tag JSON(show as follow Table-10.9.1)
Successful Response	Tag Information JSON (show as follow Table-10.9.2)

SET	
URL	POST / API/Playback/Tag /Set
Description	Add Tag label
Request Body	Record Tag JSON (show as follow Table-10.9.4)
Successful Response	The successful result response that described in 2.5

Table-10.7.1 (Search Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... "WIFI_CH1x"	string array	Each array bit represents a channel with a string.

	The number of channels depends on the capabilities of the device.		
start_date		string	Search startdate. The date format is MM/DD/YYYY
end_date		string	Search end date. The date format is MM/DD/YYYY
start_time		string	Search start time. The time format is hh:mm:ss
end_time		string	Search end time. The time format is hh:mm:ss
Keyword	0-39	string	Keywords, when searching tags, only search tags containing keywords

Table-10.7.2 (Response Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Pre-play		string	Play the video from the label time forward ("5s")("10s")("30s")("1Min")("2Min")("5Min")("10Min")
Post-play		string	Play the recording at the end of the tag time ("5s")("10s")("30s")("1Min")("2Min")("5Min")("10Min")
all_tag_info		string array	Information JSON show as follow Table-10.7.3
all_tag_num	0-5000	int	How many matching tags were found

Table-10.7.3 (all_tag_info Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Tag_date		string	The date format is MM/DD/YYYY
Tag_time		string	The time format is hh:mm:ss

channel		String	which channel this tag belongs to
chNum		int	channel number
label_id		int	Tag ID
record_id		int	Video ID
Tag_name	0-39	string	label name

Table-10.7.4 (Add Tag Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Tag_date		string	The date format is MM/DD/YYYY
Tag_time		string	The time format is hh:mm:ss
channel		String array	which channel this tag belongs to
chNum		int	channel number
label_id		int	Tag ID
record_id		int	Video ID
Tag_name	0-39	string	label name
operate	0-2	int	0-set, 1-delete, 2-modify name

Example:

Request message:

POST /API/ Playback/Tag/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "start_date": "04/14/2021",
    "start_time": "00:00:00",
    "end_date": "04/14/2021",
    "end_time": "23:59:59",
    "channel": [
      "CH1",
      "CH2",
      "CH3",
      "CH4",
      "CH5",
      "CH6",
      "CH7",
```

```
        "CH8"
    ],
    "Keyword": ""
}
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "version": "1.0",
  "result": "success",
  "data": {
    "Pre-play": "30s",
    "Post-play": "30s",
    "all_tag_num": 2,
    "all_tag_info": [
      {"tag_info": {
        "Tag_date": "04/14/2021",
        "Tag_time": "09:39:56",
        "channel": "CH1",
        "chNum": 0,
        "label_id": 1,
        "record_id": 0,
        "Tag_name": "Tag"
      }},
      {"tag_info": {
        "Tag_date": "04/14/2021",
        "Tag_time": "09:40:02",
        "channel": "CH3",
        "chNum": 2,
        "label_id": 2,
        "record_id": 0,
        "Tag_name": "Tag11"
      }}
    ]
  }
}
```

11 MAINTENANCE

11.1 Log

GET	
URL	POST/API/Maintenance/Log/Search
Description	Get the system log information
Request Body	Request JSON (show as follow Table-11.1.1)
Successful Response	Log Information JSON (show as follow Table-11.1.2)

Table-11.1.1(Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
start_date	The date format is MM/DD/YYYY	string	Start date of search.
start_time	The time format is hh:mm:ss	string	Start time of search.
end_date	The date format is MM/DD/YYYY	string	Enddate of search.
end_time	The time format is hh:mm:ss	string	Endtime of search.
main_type	"System", "Operate", "Alarm", "Account", "Record", "Storage", "AI", "All"	string	log master type
channel_max	Ip:1	int	Maximum number of channels

Table-11.1.2 (Log Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
log		JSON array	Single LogInformation JSON show as follow Table-11.1.3

Table-11.1.3 (Single LogInformation JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
start_date		string	log date The date format is MM/DD/YYYY
start_time		string	log time The time format is hh:mm:ss
end_time		string	log time The time format is hh:mm:ss
main_type	"System", "Operate", "Alarm",	string	Log master type. 1. When requesting for User, only give Alarm log (A01).

	"Account", "Record", "Storage", "AI", "All"		2. Give all logs when requested by User (NVR)
ai_main_type	"Face Detection" " Human & Vehicle"	string	AI log main type
sub_type	System: "SystemStartup", "Shutdown", "Reboot", "SystemMaintain", "Update", "ChangeTime", "Ntp" "All" Operate: "RECParameters", "ScheduleRecord", "MainstreamSettings", "NetworkSettings", "SubstreamSettings" "EmailSetting", "DdnsSettings", "ColorSettings", "DisplaySettings", "VideoSettings", "VideoCover", "Motion", "I/O", "HddSettings", "PtzSettings", "SerialPort" "NtpSettings", "DstSettings", "GeneralSetup", "MultiUser", "Maintenance", "AbnormalSettings", "EmailSchedual", "RtspSettings", "IntelligentSchedual", "IpcChannel" "FtpSettings",	string	log subtype 1. 1. Only IPC supports subtype query, NVR does not support this function 2. In order to unify the interface, the NVR reserves this interface, and the All subtype is also included in the query to facilitate future expansion.

"ImageControl", "MobileStream", "ProtocolConfigure", "CaptureSettings", "CaptureSchedual" "IntelligentSettings", "IntelligentSettings2", "IntelligentSettings3", "AlarmLinkagePTZ", "IntelligentSettings4", "IntelligentSettings5", "IntelligentSettings6", "CloudStorage", "PirAlarmSettings", "AlarmStream", "IpcPreviewCtrl" "IPV6", "ProtocolInfo", "HttpsSettings", "DeterrenceSettings", "SdSetting", "OdSetting", "FtpSchedual " "IpFilter", "AlarmSchedule", "EmailTest" "ExportParam" "ImportParam" "LoadDefault" "IpcReboot" "IpcExport" "IpcImport" "IpcLoadDefault" "IpcUpgrade" "SearchLog" "ChannelAdd", "ChannelDelete", "Modify", "IpcSearch" "PtzControl" "ManualAlarm" "BackupLog" "CrosscountSearch" "DefaultImgControl"		
--	--	--

	<p> “DefaultColor” “IpChannelDelete” “DisplaySettings” “MainstreamSettings” “PlatformSettings” “ChannelModify” “AbnormalSettings” “PosAdd” “PosDelete” “PosModify” “PosSettings” “SmartHomeSettings” “VersionSettings” “SNMPSettings” "All" Alarm: "MotionStart", "MotionEnd", "IoAlarmStart", "IoAlarmEnd", "VideoLoss", "PidStart", "PidEnd", "LcdStart", "LcdEnd", "SodStart", "SodEnd", "PirStart", "PirEnd", "Pd&VdStart", "Pd&VdEnd", "FdStart", "FdEnd", "CcStart", "CcEnd", "CdStart", "CdEnd", "QdStart", "QdEnd", "ShellBroken", "SdStart", "SdEnd", </p>		
--	--	--	--

	<p>"OdStart", "OdEnd" "All"</p> <p>Account: >Login", Logout", AddUser", DeleteUser", ModifyUser", LockScreen", Unlock", UsernameError", PasswordError", All"</p> <p>Record: Schedule Record Start", ManualRecordStart", RecordStop", RecordSearch", Playback", RecordBackup", PictureSearch", PicturePlayback", PictureBackup" "ManulCapture" InstantPlayback" All"</p> <p>Storage: FormatHDD", NoSpaceOnDisk", DiskError", Auto", HddPartitionChanged" HddSmart" All"</p> <p>AI Face Detection:</p>		
--	---	--	--

	"Stranger" "" "All" Human & Vehicle: "Human", "Vehicle", "PID[Human]" "LCD[Human]" "PID[Vehicle]" "LCD[Vehicle]" "All" Secure: "LoginLock" "PasswordInsecure" "ReadUserParamFailed" "ReadFactoryParamFailed" "SessionBusy" "UpgradeVerifyFailed", "CustomCartExpired" All: "All"		
channel		string	The channel associated with the log Note: Depending on the log type, this field is optional
record		bool	Is there a video Note: Depending on the log type, this field is optional
result_code	"operate_success", "no_permission", "lack_of_resource", "network_error", "exception_error", "operate_failed", "file_error", "memory_not_enough", "parameter_error", "para_check_error", "write_flash_error", "no_hdd", "auth_failed",	string	

	"no_support", "unknown_error", "no_udisk", "no_upgrade_package", "soft_is_new", "software_packet_error", "language_version_error", "file_name_too_long", "update_failed", "event_coverd", "backup_search_dir_too_long", "backup_no_support_mjpeg", "connect_close", "file_invalid", "sapce_shortage", "data_error", "user_not_exist", "first_login", "weak_password", "time_abnormal", "passwd_expired", "version_incompatible", "in_userinterface", "in_upgrading", "user_locked", "part_success"		
user		string	Username of the operating user
ip		String	Operating user's ip address
time_src		String	When modifying time, the old time
time_dst		String	When modifying the time, the new time
user_id	0-> Administrator, 1-> user1, 2-> user2... 6-> user6	int	User ID of the operating user
ipc_ip		string	The ip address of the operated ipc
dst_user		string	Username of the user being manipulated
dest_userid	0-> Administrator, 1->user1 2->user2...6->user6	int	User ID of the user being manipulated
hddid		int	HDD serial number

model		string	Hard disk related
serial_no		string	Hard disk related

11.2 Load Default Parameter

OPRATE	
URL	POST /API/Maintenance/Reset/Range
Description	Query whether the system needs to be restarted
Request Body	Request JSON (show as follow Table-11.2.1)
Successful Response	Load Default ParameterJSON(show as follow Table-11.2.2)

OPRATE	
URL	POST /API/Maintenance/Reset/Set
Description	Restore system default parameters
Request Body	Request JSON (show as follow Table-11.2.1)
Successful Response	The successful result response that described in 2.5

Table-11.2.1(Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel		bool	Restore channel related parameters
record		bool	Restore record related parameters
alarm		bool	Restore alarm related parameters
ai		bool	Restore ai related parameters
network		bool	Restore network related parameters
device		bool	Restore device related parameters
system		bool	Restore system related parameters
system_maintain		bool	Restore systemmaintain related parameters (for A01 only)
attendance		bool	Restore attendance related parameters (for A01 only)
secondary_authentication	Max length: 16 byte	string	Verify the admin user password for secondary authentication
intelligent		bool	Restore intelligent related parameters
except_network_param		bool	
base_secondary_authentication		Json Object	Encryption auth, Table-12.7.2.4

Table-11.2.2(Load Default Parameter JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

need_reboot		bool	Whether you need to reboot
-------------	--	------	----------------------------

11.3 Auto Reboot

GET	
URL	POST /API/Maintenance/AutoReboot/Get
Description	Get the system information
Request Body	None
Successful Response	Auto Reboot information JSON (show as follow Table-11.3)

SET	
URL	POST /API/Maintenance/AutoReboot/Set
Description	Set NTP information
Request Body	Auto Reboot information JSON (show as follow Table-11.3)
Successful Response	The successful result response that described in 2.5

Table-11.3(Auto Reboot Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
auto_reboot		bool	auto restart switch Automatic maintenance switch,.
period_mode	"EveryDay", "EveryWeek", "EveryMonth"	string	maintenance cycle mode
week	"Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun"	string	Weekdays of the week.(When MaintainPeriod is Week)
day	1~31	int	Number of months.(When MaintainPeriod is Month)
time		string	Restart time. Format: hours : minutes (Each digit must be written with two digits)

11.4 Ftp Upgrade

GET	
URL	POST /API/Maintenance/FtpUpgrade/Range
Description	Get the information of Ftp Upgrade
Request Body	none
Successful Response	Ftp upgradeinformation JSON (show as follow Table-14.4.1)

GET	
URL	POST /API/Maintenance/FtpUpgrade/Get
Description	Get the information of Ftp Upgrade
Request Body	none
Successful Response	Ftp upgrade response JSON (show as follow Table-14.4.2)

SET	
URL	POST /API/Maintenance/FtpUpgrade/Set
Description	Get the information of Ftp Upgrade
Request Body	Ftp upgrade information JSON (show as follow Table-14.4.2)
Successful Response	The successful result response that described in 2.5

SET	
URL	POST /API/Maintenance/FtpUpgrade/Check
Description	Get the information of Ftp Upgrade
Request Body	none
Successful Response	Ftp upgrade result JSON (show as follow Table-14.4.3)

SET	
URL	POST /API/Maintenance/FtpUpgrade/Upgrade
Description	Start ftp upgrade
Request Body	none
Successful Response	The successful result response that described in 2.5

GET	
URL	POST /API/Maintenance/FtpUpgrade/Progress
Description	It is used to download percent
Request Body	none
Successful Response	Ftp upgrade result JSON (show as follow Table-14.4.4)

Table-11.4.1 (FtpUpgradeRange Info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ftp_addr	{"min_len":0,"max_len":64}	string	ftp server address
ftp_port	{"min":1, "max":65535}	int	ftp server port
username	{"min_len":0,"max_len":64}	string	username
user_pwd	{"min_len":0,"max_len":32}	string	login password
user_pwd_empty	{"type": "bool", "mode": "rw" }	bool	Whether the password is empty
ftp_path	{"min_len":0,"max_len":128 }	string	ftp upgrade file path
ftp_buttons	["Save","Refresh","Check"," Upgrade"]	string	Whether the control interface button is displayed

online_upgrade	{“type”:”bool” }	bool	It is used for the compatibility of old and new APIs, and it is a functional interface to judge whether to use FTP and HTTP online upgrade.
----------------	------------------	------	---

Table-11.4.2(FtpUpgrade Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ftp_addr		string	ftp server address
ftp_port		int	ftp server port
username		string	username
user_pwd		string	login password
user_pwd_empty		bool	Whether the password is empty
ftp_path		string	ftp upgrade file path

Table-14.4.3 (FtpUpgrade Response JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
has_new_firmware		bool	Is there a new upgrade firmware
current_ver		string	
new_version		string	

Table-11.4.4 (FtpUpgradeResponse JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
upgrade_percent	0~100	int	Upgrade progress
upgrade_state	“start” “checkVersion” ”downloadStart” “upgrade_download” ” “upgrade_upgrade_s ucceed” “downloadFailure” “upgrade_upgrading” ”	string	Upgrade status
upgrade_result	“ok” “finish”	string	Upgrade result

Example:

Request message:

POST/API/Maintenance/FtpUpgrade/Range HTTP/1.1

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: xx

Connection: keep-alive

```
{
  "result": "success",
  "data": {
    "ftp_addr": {
      "type": "string",
      "mode": "rw",
      "min_len": 0,
      "max_len": 64
    },
    "ftp_port": {
      "type": "int32",
      "mode": "rw",
      "min": 1,
      "max": 65535
    },
    "username": {
      "type": "string",
      "mode": "rw",
      "min_len": 0,
      "max_len": 64
    },
    "user_pwd": {
      "type": "string",
      "mode": "rw",
      "min_len": 0,
      "max_len": 32
    },
    "user_pwd_empty": {
      "type": "bool",
      "mode": "rw"
    },
    "ftp_path": {
      "type": "string",
      "mode": "rw",
      "min_len": 0,
      "max_len": 128
    },
  }
}
```

```

"cur_version": {
  "type": "string",
  "mode": "rw",
  "min_len": 0,
  "max_len": 64
},
"new_version": {
  "type": "string",
  "mode": "rw",
  "min_len": 0,
  "max_len": 64
},
"ftp_button": {
  "description": "Used to control whether the button is displayed!",
  "type": "string",
  "items": ["Save", "Refresh", "Check", "Upgrade"]}
}
}
}

```

11.5 Device Reboot

SET	
URL	POST /API/Maintenance/DeviceReboot/Set
Description	Restart the device immediately
Request Body	Device Reboot Information JSON (show as follow Table-11.5.1)
Successful Response	The successful result response that described in 2.5

Table-11.5.1 (Device Reboot Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
base_secondary_authentication		Json Object	Encryption auth, Table-12.7.2.4

Example:

Request message:

POST /API/ Maintenance/DeviceReboot/Set HTTP/1.1

```

{
  "data":{
    "secondary_authentication":"11111"
  }
}

```

```
}
```

Response message:

HTTP/1.1 200 OK

```
{  
  "version": "1.0"  
  "result": "success"  
}
```

11.6 Import/Export Parameter

11.6.1 Description

For both DVR Camara.

11.6.2 Syntax

GET	
URL	POST /API/Maintenance/ParamManagement/Get
Description	Export the param configure file
Request Body	show as follow Table-11.6.1
Response	JSON (show as follow Table-11.6.2)

SET	
URL	POST /API/Maintenance/ ParamManagement/Set
Description	Import the param configure file
Request Body	JSON (show as follow Table-11.6.3)
Successful Response	JSON (show as follow Table-11.6.4)

11.6.3 Parameters

Table-11.6.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
base_secondary_authentication		Json Object	Encryption auth, Table-12.7.2.4

Table-11.6.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
param		string	Base64 If no device is requested, there is no need

			to reply to this item
--	--	--	-----------------------

Table-11.6.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
secondary_authentication	Max length: 16 byte	string	Verify the admin user password for secondary authentication
param		string	Base64

Table-11.6.4 (Set Successful Response JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
state	"Success", "Failed"	string	Status of IPC operations

11.7 IPC Maintenance

11.7.1 Description

For IPC Camara.

11.7.2 Syntax

OPRATE	
URL	POST /API/IPCMaintaint/IPCReset/Get
Description	For NVR request to Restore IPC's default parameters
Request Body	
Successful Response	The successful result response that described in Table-11.7.1

OPRATE	
URL	POST /API/IPCMaintaint/IPCReboot/Get
Description	For NVR request to Reboot IPC
Request Body	
Successful Response	The successful result response that described in Table-11.7.1

OPRATE	
URL	POST /API/IPCMaintaint/IPCParamManagement/Get
Description	For NVR request to import or export IPC's parameters
Request Body	
Successful Response	The successful result response that described in Table-11.7.1

OPRATE	
URL	POST /API/IPCMaintaint/IPCReset/Set
Description	For NVR request to Restore IPC's default parameters
Request Body	Request JSON (show as follow Table-11.7.4)
Successful Response	The successful result response that described in Table-11.7.7

OPRATE	
URL	POST /API/IPCMaintaint/IPCReboot/Set
Description	For NVR request to Reboot IPC
Request Body	Request JSON (show as follow Table-11.7.4)
Successful Response	The successful result response that described in Table-11.7.7

OPRATE	
URL	POST /API/IPCMaintaint/IPCParamManagement/Import
Description	For NVR request to import IPC's parameters
Request Body	Request JSON (show as follow Table-11.7.4)
Successful Response	The successful result response that described in Table-11.7.7

OPRATE	
URL	POST /API/IPCMaintaint/IPCParamManagement/Export
Description	For NVR request to export IPC's parameters
Request Body	Request JSON (show as follow Table-11.7.4)
Successful Response	The successful result response that described in Table-11.7.7

OPRATE	
URL	POST /API/IPCMaintaint/IPCUpgrade/Get
Description	For NVR request to Upgrade IPC
Request Body	
Successful Response	The successful result response that described in Table-11.7.1

GET	
URL	POST /API/IPCMaintaint/IPCUpgrade/Token
Description	IPC upgrade get token
Request Body	show as follow Table-11.7.10
Successful Response	The successful result response that described in Table-11.7.11

POST	
URL	POST /API/IPCMaintaint/IPCUpgrade/Upgrade
Description	IPC Upgrade
Request Body	file data
Successful Response	The successful result response that described in Table-11.12

GET	
URL	POST /API/IPCMaintaint/FtpIpcUpgrade/Range
Description	Get the information of Ftp IPC Upgrade
Request Body	
Successful Response	FtpIPCUpgradeRange Info JSON (show as follow Table-11.7.14)

GET	
URL	POST /API/IPCMaintaint/FtpIpcUpgrade/Get
Description	Get the information of Ftp IPC Upgrade
Request Body	none
Successful Response	FtpIPCUpgrade Info JSON (show as follow Table-11.7.17)

SET	
URL	POST /API/IPCMaintaint/FtpIpcUpgrade/Set
Description	Get the information of Ftp IPC Upgrade
Request Body	Ftp IPC Upgrade Set Info JSON (show as follow Table-11.7.18)
Successful Response	The successful result response that described in 2.5

SET	
URL	POST /API/IPCMaintaint/FtpIpcUpgrade/Check
Description	Get the information of Ftp IPC Upgrade
Request Body	Ftp IPC Upgrade Check Info JSON (show as follow Table-11.7.19)
Successful Response	Ftp IPC Upgrade Check Info JSON (show as follow Table-11.7.19)

SET	
URL	POST /API/IPCMaintaint/FtpIpcUpgrade/Upgrade
Description	Start Ftp IPC upgrade
Request Body	Ftp IPC Upgrade Info JSON (show as follow Table-11.7.20)
Successful Response	The successful result response that described in 2.5

GET	
URL	POST /API/IPCMaintaint/FtpIpcUpgrade/Progress
Description	It is used to get upgrade proess
Request Body	Ftp IPC Upgrade Info JSON (show as follow Table-11.7.20)
Successful Response	Ftp IPC upgrade result JSON (show as follow Table-11.7.21)

11.7.3 Parameters

Table-11.7.1 (Channel Information JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
channel_info		JSON Object	SingleInformation JSON show as follow Table-11.7.2

Table-11.7.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-11.7.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-11.7.3(Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ip_address	Max length: 63byte	string	IP address/domain name
state	"Offline", "Online"	string	IPC status
software_version	Max length: 40byte	string	Firmware version
file_type	Max length: 64 byte	string	Upgrade file extension

Table-11.7.4 (Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
password	Max length: 16 byte	string	Verify the admin user password for secondary authentication
channel_info		JSON Object	SingleInformation JSON show as follow Table-11.7.5
base_secondary_authentication		Json Object	Encryption auth, Table-12.7.2.4

Table-11.7.5 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-11.7.6
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	

...		Json Object	
-----	--	-------------	--

Table-11.7.6(Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
reset_switch		bool	Restore default switch (this field is only used when IPC is restored to default)
reboot_switch		bool	Restart switch (this field is only used when IPC is restarted)
ImportExportSwitch		bool	Parameter export and import switch (this field is only used for IPC parameter export and import)
param		string	Base64 (this field is only used when importing parameters)

Table-11.7.7 (Set Successful Response JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON Object	SingleInformation JSON show as follow Table-11.7.8

Table-11.7.8 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-11.7.9
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-11.7.9 (Single Channel Information JSON)

COMMENT	VALUE		
	RANGE	TYPE	
state	"Success", "Failed"	string	Status of IPC operations
param		string	Base64 (this field is only used when exporting parameters)

Table-11.7.10 (IPC upgrade get token JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

file_name		string	upgrade file name
file_size		int	Upgrade file size
ipc_channels		int array	Channels that need to be upgraded
upgrade_head		int array	1KB of data before the upgrade file
base_secondary_authentication		Json Object	Encryption auth, Table-12.7.2.4

Table-11.7.11(IPC upgrade get token JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
upgrade_token		string	Upgrade token

Table-11.7.12 (IPC upgrade JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ChannelInfo		Json array	JSON show as follow Table-11.7.13

Table-11.7.13 (IPC upgrade JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel		string	channel number
result		string	upgrade result
reason		string	Upgrade error reason
error_code		string	The error code corresponding to the cause of the upgrade error

Upgrade error code:

upgrading	IPC equipment is being upgraded
ver_same	The current version is already the latest version
connet_close	Abnormal device connection
read_file_fail	error reading file
ver_err	Upgrade file mismatch
invalid_file	The file is invalid, please upload a legal file
file_error	Error creating file
not_exist	The device is offline
failed	Upgrade failed

Table-11.7.14 (Ftp IPC UpgradeRange Info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_max		int	Maximum number of channels
ftp_auto_upgrade		bool	ftp automatic upgrade IPC switch
check_for_updates		bool	FTP automatic upgrade IPC prompt switch

ftp_buttons	["Save","Refresh","Check","Upgrade"]	string	Whether the control interface button is displayed
check_chns		string array	IPC to be detected
channel_info		JSON Object	SingleInformation JSON show as follow Table-11.7.15
online_upgrade		bool	A new interface for judging whether to use FTP or HTTP to upgrade IPC online

Table-11.7.15 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
IP_CH1		Json Object	JSON show as follow Table-11.7.16
...		Json Object	

Table-11.7.16(Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
sup_ftp_auto_upgrade		bool	Whether the channel supports FTP upgrade
ftp_ipc_new_ver		bool	Is there a new version

Table-11.7.17(Ftp IPC Upgrade Info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ftp_auto_upgrade		bool	ftp automatic upgrade IPC switch
check_for_updates		bool	FTP automatic upgrade IPC prompt switch
channel_info		JSON Object	SingleInformation JSON show as follow Table-11.7.15

Table-11.7.18 (Ftp IPC Upgrade Set Info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ftp_auto_upgrade		bool	ftp automatic upgrade IPC switch
check_for_updates		bool	FTP automatic upgrade IPC prompt switch

Table-11.7.19 (Ftp IPC Upgrade Check Info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
check_chns		string array	IPC to be detected

Table-11.7.20 (Ftp IPC Upgrade Info JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
upgrade_chns		string array	IPCs that need to be upgraded

Table-11.7.21 (Ftp IPC Upgrade result JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cur_ipc		string array	IPC is currently being upgraded
upgrade_percent		int	FTP download IPC upgrade file progress
upgrade_state		string	FTP upgrade IPC status
upgrade_result		string	show as follow Table-11.7.22

Table-11.7.22(Ftp IPC Upgrade state JSON)

IPC_UPGRADE_FILE_DOWNLOADING	download upgrade file
IPC_WAIT_FOT_UPGRADING	The upgrade file download is complete, waiting for the upgrade
IPC_UPGRADE_FILE_DOWNLOAD_FAILED	Upgrade file download failed
IPC_UPGRADE_FINISH	A single IPC upgrade is complete
IPC_UPGRADE_ALL_FINISH	All IPC upgrades completed

Table-11.7.23(Ftp IPC Upgrade result JSON)

downloading	The upgrade file is downloading
wait_for_upgrading	The upgrade file download is complete, waiting for the upgrade
download_failed	Upgrade file download failed
success	IPC upgrade successful
upgrading	IPC equipment is being upgraded
ver_same	The current version is already the latest version
connet_close	Abnormal device connection
read_file_fail	error reading file
invalid_file	The file is invalid, please upload a legal file
ver_err	Upgrade file mismatch
failed	Upgrade failed
param_error	Parameter error
file_too_long	The upgrade file is too large
not_exist	The device is offline
all_finish	update completed

11.8 System Upgrade

GET	
URL	POST /API/Maintenance/SystemUpgrade/Token

Description	System upgrade get token
Request Body	show as follow Table-11.8.1
Successful Response	The successful result response that described in Table-11.8.2

POST	
URL	POST /API/Maintenance/SystemUpgrade/Upgrade
Description	Upgrade system
Request Body	file data
Successful Response	The successful result response that described in 2.5

POST	
URL	POST /API/Maintenance/SystemUpgrade/VersionCheck
Description	Version Check (nvr needs to verify the version information in the ftp and http upgrade configuration files of ipc)
Request Body	show as follow Table-11.8.3
Successful Response	The successful result response that described in 2.5

Table-11.8.1 (System upgrade get token JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
secondary_authentication	Max length: 16 byte	string	Verify the admin user password for secondary authentication
file_name		string	upgrade file name
file_size		int	Upgrade file size
upgrade_head		Int array	1KB of data before the upgrade file

Table-11.8.2 (System upgrade get token JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
upgrade_token		string	upgrade token
upgrade_timeout	0-3600	int	Upgrade timed out

Table-11.8.3 (System upgrade check version)

KEY	VALUE		COMMENT
	RANGE	TYPE	
FirewareVersion		string	software version
FirewarePack		string	package name

Upgrade error code:

in_user_interface	in the user interface
-------------------	-----------------------

updating	The board is being upgraded
lack_memory	Not enough storage
file_error	file error
no_need_upgrade	The current version is already the latest and does not need to be updated
size_error	File size exceeded. file_limit returns the maximum file size
file_invalid	The file is invalid, please upload a legal file

11.9 Developer Mode

RANGE	
URL	POST /API/Maintenance/DeveloperMode/Range
Description	Developer Mode
Request Body	
Successful Response	The successful result response that described in Table-11.9.1

GET	
URL	POST /API/Maintenance/DeveloperMode/Get
Description	Developer Mode
Request Body	
Successful Response	The successful result response that described in Table-11.9.1

SET	
URL	POST /API/Maintenance/DeveloperMode/Set
Description	Developer Mode
Request Body	show as follow Table-11.9.1
Successful Response	The successful result response that described in 2.5

DELETE	
URL	POST /API/Maintenance/DeveloperMode/Clear
Description	Developer Mode
Request Body	show as follow Table-11.9.2
Successful Response	The successful result response that described in 2.5

TOKEN	
URL	POST /API/Maintenance/DeveloperMode/Token
Description	Developer Mode
Request Body	None
Successful Response	Table-11.9.3

DOWNLOAD	
URL	POST /API/Maintenance/DeveloperMode/Download
Description	Developer Mode
Request Body	show as follow Table-11.9.4
Successful Response	None

Table-11.9.1 (Developer Mode JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
telnet_switch		bool	Telnet switch
export_disk_switch	“Shut Off” “Output To Terminal” “Output To Disk”	string	print log output
enable_export		bool	Whether to display the export button
enable_delete		bool	Whether to show the delete button

Table-11.9.2 (Developer Mode JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
base_secondary_authentication		Json Object	Encryption auth, Table-12.7.2.4

Table-11.9.3 (Developer Mode JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
x-download-token		string	

Table-11.9.4 (Developer Mode JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
x-download-token		string	
x-csrftoken		string	

Download implementation method:

1. Use /API/Maintenance/DeveloperMode/Token to obtain x-download-token, and its value is stored

in the header.

2. Send /API/Maintenance/DeveloperMode/Download with x-download-token and X-csrfToken to download the file.

11.10 Device Shutdown

SET	
URL	POST /API/Maintenance/DeviceShutdown/Set
Description	Shutdown the device immediately
Request Body	Device Shutdown Information JSON (show as follow Table-11.10.1)
Successful Response	The successful result response that described in 2.5

Table-11.10.1 (Device Reboot Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
base_secondary_authentication		Json Object	Encryption auth, Table-12.7.2.4

Example:

Request message:

POST /API/ Maintenance/DeviceShutdown/Set HTTP/1.1

```
{
  "data":{
    "base_secondary_authentication":"11111"
  }
}
```

Response message:

HTTP/1.1 200 OK

```
{
  "version": "1.0"
  "result": "success"
}
```

11.11 Http Upgrade

RANGE	
URL	POST /API/ Maintenance/AutoUpgrade/Range
Description	Http Upgrade

Request Body	None
Successful Response	The successful result response that described in Table-11.11.1

TEST	
URL	POST /API/ Maintenance/AutoUpgrade/Check
Description	Http Upgrade
Request Body	None
Successful Response	The successful result response that described in 2.5

SET	
URL	POST /API/ Maintenance/AutoUpgrade/Set
Description	Http Upgrade
Request Body	None
Successful Response	The successful result response that described in 2.5

Table-11.9.1 (Developer Mode JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
auto_buttons	“Check”	string	Whether to display the Check button

11.12 Defogging Fan

GET	
URL	POST /API/Maintenance/DefoggingFan/Get
Description	Get fan switch information
Request Body	None
Successful Response	Defogging Fan information JSON (show as follow Table-11.10)

SET	
URL	POST /API/Maintenance/DefoggingFan/Set
Description	Set fan switch
Request Body	Defogging Fan information JSON (show as follow Table-11.10)
Successful Response	The successful result response that described in 2.5

Table-11.10(Defogging Fan Information JSON)

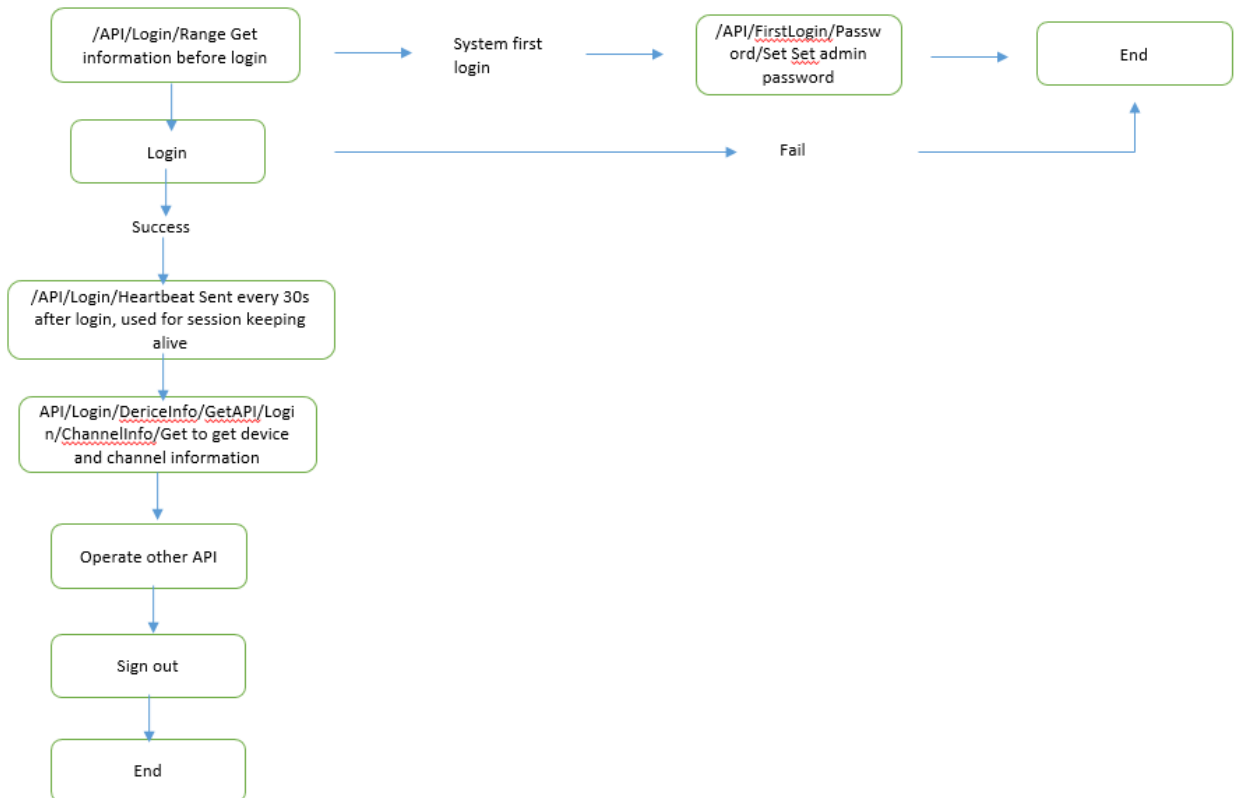
KEY	VALUE		COMMENT
	RANGE	TYPE	
defogging_fan		bool	Defogging Fan switch

12 LOGIN

12.1 Login

This chapter includes APIs such as login, logout, heartbeat, device information acquisition after login, and channel information acquisition.

Client login and subsequent process reference:



The /API/Login/Range interface does not authenticate, and returns whether the system is logged in for the first time (admin without a password), and some information for display. If it is the first login to the system, the client calls the /API/FirstLogin/Password/Set interface to set the admin password before logging in to the system.

12.1.1 Description

Range	
URL	POST/API/Login/Range
Description	Get the device info before login
Request Body	None

Successful Response	Range Information JSON (show as follow Table-12.1.1)
---------------------	--

Get	
URL	POST/API/Login/DeviceInfo/Get
Description	Get device information
Request Body	Device Information JSON (show as follow Table-12.1.2.x)
Successful Response	Device Information JSON (show as follow Table-12.1.2)

Get	
URL	POST/API/Login/ChannelInfo/Get
Description	Get channel information
Request Body	None
Successful Response	Device Information JSON (show as follow Table-12.1.3)

Set	
URL	POST/API/FirstLogin/Password/Set
Description	Device first login setting password
Request Body	show as follow Table-12.1.5
Successful Response	The successful result response that described in 2.5

Set	
URL	POST/API/Login/Password/Set
Description	Set the user password on the login interface (for ordinary users to log in and set the password for the first time, only for A01)
Request Body	show as follow Table-12.1.5
Successful Response	The successful result response that described in 2.5

Range	
URL	POST/API/Web/Login
Description	login
Request Body	show as follow Table-12.1.10
Successful Response	Range Information JSON (show as follow Table-12.1.6)

Range	
URL	POST /API/Login/Heartbeat
Description	Heartbeat API, send a heartbeat request every 30s after login to ensure that the heartbeat will not expire after timeout
Request Body	show as follow Table-12.1.9
Successful Response	The successful result response that described in 2.5

12.1.2 Syntax

Table-12.1.1(rangeJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
device_type	“IPC”	string	Used to determine the current device before login
cur_lang		string	current display language
username	Max length: 16 byte	string	username
password		string	password
pwd_tip	“ipc_tip” “nvr_tip” “dvr_tip”	string	Password prompt box content selection
password_tips		bool	Whether to display the password prompt
lang_strs		Json Object	List of languages supported for selection, the language types are shown in Table-12.1.1.1
default_lang		string	default display language
custom_name		int	Customer name, not A01
custom_logo		int	Customer logo, not used for A01
first_login_flag		bool	The sign of logging in to IE with an empty password, that is, whether it is the first time to log in
wifi_ip		string	A01 not used
web_title		string	"Web Viewer", not used for A01
site_version		string	Displayed version number
support_recover_password		bool	Support password retrieval function
use_recover_password		bool	Use the password retrieval function
default_username		string	When there is no password, it is displayed when changing the password
password_enc		bool	Whether the password is encrypted for transmission
http_api_version		String	The currently supported API version, the current version is: V1.0.

Table-12.1.1.1(lang_strs JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CHS	中文	string	
CHT	繁体中文	string	
CSY	ČEŠTINA	string	

DAN	DANSK	string	
ENU	English	string	
FIN	FINNISH	string	
FRA	FRANÇAIS	string	
DEU	DEUTSCH	string	
ELL	ΕΛΛΗΝΙΚΑ	string	
HEB	עברית	string	
HUN	MAGYAR	string	
ITA	ITALIANO	string	
JPN	日本語	string	
PLK	POLSKI	string	
PTG	PORTUGUÊS	string	
RUS	РУССКИЙ	string	
ESN	ESPAÑOL	string	
THA	ไทย	string	
SLV	SLOVENIA	string	
ROM	ROMANIAN	string	
BRG	BULGARIAN	string	
ARA	ARABIC	string	
HIN	HINDI	string	
VIE	VIETNAM	string	
HOL	HOLLAND	string	
TUR	TURKEY	string	
POS	PERSIAN	string	
SVE	SVENSKA	string	
KOR	KOREAN	string	
IND	INDONESIAN	string	
PTB	PORTUGUESE	string	
FLM	FLEMISH	string	
UKA	Українська	string	
SLK	Slovensky	string	

Table-12.1.2.x(request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
requestor	“NVR”	string	Request sources: eg: nvr, etc.
front_chn_index		int	The requested channel number. Usage scenario: When DVR/NVR is connected to multi-channel devices, bring this field when you want to get the front-end ipc device information. When the connected device receives this field, it will request the device information of

			the corresponding channel ipc and forward it to the connector.
--	--	--	--

Table-12.1.2(respond JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_num		int	Total number of channels
analog_channel_num		int	Total Analog Channels
max_analog_chn		int	The number of analog channels, if it is greater than 0, it is dvr/xvr
stream_profile	“Mainstream”, ”Substream”, “Mobilestream”	string array	Support several stream switching
support_ftp_upgrade		bool	Whether to support ftp upgrade
support_http_upgrade		bool	Whether to support http upgrade
ftp_upgrade_url		string	ftp upgrade url
http_upgrade_url		string	http upgrade url
support_cloud_upgrade		bool	Whether to support cloud upgrade
push_type	“VVPush”, “Baidu”“RSPush” “TutkPush”	string	Push type (A01 is not used)
default_stream	“Mainstream”, “Substream”	string	IE default stream type
client_logout_time	“0s”“30s”“1m”5m ““30m”	string	(A01 not used)
pushinfo_type	“Motion”“Io” “Videoloss”“Pir”	string array	
ptz_support	true,false	bool	
local_alarmin_num		int	On-board alarm input
local_alarmout_num		int	On-board alarm output
suggested_modify_pwd		bool	Is it recommended to change the password
support_face_config		bool	Whether to support face parameter query and setting
support_ie_down_snap		bool	Whether to support IE download board screenshot
support_human_vehicle_search		bool	Does it support pedestrian and vehicle search functions?
support_lpd_enhance		bool	Whether to enhance the function of vehicle recognition
preview_num		int	Main stream preview channel number
mac_addr	Max length:35byte	string	MAC address

support_io_record		bool	Whether to support determining whether the program has IO alarms based on the total number of IO alarms at the front and rear ends
support_manual_record		bool	Does it support manual recording type search
support_sound_record		bool	Sound alarm display flag: 0 hidden, 1: displayed
support_playback_new_rec_detail		bool	Support NewRecordDetailRsp (New_GetRecFileList) search
support_playback_new_rec_file_download		bool	Support NewRecordDetailRsp (New_GetRecFileList) video download, support video file lock query and lock setting
support_more_chn_playback		bool	Does the IE playback page support multi-channel playback (more than 4 channels)
support_video_cover_record		bool	Whether to display video masking alarm
support_smart_record		bool	Indicates whether or not the display intelligence analysis page
support_substream_playback		bool	substream playback
support_pir_record		bool	support PIR
support_flood_light		bool	support FloodLigh
support_speaker		bool	support the speaker
show_certificate_remain_time		bool	true: the validity period of the custom certificate is less than 8 days
certificate_remain_time	0-10	int	Valid days for custom certificates
media_port			see communication matrix
media_external_port			see communication matrix
local_ip		string	
upgrade_file_max_size	0-100	int	The maximum size of the board upgrade file, in MB
upgrade_head_trans_size	0-3	int	The length of the upgrade header information on the board side, in k
support_audio_volume		bool	Whether to support volume adjustment
talkback	"TalkHalf", "TalkFull"	string	
sound		bool	sound switch display
enable_encryption		bool	private protocol encryption
device_preview_ability	"manual_alarm" "face"	string array	Controls the set of capabilities displayed on the side of the preview

device_preview_AI_item	"FR" "Human" "Vehicle" "PID_Human" "PID_Vehicle" "LCD_Human" "LCD_Vehicle" "Non-Vehicle" "PID_Non-Vehicle" "LCD_Non-Vehicle"	string array	Control the set of options displayed on the side of the AI preview (after the update, this field is also used to judge the type in the playback smart image search, and the smart type judgment in the smart data statistics interface)
device_main_menu	"localSetting" "remoteSetting" "playback", "preview"	string array	Controls the set of capabilities displayed on the upper right side of the preview
support_ai_pic_report		bool	Does it support AI face, car and image reporting?
support_repeat_visitor		bool	Whether to support repeat customer function (for NVR only)
support_face_attendance		bool	Whether to support face attendance function (for NVR only)
support_heat_map		bool	Does it support the heat map function (for NVR only)
support_fisheye		bool	Whether to support fisheye function (for NVR only)
support_binoculars		bool	Whether to support binocular function (for NVR only)
support_cc_scenario		bool	Whether to support over-the-line scene application
support_attendance_scenario			Does it support face-based real-time attendance scene application?
support_face_attribute		bool	Whether to support face attributes
support_ai_pidlcd		bool	Whether to support AI's perimeter and tripwire
support_ai_cc		bool	Whether to support AI's crossing count
support_disarming		bool	Does it support one-key disarm function
dev_type		unsigned long long	Equipment type
device_type	Max length:24byte	string	Device model
dev_play_backcap		unsigned char	Playback mode: 0-old playback mode, 1-fast forward and fast rewind

			only take I frame mode, 2-fast forward and fast rewind jump I frame mode
support_PidLcd_search		bool	Does it support Pid Lcd search function
support_aac		bool	Whether to support AAC audio format
p2p_id		string	P2P ID (A01 is not used)
support_hls_server		bool	Support single-channel HLS code stream
videoloss_status		bool	Whether to display the missing logo
nocamera_display		bool	Whether to display nocamera text
FR_model_version		int	IPC face recognition model version
FD_model_version		int	IPC face detection model version
support_voice_prompts		bool	Whether to support voice broadcast
support_ANR		bool	Whether to support disconnected recording
support_siren_audio		bool	Whether to support white light alarm sound type selection
support_floodlight_color_image_ctrl		bool	Whether to support white light forced color function
support_param_import_export		bool	Whether to support parameter import and export
support_occlusion_detection		bool	Does it support lens occlusion alarm?
support_sound_detection		bool	Whether to support sound detection
support_osd_transparency		bool	Whether to support osd transparency switch
encode_type_ability	“H.264” “H.265” “H.264+” “H.265+”	string array	Supported encoding types
ptz_ability		Json Object	The supported ptz capabilities are shown in Table-12.1.11
localset		Json Object	See Table-12.1.12 for the default format of local settings
localsetRange		Json Object	See Table-12.1.13 for the format range of local settings
support_get_fr_group		bool	Whether to support face group acquisition, that is, support the GetId method
support_get_lpr_group		bool	Whether to support license plate group acquisition, that is, support the GetId method

manufacturer	Max length:32byte	String	manufacturer name
wireless_dev		bool	Determine whether it is a wireless device, wireless device true, wired false
support_top_online		bool	Support online channel auto-completion function
support_digitchannel_autofill		bool	Support digital channel auto-completion function
magic_channel		string array	Analog channels that have been converted to digital channels
accepted_compress_encodi	"gzip"	string array	API body compression, support gzip compression
support_partial_request		Json Object	API that supports paged data transfer

Table-12.1.3(channel info respond JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_param		Json array	Single Channel Information JSON show as follow Table-12.1.4

Table-12.1.4(single channel JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel		string	
channel_name		string	
channel_alias		string	channel alias
videoloss		bool	video loss sign
connect_status	"NotConfigured" "Online", "Offline" "Sleep" "NotPaired"	string	channel connection status
ability	A01 "FaceAlarm" "BehicleAlarm" "Fisheye" "Binoculars" "FloodLight", "AudioAlarm", "TalkHalf", "TalkFull" "Speaker"	Neutral IPC、NVR/DVR "Fisheye" "Binoculars" "Ptz" "NewPtz" (NVR/DVR not used) "HkPtz" "Iris" "Mainstream"	string array

	"AlarmOutNum" "Loud" "Ptz", "NewPtz" "HkPtz" "Iris" "Mainstream" "Substream" " Mobilestream "	"Substream" "Mobilestream" "FloodLight" "AudioAlarm" "AlarmOutNum" "TalkHalf" "TalkFull" "Color" "RedBlueLight" "NotAutoReconnect" "Pir" "Cover" "Eventstream"		
alarm_in_num			int	Alarm input supported by IPC channel
alarm_out_num			int	Alarm output supported by IPC channel
show_ptz_setting			bool	Control preview interface PTZ setting button display (A01 not used)
intelligent_ability		"CrossCount" "HeatMap" "SOD" "LCD" "PVD" "FD" "PID" "CrowdDensity" "LPD" "RSD" "QD" "AttributeDetect"		
talk_audio_ability		"G711_A" "G726_16KBPS_ASF" "G726_24KBPS_ASF" "G726_32KBPS_ASF" "G726_40KBPS_ASF" "AAC"		The intercom audio format supported by Ipc (if this field is not passed, the default is g711a, and if multiple fields are passed, the first one is used by default. Currently, only wireless models are used, and subsequent dvr and nvr can be reused if necessary)
wireless_ipc_type			int	Wireless IPC power supply type, battery 1, long-term power supply 2

Table-12.1.5(password request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
password	Max_length:16	string	
base_enc_password		Json Object	Encrypted Password, Table-12.7.2.3
base_secondary_authentication		Json Object	Encryption auth, Table-12.7.2.4
support_recover_pwd		bool	Whether to support password recovery settings

(1) Request

URL : POST / API / FirstLogin / Password / Set

```
{
  "version": "1.0",
  "data": {
    "base_enc_password": {
      "seq": 0,
      "peer_key" : "060kL4pXa9Br5Mt+/zsii4GllHfeXc/yR0jmP2Zfykzc=",
      "cipher" : "ObjEvTI4Lr8jsytAHx8bSXPnk7cuvIFYGQCjIUH2S/sVPnNQO"
    }
  }
}
```

(2) Response (JSON)

```
{
  "result":"success",
  "data": {}
}
```

Table-12.1.9(heartbeat request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
keep_alive		bool	Keep alive

Table-12.1.10(request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
oem_type		int	Login isolation flag, corresponding to the value passed by the client:114(B09)、144(B14)、148(B25)、298(B07)、221(B07)、309(B52-1)、310(B52)、313(B52-1\B52)

Table-12.1.11(ptz_ability JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ptz_version	"1.0" "2.0"	string	Dome new page version
btn_autofocus		bool	auto focus
quick_use	"Ptz_QuickUse_3DPosition" "Ptz_QuickUse_AutoFocus" "Ptz_QuickUse_PtzReset" "Ptz_QuickUse_WatchMode" "Ptz_QuickUse_ManualHumanTrace" "Ptz_QuickUse_LensReset"	string array	Shortcut button
iris_minus_add		bool	

Table-12.1.12(localset JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
RecFileType		string	Video download default format, neutral default is MP4
CapFileType		string	The default format of the snapshot, the neutral default is JPG
SwitchTime		int	Default interval time, neutral default interval is 10

Table-12.1.13(localsetRange JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
RecFileType	"RF" "MP4" "AVI"	string	Video Download Format Options
CapFileType	"JPG" "BMP" "PNG"	string	Snapshot Format Options
SwitchTime	1-60	int	interval time range

Table-12.1.14(partial_request JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
/API/IPCMaintaint/IPC Disk/Get		Json Object	API that supports page-by-page access to IPC SDK information
/API/IPCMaintaint/IPC Disk/Format		Json Object	Single JSON show as follow Table-12.1.15

Table-12.1.15

KEY	VALUE		COMMENT
	RANGE	TYPE	
type	"channel_list"	string	type of data
quota	0-32	int	The size of each page of data after paging
total		int	total size of data

Example:

Partial request(JSON)

```
support_partial_request:{
  "/API/IPCMaintaint/IPCDisk/Get":{
    "type": "channel_list",
    "quota" : 16,
    "total" : 256
  }
  "/API/IPCMaintaint/IPCDisk/Format":{
    "type": "channel_list",
    "quota" : 16,
    "total" : 256
  }
}
```

12.1.3 LOGIN

This interface is a login interface, and the client uses digest authentication to log in; when the login is successful, in the http header, two fields are returned, Set-cookie and X-csrfToken;

For example

Set-Cookie:

```
session=54f47bdcec65156e41b51d945bdcb4b6a9c77fae237b1c40d74c08db4a5eeb8f;HttpOnly;path=/
```

X-csrfToken: 93e2a23d518767b646ffd9a58b24b48665923ed797dbbb6b1e9760ebb9335b14

After the client logs in, when accessing static files, you need to bring cookies in the http header, for example

Cookie: session=54f47bdcec65156e41b51d945bdcb4b6a9c77fae237b1c40d74c08db4a5eeb8f;

When accessing the API, you also need to include the X-csrf token field in the http header, for example:

X-csrf token: 93e2a23d518767b646ffd9a58b24b48665923ed797dbbb6b1e9760ebb9335b14.

After the login is successful, the server maintains the session callback, which is 5 minutes by default and can be configured. When the client accesses the server with a session, the server refreshes the session expiration time.

The digest authentication algorithm currently uses rfc 2617, which will be modified to rfc7616 and compatible with rfc2617

See Table-12.1.6 for the return of successful login (NVR, DVR, IPC are not implemented temporarily)

Table-12.1.6 (login success json)

KEY	VALUE		COMMENT
	RANGE	TYPE	
last_login_time	Max_length:64	string	Last successful login time (time stamp)
last_login_ip	Max_length: 64	string	Last login successful IP address
fail_login_count		int	The number of failed logins since the last successful login
pwd_remain_time		int	The number of days remaining on the password (IPC defaults to 90 days)

Login failure return see Table-12.1.8

Table-12.1.7 error_code

error_code	COMMENT
no_permission	No remote login permission
login_failed_or_block	Locked after five or more failed logins
black_ip	This IP is set to blacklist
verify_failed	Incorrect username or password within five times - login failed
device_reboot	system is restarting

Table-12.1.8 (login failed json)

KEY	VALUE		COMMENT
	RANGE	TYPE	
block_remain_time		int	Indicates the time at which the system will be locked for too many login errors

12.1.4 Recover Password

RANGE	
URL	POST /API/RecoverPassword/Range
Description	It is used to get the RecoverPassword config parameters
Request Body	NULL
Successful Response	Channel Information JSON (show as follow Table-12.1.4.1)

GET	
URL	POST /API/RecoverPassword/Get
Description	It is used to get the RecoverPassword parameters
Request Body	Channel Information JSON (show as follow Table-12.1.4.1)
Successful Response	The successful result response that described in 2.5

SET	
URL	POST /API/RecoverPassword/Set
Description	It is used to set the RecoverPassword parameters
Request Body	Channel Information JSON (show as follow Table-12.1.4.1)
Successful Response	The successful result response that described in 2.5

RANGE	
URL	POST /API/RecoverPassword/Authorization/Range
Description	It is used to get the RecoverPassword Authorization parameters
Request Body	NULL
Successful Response	Channel Information JSON (show as follow Table-12.1.4.2)

GET	
URL	POST /API/RecoverPassword/Authorization/Get
Description	It is used to get the RecoverPassword Authorization parameters
Request Body	Channel Information JSON (show as follow Table-12.1.4.2)
Successful Response	The successful result response that described in 2.5

SET	
URL	POST /API/RecoverPassword/Authorization/Set
Description	It is used to set the RecoverPassword Authorization parameters
Request Body	Channel Information JSON (show as follow Table-12.1.4.2)
Successful Response	The successful result response that described in 2.5

SET	
URL	POST /API/RecoverPassword/Certificate/Export
Description	It is used to set the RecoverPassword Authorization parameters
Request Body	Channel Information JSON (show as follow Table-12.1.4.1)

Successful Response	The successful result response that described in 2.5
---------------------	--

SET	
URL	POST /API/RecoverPassword/Email/Send
Description	It is used to set the RecoverPassword Authorization parameters
Request Body	A button for send email_code
Successful Response	The successful result response that described in 2.5

Table-12.1.4.1 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
answer_flag		bool	Support question and answer verification to retrieve
email_flag		bool	Support email retrieval
certificate_flag		bool	Support certificate retrieval
super_pwd_flag		bool	Support super password
email	1-64	string	Mail
questions		Json array	JSON show as follow Table-12.1.4.1.x
answers		Json array	JSON show as follow Table-12.1.4.1.y
mode	"Answer", "Email", "Certificate", "SuperPwd"	string	Only the selected password recovery method will be displayed
email_code	1-64	string	email verification code
certificate	1-128	string	certificate code
super_pwd	1-16	string	Super password
system_time		string	date time
mac_address	0-35	string	Mac address

Table-12.1.4.1.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
1		Json Object	Questions corresponding to each verification
...		Json Object	
3		Json Object	
...		Json Object	
5		Json Object	
...		Json Object	

Table-12.1.4.1.y (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

1		string	Answers to questions for 12.1.4.1.x
...		string	
3		string	
...		string	
5		string	
...		string	

Example:

ALL request and response message:

1. /API/RecoverPassword/Range

Response example:

```
{
  "result": "success",
  "data": {
    "answer_flag": {
      "type": "bool"
    },
    "email_flag": {
      "type": "bool"
    },
    "certificate_flag": {
      "type": "bool"
    },
    "super_pwd_flag": {
      "type": "bool"
    },
    "questions": {
      "type": "array",
      "size": 3,
      "items": [
        {
          "type": "int32",
          "items": [
            1,
            2,
            3,
            4,
            5,
            6,
            7,
            8,
            9,
            10,
```

```

        11,
        12,
        13,
        14,
        15
    ]
}
]
},
"answers": {
    "type": "array",
    "size": 3,
    "items": [
        {
            "type": "string",
            "min_len": 1,
            "max_len": 64
        }
    ]
},
"email": {
    "type": "string",
    "min_len": 1,
    "max_len": 64
}
}
}

```

2. /API/RecoverPassword/Get #get problem questions

Response instance:

```

{
    "result": "success",
    "data": {
        "questions": [
            5,
            4,
            3
        ]
    }
}

```

3. /API/RecoverPassword/Set #Set the configuration of retrieving the password, log in for the first time and go

through the Set interface

send instance:

```
{
  "data": {
    "answer_flag": true,
    "email_flag": true,
    "certificate_flag": true,
    "super_pwd_flag": true,
    "questions": [5, 4, 3],
    "answers": ["111", "222", "333"],
    "email": "123456@qq.com"
  }
}
```

4. /API/RecoverPassword/Authorization/Range

example response:

```
{
  "result": "success",
  "data": {
    "mode": {
      "type": "string",
      "items": [
        "Answer",
        "Email",
        "Certificate",
        "SuperPwd"
      ]
    },
    "questions": {
      "type": "array",
      "size": 3,
      "items": [
        {
          "type": "int32",
          "items": [
            1,
            2,
            3,
            4,
            5,
            6,
          ]
        }
      ]
    }
  }
}
```

```

        7,
        8,
        9,
        10,
        11,
        12,
        13,
        14,
        15
    ]
}
]
},
"answers": {
    "type": "array",
    "size": 3,
    "items": [
        {
            "type": "string",
            "min_len": 1,
            "max_len": 64
        }
    ]
},
"email_code": {
    "type": "string",
    "min_len": 1,
    "max_len": 64
},
"certificate": {
    "type": "string",
    "min_len": 1,
    "max_len": 128
},
"super_pwd": {
    "type": "string",
    "min_len": 1,
    "max_len": 16
}
}
}

```

5. /API/RecoverPassword/Authorization/Get #Get verification methods and questions
example response:

```

{
  "result": "success",
  "data": {
    "mode": "Answer",
    "questions": [
      5,
      4,
      3
    ]
  }
}

```

6. /API/RecoverPassword/Authorization/Set #Verify each way
send example:

```

{
  "data":{
    "mode": "Answer",
    "answers": ["111", "222", "333"],
    "password": "admin123."
  }
}

```

7. /API/RecoverPassword/Certificate/Export #export certificate
example response:

```

{
  "result": "success",
  "data": {
    "certificate": "001612531125"
  }
}

```

8. /API/RecoverPassword/Email/Send #send email

12.2 Get Device Page

12.2.1 Description

Get the layout information of the client interface.

Get	
URL	POST/API/Login/DevicePage/Get
Description	Get the page list

Request Body	none
Successful Response	Login Information JSON (show as follow Table-12.2.1)

12.2.2 Syntax

Table-12.2.1(JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
		json	Show Table-12.2.2

Table-12.2.2(JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
main		Json array	Show Table-12.2.3

Table-12.2.3(JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
title	Max length 64	string	A menu “Display,”Record”,”Alarm”, ”AI”, ”N etwork”,”Storage”,”System”,”Intellige nt”
sub		Json array	Show Table-12.2.4

Table-12.2.4(JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
title	Max length 64	string	Secondary menu “
pages		Json array	Show Table-12.2.5

Table-12.2.5(JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
title	Max length 64	string	Level 3 menu
page	Max length 64	string	Confirm this with the client

12.2.3 Example

Request message:

POST /API/Login/DevicePage/GetHTTP/1.1

Accept: */*

Content-type: application/json

```
{
  "version": "1.0",
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: xx

Connection: keep-alive

```
{
  "version": "1.0",
  "main": [ //first level menu
    {
      "title": "Channel",
      "sub": [ //secondary menu
        {
          "title": "Channel",
          "pages": [ //Level 3 menu, corresponding to a single page
            {
              "title": "IP Channels", //name
              "page": "chn_ipChn" //web page
            },
            {
              "title": "Protocol Manage",
              "page": "chn_protocolMng"
            }
          ]
        },
        ...
      ]
    },
    ...
  ]
}
```

The neutral standard configuration is as follows:

Main menu [Level 1 menu]

["Channel","Record","Alarm","AI","Network","Device","System","Intelligent"]

Submenu [secondary menu]

...

Level 3 menu [web page]

Title	Page
Channel related, page chn_	
Analog Channels	chn_analogChn
IP Channels	chn_ipChn
Wireless Camera	chn_wireChn
Protocol Manage	chn_protocolMng
Live	chn_osd
Image Control	chn_imgCtrl
Video Cover	chn_videoCover
Video Crop	chn_videoCrop
PTZ	chn_ptz
FloodLight	chn_floodlight
PIR	chn_pir
PID	chn_pid
LCD	chn_lcd
SOD	chn_sod
PD	chn_pd
FD	chn_fd
CC	chn_cc
Sound_Detection	chn_sound_detection
Occlusion_Detection	chn_occlusion_detection
Intelligent_Analysis	chn_intelligent_analysis
Stream related, page stm_	
Mainstream	stm_main
Substream	stm_sub
Mobilestream	stm_mob
Capture	stm_capture
Recording related, page rec_	
Record Parameters	rec_param
Schedule, page schedules_	
Record Schedule	schedules_rec
Capture Schedule	schedules_capture
Email Schedule	schedules_email
Ftp Schedule	schedules_ftp
CloudSchedule	schedules_cloud
Intelligent Schedule	schedule_intelligent

Ai_Schedules	ai_schedules
Network related, page net_	
General	net_general
PPPoE	net_pppoe
Wireless	net_wireless
Port Configuration	net_port_conf
DDNS	net_ddns
Email	net_email
Ip Filter	net_ipFilter
Alarm related, page alarm_	
IO	alarm_io
Motion	alarm_motion
Exception	alarm_exception
PTZLinkage	alarm_ptzLinkage
PIR	alarm_pir
PID	alarm_pid
LCD	alarm_lcd
SOD	alarm_sod
PD	alarm_pd
FD	alarm_fd
CC	alarm_cc
Sound Detection	alarm_sound_detection
Occlusion Detection	alarm_occlusion_detection
Event related, page Event(IPC)	
Setup subpage	
Motion	chn_motion
PIR	chn_pir
FloodLight	chn_floodlight
Sound Detection	chn_sound_detection
Occlusion Detection	chn_occlusion_detection
Voice Alarm	alarm_voice
Alarm subpage	
Motion	alarm_motion
IO	alarm_io
PIR	alarm_pir
Sound Detection	alarm_sound_detection
Occlusion Detection	alarm_occlusion_detection
Ai related, page AI	

FD(Face Detection)	chn_ai_fd
PD&VD(Human & Vehicle Detection)	chn_ai_pvd
PID(Perimeter Intrusion Detection)	chn_ai_pid
LCD(Line Crossing Detection)	chn_ai_lcd
CC(Cross Counting)	chn_ai_cc
HM(Heat Map)	chn_heat_map
SOD(Goods Lost Legacy)	chn_ai_sod
CD(Crowd Density Detection)	chn_ai_cd
QD(Queue Lenght Detection)	chn_ai_qd
Model Configuration	ai_model_configuration
Database Management	ai_database_management
FR(Face Recognition)	alarm_ai_fr
AD(Attribute Detection)	alarm_ai_attribute
PD&VD(Human&Vehicle Detection)	alarm_ai_pvd
PID(Perimeter Intrusion Detection)	alarm_ai_pid
LCD(Line Crossing Detection)	alarm_ai_lcd
CC(Cross Counting)	alarm_ai_cc
FD(Face Detection)	alarm_ai_fd
FA(Face Attendance)	alarm_ai_fa
SOD(Goods Lost Legacy)	alarm_ai_sod
CD(Crowd Density Detection)	alarm_ai_cd
QD(Queue Lenght Detection)	alarm_ai_qd
LPD(License Plate Detection)	alarm_ai_lpd
RSD(Rare Sound Detection)	alarm_ai_rsd
Face Detection	ai_fd_statistics
Human&Vehicle Detection	ai_pvd_statistics
Heat Map Statistics	ai_hm_statistics
Cross Counting Statistics	ai_cc_statistics
Storage related, page storage_	
Disk	storage_disk
Disk Group	storage_diskGroup
Cloud	storage_cloud
Ftp	net_ftp
System related, page sys_	
General	sys_general
Date and Time	sys_date_time
Output Configuration	sys_output_conf
Multi-User	sys_user
Upgrade	sys_upgrade
IPC Upgrade	sys_ipcUpgrade

IPC Load Default	sys_ipcLoad_default
Reboot IPC	sys_ipcAuto_reboot
Information	sysInfo_base
Channel Information	sysInfo_chn
Record Information	sysInfo_rec
Network State	sysInfo_net
Others	
Auto Reboot	sys_auto_reboot
Parameter Management	sys_param_mgt
Load Default	sys_load_default
Log	sys_maint_log
Sys Defogging Fan	sys_defogging_fan

12.3 Preview Control

12.3.1 Description

Get	
URL	POST/API/PreviewChannel/PTZ/Get
Description	Get PTZControlinformation
Request Body	ChannelRequest JSON(show as follow Table-12.3.2.x)
Successful Response	Information JSON (show as follow Table-12.3.2.2)

Get	
URL	POST/API/PreviewChannel/Floodlight2AudioAlarm/Get
Description	Get LightSireninformation
Request Body	ChannelRequest JSON(show as follow Table-12.3.2.x)
Successful Response	Information JSON (show as follow Table-12.3.2.3)

Get	
URL	POST/API/PreviewChannel/DualTalk/Get
Description	Get Dualtalk information
Request Body	ChannelRequest JSON(show as follow Table-12.3.2.x)
Successful Response	Information JSON (show as follow Table-12.3.2.5)

Get	
URL	POST/API/PreviewChannel/ManualAlarm/Get
Description	Get Manulalarm information
Request Body	None

Successful Response	Information JSON (show as follow Table-12.3.2.4)
---------------------	--

Control	
URL	POST/API/PreviewChannel/PTZ/Control
Description	Control PTZ
Request Body	Request JSON (show as follow Table-12.3.2.2)
Successful Response	The successful result response that described in 2.5

Control	
URL	POST/API/PreviewChannel/PTZ/Control/Progress
Description	GET PTZ status
Successful Response	Response JSON (show as follow Table-12.3.2.2)

Control	
URL	POST/API/PreviewChannel/Floodlight2AudioAlarm/Set
Description	Control LightSiren
Request Body	Request JSON (show as follow Table-12.3.2.3)
Successful Response	The successful result response that described in 2.5

Control	
URL	POST/API/PreviewChannel/DualTalk/Set
Description	Control DualTalk
Request Body	Request JSON (show as follow Table-12.3.2.5)
Successful Response	The successful result response that described in 2.5

Control	
URL	POST/API/PreviewChannel/ManualAlarm/Set
Description	Control Manulalarm
Request Body	Request JSON (show as follow Table-12.3.2.4)
Successful Response	The successful result response that described in 2.5

12.3.2 Syntax

Table-12.3.2.x (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"... "CH1x" "IP_CH1"... "IP_CH1x" "WIFI_CH1"... "WIFI_CH1x"	String	

	The number of channels depends on the capabilities of the device.		
--	---	--	--

Table-12.3.2.1(ptz_paramJSON)

KEY	VALUE	TYPE	COMMENT
	RANGE		
channel		string	
ptz_version	"1.0" "2.0"		PTZ new page version
cmd	PTZ_CMD_UP = 1, PTZ_CMD_DOWN, PTZ_CMD_LEFT, PTZ_CMD_RIGHT, PTZ_CMD_UPLEFT, PTZ_CMD_UPRIGHT, PTZ_CMD_DOWNLEFT, PTZ_CMD_DOWNRIGHT, PTZ_CMD_ZOOMIN, PTZ_CMD_ZOOMOUT, PTZ_CMD_FOCUSNEAR, PTZ_CMD_FOCUSFAR, PTZ_CMD_IRISOPEN, PTZ_CMD_IRISCLOSE, PTZ_CMD_AUTOSCAN, PTZ_CMD_CRUISE, PTZ_CMD_CHANGECRUISETIME, PTZ_CMD_SETPRESET, PTZ_CMD_CLEARPRESET, PTZ_CMD_CALLPRESET, //20 PTZ_CMD_AUXILIART_OPEN, PTZ_CMD_AUXILIART_CLOSE, PTZ_CMD_LOCKFOCUS, PTZ_CMD_LINESCANSETA, PTZ_CMD_LINESCANSETB, PTZ_CMD_LINESCANSETSPEED , /* speed >=0 && speed <= PTZ_CMD_LINESCANSMALLRAD, // Inner arc PTZ_CMD_LINESCANBIGRAD,	int	(Only A01) (only control)

	<pre>// Outer arc PTZ_CMD_DEFAULT, PTZ_ZOOM_POSITION, PTZ_FOCUS_POSITION, PTZ_ZOOM_MOVE, PTZ_FOCUS_MOVE, PTZ_AUTO_FOCUS, PTZ_AREA_FOCUS, PTZ_REFRESH_INFOR, PTZ_CALIBRATION_SETTING, PTZ_TOUR_START, PTZ_TOUR_STOP, PTZ_PATTERN_START, PTZ_PATTERN_STOP, PTZ_PATTERN_RECORD_START , PTZ_PATTERN_RECORD_STOP, PTZ_NET_OSD, PTZ_3D_POSITION, PTZ_LIGHT, PTZ_RAIN, PTZ_CALCPADNUM_SETTING, PTZ_CMD_NEWSCENECALIB, PTZ_PRIVACYMODE, PTZ_TRACE_MODE, PTZ_Count</pre>		
cmd	<pre>“Ptz_Cmd_Up”, “Ptz_Cmd_Down”, “Ptz_Cmd_Left”, “Ptz_Cmd_Right”, “Ptz_Cmd_UpLeft”, “Ptz_Cmd_UpRight”, “Ptz_Cmd_DownLeft”, “Ptz_Cmd_DownRight”, “Ptz_Cmd_ZoomAdd”, “Ptz_Cmd_ZoomMinus”, “Ptz_Cmd_FocusAdd”, “Ptz_Cmd_FocusMinus”, “Ptz_Cmd_IrisAdd”, “Ptz_Cmd_IrisMinus”, “Ptz_Cmd_CircleCenter”, “Ptz_Cmd_Cruise”, “Ptz_Change_CruiseTime”, “Ptz_Cmd_AddPreset”,</pre>	String	(Not forA01)(only control)

<p> “Ptz_Cmd_ClearPreset”, “Ptz_Cmd_CallPreset”, “Ptz_Cmd_AuxiliartOpen”, “Ptz_Cmd_AuxiliartClose”, “Ptz_Cmd_LockFocus”, “Ptz_LineScan_StartPoint”, “Ptz_LineScan_EndPoint”, “Ptz_Cmd_LineScanSetSpeed”, “Ptz_LineScan_CruiseStart”, “Ptz_LineScan_CruiseStop”, “Ptz_Btn_Default”, “Ptz_Zoom_Position”, “Ptz_Focus_Position”, “Ptz_Zoom_Move”, “Ptz_Focus_Move”, “Ptz_Btn_AutoFocus”, “Ptz_AreaFocus”, “Ptz_Btn_Refresh”, “Ptz_CalibRationSetting”, “Ptz_Tour_Start”, “Ptz_Tour_Stop”, “Ptz_Pattern_CruiseStart”, “Ptz_Pattern_CruiseStop”, “Ptz_Pattern_RecordStart”, “Ptz_Pattern_RecordStop”, “Ptz_Net_Osd”, “Ptz_3Dposition”, “Ptz_Light”, “Ptz_Rain”, “Ptz_CalcpadNumSetting”, “Ptz_Cmd_NewSceneCalib”, “Ptz_Cmd_DefCruise_Start”, “Ptz_Cmd_DefCruise_Stop”, “Ptz_WatchPoint_Add”, “Ptz_Privacy_Mode”, “Ptz_Trace_Mode” </p>		
--	--	--

Table-12.3.2.2(respond ptz_paramJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cmd	“Ptz_Cmd_Up”, “Ptz_Cmd_Down”, ...	string	Cmd show as table 12.3.2.1

	... "Ptz_Privacy_Mode", "Ptz_Trace_Mode"		
state	"Stop""Start"	String	Used for button press and release
zoom_step	1,5,20	Int	Zoom adjust steps
focus_step	1,5,20	Int	Focus adjust steps
zoom_slider	Depending on the len	Int	Zoom slider
focus_slider	Depending on the len	Int	Focus slider
isctl		Bool	Gray screen or not: true: not, false: yes
speed	0-10; 0-100(DVR/NVR only)	int	(only control)
preset_point	0-255	int array	The maximum length of the array is 255, Every array is a preset point, Every array value (0-255) means the cruise time of preset point, 0 means no preset point.
cruise_state		bool	Cruise status of preset point
line_scan_state		bool	Cruise status of line scan
trace_preset_point		Array	TOUR preset point setted
preset_point_no	1-255	int	(only control)
ctl_stop		bool	true: stop PTZ control (only control)
preset_point_time	1-255	int	Cruise time of single preset point (only control)
preset_point_name	1-31	String	Name of preset point, ranges from 1 to 31 bytes
zoom_minus_add	"Ptz_Cmd_ZoomMinus" "Ptz_Cmd_ZoomAdd"	object	Range used to display zoom operation buttons
focus_minus_add	"Ptz_Cmd_FocusMinus" "Ptz_Cmd_FocusAdd"	object	Range used to display focus operating buttons
iris_minus_add	"Ptz_Cmd_IrisMinus" "Ptz_Cmd_IrisAdd"	object	Range used to display IRIS operating buttons
btn_autofocus	"Ptz_Btn_AutoFocus"	object	used to display autofocus operating button
btn_default	"Ptz_Btn_Default"	object	used to display default operating button
btn_refresh	"Ptz_Btn_Refresh"	object	used to display refresh operating button
preset_point_cruise_button	"Ptz_Cmd_Cruise"	object	used to display cruise button
is_req_progress		bool	if to progress the cruise command
ptz_direction_control	"Ptz_Cmd_Up" "Ptz_Cmd_Down" "Ptz_Cmd_Left" "Ptz_Cmd_Right" "Ptz_Cmd_UpLeft"	object	Range used to display the direction

	"Ptz_Cmd_UpRight" "Ptz_Cmd_DownLeft" "Ptz_Cmd_DownRight" "Ptz_Cmd_CircleCenter"		
advanced_mode	"Mode_Preset_Point" "Mode_Watch_Mode" "Mode_Line_Scan" "Mode_Trace" "Mode_Pattern_Scan"	object	Range used to display modes of advanced function
trace_number	0-3	int	Tour 1-4 trace
pattern_scan_number	0-3	int	Pattern scan 1-4 trace
pattern_scan_number_isset	0-3	Array	Pattern scan 1-4 if set or not
line_scan_speed	"Low", "Middle", "High"	String	Line scan speed
line_scan_area		bool	If set line scan
trace_interval	"5s""6s""7s"....."59s" "60s"	String	Tour trace interval
utc_protocol		string	Utc control protocol
utc_cmd	"Coax_Cmd_Menu" "Coax_Cmd_Up" "Coax_Cmd_Left" "Coax_Cmd_Right" "Coax_Cmd_Down"	string	Utc control command
current_cruise_mode	"Mode_Default_Cruise" "Mode_Preset_Point" "Mode_Watch_Mode" "Mode_Line_Scan" "Mode_Trace" "Mode_Pattern_Scan" "Mode_Restore_Btn"	string	Current cruise mode
belt_times_use	0-100	int	Belt's life, above 90, indicating life insufficient; 100 unable to cruise
quick_use	"Ptz_QuickUse_3D Position" "Ptz_QuickUse_AutoFocus"	string	Used to display shortcut buttons

	"Ptz_QuickUse_PtzReset" "Ptz_QuickUse_WatchMode" "Ptz_QuickUse_ManualHumanTrace" "Ptz_QuickUse_LensReset"		
preset_point_obj	0-256	Array	Number of preset array JSON show as follow Table-12.3.2.6
watch_mode_mode	"Mode_Default_Cruise" "Mode_Watch_Point" "Mode_Line_Scan" "Mode_Trace" "Mode_Pattern_Scan"	string	Selections in watch mode
watch_mode_num	0 - 255	int	num number of common modes in Watchmode (except line scans)
watch_mode_num	"low" "middle" "high"	string	Speed of line scan in watch mode

Table-12.3.2.3(light_siren_paramJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel		string	
floodlight_switch	true false	bool	Floodlight real-time switch, false: close true: open
floodlight_mode	0 - 1	int	Floodlight mode, 0: lighting, 1: blink
floodlight_value	1 - 100	int	Brightness of floodlight
floodlight_strobe_frequency	0 - 2	array	Enable in blink mode, 0:low; 1:middle; 2:high
floodlight_value_range	1-100	int	Floodlight brightness range
audioAlarm_switch	true false	bool	Audio real-time switch false: close true: open
audioAlarm_value	1 - 10	int	Audio value
audioAlarm_value_range	1-10	int	Audio value range
operation_type	"Floodlight", "AudioAlarm" "RedBlueLight" "All"	string	Operation type

redBlueLight_switch		bool	redBlueLight_switch
---------------------	--	------	---------------------

Table-12.3.2.4(manulalarm_param JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Local->1		bool	Local->1 channel alarm switch
.....		
Local->x		bool	Local->x channel alarm switch
IP_CH1->1		bool	IP_CH1->1 channel alarm switch
.....		
IP_CHx->x		bool	IP_CHx->x channel alarm switch

Table-12.3.2.5(dualtalk_param JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string	
action		int	1-open, 0-close

Table-12.3.2.6(preset_point_obj_paramJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
no	1-256	string	Number of preset point,max 255points can be configured
name	1-31	string	Name of array preset:range from 1-31bytes
add	true false	bool	If the preset point is set


```

213", "add": false}, {"no": 214, "name": "point
215", "add": false}, {"no": 216, "name": "point
217", "add": false}, {"no": 218, "name": "point
219", "add": false}, {"no": 220, "name": "point
221", "add": false}, {"no": 222, "name": "point
223", "add": false}, {"no": 224, "name": "point
225", "add": false}, {"no": 226, "name": "point
227", "add": false}, {"no": 228, "name": "point
229", "add": false}, {"no": 230, "name": "point
231", "add": false}, {"no": 232, "name": "point
233", "add": false}, {"no": 234, "name": "point
235", "add": false}, {"no": 236, "name": "point
237", "add": false}, {"no": 238, "name": "point
239", "add": false}, {"no": 240, "name": "point
241", "add": false}, {"no": 242, "name": "point
243", "add": false}, {"no": 244, "name": "point
245", "add": false}, {"no": 246, "name": "point
247", "add": false}, {"no": 248, "name": "point
249", "add": false}, {"no": 250, "name": "point
251", "add": false}, {"no": 252, "name": "point
253", "add": false}, {"no": 254, "name": "point
255", "add": false}], "watch_mode_time": 20, "watch_mode_mode": "Mode_Default_Cruise", "watch_mode_num": 0, "line_scan_area": false, "line_scan_speed": "low", "trace_interval": 20, "trace_number": 0, "trace_preset_point": [[], [], [], []], "pattern_scan_number_isset": [true, false, false, false], "pattern_scan_number": 0, "current_cruise_mode": "No_Cruise"}
}

```

12.3.3.2 requestcontrol ptz

POST/API/PreviewChannel/PTZ/Control HTTP/1.1

Accept: */*

Content-type: application/json

```

{
  "version": "1.0",
  "data": {
    "channel": "IP_CH4",
    "preset_index": 1,
    "command_flag": true,
    "speed": 5,
    "cmd": 19,
    "ctl_stop": false
  }
}

```

POST/API/PreviewChannel/PTZ/Control HTTP/1.1

Accept: */*

Content-type: application/json

```
{
  "version": "1.0",
  "data": {
    "channel": "CH1",
    "cmd": "Ptz_PresetPoint_ChangeName",
    "speed": 5,
    "zoom_step": 1,
    "zoom_slider": 50,
    "focus_step": 1,
    "focus_slider": 50,
    "preset_point_no": 3,
    "preset_point_name": "sdcsdc",
    "preset_point_time": 3
  }
}
```

POST/API/PreviewChannel/PTZ/Control HTTP/1.1

Accept: */*

Content-type: application/json

```
{
  "version": "1.0",
  "data": {
    "channel": "CH1",
    "cmd": "Ptz_Cmd_AddPreset",
    "speed": 5,
    "zoom_step": 1,
    "zoom_slider": 50,
    "focus_step": 1,
    "focus_slider": 50,
    "preset_point_no": 3,
    "preset_point_name": "sdcsdc",
    "preset_point_time": 3
  }
}
```

POST/API/PreviewChannel/PTZ/Control HTTP/1.1

Accept: */*

Content-type: application/json

```
{
  "version": "1.0",
  "data": {
```

```
    "channel": "CH1",
    "cmd": "Ptz_WatchMode_CruiseStop",
    "speed": 5,
    "zoom_step": 1,
    "zoom_slider": 50,
    "focus_step": 1,
    "focus_slider": 50,
    "watch_mode_time": 15,
    "watch_mode_mode": "Mode_Pattern_Scan",
    "watch_mode_num": 2
  }
}
```

POST/API/PreviewChannel/PTZ/Control HTTP/1.1

Accept: */*

Content-type: application/json

```
{
  "version": "1.0",
  "data": {
    "channel": "CH1",
    "cmd": "Ptz_WatchMode_CruiseStart",
    "speed": 5,
    "zoom_step": 1,
    "zoom_slider": 50,
    "focus_step": 1,
    "focus_slider": 50,
    "watch_mode_time": 15,
    "watch_mode_mode": "Mode_Line_Scan",
    "watch_mode_num": "high"
  }
}
```

POST/API/PreviewChannel/PTZ/Control HTTP/1.1

Accept: */*

```
{
  "version": "1.0",
  "data": {
    "channel": "CH1",
    "cmd": "Ptz_QuickUse_WatchModeStart",
    "speed": 5,
    "zoom_step": 1,
    "zoom_slider": 50,
    "focus_step": 1,
    "focus_slider": 50
  }
}
```

POST/API/PreviewChannel/PTZ/Control HTTP/1.1

Accept: */*

```
{
  "version": "1.0",
  "data": {
    "channel": "CH1",
    "cmd": "Ptz_Tour_Start",
    "speed": 5,
    "zoom_step": 1,
    "zoom_slider": 50,
    "focus_step": 1,
    "focus_slider": 50,
    "trace_number": 1,
    "trace_preset_point": [
      [
        1,
        2
      ],
      [
        5,
        8
      ],
      [],
      []
    ],
    "trace_interval": 15
  }
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "result": "success",
  "data": {}
}
```

12.4 LogOut

Request message:

POST/API/Web/Logout HTTP/1.1

Accept: */*

Cookie: session=xxxxx

X-csrfToken: xxxx

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: xx

Connection: keep-alive

Content-Type: application/json

```
{  
  "result": "success"  
}
```

12.5 Download root certificate File(A01)

12.5.1 Description

Get	
URL	POST/API/Login/CertFile/Token
Description	Get token
Request Body	None
Successful Response	Information JSON (show as follow Table-12.5.1.1)

Get	
URL	POST/API/Login/CertFile/Get
Description	Download root certificate file
Request Body	Request Data (show as follow example 12.5.3.2)
Successful Response	Root certificate data (show as follow example 12.5.3.2)

12.5.2 Syntax

Table-12.5.1.1(Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
token	0-32	string	Token value

12.5.3 Example

12.5.3.1 Gettokenvalue

```
POST/API/Login/CertFileToken HTTP/1.1
Accept: */*
Content-type: application/json
{}
```

Response message:

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "result": "success",
  "data": {
    "token": " 6aea9ef8744f"
  }
}
```

12.5.3.2 Download root certificate data

```
POST/ API/Login/CertFile HTTP/1.1
Accept: */*
Content-disposition: attachment; filename="root_cert.cer"
```

```
Body:
Token=6aea9ef8744f
```

Response message:

```
HTTP/1.1 200 OK
Content-disposition: attachment; filename="root_cert.cer"
```

```
Body:
-----BEGIN CERTIFICATE-----
MIIEDjCCAnagAwIBAgIBATANBgkqhkiG9w0BAQsFADAvMQswCQYDVQQGEwJDTjEP
MA0GA1UECgwGSHVhd2VpMQ8wDQYDVQQDDAZIdWF3ZWkwHhcNMjAwMjEzMTEw
WhcNMzAwMjEzMTEwWjAvMQswCQYDVQQGEwJDTjEPMA0GA1UECgwGSHVhd2Vp
MQ8wDQYDVQQDDAZIdWF3ZWkwggGiMA0GCSqGSIb3DQEBAQUAA4IBjwAwggGKAoIB
gQCj8ai3pFrK914LlzhWLvWqQavhj+3yro00+4K/TugZIYd/DRA8d1Zx4Js+6nPA
SuO/64ej2uYszHUVVXUYRI/BBVfF+V4zUW7C6YhArsuAfOFFcmzUC1vtrgBCYpsq
JWQl6asXeduMwOILtJG8gErcm0QkMRWxl1FQ7v3i9CyYoWLF89Ck3w6b6N8A3dIC
```

DNFLDZxLVtl+fWKb41Ed9vpmQ9jB0d2wGptA7JpnhUe5Vvo527MuoA/qN69KjylE
 1nU/+jlnUIEuEai0cZH+IQ5Aog0zhyUHgOp1RDxWYZedEjK3IXg+KOXDa/9s6BKy
 xgkqZCON6zLmS3Szi4AVzsEM5YXs08CjHi3vs7ecijfdD+NOwxCKPsr4GA27PWS
 9wMFGGgeR7GDR+S55GGjDM1UMq8Lj3PRN0AQh58xNbFc+BZwcAmw4veCxyeY4Trl
 0wopk3u1q+c+UNI3VTsTiNJQvLreFR9NRkDvYIcOFudyQSAwapIm/XDPA4QjqDPL
 BVMCAwEAAaM1MDMwCwYDVR0PBAQDAgKkMA8GA1UdEwEB/wQFMAMBAf8wEwYDVR0I
 BAwwCgYIKwYBBQUHAwEwDQYJKoZIhvcNAQELBQADggGBAAMYCNGfKZGvnsHqWTP5
 rtCyH6kIxxbYvS+nUJZCQXCDCDTPDx2qz8CPi6TFMcR+rQbkKSRynU5kYev3bkDfZ
 eJpBfhErnzoz/06QeR9xcJRqDEAKH1S+No5Jx+KtRm4AkDKMEbRx6ZMZycKGq5q
 pWPmjulm/uM+PWtQSsoBa1NVUZH2OIIoKxNYJMU7vOzHcozuUIzJLPNzaMRC2Kg
 b6/XVurS7NjxGLs/QN7KZBa+fW2Bse2C0FbKtOiw143G/eFF37ASa+SqOCOCS51i
 75Q77jNPJxkapyrzGPe2Y68A2BxfNEzvdeB06CNtnOFMpJbkxGEGQsvrPeFzp6g
 3eFEmpgvgSuffgt3C5IL39hDMcDnDxHqc9B99ZbW8LyCGsYXLjXWzEm6dkJIXF0z
 /7e7SxqOXHeBj8FGjHzUcURlhiLEf+lOvbVwBgGZ2Tjnb4Y2bbNLA5h6egovxCjy
 k8SoeEMZ+uJEnZKugx2w4G6ATB73BvNT2PSj1yEgtnLOFw===
 -----END CERTIFICATE-----

12.6 Account Rules

12.6.1 Description

Get	
URL	POST/API/AccountRules/Get
Description	Get user rule restrictions
Request Body	None
Successful Response	Information JSON (show as follow Table-12.6.2.1)

12.6.2 Syntax

Table-12.6.2.1 (Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
username		Object	namig rule, Information JSON show as follow Table-12.6.2.2
password		Object	Password rule, Information JSON show as follow Table-12.6.2.2

Table-12.6.2.2 (Username/Password Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

min_length	1-8	int	minimum length
max_length	16-16	int	maximum length
character_combinations_num	1-4	int	At least the number of character combinations
character_combinations	"ALPHA", "alpha", "Alpha", "digit", "special"	array	Character combination list
special		string	Special character
not_same_username		bool	Can be the same as the user name
deny_password		array	The password cannot be set the same as in the array

12.6.3 Example

Request message:

POST /API/AccountRules/Get HTTP/1.1

Accept: */*

Content-type: application/json

```
{
  "version": "1.0",
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "result": "success",
  "data": {
    "username": {
      "min_length": 1,
      "character_combinations": [
        "Alpha",
        "digit",
        "special"
      ],
      "special": "_"
    },
    "password": {
      "min_length": 8,
      "not_same_username": true,

```



```

        "character_combinations_num": 2,
        "character_combinations": [
            "ALPHA",
            "alpha",
            "digit",
            "special"
        ],
        "special": "~!@#$%^&*()-_+=+\\[[{}];\"',<.>/?"
    }
}
}

```

12.7 Request pubkey or randbyte

12.7.1 Description

Get	
URL	POST/API/Maintenance/TransKey/Get
Description	The user password is requested to transmit the encrypted key or a random number for the PBKDF2_SHA256
Request Body	Information JSON (show as follow Table-12.7.2.1)
Successful Response	Information JSON (show as follow Table-12.7.2.2)
Get	
URL	POST/API/Login/TransKey/Get
Description	The user password is requested to transmit the encrypted key. Used before login when device isnot activated.
Request Body	Information JSON (show as follow Table-12.7.2.1)
Successful Response	Information JSON (show as follow Table-12.7.2.2)

(1) Request

URL : POST/ API / Login / TransKey / Get

```

{
  "version": "1.0",
  "data": {
    "type": [
      "base_x_public"
    ]
  }
}

```

(2) Response (JSON)

```

{

```

```

"result": "success",
"data": {
  "Key_lists": [
    {
      "type": "base_x_public",
      "key": "0mkSa7soBJ/WewVVn3J8Y/TsI9+MvAY+8Elds6UqNTyw=",
      "seq": 0
    }
  ]
}

```

12.7.2 Syntax

Table-12.7.2.1 (Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
type	"base_salt", " base_x_public "	array	The maximum length of the array is 5." base_salt", " base_x_public " Can request more than one

Table-12.7.2.2 ((Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
key_lists		JSON array	Salt and x_pubkey Information JSON show as follow Table-12.7.2.3

Table-12.7.2.3 (Salt and x_pubkey Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
type	" base_salt ", " base_x_public "	string	"base_salt": PBKDF2_SHA256 Secondary certification. " base_x_public " Password transmission
key	0-1024	string	The random number is hex string, which needs to be converted. The public key is plaintext. Base64 transmission is adopted.
iter	0-1000000	int	Number of iterations of PBKDF2_SHA256
seq	0-1000000	int	Each key corresponds to a seq, which must be passed in at the same time as

			the encrypted data is transmitted
--	--	--	-----------------------------------

Table-12.7.2.3 (base_enc_password JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cipher	0-1024	string	Encrypted password value(Encryption using the X25519 key derived by the 12.7.1 API), Use base64 transfer
seq	0-1000000	int	The seq returned using the 12.7.1 API
peer_key	0-1024	string	The X25519 public key encrypted by the client, Use base64 transfer

Table-12.7.2.4 (base_secondary_authentication JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cipher	44	string	Encrypted password value(Encryption using the key returned by the 12.7.1 API), Use base64 transfer。 PBKDF2 outlen=32, base64 is 44
seq	0-1000000	int	The seq returned using the 12.7.1 API

12.7.3 Example

Request message:

POST/API/Maintenance/TransKey/Get

Accept: */*

Content-type: application/json

```
{
  "version": "1.0",
  "data": {
    "type": [
      "base_salt",
      " base_x_public "
    ]
  }
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```

{
  "result": "success",
  "data": {
    "Key_lists": [
      {
        "type": "base_salt",
        "key": " NZLNvhZyfhHdVsUGL8GfVbUifQHIDiJ3oueTLiAGN54= ",
        "iter": 10086,
        "seq": 19
      },
      {
        "type": " base_x_public",
        "key": " 0mkSa7soBJ/WewVVn3J8Y/TsI9+MvAY+8Elds6UqNTyw= ",
        "seq": 0
      }
    ]
  }
}

```

12.7.4 INSTRUCTION

Passwords are transmitted in various scenarios. For example, when formatting a hard disk, enter the current password for secondary authentication, set a user password, and set an email password.

Scenarios	Key type requested	Algorithm	API requesting
Secondary certification	base_salt	PBKDF2_SHA256	/API/Maintenance/TransKey/Get
First time to set admin pasword	base_x_public	RSA	/API/Login/TransKey/Get /API/FirstLogin/Password/Set
Modify the common user's initial password	base_x_public	RSA	/API/Maintenance/TransKey/Get /API/Login/Password/Set
Add a user Modify use name and password	base_x_public	RSA	/API/Maintenance/TransKey/Get /API/SystemConfig/User/Set

(1) The client uses the PBKDF2_SHA256 encryption password for secondary authentication as follows:

```

salt = base64_decode(key);
outlen = 32;
enc = PBKDF2(salt, pwd, iter, outlen); -----> openssl PKCS5_PBKDF2_HMAC , and EVP_sha256()
cipher = base64_encode(enc);

```

(2) The client uses X25519 to encrypt and transmit the password as follows:

encryption algorithm: aes_256_gcm

source of Key and iv: the secret is calculated through X25519, and then the key and iv are expanded through hkdf

```
evp_pkey = (my_pri, my_pub) = generate_X25519_key;
key_1 = remove "0" from key;
device_peer_key = base64_decode(key_1);
secret = X25519_derive(device_peer_key, evp_pkey);
aes_256_gcm_key = hkdf_expand(secret, label = "expand key", out_len = 16);
aes_256_gcm_iv = hkdf_expand(secret, label = "expand iv", out_len = 12);
```

Output:

```
peer_key = "0" + base64(EVP_PKEY_get_raw_public_key (evp_pkey)) // Here to pass the public key to the
devicecipher = base64( aes_256_gcm(password, key, iv) + aes_256_gcm_iv + tag )
// tag_len = 16, key_len = 16, iv_len = 12
```

Below is the web page we provide to calculate cipher:

1. Calculate the cipher for secondary authentication cipher

[pbkdf2.html](#)

2. Calculate the cipher for the password encryption field

[x25519.html](#)

These pages are designed to help you determine whether the cipher calculation is correct

Instructions for using the gadget:

1. [pbkdf2.html](#)

Open the gadget page, enter the value of word, salt, iter in turn, and click commit to start running. Where word is the password that needs to be encrypted, salt is the random string used to mess with the real password, and iter is the number of iterations. The final result of the run will be printed in Result, and the encrypted value is the calculated ciphertext.

2. [x25519.html](#)

Open the gadget web page, input the value of word, peer public in turn, and click commit to start running. Where word is the password to be encrypted and peer public is the public key obtained from the nvr. The final result of the run is printed in Result. In encrypted, the value of public is the public key generated by the client, which will be sent to the nvr for decryption, and the value of encrypted is the calculated ciphertext.

Guidelines for the development of specific secondary authentication scenarios are provided in Appendixes A.2
 Guidance for the development of specific scenarios for password encryption are provided in Appendixes A.3

13 MutexParam

13.1 MutexParam

GET	
URL	POST/API/MutexParam/Get
Description	Get the system log information
Request Body	Request JSON (show as follow Table-13.1.1)
Successful Response	Factory Information JSON (show as follow Table-13.1.1)

Table-14.1.1(Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_C HN_NUM	array	Channel number to enable video uploading to cloud
max_cloud_video_upload_num	MAX_CLOUD_VI DEO_RECORD_N UM	int	Maximum number of video recordings of cloud storage

14 Function

14.1 Snapshot

Range	
URL	POST /API/Snapshot/Range
Description	Get the information of Snapshot
Request Body	None
Successful Response	Snapshotinformation JSON (show as follow Table-14.1.1)

GET	
URL	POST /API/Snapshot/Get
Description	Get the information of Snapshot
Request Body	Snapshotinformation JSON (show as follow Table-14.1.2)

Successful Response	Snapshot information JSON (show as follow Table-14.1.3)
---------------------	---

Table-14.1.1 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON array	Single Channel Information JSON show as follow Table-14.1.2

Table-14.1.2 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel		string	Channel number
snapshot_resolution	Snapshot resolution at most 1920*1080	string	Snapshot resolution
reset_session_timeout		bool	Reset session timeout,true by default The field is valid only when preview/replay session timeout option is turned on.

Table-14.1.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel		string	Channel number
snapshot_resolution	Snapshot resolution at most 1920*1080	string	Snapshot resolution
img_time		int	UTC time
img_encodes	“Base64”	string	Codec type
img_format	“Jpeg”	string	Image type
img_data		string	Image data

Example:

Request message:

POST/API/Snapshot/Range HTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Access-Control-Allow-Origin: *

Content-Length: xx
Connection: keep-alive

```
{
  "version": "1.0",
  "channel_info": {
    "type": "array",
    "min_size": 0,
    "max_size": 8,
    "items": [
      {
        "channel": "IP_CH1",
        "snapshot_resolution": {
          "description": "The resolution of the image is no more than 1920*1080",
          "type": "string",
          "mode": "rw",
          "items": [
            "640 x 480",
            "1280 x 720",
            "1920 x 1080"
          ]
        }
      }
    ]
  },
  {
    "channel": "IP_CH3",
    "snapshot_resolution": {
      "description": "The resolution of the image is no more than 1920*1080",
      "type": "string",
      "mode": "rw",
      "items": [
        "320 x 240",
        "640 x 480",
        "1280 x 720",
        "1920 x 1080"
      ]
    }
  }
],
  "result": "success"
}
```


14.2 Request I Frame

Note: NVR privateprotocol change to API requesting IFame command

Range	
URL	POST /API/RequestIDR
Description	Request I Frame
Request Body	Channel Information JSON (show as follow Table-14.2.1)
Successful Response	The successful result response that described in 2.5

Table-14.2.1 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
chn_no		int	
stream_type	"Mainstream", "Substream", "Mobilestream"	string	

Example:

Request message:

POST/API/RequestIDR HTTP/1.1

```
{
  "data": {
    "chn_no": 0,
    "stream_type": "Mainstream"
  }
}
```

14.3 ANR

SET	
URL	POST/API/ANRConfig/GetClientMac
Description	Get Client Mac
Request Body	Request JSON(show as follow Table-14.3.1)
Successful Response	The successful result response that described in 2.5

SET	
URL	POST/API/ANRConfig/SetANRInfo
Description	Set ANR Info
Request Body	Request JSON(show as follow Table-14.3.2)
Successful Response	The successful result response that described in 2.5

GET	
URL	POST/API/ANRConfig/GetANRTimeInfo

Description	Get ANR TimeInfo
Request Body	Request JSON(show as follow Table-14.3.2)
Successful Response	Parameter Information JSON (show as follow Table -14.3.3)

Table-14.3.1(Mac info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
mac_address		string	Client Mac

Table-14.3.2(ANR info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
enable		int	ANR switch
device_flag		string	Client Mac

Table-14.3.3

KEY	VALUE		COMMENT
	RANGE	TYPE	
start_time		unsigned int	
end_time		unsigned int	

Example:

Request message:

POST/ API/ANRConfig/GetANRTimeInfo

```
{
  "data": {
    "device_flag ": "88-DF-58-18-4F-47 "
  }
}
```

Response message:

HTTP/1.1 200 OK

```
{
  "result": "success",
  "data": {
    "start_time":1481290144,
    "end_time":1481290249
  }
}
```

14.4 ETR

SET	
URL	POST/API/StreamConfig/EventStreamState/Set

Description	Get Client Mac
Request Body	Request JSON(show as follow Table-14.4.1)
Successful Response	The successful result response that described in 2.5

Table-14.4.1(ETR info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
EventStreamState		bool	Event stream state

15 Event

15.1 event check

Get	
URL	POST /API/Event/Check
Description	Get the push server parameters
Request Body	See Table-15.1.1
Successful Response	Show as Table-15.1.2

Table-15.1.1(request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
reader_id		unsigned int	Optional,without input, the local side checks all alarm information and outputs readerID
sequence		unsigned int	Optional,without input, the local side checks all alarm information and outputs readerID
lap_number		unsigned int	Optional,without input, the local side checks all alarm information and outputs readerID

Table-15.1.2(alarm JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
reader_id		int	Optional , when request no readerID , readerID created and return.
sequence		int	The current latest sequence
lap_number		unsigned	Current sequence position lap number

		int	
subscribe_record_sigin		bool	Optional, subscribe to the video flag by default. If the field passes true or does not pass this field, it means that the video flag is subscribed. If the field passes false, it means that the video flag is not subscribed
subscribe_intelligence		bool	Subscribe to smart alert events by default. Fields that pass true or do not pass this field indicate subscription, and fields that pass false indicate no subscription to smart alert events
alarm_list		Json array	Single alarm show as table
heat_alarm	“HeatAlarm”	string	Optional, when there is no alarm list, output heartbeat alarm message
videoloss_status		bool	Dynamic display videoloss status

Table-15.1.2(alarm JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
system_alarm		Json array	Optional see table15.1.3 (Optional)
channel_alarm		Json array	Single channel alarm see table15.1.5 (Optional)
time		string	“MM/DD/YYYYhh:mm:ss”

Table-15.1.3 System alarm(Json)

KEY	VALUE		COMMENT
	RANGE	TYPE	
device_name	Max_length:31	string	Device name (optional)
hdd		json	see table15.1.4 (optional)

Table-15.1.4 hdd alarm(Json)

KEY	VALUE		COMMENT
	RANGE	TYPE	
hdd_alarm_type	“Ok”,”Error”	string	HDD alarm type
hdd_alarm_info	“Full” “Bad” ”Unformat” “Readonly” “Warning”,	string	HDD error alarm info

	"NoDisk"		
--	----------	--	--

Table-15.1.5 single channel alarm json

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel		string	
motion_alarm		bool	(Optional)
record_flag		json	Record flag (optional) See Table-15.1.6
camera_connect_status		json	(optional) Table-15.1.7
io_alarm		bool	(optional)
pir_alarm		bool	(optional)
low_power		bool	Low power (optional)
charge_power		bool	Power charging (optional)
channel_name		string	(optional+)
Floodlight_Audio Alarm		json	(optional) Table-15.1.8
ptz_alarm		json	(optional) Table-15.1.9
videoloss		bool	(optional)
int_alarm		Json Object	(optional) Table-15.1.10
alarm_state		Json array	(optional) Table-15.1.10
motion_smart_alarm		Json Object	Table-15.1.13
alarm_out_state		Json Array	(NVR/DVR) see Table-15.1.14
wireless_ipc_type		int	(Wireless only) wireless IPC type,battery 1, DC power 2
remote_pair_state	Max_length:20	string	(Wireless only) report the result of remote pairing Pair success Pair fail Timeout Pair is not supported

Table-15.1.6record flag camera json

KEY	VALUE		COMMENT
	RANGE	TYPE	
m	"R""G"	string	Motion Alarm "R": alarm recording status "G": alarm

			not recording status
i	“R”“G”	string	IO Alarm “R”:alarm recording “G”:alarm not recording
p	“R”“G”	string	PIR Alarm “R”:alarm recording “G”:alarm not recording
s	“R”“G”	string	Intelligent Alarm “R”:alarm recording “G”:alarm not recording
r	“SR”“MR”	string	Recording “SR”:schedule recording “MR”:manual recording
h	“R”	string	SDcardAlarm “R” :HD alarm recording
c	“R”“G”	string	VideoTampering “R”:alarm recording “G”:alarm not recording

Table-15.1.7camera_connect_status json

KEY	VALUE		COMMENT
	RANGE	TYPE	
connect_status	“NotConfigured” “Online”, ”Offline” “Sleep” “NotPaired”	string	Channel connecting
ability	A01	Normal IPC、 NVR/DVR	
	“FaceAlarm” “VehicleAlarm” “Fisheye” “Binoculars” “Light”, ”Audio”, ”DualTalk”, “Siren”, “Ptz”, “Newptz” “Iris” “Hkptz” “Mainstream”	“Fisheye” “Binoculars” “Ptz” “NewPtz” (not for NVR/DVR) “HkPtz” “Iris” “Mainstream” “Substream” “Mobilestream” “FloodLight” “AudioAlarm” “AlarmOutNum” “TalkHalf” “TalkFull”	

	“Substream” “IO”	“Color” “RedBlueLight” “NotAutoReconnect” “Pir” “Cover”		
protocol	Max length: 15byte		string	IPC access protocol
input_num				Alarm input
output_num				Alarm output
intelligent_ability	“CrossCount” “HeatMap” “SOD” “LCD” “PVD” “FD” “PID” “CrowdDensity” “LPD” “RSD” “QD” “AttributeDetect”			

Table-15.1.8 Floodlight param json

KEY	VALUE		COMMENT
	RANGE	TYPE	
floodlight_switch	true false	bool	Floodlight real-time switch false: close true: open
flood_light_value	1 - 100	int	Floodlight brightness
floodlight_value_range	1 - 100	Object	Floodlight brightness range
audioAlarm_switch	true false	bool	Audio real-time switch false: close true: open
audioAlarm_value	1 - 10	int	Audio value
audioAlarm_value_range	1 - 10	Object	Audio value range

Table-15.1.9 ptz alarmjson

KEY	VALUE		COMMENT
	RANGE	TYPE	
cur_zoom_value		int	Zoom current position (not in use yet)
zoom_step	0 - 2	int	0: 1step 1: 5steps 2: 20steps(not in use yet)
cur_focus_value		int	Focus current position(not in use yet)
focus_step	0 - 2	int	0: 1 steps 1: 5steps 2: 20steps(not in use yet)

			in use yet)
auto_focus_state		int	0:focus finished, 1: focusing(not in use yet)
shift_range		int	(not in use yet)
ptz_cruise_state		bool	Preset point cruise state
ptz_line_scan_state		bool	Line scan cruise state

Table-15.1.10 alarm param json

KEY	VALUE		COMMENT
	RANGE	TYPE	
alarm_val	true false	bool	
int_subtype	video_tamper sod lcd pid pd fd sound avd pd_vd cc	string	“video_tamper”: video cover detetion “sod”:Objects lost or left detection “lcd”: line crossing detection “pid”: Perimeter intruding detection “pd”: Pedestrian detection “fd”: Face Detection “sound”: Sound detection “avd”: avd detection “pd_vd”: Person&Vehicle detection “cc”: cross counting

Table-15.1.11 error_code

error_code	COMMENT
readerID_invalid	
position_invalid	

Table-15.1.12 talkback_alarm json

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	1 - 127	int	Intercome channel
talkback_close	true false	bool	If close takback
error_code	"localuser_operating_cannot_talkback" "localuser_close_talkback"	string	Talkback closed reason localuser_operating_cannot_talkback:the user is setting interface localuser_close_talkback:user close the talk back

Table-15.1.13 motion samrt alarm json

KEY	VALUE		COMMENT
	RANGE	TYPE	
MDtime		string	“MM/DD/YYYYhh:mm:ss”
MDState		string	

		array	
--	--	-------	--

Table-15.1.14 alarm_out_state alarm json

KEY	VALUE		COMMENT
	RANGE	TYPE	
Local->1		bool	Local->1 channel alarm switch
.....		
Local->x		bool	Local->x channel alarm switch
IP_CH1->1		bool	IP_CH1->1 channel alarm switch
.....		
IP_CHx->x		bool	IP_CHx->x channel alarm switch

15.1 event check instruction and demo

Event notification, using a circular query. That is, the client sends a query, the server replies, and the client immediately inquires again. Attention: the client can query without interruption, or request after a while, but the interval can not exceed 30s.

The server maintains a reader id , sequence, lap_number for each alarm link, no parameter is passed on the first request, and the server sends the current state of the device and return reader_id. Each subsequent request by the client uses the last return from the server (reader_id, sequence, lap_number) to request the alert status, if there is no current alarm, the server will block for about 10s and return a heat_alarm heartbeat message.

If the server does not receive the next request for more than 30 seconds, the reader_id will be invalid, the server returns an error, in which case the client remakes the first request.

For alarms such as motion_alarm, note that there are true and false values. When it is true, it means that there is an alarm. When it is false, there is no alarm.

(1) First request guidance:

```
POST /API/Event/Check?2020-10-17%2014:21:33 HTTP/1.1
```

```
Content-Type: application/json; charset=UTF-8
```

```
Accept: application/json, text/javascript, */*; q=0.01
```

```
X-csrfToken: 14559384d58d55d9d80bf4baf048684f366c77905d665d99ba1bde2cdf81b08
```

```
Content-Length: 2438
```

```
Cookie: session=cc3ec99c6f1295489c86f8842e1dd719c194c637af5b965a23932c41188a3acc
```

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

Reply:

```
HTTP/1.1 200 OK
```

```
Content-Type: application/json
```

Content-Length: 2976

```
{ "result": "success", "data": { "alarm_list": [ { "time": "10/17/2020 14:16:31", "system_alarm": { "device_name": "NVR800", "hdd": { "hdd_alarm_type": "Ok", "hdd_alarm_info": "" }, "channel_alarm": { "channel": "CH1", "motion_alarm": false, "io_alarm": false, "ptz_alarm": { "ptz_cruise_state": false, "ptz_line_scan_state": false }, "videoloss": false, "camera_connect_status": { "connect_status": "Online", "offline_reason": "Online", "ability": [ "Ptz", "Mainstream", "Substream" ], "protocol": "Onvif", "alarm_in_num": 0, "alarm_out_num": 0 }, "channel_name": "ok", "camera_type": "dome", "record_flag": { "r": "SR" } }, { "channel": "CH2", "motion_alarm": false, "io_alarm": false, "ptz_alarm": { "ptz_cruise_state": false, "ptz_line_scan_state": false }, "videoloss": true, "camera_connect_status": { "connect_status": "NotConfigured", "offline_reason": "NotConfigured", "ability": [], "alarm_in_num": 0, "alarm_out_num": 0 }, "channel_name": "", "camera_type": "Uknown", "record_flag": {} }, { "channel": "CH3", "motion_alarm": false, "io_alarm": false, "ptz_alarm": { "ptz_cruise_state": false, "ptz_line_scan_state": false }, "videoloss": true, "camera_connect_status": { "connect_status": "NotConfigured", "offline_reason": "NotConfigured", "ability": [], "alarm_in_num": 0, "alarm_out_num": 0 }, "channel_name": "", "camera_type": "Uknown", "record_flag": {} }, { "channel": "CH4", "motion_alarm": false, "io_alarm": false, "ptz_alarm": { "ptz_cruise_state": false, "ptz_line_scan_state": false }, "videoloss": true, "camera_connect_status": { "connect_status": "NotConfigured", "offline_reason": "NotConfigured", "ability": [], "alarm_in_num": 0, "alarm_out_num": 0 }, "channel_name": "", "camera_type": "Uknown", "record_flag": {} }, { "channel": "CH5", "motion_alarm": false, "io_alarm": false, "ptz_alarm": { "ptz_cruise_state": false, "ptz_line_scan_state": false }, "videoloss": true, "camera_connect_status": { "connect_status": "NotConfigured", "offline_reason": "NotConfigured", "ability": [], "alarm_in_num": 0, "alarm_out_num": 0 }, "channel_name": "", "camera_type": "Uknown", "record_flag": {} }, { "channel": "CH6", "motion_alarm": false, "io_alarm": false, "ptz_alarm": { "ptz_cruise_state": false, "ptz_line_scan_state": false }, "videoloss": true, "camera_connect_status": { "connect_status": "NotConfigured", "offline_reason": "NotConfigured", "ability": [], "alarm_in_num": 0, "alarm_out_num": 0 }, "channel_name": "", "camera_type": "Uknown", "record_flag": {} }, { "channel": "CH7", "motion_alarm": false, "io_alarm": false, "ptz_alarm": { "ptz_cruise_state": false, "ptz_line_scan_state": false }, "videoloss": true, "camera_connect_status": { "connect_status": "NotConfigured", "offline_reason": "NotConfigured", "ability": [], "alarm_in_num": 0, "alarm_out_num": 0 }, "channel_name": "", "camera_type": "Uknown", "record_flag": {} }, { "channel": "CH8", "motion_alarm": false, "io_alarm": false, "ptz_alarm": { "ptz_cruise_state": false, "ptz_line_scan_state": false }, "videoloss": true, "camera_connect_status": { "connect_status": "NotConfigured", "offline_reason": "NotConfigured", "ability": [], "alarm_in_num": 0, "alarm_out_num": 0 }, "channel_name": "", "camera_type": "Uknown", "record_flag": {} } ] }, "reader_id": 3, "sequence": 58, "lap_number": 0 }
```

(2) Request again guidance:

POST /API/Event/Check?2020-10-17%2014:21:33 HTTP/1.1

Content-Type: application/json; charset=UTF-8

Accept: application/json, text/javascript, */*; q=0.01

X-csrfToken: 14559384d58d55d9d80bf4baf048684f366c77905d665d99ba1bde2cdf81b08

Content-Length: 2438

Cookie: session=cc3ec99c6f1295489c86f8842e1dd719c194c637af5b965a23932c41188a3acc

```
{
  "data": {
    "reader_id": 3,
    "sequence": 58,
    "lap_number": 0
  }
}
```

```
}
```

Reply:

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: 97

```
{
  "result": "success",
  "data": {
    "heat_alarm": "HeatAlarm",
    "reader_id": 3,
    "sequence": 58,
    "lap_number": 0
  }
}
```

15.2 Http listening

GET	
URL	POST /API/AlarmConfig/EventPush/Get
Description	It is used to get the ALL ALARM config parameters
Request Body	See Table -1.1.1.1
Successful Response	Channel Information JSON (show as follow Table-1.1.1.4)

Table-1.1.1.1 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
params		Json Object	Single Channel Information JSON show as follow Table-1.1.1.2

Table-1.1.1.2 (rule_number Information JSON)

name		string	Channel name
table		Json Object	Single Channel Information JSON show as follow Table-1.1.1.3

Table-1.1.1.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
username		string	Username, keep empty if no
password		string	password, kepp empty if no
addr		string	Server address
port	1-65535	int	Sever port
url	0-128	string	Sever API interface, keep empty if no
enable	true, false	bool	Switch of alarm push
method	GET POST	String	HTTP push method
keep_alive_interval	OFF 1min 5min 10min	String	Keep alive interval
push_way	HTTP UDP	string	Push way
udp_method	Unicast Multicast Broadcast	string	UDP push method
udp_addr		string	UDP server address
udp_port	1-65535	int	UDP sever port

15.2 Http listening push related instruction

15.2.1 Push alerts through Get

Push the alert event request	
URL	GET /API/AlarmEvent/EventPush?EventType=xx&EventTime=xx&EventAction=xx&MACA ddress=xx
Description	It is used to get the ALL ALARM config parameters
Request Param	See Table -1.1.1.10
Successful Response	Channel Information JSON (show as follow Table-1.1.1.4)

Table -1.1.1.10

KEY	VALUE		COMMENT
	RANGE	TYPE	
EventType	VideoMotion VideoBlind SoundDetect	string	IPC alarm event type

	SOD LCD PID PD_VD FD CC AD CD QD LPD RSD		
EventTime		string	Event triggered time
EventAction	Start stop	string	IPC alarm status
MACAddress		string	IPC MAC address,like D094662C1A8D

The client receives IPC data when alarm triggered:

```
GET /API/AlarmEvent/EventPush?EventType=VideoMotion&EventTime=2022-5-26
11:5:0&EventAction=start&MACAddress=5C-F2-07-49-0F-24 HTTP/1.1
Host: 172.16.8.63:123
Accept: */*
Content-Type: application/json;charset=UTF-8
```

15.2.2 Push alerts via POST

Push alerts event requests	
URL	POST /API/AlarmEvent/EventPush
Description	It is used to get the ALL ALARM config parameters
Request Body	See Table -1.1.1.4
Successful Response	Channel Information JSON (show as follow Table-1.1.1.4)

Table-1.1.1.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
EventType	VideoMotion VideoBlind SoundDetect SOD LCD PID PD_VD	string	IPC event type

	FD CC AD CD QD LPD RSD		
EventTime		string	Alarm triggered time
EventAction	Start stop	string	IPC alarm status
ChannelName		string	Channel number
DeviceName		string	IPC device name
IPAddress		string	IPC IP address
MACAddress		string	IPC MACaddress
PicData		Json Object	Push data of intelligent alarm is shown in Table-1.1.1.5 below

Table-1.1.1.5 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
SnapedObjInfo	t	Json Object	Pd&vd、PID、LCD Push data is shown in Table-1.1.1.6 below
PlateInfo		Json Object	LPD push data is shown in Table-1.1.1.7
FaceInfo		Json Object	FD push data is shown in Table-1.1.1.8

Table-1.1.1.6

KEY	VALUE		COMMENT
	RANGE	TYPE	
Chn	0	int	IPC use 0 channel only
StrChn	"CH1"	string	Channel name
StartTime		long long	Start time, used for image
EndTime		long long	End time, used for image
SnapId		int	Image id
Type		int	Type ACPT_Face = 0, ACPT_HumanBody = 1, ACPT_Vehicle = 2, ACPT_PIDHumanBody = 3, ACPT_PIDVehicle = 4, ACPT_LCDHumanBody = 5, ACPT_LCDVehicle = 6,
ObjectImage		string	Base64 Encoding image data
Background		string	Base64 Coded background data

Table-1.1.1.7

KEY	VALUE		COMMENT
	RANGE	TYPE	
Id	1-15characters	string	license plate number, unique identification
GrpId		int	Group ID
SnapId		unsigned int	Snap ID while IPC capturing
Type	0	int	Type of face capture object
StrChn		string	channel
StartTime		unsigned long long	Capture start time,for example : 2018-10-25 13:08:36, Unix time stamp (displayed in UTC time zone)
EndTime		unsigned long long	Capture end time , for example : 2018-10-25 13:08:46, Unix stamp (displayed in UTC time zone)
Chn		int	channel
PlateColor	0-5	int	Plate color, values as below: enum AIPlateColor_e { APC_Blue = 0, // blue APC_Green = 1, // green APC_Yellow = 2, // yellow APC_Black = 3, // black APC_White = 4, // white APC_Other = 5, // other APC_Max };
CarBrand		string	Car brand
CarType		string	Car type
Sex		int	Gender of car owner: 0-male, 1-female
Owner	"Mike"	string	Owner name
IdCode	"415025199203050916 "	string	ID code
Job	"Software"	string	Job
Phone	"12345678902"	string	Phone number
Domicile	"Guangdong,Zhuhai,Xi angzhou ..."	string	Current address
Remark	"Detail of this person ..."	string	remark
PlateImg	"base64(imgData)"	string	Plate license image
BgImg	"base64(imgData)"	string	Background imgae

Table-1.1.1.8

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
Id		int	Face Id,unique identification (nvr only)
GrpId		int	Group Id(nvr only)
SnapId		unsigned int	IPC capturing image ID
Type	0	int	Type of face capture object
StartTime		unsigned long long	Start time to capture, for example: 2018-10-25 13:08:36, Unix timestamp (displayed in UTC time zone)
EndTime		unsigned long long	Capature end time,for example : 2018-10-25 13:08:46, Unix timestamp (displayed in UTC time zone)
Score		int	Image score (confidence coefficient)
Sex		int	Gender: 0-male, 1-female
Age		int	Age
Gender		int	Face gender: 0-female, 1-male
fAttrAge		int	Face attribute age
GlassesType		int	Glass type, 0:without glass, 1: with glass(currently no distinction between sunglasses and ordinary glasses, and all glasses are default)
Expression		int	Expression type, 0: no exp, 1: smile, 2: laugh
MouthMask		int	Mouthmask or not, 0:without, 1: with
Race		int	race, 0: yellow, 1: white, 2: black, 3: Arabian
Chn		int	channel
StrChn	<p>"CH1"... "CH1x"</p> <p>"IP_CH1"... "IP_CH1x"</p> <p>"WIFI_CH1"... "WIFI_CH1x"</p> <p>The number of channels depends on the capabilities of the device.</p>	string	channel
ModifyCnt		int	The number of modifications, the upper level is read-only, which is updated by the device in real time every time the face information is modified(nvr only)
Image2	"base64(imgData)"	string	Face capture pictures
Image4	"base64(imgData)"	string	Background capture pictures
Name	"Mike"	string	Name (nvr only)
Country	"China"	string	Country(nvr only)
Nation	"Han"	string	Nation(nvr only)
NativePlace	"Guangdong,Zhuhai"	string	Native Place(nvr only)
IdCode	"415025199203050916"	string	ID card(nvr only)

	"		
Job	"Software"	string	Job(nvr only)
Phone	"12345678902"	string	Phone(nvr only)
Email	"abcd@163.com"	string	Email(nvr only)
Domicile	"Guangdong,Zhuhai,Xiangzhou ..."	string	Current address(nvr only)
Remark	"Detail of this person ..."	string	Comment(nvr only)
FtVersion			

The client receives IPC data at the time of alarm:

```
POST /API/AlarmEvent/EventPush HTTP/1.1
Host: 172.16.8.63:123
Accept: */*
Content-Type: application/json;charset=UTF-8
Content-Length: 192
```

```
{"result": "success", "data": {"EventType": "VideoMotion", "EventTime": "2022-5-26
11:6:3", "EventAction": "start", "DeviceName": "SSC339G", "IPAddress": "172.16.11.170",
"MacAddress": "5C-F2-07-49-0F-24"}}
```

MOTION notification:

```
{
  "EventType": "VideoMotion",
  "EventTime": "2022-4-14 9:12:46",
  "EventAction": "start",
  "ChannelName": "badroom",
  "DeviceName": "IPC_123",
  "IPAddress": "172.16.11.30",
  "MACAddress": "fe80::223:63ff:fe0a:901b"
}
```

Video tamper botification:

```
{
  "EventType": "VideoBlind",
  "EventTime": "2022-4-14 9:12:46",
  "EventAction": "start",
  "ChannelName": "badroom",
  "DeviceName": "IPC_123",
  "IPAddress": "172.16.11.30",
  "MACAddress": "fe80::223:63ff:fe0a:901b"
}
```

Sound notification:

```
{
  "EventType": "SoundDetect",
  "EventTime": "2022-4-14 9:12:46",
  "EventAction": "start",
  "ChannelName": "badroom",
  "DeviceName": " IPC_123",
  "IPAddress": "172.16.11.30",
  " MACAddress  ":"fe80::223:63ff:fe0a:901b"
}
```

Intelligence notification:

```
{
  "EventType": " LCD ",
  "EventTime": "2022-4-14 9:12:46",
  "EventAction": "start",
  "ChannelName": "badroom",
  "DeviceName": "IPC_123",
  "IPAddress": "172.16.11.30",
  " MACAddress  ":"fe80::223:63ff:fe0a:901b"
  "PicData" ":{
    "SnapedObjInfo":[
      {
        "Chn": 0,
        "StrChn": "CH1",
        "StartTime": 1649936559,
        "EndTime": 1649936569,
        "SnapId": 54,
        "Type": 1,
        "ObjectImage":
        "Background":
      }
    ]
  }
}
```

15.2.3 Http listening keep alive

POST	
URL	POST /API/HttpListening/KeepLive
Description	The device initiates a live request to the client server
Request Body	See Table -1.1.1.4

Successful Response	The successful result response that described in 2.5
---------------------	--

Table-1.1.1.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
time		string	Obtain current seconds by time

Example:

Request message:

The client receives IPC data when alarm triggered

POST /API/HttpListening/KeepLive HTTP/1.1

Host: 172.16.8.63:123

Accept: */*

Content-Type: application/json;charset=UTF-8

Content-Length: 30

Response message:

```
{"result": "success", "data": {}}
```

16 Custom

16.1 Push server

Get	
URL	POST /API/Custom/PushConfig/Get
Description	Get the push server parameters
Request Body	none
Successful Response	Show as Table-16.1.1

Set	
URL	POST /API/Custom/PushConfig/Set
Description	Set the push server parameters
Request Body	Show as Table-16.1.1
Successful Response	The successful result response that described in 2.5

Table-16.1.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
server	Max length:128byte	string	Server address
port	0~65535	int	Server port

16.2 Shared Cloud(A01only)

GET	
URL	POST /API/StorageConfig/SharedCloud/Get
Description	It is used to get the Shared Cloud parameters
Request Body	none
Successful Response	Shared Cloud JSON(show as follow Table-16.2.1)

SET	
URL	POST /API/StorageConfig/SharedCloud/Set
Description	It is used to set the Shared Cloud parameters
Request Body	Shared Cloud JSON(show as follow Table-16.2.1)
Successful Response	The successful result response that described in 2.5

Table16.2.1(Shared Cloud JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
enable		bool	Switch of public cloud
sync_time		bool	Switch the public cloud to synchronize system time on
qrcode_string	Max length: 63 byte	string	Qr code for public cloud
server_address	Max length: 127 byte	string	Server address of public cloud

16.3 Write Log (A01 specific)

Set	
URL	POST/API/Maintenance/Log/Set
Description	write manual capture, manual record log to Device
Request Body	Request JSON (show as follow Table-16.3.1)
Successful Response	The successful result response that described in 2.5

Table-16.3.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
sub_type	"manual_capture" "start_manual_record" "stop_manual_record" "picture_backup" "write_logstr"	string	
channel	"CH1"... "CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"..."	string array	

	WIFI_CH1x” The number of channels depends on the capabilities of the device.		
detail_log		string	

16.4 Cloud Upgrade(A01 only)

GET	
URL	POST /API/Maintenance/CloudUpgrade/Check
Description	Cloud upgrade detect
Request Body	show as follow Table-16.4.1
Successful Response	show as follow Table-16.4.2

POST	
URL	POST /API/Maintenance/CloudUpgrade/Upgrade
Description	Upgrade system new version
Request Body	show as follow Table-16.4.3
Successful Response	The successful result response that described in 2.5

GET	
URL	POST /API/Maintenance/CloudUpgrade/Upgrade/Progress
Description	Cloud upgrade progress
Request Body	NULL
Successful Response	show as follow Table-11.10.4

Table-11.10.4 (Cloud Upgrade)

KEY	VALUE		COMMENT
	RANGE	TYPE	
upgrade_state		string	Upgrading tus and error code (show as follow Table-11.10.5)
upgrade_percent		unsigned int	Upgrading progress,only when upgrading or rade_download, will show this field

Table-11.10.5 (Cloud Upgrade)

KEY	VALUE		COMMENT
	RANGE	TYPE	
upgrade_state (error_cord)	upgrade_download /*downloading, do not disconnect the network*/ upgrade_upgrading /*upgrading,do not disconnect the network */	string	

	upgrade_memory_not_enough /*memory not enough*/ upgrade_file_error /*file error*/ upgrade_upgrade_succeed /*upgrading succeed*/ upgrade_parameter_error /*parameter error*/ upgrade_no_u_disc /*no u disk*/ upgrade_no_upgrade_software /*no software upgrading*/ upgrade_software_is_new /*no need upgrading if firmware is latest version */ upgrade_software_packet_error /* Error in package operation */ upgrade_language_version_error /* Error in firmware language */ upgrade_file_name_too_long upgrade_failed /*upgrading failed*/ upgrade_exit upgrade_uncipher upgrade_download_faild /*downloading failed*/ upgrade_download_network_error /*network error*/ upgrade_environmentvar_different /* Startup parameters with updating*/ upgrade_usbtype_error /*udisk format error*/		
--	---	--	--

Table-16.4.1(Cloud Upgrade)

KEY	VALUE		COMMENT
	RANGE	TYPE	
is_detect		bool	ture: Detect; false:only get data, not Detect

Table-16.4.2 (Cloud Upgrade)

KEY	VALUE		COMMENT
	RANGE	TYPE	
need_upgrade		bool	If new version for upgrading
current_version		string	Current version
new_version		string	New version

Table-16.4.3 (Cloud Upgrade)

KEY	VALUE		COMMENT
	RANGE	TYPE	
secondary_authentication		string	Verify admin user password for secondary authentication

16.5 Develop Mode (A01 only)

GET	
URL	POST /API/Maintenance/DevelopMode/Get
Description	It is used to get the develop parameters
Request Body	none
Successful Response	(show as follow Table-16.5.1)

SET	
URL	POST /API/Maintenance /DevelopMode /Set
Description	It is used to set the develop parameters
Request Body	(show as follow Table-16.5.1)
Successful Response	The successful result response that described in 2.5

Token	
URL	POST /API/Maintenance /DevelopMode/Token
Description	Get token
Request Body	none
Successful Response	The successful result response that described in 2.5, X-Download-Token return at respond header

Download	
URL	POST /API/Maintenance /DevelopMode /Download
Description	It is used to set the develop parameters
Request Body	(show as follow Table-16.5.2)
Successful Response	The successful result response that described in 2.5

Table16.5.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
enable_sdk_log		bool	Switch of Sdk log
enable_cloud_log		bool	Switch of cloud log
enable_dev_log		bool	Switch of device log

Table-16.5.2

KEY	VALUE		COMMENT
	RANGE	TYPE	
x-download-token	0-32	string	Token value
x-csrf-token		string	csrf-token

16.6 Cert Manage (A01 only)

SET	
URL	POST /API/NetworkConfig/CertManage/Set
Description	It is used to install or uninstall cert
Request Body	(show as follow Table-16.6.1)
Successful Response	The successful result response that described in 2.5

Table16.6.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
cert_info		json	Certificate,check in 16.6.2

Table16.6.2

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud		json	Certificate,check in 16.6.3
sdk		json	
device		json	

Table-16.6.3

KEY	VALUE		COMMENT
	RANGE	TYPE	
operate	"Upload ""Restore"	string	Set need
root_cert	0-4096	string	Uninstall not need
local_cert	0-2048	string	Uninstall not need, Device type not need
local_private_key	0-2048	string	Uninstall not need, Device type not need
local_private_key_passwd	0-128	string	Uninstall not need, Device type not need
install_unsafe_cert	true false	bool	Uninstall not need.When the root certificate contains an invalid certificate or the certificate is unsafe, the user chooses whether to install it.

16.7 Zero Channel(B09 only)

GET	
URL	POST /API/SystemConfig/Zero/Get
Description	Get System Zero Channel Information
Request Body	none
Successful Response	Zero Channel information JSON (show as follow Table-16.7.1)

SET	
URL	POST /API/SystemConfig/Zero/Set
Description	Set System Zero Channel Information
Request Body	Zero Channel information JSON (show as follow Table-16.7.1)
Successful Response	The successful result response that described in 2.5

Table-16.7.1 (Zero Channel information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
zero_chn_enable		bool	Zero Channel enable flag
bitrate_mode	" General "	string	Birate mode
fps		int	frame
birate	32, 48, 64, 80, 96, 128, 160, 192, 224, 256, 320, 384, 448, 512, 640, 768, 896, 1024, 1280, 1536, 1792, 2048, 4096, 5120, 6144, 8192, 10240, 12288, 3072	int	birate
splitmode	" 1*1 ", " 2*2 ", " 1+5", " 1+7 ", " 3*3", " 4*4"	string	splitmode
dwel_time	" 0", " 5", " 10", " 20", " 60", " 120", " 300"	string	Dwell time
zero_chn_list	1-255	int array	Each array bit represents a channel with a int
channel_name	"NONE, CH1"...CH1x" "NONE, IP_CH1"... IP_CH1x" "NONE, WIFI_CH1"... WIFI_CH1x" The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string
default_chn_name	"CH1"...CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"... WIFI_CH1x" The number of channels depends	string array	Each array bit represents a channel with a string

	on the capabilities of the device.		
--	------------------------------------	--	--

16.8 Zero Channel State(B09 only)

GET	
URL	POST /API/ PreviewChannel/ZeroChnState/Get
Description	Get PreviewChannel Zero Channel State Information
Request Body	none
Successful Response	Zero Channel State information JSON (show as follow Table-16.8.1)

SET	
URL	POST /API/ PreviewChannel/ZeroChnState/Set
Description	Set PreviewChannel Zero Channel State Information
Request Body	Zero Channel State information JSON (show as follow Table-16.8.1)
Successful Response	The successful result response that described in 2.5

Table-16.8.1 (Zero Channel information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
chn_list		int array	Channel list
split_mode		int	Split mode
preview_mode		int	Previde mode
encode_ch		int	Ecode channel

16.9 Serail(B09 only)

GET	
URL	POST /API/SystemConfig/ Serail/Get
Description	Get System Serail Information
Request Body	none
Successful Response	Serail information JSON (show as follow Table-16.9.1)

SET	
URL	POST /API/SystemConfig/ Serail/Set
Description	Set System Serail Information
Request Body	Serail information JSON (show as follow Table-16.9.1)
Successful Response	The successful result response that described in 2.5

Table-16.9.1 (Zero Channel information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
protocol	“RS485”, “RS232”	string	Protocol type

baudrate	“1200” , “2400” , “4800” “9600”	string	birate
data_bit	“5” , “6” , “7” , “8”	string	Data bit
stop_bit	“1” , “2”	string	Stop bit
parity	“None” , “Odd” , “Even” , “Mark” , “Space”	string	verify
flow_control	“None” , “Software” , “Hardware”	string	Flow control

16.10 Force I Frame

SET	
URL	POST/API/ForceIframe/Set
Description	Set the channel force I frame
Request Body	Channel Information JSON (show as follow Table-16.10.2)
Successful Response	The successful result response that described in 2.5

Table-16.10.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string.

Table-16.10.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as follow Table-16.10.x

Table-16.10.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-16.10.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-16.10.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
force_iframe	false:off, true:on	bool	If force frame
stream_type	"Mainstream","Substream"	string	Stream type

Example:

Request message:

POST /API/ForceIFrame/Set HTTP/1.1

```
{
  "data": {
    "channel_info": {
      "CH1": {
        "force_iframe": true,
        "stream_type": "Mainstream"
      },
      "IP_CH1": {
        "force_iframe": true,
        "stream_type": "Mainstream"
      }
    }
  }
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: 30

```
{
  "result": "success",
  "data": {}
}
```

17 AI

17.1 Setup

17.1.1 Face Detection

GET	
URL	POST /API/AI/Setup/FD/Get
Description	It is used to get the FD config parameters
Request Body	See Table -17.1.1.1
Successful Response	Channel Information JSON (show as follow Table-17.1.1.2)

SET	
URL	POST /API/AI/Setup/FD/Set
Description	It is used to set theFD parameters
Request Body	Channel Information JSON (show as follow Table-17.1.1.2)
Successful Response	The successful result response that described in 2.5

Table-17.1.1.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1

Table-17.1.1.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as follow Table-17.1.1.x
page_type	“ChannelConfig”, “AlarmConfig”	string	Data used to distinguish between the channel configuration page and the alarm configuration page
channel_max		int	Chanel number at most

24

Table-17.1.1.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-17.1.1.3
...		Json Object	
IP_CH1		Json Object	

...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.1.1.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", "Nonsupport"	string	Channel online status, for digital channels only Note: No this field when the channel is online
alarm_out	"Local->1"... "Local->x" "IP_CH1->1"... "IP_CH1->2"... "IP_CHx->1" "IP_CHx->2" The number of channels depends on the capabilities of the device.	array	Alarm out channel Each array bit represents aalarm output channel with a string. Camera: Local->1:enable, null: disable
latch_time	"5s","10s","20s", "30s"	string	Alarm latch tijme :Ipc value 5 10 20 30
record_enable	true, false	bool	Record channel switch
record_channel	"CH1"... "CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"...	array	Alarm out channel Channel alarm linkage switch.
post_recording	"OFF","5s","10s", "20s","30s"	string	Post recording time Ipc value 0 5 10 20 30
send_email	true, false	bool	Send Email switch
switch	true, false	bool	switch, false: close true: open
snap_mode	" RealTimeMode " "OptimalMode" "IntervalMode"	string	Snap mode, IntervalMode JSON show as follow Table-17.1.1.4
apply_mode	"FrontalView" "MultiAngle" "Customize"	string	Customize JSON show as follow Table-17.1.1.5
min_pixel	32-1080	int	minimum pixel
max_pixel	320-1080	int	Maximal pixel
face_enhance	true, false	bool	face_enhance switch

face_attribute		bool	face_attribute switch
detection_mode	"StaticMode" "MotionMode"	string	Detection mode
iva_lines	0 no line 1 with line	int	Whether the preview and playback will be underlined
rule_info		Object	Information JSON show as follow Table-17.1.1.6

Table-17.1.1.4 (snap interval mode Information JSON)

snap_num	"1" "2" "3" "Unlimited"	string	Snap number
snap_frequency		int	Snap interval, unit s/pic

Table-17.1.1.5 (apply_mode Information JSON)

roll_range	0-180	int	The following default value fields are only available in Range: "default_value_frontal": frontal "default_value_multi":multi
pitch_range	0-180	int	The following default value fields are only available in Range: : "default_value_frontal": frontal "default_value_multi":multi
yaw_range	0-180	int	The following default value fields are only available in Range: : "default_value_frontal": frontal "default_value_multi":multi
picture_quality	0-100	int	he following default value fields are only available in Range: "default_value_frontal": frontal "default_value_multi":multi

Table-17.1.1.6 (rule_info Information JSON)

rule_number1		Object	Number rule :1, Information JSON show as follow Table-17.1.1.7
--------------	--	--------	--

Table-17.1.1.7 (rule_number Information JSON)

rule_type	"A->B" "B->A"	string	Rule type
rule_kind	"Rect" "Line"	string	Rule kind
detection_range	"FullScreen" "Customize"	string	Detection range
rule_rect		Object	rectangle,Information JSON show as follow Table-17.1.1.8
rule_line		Object	unrdeline,Information JSON show as follow Table-17.1.1.9

Table-17.1.1.8 (Channel Information JSON)

x1	0-704	short	x1 coordinate point
y1	0-576	short	y1 coordinate point
x2	0-704	short	x2 coordinate point
y2	0-576	short	y2 coordinate point
x3	0-704	short	x3 coordinate point
y3	0-576	short	y3 coordinate point
x4	0-704	short	x4 coordinate point
y4	0-576	short	y4 coordinate point

Table-17.1.1.9 (rule_line Information JSON)

x1	0-704	short	x1 coordinate point
y1	0-576	short	y1 coordinate point
x2	0-704	short	x2 coordinate point
y2	0-576	short	y2 coordinate point

Example:

Request message:

POST /API/AI/Setup/FD/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "channel": ["CH1"]
  }
}
```

Response message:

```
{
  "result": "success",
```



```

"data": {
  "channel_info": {
    "CH1": {
      "status": "Online",
      "switch": true,
      "snap_mode": "IntervalMode",
      "snap_num": "Unlimited",
      "snap_frequency": 2,
      "min_pixel": 64,
      "max_pixel": 320,
      "face_enhance": true,
      "face_attribute": true,
      "apply_mode": "MultiView",
      "roll_range": 180,
      "pitch_range": 180,
      "yaw_range": 180,
      "picture_quality": 100,
      "frontal_roll_range": 30,
      "frontal_pitch_range": 30,
      "frontal_yaw_range": 45,
      "frontal_blur_stage": 100,
      "multi_roll_range": 180,
      "multi_pitch_range": 180,
      "multi_yaw_range": 180,
      "multi_blur_stage": 100,
      "detection_mode": "StaticMode",
      "rule_info": {
        "rule_number1": {
          "rule_type": "A->B",
          "rule_kind": "Rect",
          "detection_range": "FullScreen",
          "rule_rect": {
            "x1": 0,
            "y1": 0,
            "x2": 0,
            "y2": 575,
            "y3": 575,
            "y4": 0
          }
        }
      }
    }
  }
}

```

}

17.1.2 Human & Vehicle Detection

GET	
URL	POST /API/AI/Setup/PVD/Get
Description	It is used to get the PD config parameters
Request Body	See Table -17.1.2.1
Successful Response	Channel Information JSON (show as follow Table-17.1.2.2)

SET	
URL	POST /API/AI/Setup/PVD/Set
Description	It is used to set thePD parameters
Request Body	Channel Information JSON (show as follow Table-17.1.2.2)
Successful Response	The successful result response that described in 2.5
Error_Code Response	Error Information JSON (show as follow Table-17.1.2.8) (cloud_video_upload_chn_limit returned information attached to the error code)

Table-17.1.2.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	Data used to distinguish between the channel configuration page and the alarm configuration page

Table-17.1.2.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Single Channel Information JSON show as follow Table-17.1.2.x
page_type	“ChannelConfig”, “AarmConfig”	string	Data used to distinguish between the channel configuration page and the alarm configuration page, Set need only

channel_max		int	Maximal channel number
-------------	--	-----	------------------------

Table-17.1.2.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-17.1.2.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.1.2.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", "Nonsupport"	string	Channel online status, for digital channels only Note: no this field when channel on line.
alarm_out	"Local->1"... "Local->x" "IP_CH1->1"... "IP_CH1->2"... "IP_CHx->1" "IP_CHx->2" The number of channels depends on the capabilities of the device.	array	Alarm out channel Each array bit represents a alarm output channel with a string. Camera: Local->1: enable, null: disable
latch_time	"10","20","40", "60"	string	Alarm out latch time
record_enable	true, false	bool	Record channel switch
record_channel	"CH1"... "CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"...	array	Alarm out channel Channel alarm linkage switch.
post_recording	"30","60","120", "300"	string	Post recording
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Switch for channel capture FTP uploading (NVR only)
ftp_video_upload	true, false	bool	Switch for channel recording FTP uploading (NVR only)

picture_to_cloud	true, false	bool	Switch for picture uploading (NVR only)
video_to_cloud	true, false	bool	Switch for video cloud uploading (NVR only)
full_screen		bool	FullScreen switch (NVR only)
Buzzer	"0","10","20","40", "60"	string	Buzzer working time (NVR only)
show_message	true, false	bool	Show Message switch (NVR only)
switch	true, false	bool	switch, false: close true: open
sensitivity	0-100	int	Sensitivity detection(IPC G2)
sensitivity	"Low","Medium","High"	string	Sensitivity detection (IPC G1)
snap_mode	"Default" "RealTimeMode" "IntervalMode"	string	Snap mode, IntervalMode JSON show as follow Table-17.1.2.4
min_pixel	64-1080	int	Minimum pixel
max_pixel	320-1080	int	Maximal pixel
detection_type	"Off" "Pedestrian" "Vehicle" "Pedestrian & Vehicle"	string	Detection type, PD&VD (normal detection type)
detection_type	"Pedestrian" "Motor Vehicle" "Non-motorized Vehicle"	array	Detection type, PD&VD, Motor Vehicle (Non-motorized Vehicle)
iva_lines	0 no line 1 with line	int	If with line in preview and playback
rule_info		Object	Information JSON show as follow Table-17.1.2.5

Table-17.1.2.4 (snap interval mode Information JSON)

snap_num	"1" "2" "3" "Unlimited"	string	Snap number
snap_frequency		int	Snap interval, Unit s/pic

Table-17.1.2.5 (rule_info Information JSON)

rule_number1		Object	Rule number 1, Information JSON show as follow Table-17.1.2.6
--------------	--	--------	---

Table-17.1.2.6 (rule_number Information JSON)

detection_range	"FullScreen" "Customize"	string	Detection range
rule_switch	true, false	bool	/* Each channel represents up to four rules with one byte, 0 -off, 1 -used */ (IPC G1 only)
rule_type	"Normal"	string	/* Each channel is represented by one byte for up to 4 -regular type, 0 -regular mode, and 1 -guest mode/other mode */ (IPC G1 only)
rule_rect		Object	rectangle,Information JSON show as follow Table-17.1.2.7

Table-17.1.2.7 (Channel Information JSON)

x1	0-704	short	x1 coordinate point
y1	0-576	short	y1 coordinate point
x2	0-704	short	x2 coordinate point
y2	0-576	short	y2 coordinate point
x3	0-704	short	x3 coordinate point
y3	0-576	short	y3 coordinate point
x4	0-704	short	x4 coordinate point
y4	0-576	short	y4 coordinate point

Table-17.1.2.8(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CH N_NUM	array	Channel number of cloud video uploads enabled
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	Cloud video upload supports the maximum number of open channel

Example:

Request message:

POST /API/AI/Setup/PVD/Get HTTP/1.1

```
{
  "version":"1.0",
  "data":{
    "channel":["CH1"],
    "page_type":"ChannelConfig"
  }
}
```

Response message:

```

{
  "result": "success",
  "data": {
    "channel_info": {
      "CH1": {
        "switch": false,
        "snap_mode": "IntervalMode",
        "snap_num": "1",
        "snap_frequency": 2,
        "min_pixel": 64,
        "max_pixel": 320,
        "sensitivity": 60,
        "detection_type": "Pedestrian",
        "detection_mode": "StaticMode",
        "rule_info": {
          "rule_number1": {
            "detection_range": "FullScreen",
            "rule_rect": {
              "x1": 1,
              "y1": 1,
              "x2": 0,
              "y2": 576,
              "x3": 704,
              "y3": 576,
              "x4": 704,
              "y4": 0
            }
          }
        }
      }
    }
  }
}

```

Request message:

POST /API/AI/Setup/PVD/Get HTTP/1.1

```

{
  "version": "1.0",
  "channel": ["CH2"],
  "page_type": "AlarmConfig"
}

```

Response message:

```

{

```

```

"result": "success",
"data": {
  "channel_info": {
    "CH2": {
      "buzzer": "0",
      "alarm_out": [
        "Local->1",
        "IP_CH2->1",
        "IP_CH3->1",
        "IP_CH5->1"
      ],
      "latch_time": "10",
      "record_enable": true,
      "record_channel": [
        "CH2"
      ],
      "post_recording": "30",
      "show_message": true,
      "send_email": true,
      "full_screen": false,
      "ftp_picture_upload": true,
      "ftp_video_upload": false,
      "picture_to_cloud": true,
      "video_to_cloud": false
    }
  }
}

```

17.1.3 Perimeter Intrusion Detection

GET	
URL	POST /API/AI/Setup/PID/Get
Description	It is used to get the PID config parameters
Request Body	See Table -17.1.3.1
Successful Response	Channel Information JSON (show as following Table-17.1.3.2)

SET	
URL	POST /API/AI/Setup/PID/Set
Description	It is used to set thePID parameters
Request Body	Channel Information JSON (show as following Table-17.1.3.2)
Successful Response	The successful result response that described in 2.5

Error_Code Response	Error Information JSON (show as following Table-17.1.3.7) (cloud_video_upload_chn_limitreturn information attached to error code)
---------------------	--

Table-17.1.3.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	String array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	To distinguish the data whether from a channel configuration page or an alarm configuration page

Table-17.1.3.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as following Table-17.1.3.x
page_type	“ChannelConfig”, “AarmConfig”	string	To distinguish the data whether from a channel configuration page or an alarm configuration page, Requested only in Setting
channel_max		int	Maximum number of channels

Table-17.1.3.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as following Table-17.1.3.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.1.3.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online"	string	Channel online status, only for digital

	, "Nonsupport"		channels. Note: no such strings when the channel is online
alarm_out	"Local->1"... "Local->x" "IP_CH1->1"... "IP_CH1->2"... "IP_CHx->1" "IP_CHx->2" The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents a alarm output channel with a string. Camera: Local->1:enable null: disable
latch_time	"10","20","40", "60"	string	Alarm output time
record_enable	true, false	bool	Record channel switch
record_channel	"CH1"... "CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"...	array	Alarm output channel Channel alarm linkage switch.
post_recording	"30","60","120", "300"	string	Recording time-lapse
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Channel capture FTP upload switch (NVR specific)
ftp_video_upload	true, false	bool	Channel recording FTP upload switch (NVR specific)
picture_to_cloud	true, false	bool	Picture Cloud upload switch (NVR specific)
video_to_cloud	true, false	bool	Video Cloud upload switch (NVR specific)
full_screen		bool	FullScreenswitch (NVR specific)
buzzer	"0","10","20","40", "60"	string	Buzzer beeping time (NVR specific)
show_message		bool	Show Message switch (NVR specific)
switch	true, false	bool	PID enable switch, false: close true: open.
detection_type	"Motion" "Pedestrian" "Vehicle" "Pedestrian & Vehicle"	string	Detection type, human and vehicle(Normal detection type)
detection_type	"Pedestrian" "Motor Vehicle"	array	Detection type, pedestrian, motor vehicle and non-motor vehicle(machine not human type)

	"Non-motorized Vehicle"		
sensitivity	1-4	int	Detection sensitivity
iva_lines	0 not draw lines 1 draw lines	bool	Whether the preview and playback will appear the lines
rule_info		JSON object	Single Channel Information JSON show as following Table-17.1.3.4

Table-17.3.4 (rule_info Information JSON)

rule_number1		Object	Rule number 1, Information JSON show as following Table-17.1.3.5
rule_number2		Object	Rule number 2, Information JSON show as following Table-17.1.3.5
rule_number3		Object	Rule number 3, Information JSON show as following Table-17.1.3.5
rule_number4		Object	Rule number 4, Information JSON show as following Table-17.1.3.5

Table-17.3.5 (rule_number Information JSON)

rule_switch		bool	Rule switch
rule_type	"A->B", "B->A", "A<-->B"	string	Rule direction
rule_rect		Object	rectangle, Information JSON show as following Table-17.1.3.6

Table-17.1.3.6 (Channel Information JSON)

x1	0-704	short	x1 coordinates
y1	0-576	short	y1 coordinates
x2	0-704	short	x2 coordinates
y2	0-576	short	y2 coordinates
x3	0-704	short	x3 coordinates
y3	0-576	short	y3 coordinates
x4	0-704	short	x4 coordinates
y4	0-576	short	y4 coordinates

Table-17.1.3.7(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CH N_NUM	array	The channel numbers that has enabled cloud video upload function
max_cloud_video_u	MAX_CLOUD_VID	int	The maximum number of channels supported by

pload_num	EO_RECORD_NUM	cloud video upload
-----------	---------------	--------------------

Example:

Request message:

POST /API/AI/Setup/PID/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "channel": ["CH1"]
  }
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "CH1": {
        "status": "Online",
        "switch": false,
        "sensitivity": 2,
        "detection_type": "Off",
        "latch_time": "5",
        "post_recording": "5",
        "alarm_out": [],
        "send_email": false,
        "record_enable": false,
        "rule_info": {
          "rule_number1": {
            "rule_switch": false,
            "rule_type": "A->B",
            "rule_rect": {
              "x1": 0,
              "y1": 0,
              "x2": 0,
              "y2": 0,
              "x3": 0,
              "y3": 0,
              "x4": 0,
              "y4": 0
            }
          }
        }
      }
    }
  }
}
```

```

    }
  },
  "rule_number2": {
    "rule_switch": false,
    "rule_type": "A->B",
    "rule_rect": {
      "x1": 0,
      "y1": 0,
      "x2": 0,
      "y2": 0,
      "x3": 0,
      "y3": 0,
      "x4": 0,
      "y4": 0
    }
  },
  "rule_number3": {
    "rule_switch": false,
    "rule_type": "A->B",
    "rule_rect": {
      "x1": 0,
      "y1": 0,
      "x2": 0,
      "y2": 0,
      "x3": 0,
      "y3": 0,
      "x4": 0,
      "y4": 0
    }
  },
  "rule_number4": {
    "rule_switch": false,
    "rule_type": "A->B",
    "rule_rect": {
      "x1": 0,
      "y1": 0,
      "x2": 0,
      "y2": 0,
      "x3": 0,
      "y3": 0,
      "x4": 0,
      "y4": 0
    }
  }
}

```

```

    }
  }
}

```

17.1.4 Line Crossing Detection

GET	
URL	POST /API/AI/Setup/LCD/Get
Description	It is used to get the LCD config parameters
Request Body	See Table -17.1.4.1
Successful Response	Channel Information JSON (show as following Table-17.1.4.2)

SET	
URL	POST/API/AI/Setup//LCD/Set
Description	It is used to set theLCD parameters
Request Body	Channel Information JSON (show as following Table-17.1.4.2)
Successful Response	The successful result response that described in 2.5
Error_Code Response	Error Information JSON (show as following Table-17.1.4.7) (cloud_video_upload_chn_limit (return information attached to error code))

Table-17.1.4.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	To distinguish the data whether from a channel configuration page or an alarm configuration page

Table-17.1.4.2 (Channel Information JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as following Table-17.1.4.x
page_type	“ChannelConfig”, “AarmConfig”	string	To distinguish the data whether from a channel configuration page or an alarm configuration page, requested only in Setting
channel_max		int	Maximum number of channels

Table-17.1.4.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as following Table-17.1.4.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.1.4.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", “Nonsupport”	string	Channel online status, only for digital channels. Note: no such strings when the channel is online
alarm_out	“Local->1”... ”Local->x” “IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2” The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents aalarm output channel with a string. Camera: Local->1: Enable, null value: Disable
latch_time	"10","20","40", "60"	string	Alarm output time
record_enable	true, false	bool	Record channel switch
record_channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...”	array	Alarm output channel Channel alarm linkage switch.
post_recording	"30","60","120","300"	string	Recording time-lapse

send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Channel capture FTP upload switch (NVR only)
ftp_video_upload	true, false	bool	Channel recording FTP upload switch (NVR only)
picture_to_cloud	true, false	bool	Picture Cloud upload switch (NVR only)
video_to_cloud	true, false	bool	Video Cloud upload switch (NVR only)
full_screen		bool	FullScreen Switch (NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer beeping time (NVR only)
show_message		bool	Show Message Switch(NVR only)
switch	true, false	bool	Switch, false: close true: open
sensitivity	1-4	int	Detection sensitivity
detection_type	"Motion" "Pedestrian" "Vehicle" "Pedestrian & Vehicle"	string	Detection type, pedestrian and vehicle(normal detection type)
detection_type	"Pedestrian" "Motor Vehicle" "Non-motorized Vehicle"	array	Detection type, pedestrian, motor vehicle and non-motor vehicle(machine not human type) (ipc AI II version only)
iva_lines	0 not draw lines 1 draw lines	int	Live and playback whether shows line or not (ipc AI II version only)
rule_info		JSON object	Single Channel Information JSON show as following Table-17.1.4.4

Table-17.1.4.4 (rule_info Information JSON)

rule_number1		Object	Rule number 1, Information JSON show as following Table-17.1.4.5
rule_number2		Object	Rule number 2, Information JSON show as following Table-17.1.4.5
rule_number3		Object	Rule number 3, Information JSON show as following Table-17.1.4.5
rule_number4		Object	Rule number 4, Information JSON show as following Table-17.1.4.5

Table-17.1.4.5 (rule_number Information JSON)

rule_switch		bool	Rule switch
rule_type	"A->B", "B->A", "A<->B"	string	rule direction
rule_line		Object	Each channel has maximum 64 point

			coordinates, that is 32 lines. If it is an accompanying line, only the first line is used. If it is a perimeter, these 8 lines are connected in sequence to form a closed figure Information JSON show as following Table-17.1.4.6
--	--	--	---

Table-17.1.4.6 (rule_line Information JSON)

x1	0-704	short	x1 coordinates
y1	0-576	short	y1 coordinates
x2	0-704	short	x2 coordinates
y2	0-576	short	y2 coordinates

Table-17.1.4.7(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CHN_NUM	array	The channel numbers that has enabled cloud video upload function
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	The maximum number of channels supported by cloud video upload

Example:

Request message:

POST /API/AI/Setup/LCD/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "channel": ["IP_CH2"]
  }
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "IP_CH2": {
        "status": "Online",
        "switch": false,
        "sensitivity": 2,
        "latch_time": "5",
        "post_recording": "10",

```



```
"send_email": false,
"record_enable": false,
"alarm_out": [],
"rule_info": {
  "rule_number1": {
    "rule_switch": false,
    "rule_type": "A->B",
    "rule_line": {
      "x1": 0,
      "y1": 0,
      "x2": 0,
      "y2": 0
    }
  },
  "rule_number2": {
    "rule_switch": false,
    "rule_type": "A->B",
    "rule_line": {
      "x1": 0,
      "y1": 0,
      "x2": 0,
      "y2": 0
    }
  },
  "rule_number3": {
    "rule_switch": false,
    "rule_type": "A->B",
    "rule_line": {
      "x1": 0,
      "y1": 0,
      "x2": 0,
      "y2": 0
    }
  },
  "rule_number4": {
    "rule_switch": false,
    "rule_type": "A->B",
    "rule_line": {
      "x1": 0,
      "y1": 0,
      "x2": 0,
      "y2": 0
    }
  }
}
```


			following Table-17.1.5.3
page_type	“ChannelConfig”, “AarmConfig”	string	To distinguish the data whether from a channel configuration page or an alarm configuration page, Requested only in Setting
channel_max		int	Maximum number of channels

Table-17.1.5.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as following Table-17.1.5.4
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.1.5.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", “Nonsupport”	string	Channel online status, only for digital channels. Note: no such strings when the channel is online
alarm_out	“Local->1”... ”Local->x” “IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2” The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents aalarm output channel with a string. Camera: Local->1:enable null: disable
latch_time	"10","20","40", "60"	string	Alarm output time Ipc value 5 10 20 30
record_enable	true, false	bool	Record channel switch
record_channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...”	array	Alarm output channel Channel alarm linkage switch.
post_recording	"30","60","120", "300"	string	Recording time-lapse, Ipc value 0 5 10 20 30
send_email	true, false	bool	Send Email switch

ftp_picture_upload	true, false	bool	Channel capture FTP upload switch(NVR only)
ftp_video_upload	true, false	bool	Channel recording FTP upload switch(NVR only)
picture_to_cloud	true, false	bool	Picture Cloud upload switch(NVR only)
video_to_cloud	true, false	bool	Video Cloud upload switch(NVR only)
full_screen		bool	FullScreenswitch(NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer beeping time(NVR only)
show_message		bool	Show Message switch(NVR only)
switch	true, false	bool	switch, false: close true: open
type	"Motion" "Person" "Vehicle"	string	Rule type: 0-motion 1- Person 2 veichle(Normal detection type)
	"Motion" "Pedestrian" "Motor Vehicle" "Non-motorized Vehicle"	string	Detection type, pedestrian, motor vehicle and non-motor vehicle(machine not human type)
detection_type	"Pedestrian" "Motor Vehicle" "Non-motorized Vehicle"	array	Detection type, pedestrian, motor vehicle and non-motor vehicle(machine not human type)
alarm_num	1 - 255	int	The number of alarms, if exceeds the number, an alarm will be issued
start_time		string	Starting time
end_time		string	Ending time
reset_count	true, false	bool	true: clear the analysis data of the line crossing statistics
sensitivity	1-4	int	Detection sensitivity
rule_info		Object	Information JSON show as following Table-17.1.5.5

Table-17.1.5.5 (rule_info Information JSON)

rule_number1		Object	Rule number 1, Information JSON show as following Table-17.1.5.6
--------------	--	--------	---

Table-17.1.5.6 (rule_number Information JSON)

rule_switch		bool	Rule switch
rule_type	"A->B", "B->A",	string	Rule direction
rule_line		Object	rectangle, Information JSON show as following

			Table-17.1.5.7
rule_rect		Object	rectangle, Information JSON show as following Table-17.1.5.8

Table-17.1.5.7 (Channel Information JSON)

x1	0-704	short	x1 coordinates
y1	0-576	short	y1 coordinates
x2	0-704	short	x2 coordinates
y2	0-576	short	y2 coordinates

Table-17.1.5.8 (Channel Information JSON)

x1	0-704	short	x1 coordinates(left vertex x-axis coordinates)
y1	0-576	short	y1 coordinates(left vertex y-axis coordinates)
x2	0-704	short	x2 coordinates(right vertex x-axis coordinates)
y2	0-576	short	y2 coordinates(right vertex y-axis coordinates)
X3	0-704	short	X3 coordinates(left bottom x-axis coordinates)
Y3	0-576	short	Y3 coordinates(left bottom x-axis coordinates)
X4	0-704	short	X4 coordinates(right bottom x-axis coordinates)
Y4	0-576	short	Y4 coordinates(right bottom x-axis coordinates)

Table-17.1.5.9(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CH N_NUM	array	The channel numbers that has enabled cloud video upload function
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	The maximum number of channels supported by cloud video upload

Example:

Request message:

POST /API/AI/Setup/CrossCount/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "page_type": "ChannelConfig "
  }
}
```

```
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "CH5": {
        "switch": false,
        "type": "Person",
        "alarm_num": 1,
        "start_time": "03:00:00",
        "end_time": "22:22:22",
        "reset_count": false,
        "rule_info": {
          "rule_number1": {
            "rule_switch": true,
            "rule_type": "A->B",
            "rule_line": {
              "x1": 585,
              "y1": 42,
              "x2": 557,
              "y2": 538
            }
          }
        }
      }
    }
  }
}
```

17.1.6 Heat Map

GET	
URL	POST/API/AI/Setup/HeatMap/Get
Description	It is used to get the Heat Map config parameters
Request Body	See Table -17.1.6.1
Successful Response	Channel Information JSON (show as following Table-17.1.6.2)

SET	
URL	POST /API/AI/Setup/HeatMap/Set
Description	It is used to set the Heat Map parameters

Request Body	Channel Information JSON (show as following Table-17.1.6.2)
Successful Response	The successful result response that described in 2.5

Table -17.1.6.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1

Table -17.1.6.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON array	Single Channel Information JSON show as following Table-17.1.6.3
channel_max		int	Maximum number of channels

Table-17.1.6.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as following Table-17.1.6.4
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.1.6.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline", "Online", “Nonsupport”	string	Channel online status, only for digital channels. Note: no such strings when the channel is online
switch	true, false	bool	switch, false: close true: open
rule_info		Object	Information JSON show as following Table-17.1.6.5

Table-17.1.6.5 (rule_info Information JSON)

rule_number1		Object	Rule number 1, Information JSON show as following Table-17.1.6.6
--------------	--	--------	--

Table-17.5.1.6 (rule_number Information JSON)

rule_switch		bool	Rule switch
rule_rect		Object	rectangle, Information JSON show as following Table-17.1.6.7

Table-17.1.6.7 (Channel Information JSON)

x1	0-704	short	x1 coordinates
y1	0-576	short	y1 coordinates
x2	0-704	short	x2 coordinates
y2	0-576	short	y2 coordinates
x3	0-704	short	x3 coordinates
y3	0-576	short	y3 coordinates
x4	0-704	short	x4 coordinates
y4	0-576	short	y4 coordinates

Example:

Request message:

POST /API/AI/Setup/CrossCount/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "channel": ["CH5"]
  }
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "CH5": {
        "switch": true,
        "rule_info": {
          "rule_number1": {
            "rule_switch": true,
            "rule_rect": {
              "x1": 8,
```


	“AlarmConfig”		channel configuration page or an alarm configuration page
--	---------------	--	---

Table-17.1.7.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as following Table-17.1.7.x
page_type	“ChannelConfig”, “AarmConfig”	string	To distinguish the data whether from a channel configuration page or an alarm configuration page, Requested only in Setting
channel_max		int	Maximum number of channels
support_copy		bool	Whether the page supports copy(NVR/DVR only)

Table-17.1.7.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as following Table-17.1.7.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.1.7.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", “Nonsupport”	string	Channel online status, only for digital channels. Note: no such strings when the channel is online
alarm_out	“Local->1”... ”Local->x” “IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2” The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents aalarm output channel with a string. Camera: Local->1:enable null: disable

latch_time	"10","20","40", "60"	string	Alarm output time
record_enable	true, false	bool	Record channel switch
record_channel	"CH1"... "CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"..."	array	Alarm output channel Channel alarm linkage switch.
post_recording	"30","60","120", "300"	string	Recording time-lapse
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Channel capture FTP upload switch(NVR only)
ftp_video_upload	true, false	bool	Channel recording FTP upload switch(NVR only)
picture_to_cloud	true, false	bool	Picture Cloud upload switch(NVR only)
video_to_cloud	true, false	bool	Video Cloud upload switch(NVR only)
full_screen		bool	FullScreenswitch(NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer beeping time(NVR only)
show_message		bool	Show Message switch(NVR only)
switch	true, false	bool	switch, false: close true: open
sensitivity	1-4、 0-100	int	Detection sensitivity
snap_mode	"Default" "RealTimeMode" "IntervalMode"	string	Snapshot mode, IntervalMode JSON show as following Table-17.1.7.4
min_pixel	64-1080	int	Minimum pixel value
max_pixel	320-1080	int	Maximum pixel value
detection_type	"Pedestrian" "Vehicle" "Pedestrian &Vehicle"	string	Detection type, human and vehicle
detection_mode	"StaticMode" "MotionMode"	string	Detection mode
detection_range	"FullScreen" "Customize"	string	Detection range
rule_info		JSON array	Single Channel Information JSON show as following Table-17.1.7.4
iva_lines	0 not draw lines 1 draw lines	int	Whether the preview and playback will appear the lines

copy_ch	"digit""analog""wifi"	string	the sign of support channel copy(NVR/DVR only)
---------	-----------------------	--------	--

Table-17.1.7.4 (rule_info Information JSON)

rule_number1		Object	Rule number 1, Information JSON show as following Table-17.1.7.5
rule_number2		Object	Rule number 2, Information JSON show as following Table-17.1.7.5
rule_number3		Object	Rule number 3, Information JSON show as following Table-17.1.7.5
rule_number4		Object	Rule number 4, Information JSON show as following Table-17.1.7.5

Table-7.11.5 (rule_number Information JSON)

rule_rect		Object	Each channel has maximum 64 point coordinates, that is 32 lines. If it is an accompanying line, only the first line is used. If it is a perimeter, these 8 lines are connected in sequence to form a closed figure. Information JSON show as following Table-17.1.7.6
rule_switch		bool	Rule switch
rule_type	“Legacy” “Lost” “Lost &Legacy”	string	Rule type

Table-17.1.7.6 (rule_line Information JSON)

x1	0-704	short	x1 coordinates
y1	0-576	short	y1 coordinates
x2	0-704	short	x2 coordinates
y2	0-576	short	y2 coordinates
x3	0-704	short	x3 coordinates
y3	0-576	short	y3 coordinates
x4	0-704	short	x4 coordinates
y4	0-576	short	y4 coordinates

Table-17.1.7.7(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_used	0~MAX_PARA_CH N_NUM	array	The channel numbers that has enabled cloud video upload function
max_cloud_video_upload_num	MAX_CLOUD_VIDEO_RECORD_NUM	int	The maximum number of channels supported by cloud video upload

analog_channel_is_ used	0~MAX_ANALOG_ CHN	array	The analog channel number of which the intelligent switch has enabled
max_intelligent_ana log_channel_num	1	int	The maximum number of analog channels supported by the intelligent switch

Example:

Request message:

POST/API/AI/Setup/SOD/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "channel": ["IP_CH2"]
  }
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "CH1": {
        "status": "Online",
        "switch": false,
        "sensitivity": 2,
        "latch_time": "5",
        "post_recording": "10",
        "send_email": false,
        "record_enable": false,
        "alarm_out": [],
        "rule_info": {
          "rule_number1": {
            "rule_switch": false,
            "rule_type": "Legacy",
            "rule_rect": {
              "x1": 0,
              "y1": 0,
              "x2": 0,
              "y2": 0,
              "x3": 0,
              "y3": 0,
              "x4": 0,
              "y4": 0
            }
          }
        }
      }
    }
  }
}
```

```

    }
  },
  "rule_number2": {
    "rule_switch": false,
    "rule_type": "Legacy",
    "rule_rect": {
      "x1": 0,
      "y1": 0,
      "x2": 0,
      "y2": 0,
      "x3": 0,
      "y3": 0,
      "x4": 0,
      "y4": 0
    }
  },
  "rule_number3": {
    "rule_switch": false,
    "rule_type": "Legacy",
    "rule_rect": {
      "x1": 0,
      "y1": 0,
      "x2": 0,
      "y2": 0,
      "x3": 0,
      "y3": 0,
      "x4": 0,
      "y4": 0
    }
  },
  "rule_number4": {
    "rule_switch": false,
    "rule_type": "Legacy",
    "rule_rect": {
      "x1": 0,
      "y1": 0,
      "x2": 0,
      "y2": 0,
      "x3": 0,
      "y3": 0,
      "x4": 0,
      "y4": 0
    }
  }
}

```

```

    }
  }
}
}
}

```

17.1.8 Crowd Density Detection

GET	
URL	POST /API/AI/Setup/CD/Get
Description	It is used to get the CD config parameters
Request Body	See Table -17.1.8.1
Successful Response	Channel Information JSON (show as following Table-17.1.8.2)

SET	
URL	POST /API/AI/Setup/CD/Set
Description	It is used to set theCD parameters
Request Body	Channel Information JSON (show as following Table-17.1.8.2)
Successful Response	The successful result response that described in 2.5

Table-17.1.8.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1

Table-17.1.8.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as following Table-17.1.8.x
page_type	“ChannelConfig”, “AlarmConfig”	string	To distinguish the data whether from a channel configuration page or an alarm configuration page
channel_max		int	Maximum number of channels

Table-17.1.8.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as following Table-17.1.8.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.1.8.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online" , "Nonsupport"	string	Channel online status, only for digital channels. Note: no such strings when the channel is online
alarm_out	"Local->1"... "Local->x" "IP_CH1->1"... "IP_CH1->2"... " IP_CHx->1" " IP_CHx->2" The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents aalarm output channel with a string. Camera: Local->1:enable null: disable
latch_time	"5","10","20", "30"	string	Alarm output time
record_enable	true, false	bool	Record channel switch
post_recording	"30","60","120", "300"	string	Recording time-lapse
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Channel capture FTP upload switch
ftp_video_upload	true, false	bool	Channel recording FTP upload switch
picture_to_cloud	true, false	bool	Picture Cloud upload switch
light_linkage	true, false	bool	Alarm linkage switch
switch	true, false	bool	CD ENABLE switch, false: close true: open

sensitivity	1-4	int	Detection sensitivity
iva_lines	0 not draw lines 1 draw lines	bool	Whether the preview and playback will appear the lines
max_pixel	320-1080	int	Maximum resolution
min_pixel	64-1080	int	Minimum resolution
max_detection_num	1-100	int	Detection number alarm threshold
detection_range	"FullScreen" "Customize"	string	Detection range /*0: Full Screen mode, 1: Customize mode*/
rule_info		JSON object	Single Channel Information JSON show as following Table-17.1.8.4

Table-17.1.8.4 (rule_info Information JSON)

rule_number1		Object	Rule number 1, Information JSON show as following Table-17.1.8.5
rule_number2		Object	Rule number 2, Information JSON show as following Table-17.1.8.5
rule_number3		Object	Rule number 3, Information JSON show as following Table-17.1.8.5
rule_number4		Object	Rule number 4, Information JSON show as following Table-17.1.8.5

Table-17.1.8.5 (rule_number Information JSON)

rule_switch		bool	Rule switch
rule_rect		Object	Octagon, Information JSON show as following Table-17.1.8.6

Table-17.1.8.6 (Channel Information JSON)

x1	0-704	short	x1 coordinates
y1	0-576	short	y1 coordinates
x2	0-704	short	x2 coordinates
y2	0-576	short	y2 coordinates
x3	0-704	short	x3 coordinates
y3	0-576	short	y3 coordinates
x4	0-704	short	x4 coordinates
y4	0-576	short	y4 coordinates
x5	0-704	short	x5 coordinates
y5	0-576	short	y5 coordinates
x6	0-704	short	x6 coordinates
y6	0-576	short	y6 coordinates
x7	0-704	short	x7 coordinates
y7	0-576	short	y7 coordinates
x8	0-704	short	x8 coordinates

y8	0-576	short	y8 coordinates
----	-------	-------	----------------

Example:

Request message:

POST /API/AI/Setup/PID/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "channel": ["CH1"]
  }
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "result": "success",
  "data": { "channel_info": { "CH1": {
    "status": "Online",
    "switch": false,
    "sensitivity": 2,
    "iva_lines": true,
    "max_pixel": 640,
    "min_pixel": 64,
    "max_detection_num": 50,
    "rule_info": { "rule_number1": {
      "rule_switch": true,
      "rule_rect": {
        "x1": 31,
        "y1": 20,
        "x2": 38,
        "y2": 532,
        "x3": 597,
        "y3": 529,
        "x4": 590,
        "y4": 338,
        "x5": 405,
        "y5": 349,
        "x6": 404,
        "y6": 481,
        "x7": 320,
        "y7": 478,
```

```

        "x8": 315,
        "y8": 18
    }
}
}
}
}

```

17.1.9 Queue Length Detection

GET	
URL	POST /API/AI/Setup/QD/Get
Description	It is used to get the QD config parameters
Request Body	See Table -17.1.9.1
Successful Response	Channel Information JSON (show as following Table-17.1.9.2)

SET	
URL	POST /API/AI/Setup/QD/Set
Description	It is used to set the QD parameters
Request Body	Channel Information JSON (show as following Table-17.1.9.2)
Successful Response	The successful result response that described in 2.5

Table-17.1.9.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1

Table-17.1.9.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as following Table-17.1.9.x
page_type	“ChannelConfig”, “AlarmConfig”	string	To distinguish the data whether from a channel configuration page or an alarm configuration page
channel_max		int	Maximum number of channels

Table-17.1.9.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as following Table-17.1.9.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.1.9.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online" , "Nonsupport"	string	Channel online status, only for digital channels. Note: no such strings when the channel is online
alarm_out	"Local->1"... "Local->x" "IP_CH1->1"... "IP_CH1->2"... " IP_CHx->1" " IP_CHx->2" The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents aalarm output channel with a string. Camera: Local->1:enable null: disable
latch_time	"5","10","20", "30"	string	Alarm output time
record_enable	true, false	bool	Record channel switch
post_recording	"30","60","120", "300"	string	Recording time-lapse
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Channel capture FTP upload switch
ftp_video_upload	true, false	bool	Channel recording FTP upload switch
picture_to_cloud	true, false	bool	Picture Cloud upload switch
light_linkage	true, false	bool	Alarm linkage switch
switch	true, false	bool	QD enable switch, false: close true: open

sensitivity	1-4	int	Detection sensitivity
iva_lines	0 not draw lines 1 draw lines	bool	Whether the preview and playback will appear the lines
max_pixel	320-1080	int	Maximum resolution
min_pixel	64-1080	int	Minimum resolution
max_detection_num	1-100	int	Detection number alarm threshold
max_pro_time	1-3600	int	Single person processing time alarm threshold
detection_range	"FullScreen" "Customize"	string	Detection range /*0: Full Screen mode, 1: Customize mode*/
rule_info		JSON object	Single Channel Information JSON show as following Table-17.1.9.4

Table-17.1.9.4 (rule_info Information JSON)

rule_number1		Object	Rule number 1, Information JSON show as following Table-17.1.9.5
rule_number2		Object	Rule number 2, Information JSON show as following Table-17.1.9.5
rule_number3		Object	Rule number 3, Information JSON show as following Table-17.1.9.5
rule_number4		Object	Rule number 4, Information JSON show as following Table-17.1.9.5

Table-17.1.9.5 (rule_number Information JSON)

rule_switch		bool	Rule switch
rule_rect		Object	Octagon, Information JSON show as following Table-17.1.9.6

Table-17.1.9.6 (Channel Information JSON)

x1	0-704	short	x1 coordinates
y1	0-576	short	y1 coordinates
x2	0-704	short	x2 coordinates
y2	0-576	short	y2 coordinates
x3	0-704	short	x3 coordinates
y3	0-576	short	y3 coordinates
x4	0-704	short	x4 coordinates
y4	0-576	short	y4 coordinates
x5	0-704	short	x5 coordinates
y5	0-576	short	y5 coordinates
x6	0-704	short	x6 coordinates
y6	0-576	short	y6 coordinates
x7	0-704	short	x7 coordinates

y7	0-576	short	y7 coordinates
x8	0-704	short	x8 coordinates
y8	0-576	short	y8 coordinates

Example:

Request message:

POST /API/AI/Setup/PID/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "channel": ["CH1"]
  }
}
```

Response message:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "result": "success",
  "data": { "channel_info": { "CH1": {
    "status": "Online",
    "switch": false,
    "sensitivity": 2,
    "iva_lines": true,
    "max_pixel": 640,
    "min_pixel": 64,
    "max_detection_num": 10,
    "max_pro_time": 10,
    "rule_info": { "rule_number1": {
      "rule_switch": true,
      "rule_rect": {
        "x1": 36,
        "y1": 96,
        "x2": 59,
        "y2": 356,
        "x3": 170,
        "y3": 361,
        "x4": 170,
        "y4": 472,
        "x5": 465,
        "y5": 472,
        "x6": 496,
```

```

        "y6": 337,
        "x7": 446,
        "y7": 180,
        "x8": 133,
        "y8": 106
    }
}
}
}
}
}

```

17.1.10 License Plate Detection

GET	
URL	POST /API/AI/Setup/LPD/Get
Description	It is used to get the LPD config parameters
Request Body	See Table -17.1.10.1
Successful Response	Channel Information JSON (show as following Table-17.1.10.2)

SET	
URL	POST /API/AI/Setup/LPD/Set
Description	It is used to set the LPD parameters
Request Body	Channel Information JSON (show as following Table-17.1.10.2)
Successful Response	The successful result response that described in 2.5

Table-17.1.10.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	To distinguish the data whether from a channel configuration page or an alarm configuration page

Table-17.1.10.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Single Channel Information JSON show as following Table-17.1.10.x
page_type	“ChannelConfig”, “AarmConfig”	string	To distinguish the data whether from a channel configuration page or an alarm configuration page, Requested only in Setting
channel_max		int	Maximum number of channels

Table-17.1.10.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as following Table-17.1.10.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.1.10.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online" , “Nonsupport”	string	Channel online status, only for digital channels. Note: no such strings when the channel is online
alarm_out	“Local->1”... ”Local->x” “IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2” The number of channels depends on the capabilities of the device.	array	Alarm output channel Each array bit represents aalarm output channel with a string. Camera: Local->1:enable null: disable
latch_time	"10","20","40", "60"	string	Alarm output time
record_enable	true, false	bool	Record channel switch
record_channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x”	array	Alarm output channel Channel alarm linkage switch.

	“WIFI_CH1”...”		
post_recording	"0","5","10", "20","30"	string	Recording time-lapse
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Channel capture FTP upload switch(NVR only)
ftp_video_upload	true, false	bool	Channel recording FTP upload switch(NVR only)
picture_to_cloud	true, false	bool	Picture Cloud upload switch(NVR only)
video_to_cloud	true, false	bool	Video Cloud upload switch(NVR only)
full_screen		bool	FullScreenswitch(NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer beeping time(NVR only)
show_message	true, false	bool	Show Message switch(NVR only)
switch	true, false	bool	switch, false: close true: open
sensitivity	0-100	int	Detection sensitivity
snap_mode	"Default" "RealTimeMode" "IntervalMode"	string	Snapshot mode, IntervalMode JSON show as following Table-17.1.10.4
min_pixel	64-1080	int	Minimum pixel value
max_pixel	320-1080	int	Maximum pixel value
detection_mode	"StaticMode" "MotionMode"	string	Detection mode
iva_lines	0 not draw lines 1 draw lines	int	Whether the preview and playback will appear the lines
lpd_enhance		bool	Licence plate detection enhance
day_enhance_level	0-255	int	Increase level during the day
night_enhance_level	0-255	Int	Increase level at night
rule_info		Object	Information JSON show as following Table-17.1.10.5

Table-17.1.10.4 (snap interval mode Information JSON)

snap_num	"1" "2" "3" "Unlimited"	string	Screenshot quantity
snap_frequency		int	Screenshot interval, unit s/pic

Table-17.1.10.5 (rule_info Information JSON)

rule_number1		Object	Rule number 1, Information JSON show as following Table-17.1.10.6
--------------	--	--------	---

Table-17.1.10.6 (rule_number Information JSON)

detection_range	"FullScreen" "Customize"	string	Detection range
rule_rect		Object	rectangle, Information JSON show as following Table-17.1.10.7

Table-17.1.10.7 (Channel Information JSON)

x1	0-704	short	x1 coordinates
y1	0-576	short	y1 coordinates
x2	0-704	short	x2 coordinates
y2	0-576	short	y2 coordinates
x3	0-704	short	x3 coordinates
y3	0-576	short	y3 coordinates
x4	0-704	short	x4 coordinates
y4	0-576	short	y4 coordinates

Table-17.1.10.8(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
cloud_video_is_use d	0~MAX_PARA_CH N_NUM	array	The channel numbers that has enabled cloud video upload function
max_cloud_video_u pload_num	MAX_CLOUD_VID EO_RECORD_NUM	int	The maximum number of channels supported by cloud video upload

Example:

Request message:

POST /API/AI/Setup/PVD/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "channel": ["CH1"],
    "page_type": "ChannelConfig"
  }
}
```

Response message:

```
{
```

```

"result": "success",
"data": {
  "channel_info": {
    "CH1": {
      "switch": false,
      "snap_mode": "IntervalMode",
      "snap_num": "1",
      "snap_frequency": 2,
      "min_pixel": 64,
      "max_pixel": 320,
      "sensitivity": 60,
      "detection_type": "Pedestrian",
      "detection_mode": "StaticMode",
      "rule_info": {
        "rule_number1": {
          "detection_range": "FullScreen",
          "rule_rect": {
            "x1": 1,
            "y1": 1,
            "x2": 0,
            "y2": 576,
            "x3": 704,
            "y3": 576,
            "x4": 704,
            "y4": 0
          }
        }
      }
    }
  }
}

```

Request message:

POST /API/AI/Setup/PVD/Get HTTP/1.1

```

{
  "version": "1.0",
  "channel": ["CH2"],
  "page_type": "AlarmConfig"
}

```

Response message:

```

{
  "result": "success",

```

```

"data": {
  "channel_info": {
    "CH2": {
      "buzzer": "0",
      "alarm_out": [
        "Local->1",
        "IP_CH2->1",
        "IP_CH3->1",
        "IP_CH5->1"
      ],
      "latch_time": "10",
      "record_enable": true,
      "record_channel": [
        "CH2"
      ],
      "post_recording": "30",
      "show_message": true,
      "send_email": true,
      "full_screen": false,
      "ftp_picture_upload": true,
      "ftp_video_upload": false,
      "picture_to_cloud": true,
      "video_to_cloud": false
    }
  }
}

```

17.1.11 Rare Sound Detection

GET	
URL	POST /API/AI/Setup/RSD/Get
Description	It is used to get the RSD config parameters
Request Body	See Table -17.1.1.1
Successful Response	Channel Information JSON (show as following Table-17.1.1.2)

SET	
URL	POST /API/AI/Setup/RSD/Set
Description	It is used to set the RSD parameters
Request Body	Channel Information JSON (show as following Table-17.1.1.2)

Successful Response	The successful result response that described in 2.5
---------------------	--

Table-17.1.10.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1
page_type	“ChannelConfig”, “AlarmConfig”	string	To distinguish the data whether from a channel configuration page or an alarm configuration page

Table-17.1.10.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Single Channel Information JSON show as following Table-17.1.2.x
page_type	“ChannelConfig”, “AarmConfig”	string	To distinguish the data whether from a channel configuration page or an alarm configuration page, Requested only in Setting
channel_max		int	Maximum number of channels

Table-17.1.10.x (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as following Table-17.1.10.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.1.10.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline", "Online", “Nonsupport”	string	Channel online status, only for digital channels. Note: no such strings when the channel is online

alarm_out	<p>“Local->1”... ”Local->x” “IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2”</p> <p>The number of channels depends on the capabilities of the device.</p>	array	<p>Alarm output channel</p> <p>Each array bit represents a alarm output channel with a string.</p> <p>Camera: Local->1:enable null: disable</p>
latch_time	"10","20","40", "60"	string	Alarm output time
record_enable	true, false	bool	Record channel switch
record_channel	<p>“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...”</p>	array	<p>Alarm output channel</p> <p>Channel alarm linkage switch.</p>
post_recording	"0","5","10", "20","30"	string	Recording time-lapse
send_email	true, false	bool	Send Email switch
ftp_picture_upload	true, false	bool	Channel capture FTP upload switch(NVR only)
ftp_video_upload	true, false	bool	Channel recording FTP upload switch(NVR only)
picture_to_cloud	true, false	bool	Picture Cloud upload switch(NVR only)
video_to_cloud	true, false	bool	Video Cloud upload switch(NVR only)
full_screen		bool	FullScreenswitch(NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer beeping time(NVR only)
show_message	true, false	bool	Show Message switch(NVR only)
switch	true, false	bool	switch, false: close true: open
sensitivity	1-100	int	Detection sensitivity
detection_type	<p>“Baby Crying Sound” "Dog Barking" "Gunshot"</p>	array	Detection type, Baby crying, dog barking, gunshot

Example:

Request message:

POST /API/AI/Setup/PVD/Get HTTP/1.1

```
{
  "version": "1.0",
```

```
“data”:{
  "channel":["CH1"],
  "page_type":"ChannelConfig"
}
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "CH1":{
        "switch": false,
        "snap_mode": "IntervalMode",
        "snap_num": "1",
        "snap_frequency": 2,
        "min_pixel": 64,
        "max_pixel": 320,
        "sensitivity": 60,
        "detection_type": "Pedestrian",
        "detection_mode": "StaticMode",
        "rule_info": {
          "rule_number1": {
            "detection_range": "FullScreen",
            "rule_rect": {
              "x1": 1,
              "y1": 1,
              "x2": 0,
              "y2": 576,
              "x3": 704,
              "y3": 576,
              "x4": 704,
              "y4": 0
            }
          }
        }
      }
    }
  }
}
```

Request message:

POST /API/AI/Setup/PVD/Get HTTP/1.1

```
{
  "version": "1.0",
  "channel": ["CH2"],
  "page_type": "AlarmConfig"
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "CH2": {
        "buzzer": "0",
        "alarm_out": [
          "Local->1",
          "IP_CH2->1",
          "IP_CH3->1",
          "IP_CH5->1"
        ],
        "latch_time": "10",
        "record_enable": true,
        "record_channel": [
          "CH2"
        ],
        "post_recording": "30",
        "show_message": true,
        "send_email": true,
        "full_screen": false,
        "ftp_picture_upload": true,
        "ftp_video_upload": false,
        "picture_to_cloud": true,
        "video_to_cloud": false
      }
    }
  }
}
```

17.1.12 AI Func Schedule

GET

URL	POST /API/AI/Setup/AISchedule/Get
Description	It is used to get the AI Func Schedule parameters
Request Body	none
Successful Response	AI Schedule JSON (show as following Table-17.1.12.1)

SET	
URL	POST /API/AI/Setup/AISchedule/Set
Description	It is used to set the AI Func Schedule parameters
Request Body	AI Schedule JSON (show as following Table-17.1.12.1)
Successful Response	The successful result response that described in 2.5

Table-17.1.12.1 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON object	Channel Information JSON show as following Table-17.1.12.2

Table-17.1.12.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as following Table-17.1.12.3
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.1.12.3 (Single Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
category		JSON array	CategoryJSON show as following Table-17.1.12.4
AI_Schedule		bool	AI function schedule switch

Table-17.1.12.4(CategoryJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
schedule_type	"fd", "pvd", "pid", "lcd", "sod", "cc",	string	Schedule Types Allowed to be Configured Note: Depending on device capabilities, actual schedule types may be less than allowable configurations

Successful Response	Model Configuration JSON (show as following Table-17.2.1.1)
---------------------	---

SET	
URL	POST /API/AI/Model/Set
Description	It is used to set the Face Model Configuration parameters
Request Body	Model Configuration JSON (show as following Table-17.2.1.1)
Successful Response	The successful result response that described in 2.5

Table-17.2.1.1 (Model Configuration JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
auto_select_enable		bool	Whether the model is automatically selected
rows		JSON array	Model Configuration JSON(show as following Table-17.2.1.2)

Table17.2.1.2(Model Configuration JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“local” “CH1”...”CH1x” The number of channels depends on the capabilities of the device.	string	Local model or Frontend model sign
face_recognition		string	Face recognition model version display
face_detection		string	Face detection model version display
enable_face_recognition	True false	bool	Facial recognition model enable switch

Example:

Request message:

POST /API/AI/Model/Get HTTP/1.1

```
{
  "version": "1.0"
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "rows": [
      {
        "channel": "local",
        "face_recognition": "-----",
        "face_detection": "-----",
```

```

        "enable_face_recognition": false
    },
    {
        "channel": "CH1",
        "face_recognition": "V0.2.0.0.1-release",
        "face_detection": "V0.2.1.2.1-release",
        "enable_face_recognition": true
    }
]
}
}

```

17.2.2Face Group

GET	
URL	POST/API/AI/FDGroup/Get
Description	It is used to get the Face Group parameters
Request Body	See Table -17.2.2.1
Successful Response	Face Group JSON (show as following Table-17.2.2.2)
Error_Code Response	Error Information JSON (show as following Table-17.2.2.x)

GET	
URL	POST/API/AI/FDGroup/GetId
Description	It is used to get the Face Group id
Request Body	See Table -17.2.2.12
Successful Response	Face Group JSON (show as following Table-17.2.2.13)
Error_Code Response	Error Information JSON (show as following Table-17.2.2.x)

ADD	
URL	POST /API/AI/FDGroup/Add
Description	It is used to add the Face Group parameters
Request Body	See Table -17.2.2.3
Successful Response	Face Group JSON (show as following Table-17.2.2.2)
Error_Code Response	Error Information JSON (show as following Table-17.2.2.x)

Modify	
URL	POST /API/AI/FDGroup/Modify
Description	It is used to modify the Face Group parameters
Request Body	Face Group JSON (show as following Table-17.2.2.2)
Successful Response	Response JSON (show as following Table-17.2.2.9)

Remove	
URL	POST /API/AI/FDGroup/Remove
Description	It is used to remove the Face Group
Request Body	Remove Face Group JSON (show as following Table-17.2.2.7)
Successful Response	Response JSON (show as following Table-17.2.2.9)

Table-17.2.2.1 (Get Face Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		Moot
TypeFlags		int	the detection type combination of the group to get, for example: 1-Face , 2-Car, 3-Human&Vehicle (0x1 << DLDT_Face 0x1 << DLDT_Car), Currently only supports faces
DefaultVal		int	Get the default alarm parameter or the actual alarm parameter, 1-default value, 0-actual value
WithInternal		int	Whether to include internal groups. Internal groups are not open to users and are only used to store common faces. When the client wants to obtain the list of common faces, the internal group Id should be used
SimpleInfo		int	Whether to obtain brief information or complete information, and the brief information only includes the Group name、 Id、 DetectType、 Policy、 Enabled、 CanDel、 Similarity, the full information contains all parameters of the group. 1-Get brief information, 0-Complete information
GroupsId	[1,2,3....x]	array	Get the configuration information of the corresponding group, if it is empty or the field does not exist, it means just get all

Table-17.2.2.3 (Add Face Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Group		array	Add Face Group JSON show as following Table-17.2.2.4

Table-17.2.2.4 (Add Face Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

MsgId	null		
Id		int	Face group ID, no need given when adding
Name		string	Face group Name, no need given when adding
DetectType	0	int	Detection type, To add a face group, just specify the DetectType field, 0 - DLDT_Face Face, 1 - DLDT_Car Car

Table17.2.2.2(Face Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result	0, -1, -2,-23, -24	int	Request return result, See Table-17.x.x, this field is not required for Modify
channel_max		int	Maximum number of channels
Count	1-16	int	Group number, used in get
Group		JSON array	Face Group, Face Group Information JSON show as following Table-17.2.2.5

Table-17.2.2.5 (Face Group Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Id		int	Face group ID
name	" Allow List" " Block List "	string	Face group Name, group name can be customized
DetectType	0,1	int	Detection type 0: face 1: car
Policy	0-4	int	Policy attributes, 0-Allow List, 1-Block List, 2-Stranger, 3-Internal (Internal is only used to store the most recent faces for comparison, not for alarm and other functions), 4-Advance
Enabled	0,1	int	Face alarm enable 0: off 1: on
CanDel		int	The predefined group cannot be deleted and is identified by the "CanDel" field, and 0 cannot be deleted
Similarity	0-100	unsigned int	Similarity
PolicyConfigs		JSON array	Group alarm policy configuration, fixed to 2 elements, the 0th element is used for the common group, the 0th element corresponds to Allow in the advanced group, and the 1st element corresponds to Deny. PolicyConfigs JSON show

			as following Table-17.2.2.6
EnableChnAlarm	[255, 255, 255, 255]	array	Control which channels the corresponding group is allowed to alarm on, and use it according to the bit position, and the bit position indicates that the corresponding channel allows the alarm
AlarmOut		Object	Alarm output association. Information JSON show as following Table-17.2.2.10

Table-17.2.2.6 (PolicyConfigs JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ChnAlarmOut	[[255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255]], ... [[255, 255, 0, 255], [255, 255, 0, 255], [255, 255,0,255]]	array	Each element represents the alarm output associated with a channel, A total of 3 main elements, the 0th represents the local alarm output, the 1/2 represents the front-end alarm output 0/1, and each bit of the sub-element represents whether the alarm output of the channel is associated
ChnBuzzerOpt	[0, 1, 1, 2, 3, 4, 0, 1, 1, 3, 2, 2 ...]	array	Each element represents the buzzer configuration of a channel, 0-disable, 1-10s, 2-20s, 3-40s, 4-60s
LatchTimeOpt	[0, 1, 1, 2, 3, 4, 0, 1, 1, 3, 2, 2 ...]	array	Each element represents the alarm output duration of a channel, 0-0s, 1-1s, 2-2s, 3-3s, 4-5s, 5-10s, 6-20s, 7- 40s, 8-60s
SaveImg	[255, 255, 255, 255...]	array	Each element represents whether a channel saves the image configuration, and the use of bits in each element represents whether to save the face and background
SendEmail	[0, 0, 0, 0]	array	Use by bit, each bit represents whether a channel sends mail, 0-not send, 1-send
UploadToFtp	[255, 255, 255, 255]	array	Use by bits, each bit represents whether a channel uploads FTP
UploadToCloud	[0, 0, 0, 0]	array	Use by bit, each bit represents whether a channel uploads Cloud, 0-not upload, 1-upload
ShowThumbnail	[255, 255, 255, 255]	array	Use by bit, each bit represents a channel is a thumbnail display
Record	[255, 255, 255, 255]	array	Use by bit, each bit represents that a channel is video
Push	[0, 0, 0, 0]	array	Use by bit, each bit represents whether a channel sends a push, 0-no push, 1-push
AlarmSchedule	[[255, 255, 255, 255,	array	Alarm schedule, each bit represents half an hour,

	255, 255], ...], [[255, 255, 255, 255, 255, 255] ...]		a total of 48 bits, set a bit to represents recording, otherwise no recording
--	---	--	--

Table-17.2.2.7 (Remove Face Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Group	0-16	array	RemoveFace Group JSON show as following Table-17.2.2.8

Table-17.2.2.8 (Remove Face Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Id		int	Delete the face group ID, the pre-defined group cannot be deleted

Table-17.2.2.9(Response JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result	0, -1, -2,-23, - 24	int	Return a result or an error code. See Table- 17.x.x

Table-17.2.2.10

KEY	VALUE		COMMENT
	RANGE	TYPE	
Local	"Local->1" "Local->2" "Local->3" The number of Locals depends on the capabilities of the device.	string array	Local alarm linkage.
Ipc		JSON array	Front-end alarm linkage. IPC JSON show as following Table-17.2.2.11

Table-17.2.2.11 (IPC JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
Channel		int	The channel associated with the front-end alarm output
AlarmOutCnt		int	Indicates which front-end alarm output, the front-end supports 2 alarm outputs

Table-17.2.2.12 (Get Face Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		Moot
TypeFlags		int	The combination of detection types of the group to be obtained, such as: 1-face, 2-car, 3-person-vehicle (0x1 << DLDT_Face 0x1 << DLDT_Car), currently only supports faces
DefaultVal		int	Get the default alarm value or the actual alarm value, 1-default value, 0-actual value
WithInternal		int	Whether to include internal groups. Internal groups are not open to users and are only used to store common faces. When the client wants to obtain the list of common faces, the internal group Id should be used
SimpleInfo		int	Whether to obtain brief information or complete information. The brief information only includes Group Name, Id, DetectType, Policy, Enabled, CanDel, and Similarity, and the complete information includes all parameters of the group. 1-Get brief information, 0-Complete information

Table17.2.2.13(Face Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	Nul		Moot
Result	0, -1, -2,-23, -24	int	Please refer to Table-17.x.x for the result returned by the request. This field is not required when modifying
GroupsId	[1,2,3...x]	array	The obtained group ID

Table-17.2.2.x(Error Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result	0, -1, -2,-23, -24	int	Return a result or an error code. See Table-17.x.x

channel_max		int	Maximum number of channels
Count		int	
Group	0-16	array	

Example:

Request message:

POST /API/AI/FDGroup/Get HTTP/1.1

```
{
  "data": {
    "MsgId": null,
    "TypeFlags": 1,
    "DefaultVal": 0,
    "WithInternal": 1,
    "SimpleInfo": 0
    "GroupsId": [
      2,
    ]
  }
}
```

Response message:

Success:

```
{
  "data": {
    "MsgId": null,
    "Result": 0,
    "Count": 5,
    "Group": [
      {
        "Id": 2,
        "Name": "Block List",
        "DetectType": 0,
        "Policy": 0,
        "Enabled": 1,
        "CanDel": 0,
        "Similarity": 70,
        "PolicyConfigs": [
          {
            "ChnAlarmOut": [
              [ 255, 255, 0, 255 ], [ 255, 255, 0, 255 ], [ 255, 255, 0, 255 ] ],
              [ 255, 255, 0, 255 ], [ 255, 255, 0, 255 ], [ 255, 255, 0, 255 ] ],
              ...
            ]
          }
        ]
      }
    ]
  }
}
```

```

        [ [255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255] ]
    ],
    "ChnBuzzerOpt": [0, 1, 1, 2, 3, 4, 0, 1, 1, 3, 2, 2 ],
    "LatchTimeOpt": [0, 1, 1, 2, 3, 4, 0, 1, 1, 3, 2, 2 ],
    "SaveImg": [255, 255, 255, 255],
    "SendEmail": [0, 0, 0, 0],
    "UploadToFtp": [255, 255, 255, 255],
    "UploadToCloud": [0, 0, 0, 0],
    "ShowThumbnail": [255, 255, 255, 255],
    "Record": [255, 255, 255, 255],
    "Push": [0, 0, 0, 0],
    "AlarmSchedule": [
        [
            [255, 255, 255, 255, 255, 255],
            ...
        ],
        [
            [255, 255, 255, 255, 255, 255],
            ...
        ],
        ...
    ]
},
{
    "ChnAlarmOut": [
        [ [255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255] ],
        [ [255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255] ],
        [ [255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255] ],
        ...
        [ [255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255] ]
    ],
    "ChnBuzzerOpt": [0, 1, 1, 2, 3, 4, 0, 1, 1, 3, 2, 2 ],
    "LatchTimeOpt": [0, 1, 1, 2, 3, 4, 0, 1, 1, 3, 2, 2 ],
    "SaveImg": [255, 255, 255, 255],
    "SendEmail": [0, 0, 0, 0],
    "UploadToFtp": [255, 255, 255, 255],
    "UploadToCloud": [0, 0, 0, 0],
    "ShowThumbnail": [255, 255, 255, 255],
    "Record": [255, 255, 255, 255],
    "Push": [0, 0, 0, 0],
    "AlarmSchedule": [
        [
            [255, 255, 255, 255, 255, 255],
            ...

```

```

        ],
        [
            [255, 255, 255, 255, 255, 255],
            ...
        ],
        ...
    ]
}
],
"EnableChnAlarm": [255, 255, 255, 255],
"AlarmOut": {
    "Local": ["Local->1"],
    "Ipc": [{
        "Channel": 1,
        "AlarmOutCnt": 1
    }, {
        "Channel": 4,
        "AlarmOutCnt": 1
    }, {
        "Channel": 5,
        "AlarmOutCnt": 1
    }]
}
},
...
]
}
}

```

Fail:

```

{
    "data": {
        "MsgId": null,
        "Result": -1,
        "Count": 0,
        "Group": []
    }
}

```

Example:

Request message:

POST /API/AI/FDGroup/GetId HTTP/1.1

```

{
    "version": "1.0",
    "data": {

```

```

    "MsgId": "",
    "DefaultVal": 0,
    "SimpleInfo": 0,
    "TypeFlags": 1,
    "WithInternal": 0
  }
}

```

Response message:

Success:

```

{"data": {
  "MsgId": "",
  "Result": 0,
  "GroupsId": [
    2,
    3,
    4
  ]
}}

```

Example:

Request message:

POST /API/AI/FDGroup/Modify HTTP/1.1

```

{
  "data": {
    "MsgId": null,
    "Group": [
      {
        "Id": 2,
        "Name": "Block List",
        "DetectType": 0,
        "Policy": 0,
        "Enabled": 1,
        "CanDel": 0,
        "Similarity": 70,
        "PolicyConfigs": [
          {
            "ChnAlarmOut": [
              [255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255] ],
              [255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255] ],
              ...
              [255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255] ]
            ],
            "ChnBuzzerOpt": [0, 1, 1, 2, 3, 4, 0, 1, 1, 3, 2, 2 ],
            "LatchTimeOpt": [0, 1, 1, 2, 3, 4, 0, 1, 1, 3, 2, 2 ],

```

```

"SaveImg": [255, 255, 255, 255],
"SendEmail": [0, 0, 0, 0],
"UploadToFtp": [255, 255, 255, 255],
"UploadToCloud": [0, 0, 0, 0],
"ShowThumbnail": [255, 255, 255, 255],
"Record": [255, 255, 255, 255],
"Push": [0, 0, 0, 0],
"AlarmSchedule": [
  [
    [255, 255, 255, 255, 255, 255],
    ...
  ],
  [
    [255, 255, 255, 255, 255, 255],
    ...
  ],
  ...
]
},
{
  "ChnAlarmOut": [
    [ [255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255] ],
    [ [255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255] ],
    [ [255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255] ],
    ...
    [ [255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255] ]
  ],
  "ChnBuzzerOpt": [0, 1, 1, 2, 3, 4, 0, 1, 1, 3, 2, 2 ],
  "LatchTimeOpt": [0, 1, 1, 2, 3, 4, 0, 1, 1, 3, 2, 2 ],
  "SaveImg": [255, 255, 255, 255],
  "SendEmail": [0, 0, 0, 0],
  "UploadToFtp": [255, 255, 255, 255],
  "UploadToCloud": [0, 0, 0, 0],
  "ShowThumbnail": [255, 255, 255, 255],
  "Record": [255, 255, 255, 255],
  "Push": [0, 0, 0, 0],
  "AlarmSchedule": [
    [
      [255, 255, 255, 255, 255, 255],
      ...
    ],
    [
      [255, 255, 255, 255, 255, 255],
      ...
    ]
  ]
}

```



```

        ],
        ...
    ]
}
},
"EnableChnAlarm": [255, 255, 255, 255]
},
...
]
}
}

```

Response message:

```

{
  "data": {
    "MsgId": null,
    "Result": 0
  }
}

```

Example:

Request message:

POST /API/AI/FDGroup/Remove HTTP/1.1

```

{
  "data": {
    "MsgId": null,
    "Group": [
      {
        "Id": 4,
      }
    ]
  }
}

```

Response message:

```

{
  "data": {
    "MsgId": null,
    "Result": 0
  }
}

```

17.2.3 Face

ADD	
URL	POST /API/AI/Faces/Add
Description	It is used to add the Faces
Request Body	See Table-17.2.3.1
Successful Response	Faces Information JSON (show as following Table-17.2.3.5)

Modify	
URL	POST /API/AI/Faces/Modify
Description	It is used to Modify the faces
Request Body	See Table-17.2.3.1
Successful Response	Faces Information JSON (show as following Table-17.2.3.5)

Change	
URL	POST /API/AI/FDGroup/Change
Description	It is used to Change the group the face belongs to
Request Body	Faces JSON (show as following Table-17.2.3.3)
Successful Response	Faces Information JSON (show as following Table-17.2.3.5)

Remove	
URL	POST /API/AI/Faces/Remove
Description	It is used to remove the Faces
Request Body	Faces JSON (show as following Table-17.2.3.3)
Successful Response	Faces Information JSON (show as following Table-17.2.3.5)

GET	
URL	POST/API/AI/Faces/GetImagesFeature
Description	Get Image Feature Value
Request Body	Images info JSON(show as following Table-17.2.3.7)
Successful Response	Feature Value info JSON(show as following Table-17.2.3.8)

Table-17.2.3.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Count		int	Add the count of multiple face information
FaceInfo		JSON array	Add Faces JSON show as following Table-17.2.3.2

Table-17.2.3.2

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Id		int	64-bit signed face ID, unique identifier, the Id field must be valid when modifying face information
GrpId		int	Group ID
Time		int	Starting time
Similarity		unsigned int	Similarity
Sex		int	Gender: 0-male, 1-female
Age		int	Age
Chn		int	Channel
ModifyCnt		int	The number of revisions, the upper layer is read-only, and is updated in real time by the device every time the face information is modified
Image1	"base64(imgData)"	string	It must be specified when adding, which is the main face image in the database. If "Image1" and "Feature" are given during Modify, the given "Image1" and "Feature" will be used directly (note that the feature value version also needs to be given). If only "Image1" is given, "Feature" will be recalculated. If "Image1" is not given, only fields other than "Image1" and "Feature" will be modified.
Image2	null	string	This parameter is only valid for real-time alarm
Image3	null	string	This parameter is only valid for real-time alarm
Feature	"base64(feature)"	string	Characteristics
FtVersion		int	Feature value version, it is meaningful only when comparing feature values with the same version, if return feature value, please return this field
Name	"Mike"	string	Name
Country	"China"	string	Country
Nation	"Han"	string	Nation
NativePlace	"Guangdong,Zhuhai"	string	NativePlace
IdCode	"415025199203050916"	string	ID card
Job	"Software"	string	Job
Phone	"12345678902"	string	Telephone number
Email	"abcd@163.com"	string	Email address
Domicile	"Guangdong,Zhuhai, Xiangzhou ..."	string	current residence

Remark	"Detail of this person ..."	string	Evaluate
EnableChnAlarm	[255, 255, 255, 255]	array	Special permission control, which controls which channels the face is allowed to alarm, and uses it according to the bit position. When the bit position is set, it means that the corresponding channel is allowed to alarm. If this field does not exist or is empty, it means that special permission control is not used.

Table-17.2.3.3 (Faces JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Count		int	Modify the number of face groups in which the belongs face group or delete the number of face groups
Group		int	The Id of the face group to be moved to, this field is only available for Change
FaceInfo		JSON array	Add Faces JSON show as following Table-17.2.3.4

Table-17.2.3.4 (Faces JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Id	-1	int	64-bit signed face ID,
MD5	"F74C70ADB0B63B00E279E71B4143704D"	string	When deleting face information, you only need to give the "ID" or MD5 field. The MD5 field has a higher priority. If there is an MD5 field, MD5 will be used first.

Table-17.2.3.5 (Faces Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Count		int	Group Number
Result	[0, 0,...]	array	Result refer to Table-17.x.x
Id	[2, 3,...]	array	The corresponding face ID after adding successfully
MD5	["F74C70ADB0B63B00E279E71B4143704	array	The MD5 value of the corresponding face

	D", "0194F781438F2DE8 FBE5B0469895036D "]		
--	--	--	--

Table-17.2.3.6 (Faces Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Id	-1	int	64-bit signed face ID,
MD5	"F74C70ADB0B63B0 0E279E71B4143704 D"	string	When deleting face information, you only need to give the "ID" or MD5 field. The MD5 field has a higher priority. If there is an MD5 field, MD5 will be used first.

Table-17.2.3.7(Images info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Images		string array	Image data encoded by Base64

Table-17.2.3.8(Feature Value info JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
FeatureVersion		int	Eigenvalue version
Features		string array	Get eigenvalue data
Results		int array	Get eigenvalue operation result

Example:

Request message:

POST /API/AI/Faces/Add HTTP/1.1

```
{
  "data": {
    "MsgId": null,
    "Count": 2,
    "FaceInfo": [
      {
        "Id": -1,
        "GrpId": 1,
        "Time": 0,
        "Similarity": 0,
        "Sex": 0,
        "Age": 26,
        "Chn": 0,

```

```

        "ModifyCnt": 0,
        "Image1": "base64(imgData)",
        "Image2": null,
        "Image3": null,
        "Feature": "base64(feature)",
        "FtVersion": 0
        "Name": "Mike",
        "Country": "China",
        "Nation": "Han",
        "NativePlace": "Guangdong,Zhuhai",
        "IdCode": "415025199203050916",
        "Job": "Software",
        "Phone": "12345678902",
        "Email": "abcd@163.com",
        "Domicile": "Guangdong,Zhuhai,Xiangzhou ...",
        "Remark": "Detail of this person ...",
        "EnableChnAlarm": [255, 255, 255, 255]
    }
}
]
}
}

```

Response message:

```

{
  "data": {
    "MsgId": null,
    "Count": 2,
    "Result": [0, 0],
    "Id": [2, 3],
    "MD5": ["F74C70ADB0B63B00E279E71B4143704D",
"0194F781438F2DE8FBE5B0469895036D"]
  }
}

```

Example:

Request message:

POST /API/AI/Faces/Remove HTTP/1.1

```

{
  "data": {
    "MsgId": null,
    "Count": 2,
    "FaceInfo": [
      {
        "Id": 2,

```

```

        "MD5": "F74C70ADB0B63B00E279E71B4143704D"
    },
    {
        "Id": 3,
        "MD5": "0194F781438F2DE8FBE5B0469895036D"
    }
]
}
}

```

Response message:

```

{
  "data": {
    "MsgId": null,
    "Count": 2,
    "Result": [0,0],
    "Id": [ 2,3],
    "MD5": [
      "F74C70ADB0B63B00E279E71B4143704D",
      "0194F781438F2DE8FBE5B0469895036D"
    ]
  }
}

```

17.2.4 Add Compare Face Image

Add	
URL	POST /API/AI/CompareFaces/Add
Description	It is used to add compareFaces
Request Body	See Table-17.2.4.1
Successful Response	Faces JSON (show as following Table-17.2.4.3) & Like example

Table-17.2.4.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Count		int	Add the total number of comparison face images
FaceInfo		JSON array	show as following Table-17.2.4.2
WithImage	0	int	Whether need a picture, 1- yes, 0- no
WithFeature	1	int	Whether need Characteristics, 1-required, 0-not required

Table-17.2.4.2

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Id	-1	int	64-bit signed face ID, unique identification, the ID field must be valid when modifying face information
GrpId		int	Group
Time		int	Starting time
Similarity		int	Similarity
Sex	0	int	Gender: 0-male, 1-female
Age		int	Age
Chn		int	Channel
ModifyCnt		int	Number of revisions, the upper layer is read-only, and is updated in real time by the device side every time the face information is modified
Image1	"base64(imgData)"	string	It must be specified when adding, which is the main face image in the database. If "Image1" and "Feature" are given during Modify, the given "Image1" and "Feature" will be used directly (note that the feature value version also needs to be given). If only "Image1" is given, "Feature" will be recalculated. If "Image1" is not given, only fields other than "Image1" and "Feature" will be modified.
Image2	""	string	This parameter is only valid for real-time alarm
Image3	""	string	This parameter is only valid for real-time alarm
Feature	"base64(feature)"	string	Characteristics
FtVersion		int	Feature value version, it is meaningful only when comparing feature values with the same version, if return feature value, please return this field
Name	""	string	Name
Country	""	string	Country
Nation	""	string	Nation
NativePlace	""	string	NativePlace
IdCode	""	string	ID Card
Job	""	string	Job
Phone	""	string	Telephone number
Email	""	string	Email address
Domicile	""	string	current residence
Remark	""	string	Evaluate

Table-17.2.4.3(Faces JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result		int	Result refer to Table-17.x.x, GetByIndex, GetById is valid for this field
Count		int	The total number of queried faces, GetByIndex, GetById is valid for this field
FaceInfo		JSON array	Add Faces JSON show as following Table-17.2.4.4

Table-17.2.4.4

KEY	VALUE		COMMENT
	RANGE	TYPE	
Id		int	Face ID, unique identifier, GetByIndex, Id field must be valid when GetById face information, there is no such field in Search
GrpId		int	Group ID
StartTime		unsigned long long	Unix timestamps, such as: 1540444116
EndTime		unsigned long long	Unix timestamps, such as: 1540444116
Similarity		float	Similarity
Sex	0	int	Gender: 0-male, 1-female
Age		int	Age
Chn		int	Channel
StrChn	"CH1"... "CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"... WIFI_CH1x" The number of channels depends on the capabilities of the device.	string	string channel number
ModifyCnt		int	Number of revisions, the upper layer is read-only, and is updated in real time by the device side every time the face information is modified
Image1	"base64(imgData)"	string	Database face master image.
Feature	"base64(feature)"	string	Characteristics.
FtVersion		int	Feature value version, it is meaningful only when

			comparing feature values with the same version. If you return feature values, please return this field as well. The FtVersion field is valid for GetByIndex and GetById face information, but not for Search
Name	"Mike"	string	Name
Country	"China"	string	Country
Nation	"Han"	string	Nation
NativePlace	"Guangdong,Zhuhai"	string	NativePlace
IdCode	"415025199203050916"	string	ID Card
Job	"Software"	string	Job
Phone	"12345678902"	string	Telephone number
Email	"abcd@163.com"	string	Email address

Domicile	"Guangdong,Zhuhai,Xiangzhou ..."	string	current residence
Remark	"Detail of this person ..."	string	Evaluate
MD5	"F74C70ADB0B63B00E279E71B4143704D"	string	The MD5 field is valid for MD5, GetByIndex, and GetById face information, but there is no such field for Search
EnableChnAlarm		array	Special permission control. Control which channels the face is allowed to alarm, use by bit, and the bit is set to indicate that the corresponding channel is allowed to alarm. If this field does not exist or is empty, it means that the special authority control is not used, but the alarm channel setting of the group is used.

Example:

Request message:

POST /API/AI/CompareFaces/Add HTTP/1.1

```
{
  "data": {
    "MsgId": null,
    "Count": 2,
    "WithImage": 0,
    "WithFeature": 1
    "FaceInfo": [
      {
        "Image1": "base64(imgData)",
        "Feature": "base64(feature)",
        "FtVersion": 0
      }
    ]
  }
}
```

```

    },
    {
        "Image1": "base64(imgData)",
        "Feature": null,
        "FtVersion": 0
    }
]
}
}

```

Response message:

```

{
    "data": {
        "MsgId": null,
        "Result": [0, -3],
        "Count": 3,
        "FaceInfo": [
            {
                "Id": 5,
                "GrpId": 0,
                "Time": 0,
                "Similarity": 0,
                "Sex": 0,
                "Age": 0,
                "Chn": 0,
                "StrChn": "1",
                "ModifyCnt": 0,
                "Image1": "base64(imgData)",
                "Image2": null,
                "Image3": null,
                "Feature": "base64(feature)",
                "FtVersion": 0,
                "Name": "",
                "Country": "",
                "Nation": "",
                "NativePlace": "",
                "IdCode": "",
                "Job": "",
                "Phone": "",
                "Email": "",
                "Domicile": "",
                "Remark": "",
                "MD5": "294C703DB05F3B00E279E71B41437E46"
            },

```

```

    {
      "Id": -1,
      "GrpId": 0,
      "Time": 0,
      "Similarity": 0,
      "Sex": 0,
      "Age": 0,
      "Chn": 0,
      "StrChn": "1",
      "ModifyCnt": 0,
      "Image1": null,
      "Image2": null,
      "Image3": null,
      "Feature": null,
      "FtVersion": 0 ,
      "Name": "",
      "Country": "",
      "Nation": "",
      "NativePlace": "",
      "IdCode": "",
      "Job": "",
      "Phone": "",
      "Email": "",
      "Domicile": "",
      "Remark": "",
      "MD5": "397E503DB05F3B00E279E71B41437E46"
    }
  ]
}

```

17.2.5 Database face information query

Search	
URL	POST /API/AI/AddedFaces/Search
Description	It is used to searchAddedFaces
Request Body	Faces JSON (show as following Table-17.2.5.1)
Successful Response	Response JSON (show as following Table-17.2.5.3)

GetByIndex

URL	POST /API/AI/AddedFaces/GetByIndex
Description	It is used to getAddedFaces
Request Body	See Table-17.2.5.4
Successful Response	Faces JSON (show as following Table-17.2.5.1)

GetById	
URL	POST /API/AI/AddedFaces/GetById
Description	It is used to getAddedFaces
Request Body	See Table-17.2.5.5
Successful Response	Faces JSON (show as following Table-17.2.5.1) & Like example

Table-17.2.5.1(Faces JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result		int	Result refer to Table-17.x.x, GetByIndex, GetById is valid for this field
channel_max		int	Maximum number of channels
Count		int	The total number of faces queried, GetByIndex and GetById are valid for this field
FaceInfo		JSON array	Add Faces JSON show as following Table-17.2.5.2

Table-17.2.5.2

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Id		int	Face ID, unique identifier, GetByIndex, ID field must be valid when GetById face information, there is no such field in Search
GrpId		int	Group Id
StartTime		unsigned long long	Unix timestamp, such as: 1540444116
EndTime		unsigned long long	Unix timestamp, such as: 1540444116
Similarity		float	Similarity
Sex	0	int	Gender: 0-male, 1-female
Age		int	Age
Chn		int	Channel
StrChn	“CH1”...”CH1x” “IP_CH1”...”	string	String channel number

	IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.		
ModifyCnt		int	Number of revisions, the upper layer is read-only, and is updated in real time by the device side every time the face information is modified
Image1	"base64(imgData)"	string	Database face master image. The Image1 field is valid for GetByIndex and GetById face information, but not for Search
Image2	null	string	This field is only valid for real-time alarms, but not for Search
Image3	null	string	This parameter is only valid for real-time alarm, but not for Search
Feature	"base64(feature)"	string	Characteristics. The Feature field is valid for GetByIndex and GetById face information, but not for Search
FtVersion		int	Feature value version, it is meaningful only when comparing feature values with the same version. If you return feature values, please return this field as well. The FtVersion field is valid for GetByIndex and GetById face information, but not for Search
Name	"Mike"	string	Name
Country	"China"	string	Country
Nation	"Han"	string	Nation
NativePlace	"Guangdong,Zhuhai"	string	NativePlace
IdCode	"415025199203050916"	string	ID Card
Job	"Software"	string	Job
Phone	"12345678902"	string	Telephone number
Email	"abcd@163.com"	string	Email address

Domicile	"Guangdong,Zhuhai, Xiangzhou ..."	string	current residence
Remark	"Detail of this person ..."	string	Evaluate
MD5	"F74C70ADB0B63B0"	string	The MD5 field is valid for MD5, GetByIndex,

	0E279E71B4143704 D"		GetById face information, but not for Search
EnableChnAlarm		array	Special permission control. Control which channels the face is allowed to alarm, use by bit, and the bit is set to indicate that the corresponding channel is allowed to alarm. If this field does not exist or is empty, it means that the special authority control is not used, but the alarm channel setting of the group is used.

Table-17.2.5.3(Response JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result		int	Result refer to Table-17.x.x
channel_max		int	Maximum number of channels
Count		int	Search for the total number of faces added

Table-17.2.5.4

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
StartIndex	0	int	The starting index of the request, for example, the first time: 0-19, the second time: 20-39, then the StartIndex is 0, 20 respectively
Count	20	int	The number of face information requested
SimpleInfo	0	int	Obtain brief information, each face information in the response message only contains Id, GrpId and Name fields, 1-simplified mode, 0-normal mode
WithImage	1	int	Whether need a picture, 1- yes, 0- no
WithFeature	1	int	Whether to need characteristic s, 1-required, 0-not required
NeedMD5	0	int	Whether MD5 is required, MD5 will also be given in non-abbreviated mode

Table-17.2.5.5

KEY	VALUE		COMMENT
	RANGE	TYPE	

MsgId	null		
FacesId	[1, 5, 6, 20, 53, 25...]	array	A list of face IDs that need to be requested, each array bit corresponds to an ID with an integer number.
FacesMD5	["F75C70ADB0B63B00E279E71B4143704D", "B74C70ADB0B63B00E279B71B4193704F", "A29B70ADB0B63B00E2793C1B4123504D", "B34C70A3B0B53B00E279571B4143704F", "AC3C70ADB3B63B40E279EE1B41F3C04D", "B74A70ADB0B63400E279E71B4143804F" ...]	array	Face MD5 value
SimpleInfo	0	int	Obtain brief information, each face information in the response message only contains Id, GrpId and Name fields, 1-simplified mode, 0-normal mode
WithImage	1	int	Whether need a picture, 1- yes, 0- no
WithFeature	1	int	Whether need characteristics, 1-required, 0-not required
NeedMD5	0	int	Whether MD5 is required, 1-required, 0-not required, MD5 will also be given in non-abbreviated mode

Example:

Request message:

POST /API/AI/AddedFaces/Search HTTP/1.1

```
{
  "data": {
    "MsgId": null,
    "FaceInfo": [
      {
        "GrpId": 1,
        "Time": 0,
        "Similarity": 0,
        "Sex": 0,
        "Age": 0,
        "Chn": 0,

```



```
        "ModifyCnt": 0,
        "Name": "",
        "Country": "",
        "Nation": "",
        "NativePlace": "",
        "IdCode": "",
        "Job": "",
        "Phone": "",
        "Email": "",
        "Domicile": "",
        "Remark": ""
    }
]
}
}
```

Response message:

```
{
  "data": {
    "MsgId": null,
    "Result": 0,
    "Count": 153
  }
}
```

Example:

Request message:

POST /API/AI/AddedFaces/GetById HTTP/1.1

```
{
  "data": {
    "MsgId": null,
    "FacesId": [1, 5, 6, 20, 53, 25],
    "FacesMD5": ["F75C70ADB0B63B00E279E71B4143704D",
                 "B74C70ADB0B63B00E279B71B4193704F",
                 "A29B70ADB0B63B00E2793C1B4123504D",
                 "B34C70A3B0B53B00E279571B4143704F",
                 "AC3C70ADB3B63B40E279EE1B41F3C04D",
                 "B74A70ADB0B63400E279E71B4143804F"],
    "SimpleInfo": 0
    "WithImage": 1,
    "WithFeature": 1,
    "NeedMD5": 0
  }
}
```

```
}  
}
```

Response message:

```
{  
  "data": {  
    "MsgId": null,  
    "Result": 0,  
    "Count": 1,  
    "FaceInfo": [  
      {  
        "Id": 1,  
        "GrpId": 1,  
        "Time": 0,  
        "Similarity": 0,  
        "Sex": 0,  
        "Age": 26,  
        "Chn": 0,  
        "StrChn": "1",  
        "ModifyCnt": 0,  
        "Image1": "base64(imgData)",  
        "Image2": null,  
        "Image3": null,  
        "Feature": "base64(feature)",  
        "FtVersion": 0,  
        "Name": "Mike",  
        "Country": "China",  
        "Nation": "Han",  
        "NativePlace": "Guangdong,Zhuhai",  
        "IdCode": "415025199203050916",  
        "Job": "Software",  
        "Phone": "12345678902",  
        "Email": "abcd@163.com",  
        "Domicile": "Guangdong,Zhuhai,Xiangzhou ...",  
        "Remark": "Detail of this person ...",  
        "MD5": "F75C70ADB0B63B00E279E71B4143704D"  
      }  
    ]  
  }  
}
```

17.2.6 Additional Face Image

Get	
URL	POST /API/AI/ExtraFaces/Get
Description	It is used to GetExtraFaces
Request Body	See Table-17.2.6.1
Successful Response	ExtraFaces JSON (show as follow Table-17.2.6.2)

GetById	
URL	POST /API/AI/ExtraFaces/GetById
Description	It is used to getExtraFacesById
Request Body	See Table-17.2.6.1
Successful Response	Extra Faces JSON (show as follow Table-17.2.6.2)

Add	
URL	POST /API/AI/ExtraFaces/Add
Description	It is used to getAddedFaces
Request Body	Extra Faces JSON (show as follow Table-17.2.6.2)
Successful Response	Like example Add

Remove	
URL	POST /API/AI/ExtraFaces/Remove
Description	It is used to getAddedFaces
Request Body	Extra Faces JSON (show as follow Table-17.2.6.2)
Successful Response	Like example Remove

Table-17.2.6.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
FaceId		int	The Id of the additional face to be requested. Note that it is the face Id of the additional face. This field is valid when getting.
Id		int	The Id of the additional face to be requested. Note that it is not the face Id to which it belongs. This field is valid when GetById
WithImage	1	int	Do you need pictures? 1 - Yes, 0 - No
WithFeature	1	int	Do you need the characteristic value?

			1 - Yes, 0 - No
--	--	--	--------------------

Table-17.2.6.2(ExtraFaces JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result		int	See Table-17. x.x for the results. This field is valid for Get and GetById, but no field for Add and Remove
Count		int	Total faces you searched.
ExtFaceInfo		JSON array	ExtraFaces JSON show as follow Table-17.2.6.3

Table-17.2.6.3(ExtraFaces JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Id		int	Refer to additional face ID The Id of the attached face is not the face Id it belongs to
FaceId		int	The Id of the attached face, the face Id of the attached face
Image	"base64(imgData)"	string	To attach a face image and remove an additional face, just give the "Id" field, which uses the default value
Feature	"base64(feature)"	string	To add a face feature value and remove an additional face, just give the "Id" field, which uses the default value
FtVersion	0	int	The feature value version is meaningful only when the characteristic values with the same version are compared. If returning the characteristic value, please also return this field

Example:

Request message:

POST /API/AI/ExtraFaces/Add HTTP/1.1

```
{
  "data": {
    "MsgId": null,
    "Count": 2,
    "ExtFaceInfo": [
      {
        "Id": -1,
        "FaceId": 1,
        "Image": "base64(imgData)",
```

```

        "Feature": "base64(feature)",
        "FtVersion": 0
    },
    {
        "Id": -1,
        "FaceId": 1,
        "Image": "base64(imgData)",
        "Feature": "base64(feature)",
        "FtVersion": 0
    }
]
}
}

```

Response message:

```

{
  "data": {
    "MsgId": null,
    "Count": 2,
    "Result": [0, 0],
    "Id": [1, 2]
  }
}

```

Example:

Request message:

POST /API/AI/ExtraFaces/Remove HTTP/1.1

```

{
  "data": {
    "MsgId": null,
    "Count": 2,
    "ExtFaceInfo": [
      {
        "Id": 1,
        "FaceId": 0,
        "Image": null,
        "Feature": null,
        "FtVersion": 0
      },
      {
        "Id": 2,
        "FaceId": 0,
        "Image": null,
        "Feature": null,
        "FtVersion": 0
      }
    ]
  }
}

```

```

    }
  ]
}
}

```

Response message:

```

{
  "data": {
    "MsgId": null,
    "Count": 2,
    "Result": [0, 0],
    "Id": [1, 2]
  }
}

```

17.2.7 Snaped Faces Search and Match

Search	
URL	POST /API/AI/SnapedFaces/Search
Description	It is used to searchSnapedFaces
Request Body	search SnapedFaces JSON (show as follow Table-17.2.7.1)
Successful Response	SnapedFaces JSON (show as follow Table-17.2.7.3)

GetByIndex	
URL	POST /API/AI/SnapedFaces/GetByIndex
Description	It is used to getSnapedFacesByIndex
Request Body	See Table-17.2.7.4
Successful Response	SnapedFaces JSON (show as follow Table-17.2.7.3)

GetById	
URL	POST /API/AI/SnapedFaces/GetById
Description	It is used to getSnapedFacesById
Request Body	See Table-17.2.7.5
Successful Response	SnapedFaces JSON (show as follow Table-17.2.7.3)

StopSearch	
URL	POST /API/AI/SnapedFaces/StopSearch
Description	It is used to stopSearchSnapedFaces
Request Body	StopSearch SnapedFaces JSON (show as follow Table-17.2.7.7)
Successful Response	StopSearch SnapedFaces JSON (show as follow Table-17.2.7.7)

Table-17.2.7.1 (search SnapedFaces JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
StartTime		string	Search Start time, for example:"2020-07-12 00:00:00"
EndTime		string	Search End time, for example: : "2020-07-12 23:59:59"
Chn	0~MAX_PARA_CH N_NUM	array	The channel you want to search. The value stands for one channel.
AlarmGroup		array	The group to which the face to be displayed belongs during real-time capture. The value represents the group ID of the selected group. If it is blank, all (including deleted groups) will be displayed. If the group is given, the result will be filtered by the given group ID.
Similarity	0-100	int	The similarity during matching. The faces below this similarity will be filtered out [0, 100] as the normal value range. If the faces do not match, this parameter can be given - 1
Engine	0	int	Considering the superposition of the search process and operation logic (for example, after the search results are found on the matching page, you can open the face selection dialog box to add a new face. At this time, the matched cache information will be overwritten by the results of the new search and become invalid), the device side provides two search engines with equivalent functions (0 and 1) for this purpose. The data and cache of the two engines are completely independent and do not interfere with each other, The division of labor is allocated by the client itself. For example, the number 0 engine can be used to realize some temporary face selection Service (for the selection dialog box), using the data of engine 1 to display the search or matching results.
Count		int	Number of matching pictures searched
Gender		array	Gender: 0- male 1- female
fAttrAge	[0, 1, 2, 3, 4, 5, 6]	array	Age range. The value represents the corresponding age range. [[0, 17], [18, 25], [26, 30], [31, 35], [36, 40],[41, 50], [51, 127], [-1,

			0]]. [- 1, 0] indicates that the age range found is within 1 year old
Beauty	[0, 1, 2, 3, 4]	array	Face value, the value represents the corresponding value. [[0, 9], [10, 29], [30, 69], [70, 89], [90, 100]]
GlassesType	[0, 1]	array	Type of glasses, 0: no glasses, 1: wear glasses (at present, there is no distinction between sunglasses and ordinary glasses, all glasses are default)
Expression	[0, 1, 2]	array	Expression type, 0: no expression, 1: smile, 2: laugh
MouthMask	[0, 1]	array	The value represents the search type, 0: no mask, 1: with mask
Race	[0, 1, 2, 3]	array	Race, the number represents the corresponding type, 0: yellow, 1: white, 2: black, 3: Arab
FaceInfo		JSONarray	search SnapedFaces JSON (show as follow Table-17.2.7.2)

Table-17.2.7.2(search SnapedFaces JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Id		int	
MD5	"F74C70ADB0B63B00E279E71B4143704D"	string	MD5 value
Feature	"base64(feature)"	string	Feature value
FtVersion		int	The characteristic value version is meaningful only when the characteristic values with the same version are compared. If returning the characteristic value, please also return this field

Table-17.2.7.3(SnapedFaces JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result	0	int	Result see Table-17.x.x
Count		int	The total number of captured faces in this response

Table-17.2.7.4

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
MsgId	null		
Engine	0	int	Which engine was used, 0 or 1, this will match with <Search> interface Which search engine to use, 0 or 1, works with the search interface
MatchedFaces	1	int	Get the matched face or capture the face, 1 - Get the matched face, 0 - Get the captured face, only GetByIndex has this field
StartIndex		int	The starting index of the request, for example, the first time: 0-19, the second time: 20-39, then the StartIndex is 0 and 20 respectively. Only GetByIndex has this field
Count		int	The number of requested face information. Only GetByIndex has this field
SimpleInfo	1	int	Get brief information or complete information. 1 - brief information, 0 - complete information. The brief information only contains MatchedId, Similarity and UUIId at most. If no face is matched, only UUIId is included. Only GetByIndex has this field
UUIIds		array	The unique identifier of the snapshot information to be obtained. Only GetById has this field
WithFaceImage	1	int	Whether face image is needed, 1 - Yes, 0 - No
WithBodyImage	0	int	Whether the human body picture is needed, 1 - Yes, 0 - No
WithBackground	0	int	Whether background image is required, 1 - Yes, 0 - No
WithFeature	1	int	Whether the characteristic value is required, 1 - Yes, 0 - No

Table-17.2.7.5(SnapedFaces JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result	0	int	Result see Table-17.x.x
TotalCount		int	The total number of faces searched or matched successfully. Only GetByIndex has this field
Count		int	The total number of captured faces in this response
SnapedFaceInfo		array	SnapedFaces JSON (show as follow Table-17.2.7.6)

Table-17.2.7.6(SnapedFaces JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
UUIId		string	The unique ID of the captured face information, which can be directly used to obtain the detailed information of the face in the future
MatchedFaceId		int	The matched face Id. If there is no match, this field is not available. Only GetByIndex has this field
MatchedMD5		string	The matched face MD5. If there is no match, this field is not available. Only GetByIndex has this field
Chn		int	channel
StrChn	"CH1"... "CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"... WIFI_CH1x" The number of channels depends on the capabilities of the device.	string	channel
Similarity		float	The similarity with the matched face. If there is no match, this field is not available. Only GetByIndex has this field
StartTime		unsigned long long	Unix time stamp, such as: 1540444116
EndTime		unsigned long long	Unix time stamp, such as: 1540444116
FaceImage	"base64(imgData)"	string	Captured face pictures from the front-end IPC
BodyImage	"base64(imgData)"	string	Captured human body pictures from the front-end IPC
Background	"base64(imgData)"	string	Snapshot background image from front-end IPC
Feature	"base64(feature)"	string	Feature value
FtVersion		int	The feature value version is meaningful only when the characteristic values with the same version are compared. If returning the characteristic value, please also return this field
SnapId		unsignedint	The ID of the picture when the front-end IPC snaps, 32-bit unsigned integer
Type	0	int	Capture face type sign
Score		int	Picture score (confidence)
Gender		int	Gender:

			0- male 1- female
fAttrAge		int	age
Beauty		int	level of appearance
GlassesType		int	Type of glasses, 0: no glasses, 1: wear glasses (at present, there is no distinction between sunglasses and ordinary glasses, all glasses are default)
Expression		int	Expression type, 0: no expression, 1: smile, 2: laugh
MouthMask		int	Whether to wear a mask, 0: no mask, 1: with a mask
Race		int	Race, 0: Yellow, 1: White, 2: Black, 3: Arab

Table-17.2.7.7(StopSearch SnapedFaces JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Engine		array	List of engines to release cache
Result	0	int	Result see Table-17.x.x

Example:

Request message:

POST /API/AI/SnapedFaces/Search HTTP/1.1

```
{
  "msgType": "AI_searchSnapedFaces",
  "data": {
    "MsgId": null,
    "StartTime": "2018-10-20 00:00:00",
    "EndTime": "2018-10-28 23:59:59",
    "Chn": [0, 1, 2, 3, 4, 5, 6, 7, 8],
    "AlarmGroup": [1, 2, 5, 9, 13],
    "Similarity": 70,
    "Engine": 0,
    "Count": 2,
    "FaceInfo": [
      {
        "Id": 2,
        "MD5": "F74C70ADB0B63B00E279E71B4143704D",
        "Feature": "base64(feature)",
        "FtVersion": 0
      }
    ]
  }
}
```

```

        },
        {
            "Id": 5,
            "MD5": "294C703DB05F3B00E279E71B41437E46",
            "Feature": "base64(feature)",
            "FtVersion": 0
        }
    ]
}

```

Response message:

```

{
  "data": {
    "MsgId": null,
    "Result": 0,
    "Count": 600
  }
}

```

Example:

Request message:

POST /API/AI/SnapedFaces/GetByIndex HTTP/1.1

```

{
  "data": {
    "MsgId": null,
    "Engine": 0,
    "MatchedFaces": 1,
    "StartIndex": 0,
    "Count": 20,
    "SimpleInfo": 1

    "WithFaceImage": 1,
    "WithBodyImage": 0,
    "WithBackgroud": 0,
    "WithFeature": 1
  }
}

```

Response message:

```

{
  "data": {
    "MsgId": null,
    "Result": 0,
    "TotalCount": 600,
    "Count": 20,

```

```
"SnapedFaceInfo": [  
  {  
    "UUIId": 103,  
    "MatchedFaceId": 5,  
    "MatchedMD5": "294C703DB05F3B00E279E71B41437E46",  
    "Chn": 3,  
    "StrChn": "4",  
    "Similarity": 89.39759,  
    "StartTime": 1540444116,  
    "EndTime": 1540444136,  
    "FaceImage": "base64(imgData)",  
    "BodyImage": "base64(imgData)",  
    "Background": "base64(imgData)",  
    "Feature": "base64(feature)",  
    "FtVersion": 0  
    "SnapId": 2375,  
    "Type": 0,  
    "Score": 60,  
    "Gender": 0,  
    "fAttrAge": 25,  
    "Beauty": 51,  
    "GlassesType": 1,  
    "Expression": 0,  
    "MouthMask": 1,  
    "Race": 1  
  },  
  {  
    "UUIId": 126,  
    "MatchedFaceId": 2,  
    "MatchedMD5": "F74C70ADB0B63B00E279E71B4143704D",  
    "Chn": 3,  
    "StrChn": "4",  
    "Similarity": 96.87693,  
    "StartTime": 1540444116,  
    "EndTime": 1540444136,  
    "FaceImage": "base64(imgData)",  
    "BodyImage": "base64(imgData)",  
    "Background": "base64(imgData)",  
    "Feature": "base64(feature)",  
    "FtVersion": 0  
    "SnapId": 2376,  
    "Type": 0,  
    "Score": 60,  
    "Gender": 0,  
  }  
]
```

```

        "fAttrAge": 25,
        "Beauty": 51,
        "GlassesType": 1,
        "Expression": 0,
        "MouthMask": 1,
        "Race": 1
    },
    {
        ...
    },
    ...
]
}
}

```

17.2.8 Snaped Objects Search

Search	
URL	POST /API/AI/SnapedObjects/Search
Description	It is used to searchSnapedObjects
Request Body	search SnapedObjects JSON (show as follow Table-17.2.8.1)
Successful Response	SnapedObjects Response JSON (show as follow Table-17.2.8.2)

GetByIndex	
URL	POST /API/AI/SnapedObjects/GetByIndex
Description	It is used to getSnapedObjectsByIndex
Request Body	See Table-17.2.8.3
Successful Response	SnapedObjects JSON (show as follow Table-17.2.8.4)

GetById	
URL	POST /API/AI/SnapedObjects/GetById
Description	It is used to getSnapedObjectsById
Request Body	See Table-17.2.8.3
Successful Response	SnapedObjects JSON (show as follow Table-17.2.8.4)

StopSearch	
URL	POST /API/AI/SnapedObjects/StopSearch
Description	It is used to stopSearchSnapedObjects
Request Body	StopSearch SnapedObjects JSON (show as follow Table-17.2.8.6)
Successful Response	StopSearch SnapedObjects JSON (show as follow Table-17.2.8.6)

Table-17.2.8.1 (search SnapedObjects JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
StartTime	"2020-07-12 00:00:00"	string	Search <Start Time>
EndTime	"2020-07-1223:59:59"	string	Search <End Time>
Chn	0~MAX_PARA_CH N_NUM	array	The channel you searched. Value stand for Channel number
Type	[1, 2]	array	The type you searched. For example: 1 – Shape 2 – Vehicle 3 – PID Human 4 – PID Vehicle 5 – LCD Human 6 LCD Vehicle
Engine	0	int	Search Engine (0 & 1)

Table-17.2.8.2(SnapedObjects Response JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result		int	See Table-17. x.x for the search results of the captured objects (including people, cars, etc.)
Count	0	int	Actual quantity of captured objects

Table-17.2.8.3

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Engine	0	int	Which engine was used, 0 or 1, this will match with <Search> interface
StartIndex		int	Request Start Index, For example: 1 st : 0~19; 2 nd : 20-39 StartIndex should be 0, 20 seperately. And only GetByIndex have this field.
Count		int	Request the count for human or Car. Only GetByIndex have this field.
SimpleInfo	1	int	Get one simple info or whole info. 1 – Simple info 0 – Whole info

			Brief info only covers UUID, and Only GetByIndex have this field.
UUIDs	[103, 105, 190, 129]	array	Get the unique identifier of the snapshot information. Only GetById has this field
WithObjectImage	1	int	Request the image of the object (such as Human and car) 1 – Request 0 – Not request
WithBackground	0	int	Request the background image 1 – Request 0 – Not request

Table-17.2.8.4(SnapedObjects JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result	0	int	Refer to Table-17.x.x for the result
TotalCount		int	The total count you searched. Only GetByIndex have this field.
Count		int	Total response objects
SnapedObjInfo		array	SnapedObjects JSON (show as follow Table-17.2.8.5)

Table-17.2.8.5(SnapedObjects JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
UUID		string	The unique ID of the captured object information, which can be directly used to get the detailed information of the object in the future.
Chn		int	Channel
StrChn	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string	Channel
StartTime		unsigned long long	Unix Time stamp, For example:1540444116
EndTime		unsigned long	Unix Time stamp, For example:1540444116

		long	
ObjectImage	"base64(imgData)"	string	Captured images of Human and cars from the front-end IPC
Background	"base64(imgData)"	string	Captured background images of Human and cars from the front-end IPC
SnapId		unsigned int	Captured Snap ID from the front-end IPC
Type	1	int	The type for captured objects. 0 - face 1 – Human shape 2 – Vehicle Shape 3 – PID Human shape 4 – PID Vehicle shape 5 – LCD Human shape 6 – LCD Vehicle shape

Table-17.2.8.6(StopSearch SnapedObjects JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Engine		array	Engine list to release cache.
Result	0	int	Refer to Table-17.x.x for the result

Example:

Request message:

POST/API/AI/SnapedObjects/Search HTTP/1.1

```
{
  "data": {
    "MsgId": null,
    "StartTime": "2018-10-20 00:00:00",
    "EndTime": "2018-10-28 23:59:59",
    "Chn": [0, 1, 2, 3, 4, 5, 6, 7, 8],
    "Type": [1, 2],
    "Engine": 0,
  }
}
```

Response message:

```
{
  "data": {
    "MsgId": null,
    "Result": 0,
    "Count": 600
  }
}
```

```
}  
}
```

Example:

Request message:

POST/API/AI/SnapedObjects/GetByIndex HTTP/1.1

```
{  
  "data": {  
    "MsgId": null,  
    "Engine": 0,  
    "StartIndex": 0,  
    "Count": 20,  
    "SimpleInfo": 1,  
    "WithObjectImage": 0,  
    "WithBackgroud": 0  
  }  
}
```

Response message:

```
{  
  "data": {  
    "MsgId": null,  
    "Result": 0,  
    "TotalCount": 600,  
    "Count": 20,  
    "SnapedObjInfo": [  
      {  
        "UUIId": 103,  
        "Chn": 3,  
        "StrChn": "CH4",  
        "StartTime": 1540444116,  
        "EndTime": 1540444137,  
        "ObjectImage": "base64(imgData)",  
        "Background": "base64(imgData)",  
        "SnapId": 2375,  
        "Type": 1  
      },  
      {  
        "UUIId": 105,  
        "Chn": 6,  
        "StrChn": "CH7",  
        "StartTime": 1540444116,  
        "EndTime": 1540444139,  
        "ObjectImage": "base64(imgData)",  
        "Background": "base64(imgData)",  
      }  
    ]  
  }  
}
```

```

        "SnapId": 2375,
        "Type": 2
    },
    {
        ...
    },
    ...
]
}
}

```

17.2.9 License Plate Group

GET	
URL	POST/API/AI/PlateGroup/Get
Description	It is used to get the License Plate Group parameters
Request Body	See Table -17.2.9.1
Successful Response	License Plate Group JSON (show as follow Table-17.2.9.4)
Error_Code Response	Error Information JSON (show as follow Table-17.2.9.12)

GET	
URL	POST/API/AI/PlateGroup/GetId
Description	It is used to get the License Plate Group Id
Request Body	See Table -17.2.9.13
Successful Response	License Plate Group JSON (show as follow Table-17.2.9.14)
Error_Code Response	Error Information JSON (show as follow Table-17.2.9.12)

ADD	
URL	POST /API/AI/PlateGroup/Add
Description	It is used to add the License Plate Group parameters
Request Body	See Table -17.2.9.3
Successful Response	License Plate Group JSON (show as follow Table-17.2.9.4)
Error_Code Response	Error Information JSON (show as follow Table-17.2.9.12)

Modify	
URL	POST /API/AI/PlateGroup/Modify
Description	It is used to modify the License Plate Group parameters
Request Body	License Plate Group JSON (show as follow Table-17.2.9.2)
Successful Response	Response JSON (show as follow Table-17.2.9.9)

Remove	
--------	--

URL	POST /API/AI/PlateGroup/Remove
Description	It is used to remove the License Plate Group
Request Body	Remove License Plate Group JSON (show as follow Table-17.2.9.7)
Successful Response	Response JSON (show as follow Table-17.2.9.9)

Table-17.2.9.1 (Get License Plate Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		no practical significance
DefaultVal		int	Get default alarm parameters or actual alarm ammetersers 1 – Default parameters 0 – Actual parameters
SimpleInfo		int	Get one simple info or whole info. Simple info only covers Group name, Id, DetectType, Policy, Enabled, CanDel and Similarity. Whole info cover all the parameter in the group 1 – Get SimpleInfo 0 – Get WholeInfo
GroupsId	[1,2,3...x]	array	Get the configuration information of the corresponding group. If the field is empty or not, get datas of all groups

Table-17.2.9.2 (Add License Plate Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Group		array	Add License Plate Group JSON show as follow Table-17.2.9.4

Table-17.2.9.3 (Add License Plate Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Id		int	License Plate Group Id
Name		string	LP group Name
DetectType	2	int	Detect Type enum DLDetectType {

			DLDT_Face, DLDT_Object, DLDT_Plate, DLDT_All }; The license plate set is fixed to 2(DLDT_Plate), Modifying will be unvalid
--	--	--	--

Table17.2.9.4(License Plate Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result	0, -1, -2,-23, -24	int	Please see Table-17.2.9.12 for the result. No need this field when modifying
channel		string array	Channel Name. No need this field when modifying
Count	0-64	int	License Plate Group Qty
Group		JSON array	License Plate Group Information JSON show as follow Table-17.2.9.5

Table-17.2.9.5 (License Plate Group Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Id		int	LP Group ID
Name	" Allow List" " Block List "	string	LP Group name, which can be customized and can't be duplicated.
DetectType	2	int	The set is fixed to 2 (DLDT_Plate)
Policy	0-2	int	Policy attribute 0 – Allow list 1 – Block list 2 – Unknown
Enabled	0,1	int	PL alarm enable: 0 – Disable 1 – Enable
CanDel	0,1	int	Predefined groups cannot be deleted. They are identified by the "CanDel" field. 0 cannot be deleted
Similarity	0-4	unsigned int	Fault tolerance rate, the maximum number of characters allowed for error, for example, 0 means zero tolerance, require all characters

			match
PolicyConfigs		JSON array	Group alarm Policy Configs Fixed to 2 elements, only use “0” element and use-up “1”. PolicyConfigs JSON show as follow Table-17.2.9.6
EnableChnAlarm	[255, 255, 255, 255]	array	It controls which channels the corresponding group is allowed to alarm in, and is used by bit, which indicates that the corresponding channel is allowed to alarm
AlarmOut		Object	Alarm out linkage Information JSON show as follow Table-17.2.9.10

Table-17.2.2.6 (PolicyConfigs JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ChnAlarmOut	[[255, 255, 0, 255], [255, 255, 0, 255], [255, 255, 0, 255]], ... [[255, 255, 0, 255], [255, 255, 0, 255], [255, 255,0,255]]	array	Each element represents the alarm output associated with a channel, there are three main elements in total, 0 – Local alarm out 1/2 – Front-end alarm out 0/1 - each bit of the sub-element indicates whether the alarm output of the channel is related
ChnBuzzerOpt	[0, 1, 1, 2, 3, 4, 0, 1, 1, 3, 2, 2 ...]	array	Every element indicates buzzer configs for one channel 0-disable, 1-10s, 2-20s, 3-40s, 4-60s
LatchTimeOpt	[0, 1, 1, 2, 3, 4, 0, 1, 1, 3, 2, 2 ...]	array	Every element indicates alarm latch time for one channel. 0-0s, 1-1s, 2-2s, 3-3s, 4-5s, 5-10s, 6-20s, 7- 40s, 8-60s
SaveImg	[255, 255, 255, 255...]	array	Every element indicates if one channel will save image configs. Each element is used by bit to indicate whether to save face and background
SendEmail	[0, 0, 0, 0]	array	Used by bit, each bit indicates whether a channel sends mail, 0-not to send, 1-send
UploadToFtp	[255, 255, 255, 255]	array	Used by bit, each bit indicates if a channel upload to FTP
UploadToCloud	[0, 0, 0, 0]	array	Used by bit, each bit indicates whether a channel sends Cloud or not, 0-not send, 1-send
ShowThumbnail	[255, 255, 255, 255]	array	Used by bit, each bit indicates whether a channel shows thumbnail or not.

Record	[255, 255, 255, 255]	array	Used by bit, each bit indicates whether a channel record or not.
Push	[0, 0, 0, 0]	array	Used by bit, each bit indicates whether a channel sends Push notification or not, 0-not push, 1-push
AlarmSchedule	[[255, 255, 255, 255, 255, 255], ...], [[255, 255, 255, 255, 255, 255] ...]	array	Alarm Schedule, each bit stands for half an hour. total 48bits Setting means recording, otherwise not recording

Table-17.2.9.7 (Remove License Plate Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Group		array	Remove License Plate Group JSON show as follow Table-17.2.9.8

Table-17.2.9.8 (Remove License Plate Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Id		int	License Plate Group Id which need to be removed Predefined groups cannot be deleted

Table-17.2.9.9(Response JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result		integer array	Response result or error code. See Table-17.2.9.12

Table-17.2.9.10

KEY	VALUE		COMMENT
	RANGE	TYPE	
Local	"Local->1" "Local->2" "Local->3" The number of Locals depends on the capabilities of the device.	string array	Local alarm linkage

Ip		JSON array	Front-end IPC alarm linkage IPC JSON show as follow Table-17.2.9.11
----	--	------------	--

Table-17.2.9.11 (IPC JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Channel		int	Linkable channel which front-end IPC trigger alarm out
AlarmOutCnt		int	Indicates the number of front-end alarm outputs. The front-end supports 2 alarm outputs

Table-17.2.9.12(Error Code Information)

KEY	VALUE	COMMENT
AORT_SUCCESS	0	Success
AORT_NO_DB	-1	No database
AORT_DB_EXEC_FAILED	-2	Database execuattion failed
AORT_CALC_FEATURE_FAILED	-3	Feature extraction failed
AORT_CANCELED	-4	Cancelled
AORT_NO_DISK	-5	No disk
AORT_DISK_ERROR	-6	Disk error
AORT_EXIST	-7	Exit
AORT_GROUP_INVALID	-8	Group Invalid
AORT_NOT_EXIST	-9	Not exist
AORT_MORE_FILE_EXIST	-10	Record file exists
AORT_SEARCH_ERROR	-11	HDD error
AORT_OVER_MAX_COUNT	-12	over the upper limit
AORT_UPDATING_FEATURE	-13	Updating the feature value
AORT_NO_USABLE_IPC	-14	No IPC available for eigenvalue calculation
AORT_INVALID_PARAM	-15	Invalid parameters
AORT_INVALID_FORMAT	-16	Invalid format
AORT_INVALID_RES	-17	Invalid resolution
AORT_INVALID_MEM	-18	Invalid file (over-large File)
AORT_CREAT_FAILED	-19	Creation failed
AORT_MD5_NOT_MATCH	-20	MD5 not match
AORT_POS_ERROR	-21	Pos error
AORT_SIZE_ERROR	-22	Size error
AORT_NOT_READY	-23	Not ready
AORT_INVALID_DB	-24	Invalid database
AORT_GRP_INVALID_OR_INTERNAL	-25	Invalid Group or Internal pre-group not allowed

		to be deleted.
AORT_GRP_FACE_OVER_MAX	-26	The number of faces in the group have exceeded the upper limit
AORT_NO_PERMISSION	-27	No permission
AORT_GRP_PLATE_OVER_MAX	-28	The number of License Plates in the group have exceeded the upper limit

Table-17.2.9.13 (Get License Plate Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
DefaultVal		int	Get default alarm parameters or actual alarm ammetersers 1 – Default parameters 0 – Actual parameters
SimpleInfo		int	Get one simple info or whole info. Simple info only covers Group name, Id, DetectType, Policy, Enabled, CanDel and Similarity. A whole info covers all the parameters in the group 1 – Get SimpleInfo 0 – Get WholeInfo
TypeFlags	1	int	The set is fixed to 1
WithInternal	0,1	int	It is used to control whether to return the ID of an internal group. The internal group is not open to users. If you want to get the ID of this group, WithInternal - 1. General - 0

Table17.2.9.14(License Plate Group JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result	0, -1, -2,-23, -24	int	For request the result please see Table-17.2.9.12, Modify will not need this field.
GroupsId	[1,2,3...x]	array	Group Id you get

Example:

Request message:

POST /API/AI/PlateGroup/Add HTTP/1.1

```
{
  "data": {
    "Group": [
      {
        "Name": "Test Group 1"
      },
      {
        "Name": "Test Group 2"
      }
    ]
  }
}
```

Response message:

```
{
  "data": {
    "Result": [
      0,
      0
    ],
    "Group": [
      {
        "Id": 7,
        "Name": "Test Group 1",
        "Policy": 0,
        "DetectType": 2,
        "Similarity": 1,
        "CanDel": 1,
        "Enabled": 1,
        "EnableAlarm": 1,
        "PolicyConfigs": [...],
        "EnableChnAlarm": [...],
        "AlarmOut": {...}
      },
      {
        "Id": 8,
        "Name": "Test Group 2",
        "Policy": 0,
        "DetectType": 2,
        "Similarity": 1,
        "CanDel": 1,
        "Enabled": 1,
        "EnableAlarm": 1,
        "PolicyConfigs": [...],
        "EnableChnAlarm": [...],

```

```

        "AlarmOut": {...}
    }
]
}
}

```

Example:

Request message:

POST /API/AI/PlateGroup/Modify HTTP/1.1

```

{
  "data": {
    "Group": [
      {
        "Id": 7,
        "Name": " Test group1",
        "Policy": 0,
        "DetectType": 2,
        "Similarity": 1,
        "CanDel": 1,
        "Enabled": 1,
        "EnableAlarm": 1,
        "PolicyConfigs": [...],
        "EnableChnAlarm": [...],
        "AlarmOut": {...}
      },
      {
        "Id": 8,
        "Name": "Test group2",
        "Policy": 0,
        "DetectType": 2,
        "Similarity": 1,
        "CanDel": 1,
        "Enabled": 1,
        "EnableAlarm": 1,
        "PolicyConfigs": [...],
        "EnableChnAlarm": [...],
        "AlarmOut": {...}
      }
    ]
  }
}

```

Response message:

```

{
  "data": {

```

```
        "Result": [
            0,
            0
        ]
    }
}
```

Example:

Request message:

POST /API/AI/PlateGroup/Remove HTTP/1.1

```
{
  "data": {
    "Group": [
      {
        "Id": 7
      },
      {
        "Id": 8
      }
    ]
  }
}
```

Response message:

```
{
  "data": {
    "Result": [
      0,
      0
    ]
  }
}
```

Example:

Request message:

POST /API/AI/PlateGroup/GetId HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "MsgId": "",
    "DefaultVal": 0,
    "SimpleInfo": 0,
    "TypeFlags": 1,
    "WithInternal": 0
  }
}
```

```
}
```

Response message:

```
{  
  {"data": {  
    "MsgId": "",  
    "Result": 0,  
    "GroupsId": [  
      1,  
      2,  
      3,  
      4,  
      5,  
      6,  
    ]  
  }}  
}
```

Example:

Request message:

POST /API/AI/PlateGroup/Get HTTP/1.1

```
{  
  "data": {  
    "MsgId": "",  
    "DefaultVal": 0,  
    "SimpleInfo": 1,  
    "GroupsId": [  
      1,  
      2,  
      3,  
      ...  
    ]  
  }  
}
```

Response message:

```
{  
  "data": {  
    "channel": [  
      "CH1",  
      "CH2",  
      "CH3",  
      ...  
    ],  
    "Result": 0,  
    "Count": 4,  
    "Group": [  

```

```

{
  "Id": 1,
  "Name": "Allow list",
  "Policy": 0,
  "DetectType": 2,
  "Similarity": 1,
  "CanDel": 0,
  "Enabled": 1,
  "EnableAlarm": 1
},
{
  "Id": 2,
  "Name": "Block list",
  "Policy": 1,
  "DetectType": 2,
  "Similarity": 1,
  "CanDel": 0,
  "Enabled": 1,
  "EnableAlarm": 1
},
{
  "Id": 3,
  "Name": "Unknown",
  "Policy": 2,
  "DetectType": 2,
  "Similarity": 1,
  "CanDel": 0,
  "Enabled": 1,
  "EnableAlarm": 1
},
...
]
}
}

```

17.2.10 License Plate

ADD	
URL	POST /API/AI/Plates/Add
Description	It is used to add the license plates
Request Body	See Table-17.2.10.1
Successful Response	License Plates Information JSON (show as follow Table-17.2.10.5)

Modify	
URL	POST /API/AI/Plates/Modify
Description	It is used to modify the license plates
Request Body	See Table-17.2.10.1
Successful Response	License Plates Information JSON (show as follow Table-17.2.10.5)

Change	
URL	POST /API/AI/Plates/ChangeGroup
Description	It is used to change the group the license plate belongs to
Request Body	License Plates JSON (show as follow Table-17.2.10.3)
Successful Response	License Plates Information JSON (show as follow Table-17.2.10.5)

Remove	
URL	POST /API/AI/Plates/Remove
Description	It is used to remove the License Plates
Request Body	License Plates JSON (show as follow Table-17.2.3.3)
Successful Response	License Plates Information JSON (show as follow Table-17.2.3.5)

Table-17.2.10.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
PlateInfo		JSON array	Add License Plates JSON show as follow Table-17.2.3.2

Table-17.2.10.2

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Id	1-15characters	string	License Plate Number, Unique identity
GrpId		int	Group Id
PlateColor	0-5	int	Plate Color, available values are as follows: enum AIPlateColor_e { APC_Blue = 0, // Blue APC_Green = 1, // Green APC_Yellow = 2, // Yellow APC_Black = 3, // Black APC_White = 4, // White

			APC_Other = 5, // Other APC_Max };
CarBrand		string	Car Brand
CarType		string	Car Type
Sex		int	Gender of vehicle owner: 0– male. 1– female
Owner	"Mike"	string	Owner name
IdCode	"415025199203050916"	string	Identity Code
Job	"Software"	string	Job
Phone	"12345678902"	string	Phone
Domicile	"Guangdong,Zhuhai, Xiangzhou ..."	string	Address
Remark	"Detail of this person ..."	string	Remark
EnableChnAlarm	[255, 255, 255, 255]	array	Special permission control, which controls which channels the face is allowed to alarm, Used by bit, which indicates that the corresponding channel is allowed to alarm, This field does not exist or empty, which means that special permission control is not used
VoicePromptsChn	0~255	int	File index. When there is no audio file, this is 255
VoicePromptsSel		array	Playback channel, calculated by bit Bit 0 – local bit1 - front-end channel 1, bit2 – front-end channel 23 ...
VoicePromptsTm		array	Time period, 12 time periods cannot conflicts

Table-17.2.10.3 (License Plates JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Group		int	The face group to be moved to. This field is only available for ChangeGroup
PlateInfo		JSON array	Add License Plates JSON show as follow Table-17.2.10.4

Table-17.2.10.4 (License Plates JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Id	1-15 characters	string	License Plate Id

Table-17.2.10.5 (Response JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Count		int	Group count
Result	[0, 0,...]	array	Please refer to Table-17.2.9.12

Example:**Request message:**

POST /API/AI/Plates/Add HTTP/1.1

```
{
  "data": {
    "PlateInfo": [
      {
        "Id": "粤CW2763",
        "GrpId": 6,
        "PlateColor": 0,
        "Sex": 1,
        "CarBrand": "Volkswagen ",
        "CarType": "两厢车",
        "Owner": "张三",
        "IdCode": "12125180",
        "Job": "职业",
        "Phone": "15271859302",
        "Domicile": "居住地1",
        "Remark": "备注",
        "EnableChnAlarm": []
      },
      {
        "Id": "粤CK3961",
        "GrpId": 6,
        "PlateColor": 1,
        "Sex": 1,
        "CarBrand": "大众",
        "CarType": "三厢车",
        "Owner": "李四",
        "IdCode": "12125181",
```

```
        "Job": "职业",
        "Phone": "15271859303",
        "Domicile": "居住地2",
        "Remark": "备注",
        "EnableChnAlarm": []
    }
]
}
```

Response message:

```
{
  "data": {
    "Count": 2,
    "Result": [
      0,
      0
    ]
  }
}
```

Example:

Request message:

POST /API/AI/Plates/Modify HTTP/1.1

```
{
  "data": {
    "PlateInfo": [
      {
        "Id": "粤CW2763",
        "GrpId": 6,
        "PlateColor": 1,
        "Sex": 1,
        "CarBrand": "大众",
        "CarType": "两厢车",
        "Owner": "张三三",
        "IdCode": "12125180",
        "Job": "职业",
        "Phone": "15271859302",
        "Domicile": "居住地1",
        "Remark": "备注",
        "EnableChnAlarm": []
      },
      {
        "Id": "粤CK3961",
        "GrpId": 6,
```

```
        "PlateColor": 2,
        "Sex": 1,
        "CarBrand": "大众",
        "CarType": "三厢车",
        "Owner": "李四四",
        "IdCode": "12125181",
        "Job": "职业",
        "Phone": "15271859303",
        "Domicile": "居住地2",
        "Remark": "备注",
        "EnableChnAlarm": []
    }
}
]
```

Response message:

```
{
  "data": {
    "Count": 2,
    "Result": [
      0,
      0
    ]
  }
}
```

Example:

Request message:

POST /API/AI/Plates/Remove HTTP/1.1

Body:

```
{
  "data": {
    "PlateInfo": [
      {
        "Id": "粤CW2763"
      },
      {
        "Id": "粤CK3961"
      }
    ]
  }
}
```

Response message:

```
{
```

```

    "data": {
      "Count": 2,
      "Result": [
        0,
        0
      ]
    }
  }
}

```

Example:

Request message:

POST /API/AI/Plates/ChangeGroup HTTP/1.1

Body:

```

{
  "data": {
    "Group": 1,
    "PlateInfo": [
      {
        "Id": "粤CW2763"
      },
      {
        "Id": "粤CK3961"
      }
    ]
  }
}

```

Response message:

```

{
  "data": {
    "Count": 2,
    "Result": [
      0,
      0
    ]
  }
}

```

17.2.11 Database license plate information query

GetCount	
URL	POST /API/AI/AddedPlates/GetCount
Description	It is used to get added license plates count
Request Body	License Plates JSON (show as follow Table-17.2.11.1)

Successful Response	Response JSON (show as follow Table-17.2.11.3)
---------------------	--

GetId	
URL	POST /API/AI/AddedPlates/GetId
Description	It is used to get license plates id
Request Body	See Table-17.2.11.1
Successful Response	License Plates JSON (show as follow Table-17.2.11.3)

GetById	
URL	POST /API/AI/AddedPlates/GetById
Description	It is used to get license plate information by license plates id
Request Body	See Table-17.2.11.4
Successful Response	License Plates JSON (show as follow Table-17.2.11.3)

Table-17.2.11.1(License Plates JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
PlateInfo		JSON array	Get license plates count JSON show as follow Table-17.2.11.2 Use only the first element of the array when searching When GetId is selected, this field and the following GrpId field should be selected alternatively. This field is prioritily chosed compared to the GrpId field. If this field is provided, the GrpId field will be invalid.
GrpId		integer array	Only GetId will be valid.

Table-17.2.11.2

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Id	1-15characters	string	License Plate number When searching, the field is optional.
GrpId		int	Group Id When searching the field is optional If the word provided is within the valid range (only - 1 will be regarded as an invalid value), it

			will be used as the exact condition, otherwise the field will be ignored
PlateColor	0-5	int	<p>License Plate color</p> <p>Available values are as follows:</p> <pre>enum AIPlateColor_e { APC_Blue = 0, // Blue APC_Green = 1, // Green APC_Yellow = 2, // Yellow APC_Black = 3, // Black APC_White = 4, // White APC_Other = 5, // Other APC_Max</pre> <p>};</p> <p>When searching the field is optional</p> <p>If the word provided is within the valid range, it will be used as the exact condition, otherwise the field will be ignored</p>
CarBrand		string	<p>Car brand</p> <p>When searching the field is optional</p>
CarType		string	<p>Car type</p> <p>When searching the field is optional</p>
Sex		int	<p>Gender of owner</p> <p>0– male.</p> <p>1– female</p> <p>If the word provided is within the valid range, it will be used as the exact condition, otherwise the field will be ignored</p>
Owner	"Mike"	string	<p>Owner name</p> <p>When searching the field is optional</p>
IdCode	"415025199203050916"	string	<p>Id code</p> <p>When searching the field is optional</p>
Job	"Software"	string	<p>Job</p> <p>When searching the field is optional</p>
Phone	"12345678902"	string	<p>Phone number</p> <p>When searching the field is optional</p>
Domicile	"Guangdong,Zhuhai,Xiangzhou ..."	string	<p>Address</p> <p>When searching the field is optional</p>
Remark	"Detail of this person ..."	string	<p>Remark</p> <p>When searching the field is optional</p>
EnableChnAlarm	[255, 255, 255, 255]	array	<p>Special permission control, which controls which channels the face is allowed to alarm,</p> <p>Used by bit, which indicates that the corresponding channel is allowed to alarm,</p> <p>This field does not exist or empty, which means</p>

			that special permission control is not used The field is not used during the search.
--	--	--	---

Table-17.2.11.3(Response JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result		int	See Table-17.2.9.12
Count		int	Total number of eligible license plates in the database searched
PlatesId		string array	A list of eligible license plate numbers. Only the GetId result has this field
PlateInfo		JSON array	Only the GetId result has this field

Table-17.2.11.4

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
PlatesId		string array	Get the License Plate list details

Example:

Request message:

POST /API/AI/AddedPlates/GetCount HTTP/1.1

```
{
  "data": {
    "PlateInfo": [
      {
        "Id": "粤C"
      }
    ]
  }
}
```

Response message:

```
{
  "data": {
    "Result": 0,
    "Count": 2
  }
}
```

Example:

POST /API/AI/AddedPlates/GetId

(1) Fuzzy query

Request message:

```
{
  "data": {
    "PlateInfo": [
      {
        "Id": "粵CW"
      }
    ]
  }
}
```

Response message:

```
{
  "data": {
    "Result": 0,
    "Count": 1,
    "PlatesId": [
      "粵CW2763"
    ]
  }
}
```

(2) Get from the group Id list

Request message:

```
{
  "data": {
    "GrpId": [1, 2, 6]
  }
}
```

Response message:

```
{
  "data": {
    "Result": 0,
    "Count": 2,
    "PlatesId": [
      "粵CW2763",
      "粵CK3961"
    ]
  }
}
```

Example:

Request message:

POST /API/AI/AddedPlates/GetById

```
{
  "data": {
    "PlatesId": [
```



```
        "粤CW2763",
        "粤CK3961"
    ]
}
```

Response message:

```
{
  "data": {
    "Result": 0,
    "Count": 2,
    "PlateInfo": [
      {
        "Id": "粤CW2763",
        "GrpId": 1,
        "PlateColor": 0,
        "Sex": 1,
        "CarBrand": "大众",
        "CarType": "两厢车",
        "Owner": "张三",
        "IdCode": "12125180",
        "Job": "职业",
        "Phone": "15271859302",
        "Domicile": "居住地1",
        "Remark": "备注",
        "EnableChnAlarm": []
      },
      {
        "Id": "粤CK3961",
        "GrpId": 1,
        "PlateColor": 1,
        "Sex": 1,
        "CarBrand": "大众",
        "CarType": "三厢车",
        "Owner": "李四",
        "IdCode": "12125181",
        "Job": "职业",
        "Phone": "15271859303",
        "Domicile": "居住地2",
        "Remark": "备注",
        "EnableChnAlarm": []
      }
    ]
  }
}
```

17.2.12 Snaped License Plates Search and Match

SearchPlate	
URL	POST /API/AI/SnapedObjects/SearchPlate
Description	It is used to search and match license plates
Request Body	Search license plate JSON (show as follow Table-17.2.12.1)
Successful Response	Search license plate Response JSON (show as follow Table-17.2.12.2)

License plate retrieval is a subtype of Snapped Object. After the search is completed, it shares the API with Snapped Object. Please refer to Chapter 17.2.8 for the following APIs

POST /API/AI/SnapedObjects/GetByIndex

POST /API/AI/SnapedObjects/GetById

POST/API/AI/SnapedObjects/StopSearch

Table-17.2.12.1 (Search license plate JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
StartTime	"2020-07-12 00:00:00"	string	Search start time
EndTime	"2020-07-1223:59:59"	string	Search end time
Chn	0~MAX_PARA_CH N_NUM	array	The channel to search, and the value represents the channel
AlarmGroup	[1, 2]	integer array	The license plate group which the comparison result belongs to in the real-time capturing mode
PlatesId		string array	License Plate list
MaxErrorCharCnt	0-5	int	Fault tolerance. The maximum number of different strings are allowed when searching for matching strings
SortType			Sort Type <pre>enum AIResultSortType { ARST_Ascending = 0, // Ascending order ARST_Descending = 1, // Descending order ARST_None = 2, ARST_Max };</pre>
Engine	0,1	int	Search engine (0 & 1)

Table-17.2.12.2(License plate Response JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result		int	License Plate search result See Table-17.2.9.12

Count	0	int	Actual count of captured objects
-------	---	-----	----------------------------------

Example:

Request message:

POST /API/AI/SnapedObjects/SearchPlate

```
{
  "data": {
    "StartTime": "2021-08-15 00:00:00",
    "EndTime": "2021-08-17 23:59:59",
    "Chn": [2, 3],
    "AlarmGroup": [2, 3, 4],
    "PlatesId": ["3918K"],
    "MaxErrorCharCnt": 3,
    "SortType": 0,
    "Engine": 0
  }
}
```

Response message:

```
{
  "data": {
    "Result": 0,
    "Count": 129
  }
}
```

17.2.13 Snaped Faces and Objects Count Get (VHD)

Get	
URL	POST /API/AI/VhdLogCount/Get
Description	It is used to get VHD log count
Request Body	Get VHD log count JSON (show as follow Table-17.2.13.1)
Successful Response	Get VHD log count Response JSON (show as follow Table-17.2.13.2)

Table-17.2.13.1 (Get VHD log count JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
StartTime	"2020-07-12 00:00:00"	string	Search start time
EndTime	"2020-07-12 23:59:59"	string	Search end time
Chn	0~MAX_PARA_CH	array	The channel to be searched, and the value

	N_NUM		represents the channel (passing an empty array means searching all channels)
Type	[0, 1, 2]	array	Type to search, for example: 0-face, 1-Human, 2-Vehicle, 3-PID Human, 4-PID Vehicle, 5-LCD Human, 6-LCD Vehicle, 7-non-motor vehicle, 8-PID non-motor vehicle, 9-LCD non-motor vehicle, 10-license plate
Engine	0	int	Search Engine, 0 or 1

Table-17.2.13.2(Get VHD log count Response JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result		int	see Table-17. x.x for the search results of the captured objects (including human, cars, etc.)
Count	[0, 0]	int64 array	Actual number of corresponding types of captured objects

Example:

Request message:

POST/API/AI/VhdLogCount/Get HTTP/1.1

```
{
  "data": {
    "MsgId": null,
    "StartTime": "2018-10-20 00:00:00",
    "EndTime": "2018-10-28 23:59:59",
    "Chn": [0, 1, 2, 3, 4, 5, 6, 7, 8],
    "Type": [0, 2],
    "Engine": 0
  }
}
```

Response message:

```
{
  "data": {
    "MsgId": null,
    "Result": 0,
    "Count": [300, 100]
  }
}
```

17.3 Face Attendance(Only NVR)

GET	
URL	POST /API/AI/FDAttendance/Get
Description	It is used to get the Face Attendance Configuration parameters
Request Body	none
Successful Response	Face Attendance Configuration JSON (show as follow Table-17.3.1)

SET	
URL	POST /API/AI/FDAttendance/Set
Description	It is used to set the Face Attendance Configuration parameters
Request Body	Face AttendanceConfiguration JSON (show as follow Table-17.3.1)
Successful Response	The successful result response that described in 2.5

Table-17.3.1 (Face Attendance Configuration JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
fd_atd_info		JSON object	Face Attendance Configuration JSON(show as follow Table-17.3.2)

Table17.3.2(Face Attendance Configuration JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
enable	true false	bool	Enable periodic sending of attendance form
mode	“Day”“Week”“Month”	string	Send Mode
mode_week	“Mon.”“Tue.”“Wed.”“Thu.” “Fri.” “Sat.” “Sun.”	string	A day in the week mode
mode_month_day	“1” “2” “3” “4” “5” ... “26” “27” “28” “29” “30” “31”	string	A day in the Month mode
send_email		string	Sending time
on_duty_time		string	On-duty time
off_duty_time		string	Off-duty time
working_days	"Mon.", "Tue.", "Wed.", "Thu.", "Fri.", "Sat.", "Sun."	array	Working days, corresponding to Monday~Sunday respectively
channel	"CH1", "CH2", "CH3"....."CHx" " The number of channels depends on the capabilities of the device.	array	Effective channel
group	"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15", "16"	array	Effective group

Example:

Request message:

POST /API/AI/FDAttendance/Get HTTP/1.1

```
{  
  "version": "1.0"  
}
```

Response message:

```
{  
  "result": "success",  
  "data": {  
    "fd_atd_info": {  
      "enable": true,  
      "mode": "Week",  
      "mode_week": "Mon.",  
      "mode_month_day": "1",  
      "send_email": "07:30:00",  
      "on_duty_time": "09:30:00",  
      "off_duty_time": "18:30:00",  
      "working_days": [  
        "Mon.",  
        "Tue.",  
        "Wed.",  
        "Thu.",  
        "Fri.",  
        "Sat.",  
        "Sun."  
      ],  
      "channel": [  
        "CH1",  
        "CH2",  
        "CH3",  
        "CH4",  
        "CH5",  
        "CH6",  
        "CH7",  
        "CH8"  
      ],  
      "group": [  
        "1",  
        "2",  
        "4",  
        "5",
```


Example:

Request message:

POST /API/AI/AddedFaces/GetId HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "MsgId": "",
    "GrpId": [2, 3, 7, 8, 9, 10, 11, 12, 13, 16, 17, 18, 19, 20, 21]
  }
}
```

Response message:

```
{
  "data": {
    "MsgId": "",
    "Result": 0,
    "Count": 16,
    "FaceIds": [
      1,
      3,
      4,
      5,
      6,
      7,
      9,
      10,
      11,
      12,
      13,
      14,
      15,
      16,
      23,
      25
    ]
  }
}
```

17.4 Statistics (Only NVR)

17.4.1 Face Search

URL	POST /API/AI/FaceStatistics/Search
Description	It is used to search face statistics
Request Body	See Table-17.4.1.1
Successful Response	Search face statistics JSON (show as follow Table-17.4.1.5)

Get	
URL	POST /API/AI/FaceStatistics/Get
Description	It is used to get face statistics
Request Body	See Table-17.4.1.2
Successful Response	Face statistics JSON (show as follow Table-17.4.1.3)

Table-17.4.1.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Engine	0	int	Which engine was used, 0 or 1, this will match with <Search> interface
StartTime		string	Start time : "2020-07-12 00:00:00"
EndTime		string	End time: "2020-07-12 23:59:59"
Chn	[0, 1, 2, 3, 4, 5, 6, 7, 8]	array	The channel to be searched, and the value represents the channel
Group	[1, 2, 5, 9, 13....]	array	The group to search. The value represents the group ID of the selected group. If it is blank, all (including deleted groups) will be searched

Table-17.4.1.2

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Engine	0	int	Which engine was used, 0 or 1, this will match with <Search> interface
StartIndex	0	int	Request start index, For example 1 st : 0-9999 2 nd : 10000-19999 Then Start Index should be 0, 10000
Count		int	Number of requested face statistics

Table-17.4.1.3(Face statistics JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		

Result	0	int	See Table-17.x.x
channel_max		int	Max Channels
Count		int	Total number of objects in this response
Statistics		array	Face statistics JSON (show as follow Table-17.4.1.4)

Table-17.4.1.4(Face statistics JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Group		int	Group Id to which the face belongs
Time		unsigned long long	Unix time stamp, for example: 1540444116 (The display is uniformly based on UTC time zone, such as: 2020-07-12 00:00:00)
Chn		int	Channel Id to which the face belongs
StrChn	“CH1”...”CH1x” The number of channels depends on the capabilities of the device.	string	Group Id to which the face belongs

Table-17.4.1.5(Search face statistics JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result	0	int	Face statistics Search, See theTable-17.x.x
channel_max		int	Max Channels
Count		int	Total number of actual face statistics

Example:

Request message:

POST /API/AI/FaceStatistics/Search HTTP/1.1

```
{
  "data": {
    "MsgId": null,
    "Engine": 0,
    "StartTime": "2018-10-20 00:00:00",
    "EndTime": "2018-10-28 23:59:59",
    "Chn": [0, 1, 2, 3, 4, 5, 6, 7, 8],
    "Group": [1, 2, 5, 9, 13]
  }
}
```

Response message:

```
{
  "data": {
    "MsgId": null,
    "Result": 0,
    "Count": 160000
  }
}
```

Example:

Request message:

POST /API/AI/FaceStatistics/Get HTTP/1.1

```
{
  "data": {
    "MsgId": null,
    "Engine": 0,
    "StartIndex": 0,
    "Count": 10000
  }
}
```

Response message:

```
{
  "data": {
    "MsgId": null,
    "Result": 0,
    "Count": 10000,
    "Statistics": [
      {
        "Group": 1,
        "Time": 1540444116,
        "Chn": 3,
        "StrChn": "CH4",
      },
      {
        "Group": 3,
        "Time": 1540444119,
        "Chn": 4,
        "StrChn": "CH5",
      },
      ...
    ],
    ...
  }
}
```

```

}
}

```

17.4.2 Hman & Vehicle Search

Get	
URL	POST /API/AI/ObjectStatistics/Get
Description	It is used to get object statistics
Request Body	object statistics JSON (show as follow Table-17.4.2.1)
Successful Response	object statistics JSON (show as follow Table-17.4.2.2)

Table-17.4.2.1(object statistics JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Engine	0	int	Which engine was used, 0 or 1, this will match with <Search> interface
TimePoints	["2020-07-06 00:00:00", "2020-07-07 00:00:00", "2020-07-08 00:00:00", "2020-07-09 00:00:00", "2020-07-10 00:00:00", "2020-07-11 00:00:00", "2020-07-12 00:00:00", "2020-07-13 00:00:00"]	array	The time interval of the search. For example, the example gives the time point of the week of 2020.7.6. When searching, the previous time point is used as the start time, and the next time point is used as the end time (excluding the end time point). The number of time points is unlimited, which is determined by the search criteria. For example, the number of time points searched by day is 25.
Chn	[0, 1, 2, 3, 4, 5, 6, 7, 8...] The number of channels depends on the capabilities of the device.	array	Channel to be searched. The value stands for channel.
Type	[0, 1, 2, 3, 4, 5, 6]	array	Type to search, for example: 0-face, 1-Human, 2-Vehicle, 3-PID Human, 4-PID Vehicle, 5-LCD human, 6-LCD Vehicle

Table-17.4.2.2(object statistics JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result	0	int	Get object statistics result, See Table-17.x.x
ObjectCount		int array	Get object statistics count, each digit indicates the searched counts in the corresponding time interval.

Example:

Request message:

POST/API/AI/ObjectStatistics/Get HTTP/1.1

```
{
  "data": {
    "MsgId": null,
    "Engine": 0,
    "TimePoints":[
      "2020-07-0600:00:00",
      "2020-07-07 00:00:00",
      "2020-07-0800:00:00",
      "2020-07-09 00:00:00",
      "2020-07-1000:00:00",
      "2020-07-11 00:00:00",
      "2020-07-1200:00:00",
      "2020-07-13 00:00:00"
    ]
  }
  "Chn": [0, 1, 2, 3, 4, 5, 6, 7, 8],
  "Type": [1, 2]
}
```

Response message:

```
{
  "data": {
    "MsgId": null,
    "Result": 0,
    "ObjectCount": [3650, 1230, 1980, 1002, 5000, 8900, 8897]
  }
}
```

17.4.3 Cross Counting Statistics

GET	
URL	POST /API/AI/CCStatistics/Get
Description	It is used to getCC statistics
Request Body	See Table -17.4.3.1
Successful Response	Channel Information JSON (show as follow Table-17.4.3.1)

SET	
URL	POST /API/AI/CCStatistics/Set
Description	It is used to set the CCt parameters
Request Body	Channel Information JSON (show as follow Table-17.4.3.2)
Successful Response	Channel Information JSON (show as follow Table-17.4.3.2)

Table -17.4.3.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1

Table -17.4.3.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON array	Single Channel Information JSON show as follow Table-17.4.3.3
channel_max		int	Max channel quantity

Table-17.4.3.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-17.4.3.4 when the URL is a Get request JSON show as follow Table-17.4.3.5 when the URL is a Set request
...		Json Object	
IP_CH1		Json Object	
...		Json Object	

WIFI_CH1		Json Object	
...		Json Object	

Table-17.4.3.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", "Nonsupport"	string	IP Channel status Offline, Online Note: When channel is online, no field.
search_date		string	Search date
detection_type	"Motion" "Person" "Vehicle" "Non-Vehicle"	string	Detection type
cross_type	"Cross In""Cross Out"	string	Coss type
ai_cross_count		bool	Distinguish between the 1st and 2nd generation CC, and the default here is true

Table-17.4.3.5 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", "Nonsupport"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
report		int array	Number of people for CC count

Example:

Request message:

POST /API/AI/CCStatistics/SetHTTP/1.1

```
{
  "data": {
    "channel_info": {
      "CH5": {
        "search_date": "2020-08-26",
        "report_type": "Daily report",
        "detection_type": "Person",
        "cross_type": "Cross In"
      }
    }
  }
}
```

```
}  
}
```

Response message:

```
{  
  "result": "success",  
  "data": {  
    "channel_info": {  
      "CH5": {  
        "report": [  
          18,  
          29,  
          0,  
          0,  
          0,  
          0,  
          0,  
          11,  
          2,  
          4,  
          3,  
          1,  
          0,  
          11,  
          10,  
          1,  
          0,  
          0,  
          0,  
          0,  
          0,  
          0,  
          0,  
          0,  
          0,  
          0  
        ]  
      }  
    }  
  }  
}
```

17.4.4 Heat Map Statistics

GET

URL	POST /API/AI/HeatMapStatistics/Get
Description	It is used to get heat map statistics
Request Body	See Table -17.4.4.1
Successful Response	Channel Information JSON (show as follow Table-17.4.4.1)

SET	
URL	POST /API/AI/HeatMapStatistics/Set
Description	It is used to set the alarm attribute detect parameters
Request Body	Channel Information JSON (show as follow Table-17.4.4.2)
Successful Response	Channel Information JSON (show as follow Table-17.4.4.2)

Table -17.4.4.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1

Table -17.4.4.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON array	Single Channel Information JSON show as follow Table-17.4.4.3
channel_max		int	Max channels

Table-17.4.4.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-17.4.4 when the URL is a Get request JSON show as follow Table-17.4.5 when the URL is a Set request
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.4.4.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", "Nonsupport"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
date		string	Search date
start_hour	0 - 23	int	Start hour, valid when reportType is "Daily report"
end_hour	0 - 23	int	End hour, valid when reportType is "Daily report"
report_type	"Daily report" "Weekly report" "Monthly report" "Annual report"	string	Report tyep
detection_type	"Motion" "face"	string	Detection type
heat_map_type		int	Heatmap type: 1 – Space Heatmap 0 – Time heatmap

Table-17.4.4.5 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
status	"Offline","Online", "Nonsupport"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
report_type	"Daily report" "Weekly report" "Monthly report" "Annual report"	string	Report type
detection_type	"Motion" "face"	string	Detection type
heat_map_type		int	Heatmap type: 1 – Space Heatmap 0 – Time heatmap
time_heat_map	0 - 31	int array	Time heat-map data, Heatmap is valid when type is 0
heat_map_width	0 - 44	int	Space heat map width, heatmap is valid when type is 1
heat_map_height	0 - 36	int	Space heat map height, heatmap is valid when type is 1
space_heat_map	0 - 1583	int array	Space heat map data (unsigned char gray value), heat_map_ Valid when type is 1

background		string	Background image in jpg format of space heat map, heat_map_ Valid when type is 1
------------	--	--------	--

Example:

Request message:

POST /API/AI/HeatMapStatistics/SetHTTP/1.1

```
{
  "data": {
    "channel_info": {
      "CH1": {
        "date": "2020-08-26",
        "start_hour": 0,
        "end_hour": 23,
        "report_type": "Daily report",
        "detection_type": "Motion",
        "heat_map_type": 1
      }
    }
  }
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "CH1": {
        "report_type": "Daily report",
        "detection_type": "Motion",
        "heat_map_type": 1,
        "heat_map_width": 44,
        "heat_map_height": 36,
        "space_heat_map": [89,82,103,31, ... ],
        "background": "base64(imgData)"
      }
    }
  }
}
```

17.5 Snaped face or object (pedestrian, car) real time alarm

(NVR、IPC only, not for A01)

Get	
URL	POST /API/AI/processAlarm/Get
Description	It is used to Get faceor objectalarm real-time appeal
Request Body	none
Successful Response	The face or object alarm real-time appeal JSON (show as follow Table-17.5.1)

Table-17.5.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
FaceInfo		JSON array	FaceInfo JSON show as follow Table-17.5.2
SnapedObjInfo		JSON array	SnapedObjInfoJSON show as follow Table-17.5.3

Table-17.5.2(FaceInfoJSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Id		int	Face Id, unique identifier (nvr only)
GrpId		int	Group Id (nvr only)
SnapId		unsigned int	The Snap Id for captured image for front-end IPC
Type	0	int	Type for Face captured objects
StartTime		unsigned long long	Capturing start time, such as: 13:08:36, 2018-10-25, Unix time stamp (displayed uniformly according to UTC time zone)
EndTime		unsigned long long	Capturing end time, such as: 13:08:36, 2018-10-25, Unix time stamp (displayed uniformly according to UTC time zone)
Similarity		float	Similarity such as: 93.56932(nvr only)
Score		int	Image score (confidence) (nvr only)
Sex		int	Gender: 0- male 1- female
Age		int	Age
Gender		int	Gender for Face attribute: 0- male 1- female
fAttrAge		int	Age for face attribute
Beauty		int	Beauty
GlassesType		int	Glasses Type 0: no glasses 1: wear glasses (at present, there is no distinction between sunglasses and ordinary glasses, Default

			is all glasses)
Expression		int	Emotion type: 0: no expression 1: Smile 2: Laugh
MouthMask		int	Mask type 0: no mask 1: wear mask
Race		int	Race type: 0: Yellow, 1: White, 2: Black, 3: Arab
Chn		int	Channel
StrChn	<p>“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x”</p> <p>The number of channels depends on the capabilities of the device.</p>	string	Channel
ModifyCnt		int	The number of modifications, the upper layer is read-only, and the device side updates the face information in real time each time it modifies it (nvr only)
Image1	"base64(imgData)"	string	Imported face image
Image2	"base64(imgData)"	string	Captured face pictures
Image3	"base64(imgData)"	string	Captured Human pictures
Image4	"base64(imgData)"	string	Captured background pictures
Feature	"base64(feature)"	string	eigenvalue (nvr only)
Name	"Mike"	string	Name (nvr only)
Country	"China"	string	Country (nvr only)
Nation	"Han"	string	Nationality (nvr only)
NativePlace	"Guangdong,Zhuhai"	string	Native Place (nvr only)
IdCode	"415025199203050916"	string	Id code (nvr only)
Job	"Software"	string	Job (nvr only)
Phone	"12345678902"	string	Phone (nvr only)
Email	"abcd@163.com"	string	Email (nvr only)
Domicile	"Guangdong,Zhuhai, Xiangzhou ..."	string	residence (nvr only)

Remark	"Detail of this person ..."	string	Remark (nvr only)
--------	-----------------------------	--------	-------------------

Table-17.5.2(SnapedObjInfo JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
chn		string	Channel
StrChn	"CH1"... "CH1x" "IP_CH1"... IP_CH1x" "WIFI_CH1"... WIFI_CH1x" The number of channels depends on the capabilities of the device.	string	Channel
GrpId		int	Group Id
StartTime		unsigned long long	Capturing start time, such as: 13:08:36, 2018-10-25, Unix time stamp (displayed uniformly according to UTC time zone)
EndTime		unsigned long long	Capturing end time, such as: 13:08:36, 2018-10-25, Unix time stamp (displayed uniformly according to UTC time zone)
ObjectImage	"base64(imgData)"	string	Imported face image
Background	"base64(imgData)"	string	Captured face pictures
SnapId		unsigned int	Snap Id of capturing image for front-end IPC
Type		int	Captured object Type 0- face 1- Human 2- Vehicle 3- PID human 4- PID vehicle 5- LCD human 6- LCD vehicle

Example:

Response message:

```

{
  "data": {
    "FaceInfo": [
      {
        "Id": 1,
        "GrpId": 1,

```

```

    "SnapId": 1250,
    "StartTime": 1540444116,
    "EndTime": 1540444126,
    "Similarity": 93.56932,
    "Score": 87,
    "Gender": 0,
    "fAttrAge": 26,
    "Chn": "CH1",
    "ModifyCnt": 0,
    "Image1": "base64(imgData)",
    "Image2": "base64(imgData)",
    "Image3": "base64(imgData)",
    "Image4": "base64(imgData)",
    "Feature": "base64(feature)",
    "Name": "Mike",
    "Country": "China",
    "Nation": "Han",
    "NativePlace": "Guangdong,Zhuhai",
    "IdCode": "415025199203050916",
    "Job": "Software",
    "Phone": "12345678902",
    "Email": "abcd@163.com",
    "Domicile": "Guangdong,Zhuhai,Xiangzhou ...",
    "Remark": "Detail of this person ..."
  },
  "SnapedObjInfo": [
    {
      "Chn": "CH1",
      "StartTime": 1540444116,
      "EndTime": 1540444137,
      "ObjectImage": "base64(imgData)",
      "Background": "base64(imgData)",
      "SnapId": 2375,
      "Type": 1
    }
  ]
}

```

17.6 Attribute Detection

GET

URL	POST/API/AI/Alarm/AttributeDetect/Get
Description	It is used to get the alarm attribute detect config parameters
Request Body	See Table -17.6.1
Successful Response	Channel Information JSON (show as follow Table-17.6.2)

SET	
URL	POST /API/AI/Alarm/AttributeDetect/Set
Description	It is used to set the alarm attribute detect parameters
Request Body	Channel Information JSON (show as follow Table-17.6.2)
Successful Response	The successful result response that described in 2.5

Table -17.6.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1

Table -17.6.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_info		JSON array	Single Channel Information JSON show as follow Table-17.6.3
channel_max		int	Max channels

Table-17.6.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CH1		Json Object	JSON show as follow Table-17.16.4
...		Json Object	
IP_CH1		Json Object	
...		Json Object	
WIFI_CH1		Json Object	
...		Json Object	

Table-17.6.4 (Channel Information JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
status	"Offline","Online", "Nonsupport"	string	Channel online status, only for digital channels. Note: This field is not available when the channel is online
alarm_type	"Close" "NO Mask" "Wear Mask"	string	Alarm type: 0 – Close 1 – No Mask 2 – Wear Mask
alarm_out	“Local->1”... ”Local->x” “IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2” The number of channels depends on the capabilities of the device.	array	Alarm out channel Each array bit represents a alarm output channel with a string. Camera: Local->1:Enable, empty, disable
latch_time	"10","20","40", "60"	string	Latch time for IPC alarm out Value: 5 10 20 30
record_enable	true, false	bool	Record channel switch
record_channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...”	array	Record channel Channel alarm linkage switch.
post_recording	"30","60","120", "300"	string	Post recording time Ipc value: 0 5 10 20 30
send_email	true, false	bool	Send Email switch
full_screen		bool	FullScreen switch (NVR only)
buzzer	"0","10","20","40", "60"	string	Buzzer time (NVR only)
show_message		bool	Show Message switch (NVR only)

Example:

Request message:

POST /API/AI/Setup/CrossCount/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "channel": ["CH5"]
  }
}
```

```
}  
}
```

Response message:

```
{  
  "result": "success",  
  "data": {  
    "channel_info": {  
      "CH4": {  
        "alarm_type": "Wear Mask",  
        "buzzer": "10",  
        "alarm_out": [  
          "Local->1",  
          "IP_CH1->1",  
          "IP_CH4->1",  
          "IP_CH5->1"  
        ],  
        "latch_time": "10",  
        "record_enable": true,  
        "record_channel": [  
          "CH1",  
          "CH2",  
          "CH3",  
          "CH4",  
          "CH5",  
          "CH6",  
          "CH7",  
          "CH8"  
        ],  
        "post_recording": "60",  
        "show_message": true,  
        "send_email": true,  
        "full_screen": true,  
      }  
    }  
  }  
}
```

17.7 Repeat Customer

GET	
URL	POST/API/AI/SnapedFeaturesId/Get
Description	It is used to get the SnapedFeaturesId parameters
Request Body	See Table -17.7.1
Successful Response	SnapedFeatures Information JSON (show as follow Table-17.7.2)

GET	
URL	POST /API/AI/FilterSnapedFaces/Get
Description	It is used to get the FilterSnapedFaces parameters
Request Body	See Table -17.7.3
Successful Response	FilterSnapedFaces Information JSON (show as follow Table-17.7.6)

GET	
URL	POST /API/AI/MatchAddedFaces/Get
Description	It is used to get the MatchAddedFaces parameters
Request Body	See Table -17.7.7
Successful Response	MatchAddedFaces Information JSON (show as follow Table-17.7.8)

Table -17.7.1

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Engine	0	int	Search Engine, 0 or 1 Please be consistent with below API /API/AI/SnapedFaces/Search
StartIndex	0	int	Requerst start index, For example 1 st : 0-9999 2 nd : 10000-19999 Then Start Index should be 0, 10000
Count	1000	int	Number of requested capturing If data is small, you can request at one time

Table-17.7.2 (SnapedFeatures Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result	0	int	Get the results of all eligible Ids and feature Ids. 0 means success. See Table-17. x.x for details
UUIds	[200053, 200054, ...]	unsigned int array	The elements in the array represent the unique identification of the captured face
FtIds	[58, 59, ...]	unsigned int	The element in the array represents the

		array	eigenvalue Id of the captured face, which corresponds to the Id in "UIIds" one by one
--	--	-------	---

Table -17.7.3

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Engine	0	int	Search Engine, 0 or 1 Please be consistent with below API /API/AI/SnapedFaces/Search
MinInterval	5	int	The time interval of face appearance, in seconds. If it is less than this value, it should not be included in the statistics. The "bValid" field in the response results indicates whether it should be included in the statistics
Similarity	70	unsigned int	Similarity used in filtering
Filter		Json Object	The target face to be filtered. JSON show as follow Table-17.7.4
FtIdSet		Json Object	The collection that needs to be filtered. JSON show as follow Table-17.7.5

Table-17.7.4

KEY	VALUE		COMMENT
	RANGE	TYPE	
UUIId	200053	unsigned int	Unique identification for capturing faces, corresponding to UUIId in/API/AI/SnapedFeaturesId/Get
FtId	58	unsigned int	The eigenvalue Id of the captured face, corresponds to the FtId in/API/AI/SnapedFeaturesId/Get

Table-17.7.5

KEY	VALUE		COMMENT
	RANGE	TYPE	
UIIds	[200053, 200054, ...]	unsigned int array	The elements in the array represent the unique identification of the captured face
FtIds	[58, 59, ...]	unsigned int array	The element in the array represents the eigenvalue Id of the captured face, which corresponds to the Id in "UIIds" one by one

Table-17.7.6 (FilterSnapedFaces Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

MsgId	null		
Result	0	int	Returns the result of the face Id matching the filter in the target set. 0 indicates success. See Table-17. x.x for details
Count	6	int	Number of filtered faces
MatchedIds	[200053, 200059, ...]	unsigned int array	The elements in the array represent the unique identification of the captured face
Counting	[1, 0, 0, 1, 1, ...]	int array	The element in the array indicates whether the corresponding face can be used for counting statistics. If the occurrence interval is less than the minimum interval, the value is 0, which should not be included in the statistics. Otherwise, it is 1, which should be included in the statistics

Table -17.7.7

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Engine	0	int	Search Engine, 0 or 1 Please be consistent with below API /API/AI/SnapedFaces/Search
Similarity	70	int	Similarity you used during the comparison
GrpIds	[1, 2, 3, 5]	int array	Which groups of faces in the base database are used to identify the identity. If the value is not given or empty, it means that all groups (excluding Internal) are used. Generally, the value is not given
UIIds	[200053, 200059, 210010, ...]	unsigned int array	The elements in the array represent the unique identification in the captured face

Table-17.7.8 (MatchAddedFaces Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
MsgId	null		
Result	0	int	Get the result of a repeat customer Id 0 - success. Details, please see Table-17. x.x
FaceIds	[3, -1, -1, 2, ...]	long longarray	The matched face in the base database. The elements in this array correspond to the requested "UIIds" array one by one. "-1" - no matched face in the base database

Example:

Request message:

POST /API/AI/SnapedFeaturesId/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "MsgId": "",
    "StartIndex": 0,
    "Engine": 1,
    "Count": 1011
  }
}
```

Response message:

```
{
  "data": {
    "MsgId": null,
    "Result": 0,
    "UIIds": [200053, 200054, ...],
    "FtIds": [58, 59, ...]
  }
}
```

Request message:

POST /API/AI/FilterSnapedFaces/Get HTTP/1.1

```
{
  "version": "1.0",
  "data": {
    "MsgId": "",
    "Engine": 0,
    "MinInterval": 0,
    "Similarity": 70,
    "Filter": {
      "UIId": 20402,
      "FtId": 20402
    },
    "FtIdSet": {
      "UIIds": [20402, 20403, 20408, 20404, 20405, ...],
      "FtIds": [20402, 20403, 20408, 20404, 20405, ...]
    }
  }
}
```

```
}  
}
```

Response message:

```
{  
  "data": {  
    "MsgId": "",  
    "Result": 0,  
    "Count": 189,  
    "MatchedIds": [20402, 20404, 20406, 20410, 20412,...],  
    "Counting": [1, 1, 1, 1, 1, 1, 1, 1,...]  
  }  
}
```

Request message:

POST /API/AI/MatchAddedFaces/Get HTTP/1.1

```
{  
  "version": "1.0",  
  "data": {  
    "MsgId": "",  
    "Similarity": 70,  
    "GrpIds": [],  
    "Engine": 1,  
    "UUIIds": [21411, 21409, 21408, 21407, 21405, ...]  
  }  
}
```

Response message:

```
{  
  "data": {  
    "MsgId": "",  
    "Result": 0,  
    "FaceIds": [25, 25, 25, -1, 25, ...]  
  }  
}
```

17.8 Cross Counting Scenario

17.8.1 RealTime Info

GET	
URL	POST/API/AI/Scenario/CC/RealTime/Get
Description	It is used to get the CrossCountingApp RealTimeInfo
Request Body	See Table -17.8.1.1
Successful Response	Cross Counting RealTime Information JSON (show as follow Table-17.8.1.2)

SET	
URL	POST/API/AI/Scenario/CC/RealTime/Set
Description	It is used to clear the CrossCountingApp RealTimeInfo
Request Body	See Table -17.8.1.1
Successful Response	Cross Counting RealTime Information JSON (show as follow Table-17.8.1.2)

Table-17.8.1.1 (Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
msgType	"get_CCScenario_RTData" "clear_CCScenario_RTData"	string	Get or Clear CC RealTimeInfo
clear_type	"Channel","Group", "All Channels","All Groups"	string	Clear_ type of CC RealTimeInfo
chnId		int	When <clear_type> select Channel, need to request channel number
groupId		int	When <clear_type> select Group, need to request Group Id.

Table-17.8.1.2 (Cross Counting RealTime Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
CCScenarioRTInfo	"Channels", "Groups"	Object array	Cross Counting RealTime Information JSON show as follow Table-17.8.1.3

Table-17.8.1.3 (Cross Counting RealTime Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Channels		Object array	JSON show as follow Table-17.8.1.4
Groups		Object array	JSON show as follow Table-17.8.1.5

Table-17.8.1.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

ChnId		int	Channel Id
EnterCnt		int	Number of entrants
ExitCnt		int	Number of departures
StayCnt		int	Number of stays
AvailableCnt		int	Available capacity
ObjType		int	Triggered count Type: 0: Motion 1: Persion 2: Vehicle

Table-17.8.1.5 (Group Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
GroupId		int	Group Id
EnterCnt		int	Number of entrants
ExitCnt		int	Number of departures
StayCnt		int	Number of stays
AvailableCnt		int	Available capacity
ObjType		int	Triggered count Mode: 0: Motion 1: Persion 2: Vehicle
ChnDetail		Object array	Live channel data in the group JSON show as follow Table-17.8.1.6

Table-17.8.1.6 (Chn Detail JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ChnId		int	Channel Id
EnterCnt		int	Number of entrants
ExitCnt		int	Number of departures

Example:

Request message:

POST /API/AI/Scenario/CC/RealTime/Get

```
{
  "version": "1.0",
  "data": {
    "msgType": "get_CCScenario_RTData"
  }
}
```

POST /API/AI/Scenario/CC/RealTime/Set

```
{
```

```

"version": "1.0",
"data": {
  "msgType": "clear_CCScenario_RTData",
  "clear_type": "Group",
  "groupId": 1
}
}

```

Response message:

```

{
  "result": "success",
  "data": {
    "CCScenarioRTInfo": [
      {
        "Channels": [
          {
            "ChnId": 1,
            "EnterCnt": 0,
            "ExitCnt": 0,
            "StayCnt": 0,
            "AvailableCnt": 10,
            "ObjType": 0
          }
        ],
        "Groups": [
          {
            "GroupId": 0,
            "EnterCnt": 0,
            "ExitCnt": 0,
            "StayCnt": 0,
            "AvailableCnt": 10,
            "ObjType": 1,
            "ChnDetail": [
              {
                "ChnId": 0,
                "EnterCnt": 0,
                "ExitCnt": 0
              }
            ]
          }
        ]
      }
    ]
  }
}

```

```

    ]
  }
}

```

17.8.2 Map

GET	
URL	POST /API/AI/Scenario/CC/MapConfig/Get
Description	It is used to get the Cross Counting Scenario Map
Request Body	See Table -17.8.2.1
Successful Response	Map Information JSON (show as follow Table-17.8.2.2)

SET	
URL	POST /API/AI/Scenario/CC/MapConfig/Set
Description	It is used to set the Cross Counting Scenario Map
Request Body	Map Information JSON (show as follow Table-17.8.2.2)
Successful Response	The successful result response that described in 2.5

Table-17.8.2.1 (Map Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
GroupId		int	Group Id of Map you need to search

Table-17.8.2.2 (Map Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
GroupId		int	Group Id
RefWidth	0~1920	int	Ref Width for coordinates
RefHeight	0~1080	int	Ref Height for coordinate
CamPos		Object array	Cameras' positions JSON show as follow Table-17.8.2.3
MapImage	"base64(imgData)" 0~5 * 1024 * 1024	string	Map Image, only support png/jpg/bmp format fort the Map image.

Table-17.8.2.3 (CamPos Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
ChnId		int	Channel Id

XPos		int	X-axis coordinate in the map
YPos		int	Y-axis coordinate in the map

Example:

Request message:

POST/API/AI/Scenario/CC/MapConfig/Get

```
{
  "data": {
    "GroupId": 0
  }
}
```

Response message:

```
{
  "data": {
    "RefWidth": 1920,
    "RefHeight": 1080,
    "CamPos": [
      {
        "ChnId": 0,
        "XPos": 362,
        "YPos": 936
      },
      {
        "ChnId": 3,
        "XPos": 1830,
        "YPos": 289
      },
      .....
    ]
    "MapImage": base64
  }
}
```

Request message:

POST/API/AI/Scenario/CC/MapConfig/Set

```
{
  "data": {
    "GroupId": 0,
    "RefWidth": 1920,
    "RefHeight": 1080,
    "CamPos": [
      {
```

```

        "ChnId": 0,
        "XPos": 100,
        "YPos": 100
    },
    {
        "ChnId": 2,
        "XPos": 100,
        "YPos": 100
    }
]
"MapImage": base64 (optional)
}
}

```

Response message:

```

{
    "result": "success",
    "data": {}
}

```

17.8.3 Statistics

GET	
URL	POST /API/AI/Scenario/CC/Statistics/Get
Description	It is used to get CC Scenario Statistics
Request Body	See Table -17.8.3.1
Successful Response	Response Information JSON (show as follow Table-17.8.3.2)

SET	
URL	POST /API/AI/Scenario/CC/Statistics/Set
Description	It is used to set CC Scenario Statistics
Request Body	Request Information JSON (show as follow Table-17.8.3.3)
Successful Response	Response Information JSON (show as follow Table-17.8.3.4)

Table-17.8.3.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	“CH1”...”CH1x” “IP_CH1”...” IP_CH1x” “WIFI_CH1”...” WIFI_CH1x” The number of channels depends on the capabilities of	String array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1

	the device.		
--	-------------	--	--

Table-17.8.3.2 (Response Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Channels		Int array	Allowed to search Channels
Groups		Int array	Allowed to search Groups
ReportType	"Day","Week", "Month","Year"	string	Allowed to search ReportType
DetectionType	"Motion","Person", "Vehicle"	string	Detection Type
Date	10	string	Current Date

Table-17.8.3.3 (Request Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Channels		Int array	Request Channels, Response “Empty” when requesting Groups
Groups		Int array	Request Groups, Response “Empty” when requesting Channels
Date	10	string	Request Date
ReportType	"Day","Week", "Month","Year"	string	Request Report Type
ChnObjType	0~2	Int array	Detect Type (Only response “empty” when searching Group, matched with Channels) 0: "Motion", 1: "Person", 2: "Vehicle"
GrpObjType	0~2	Int array	Detect Type (Only response “empty” when searching Channel, matched with Groups) 0: "Motion", 1: "Person", 2: "Vehicle"

Table-17.8.3.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Channels		Object array	Channels Statistics Info JSON show as follow Table-17.8.3.5
Groups		Object array	Group Statistics Info JSON show as follow Table-17.8.3.6

Table-17.8.3.5 (Channels Information JSON)

KEY	VALUE	COMMENT
-----	-------	---------

	RANGE	TYPE	
ChnId		int	Channel Number
Num		Object array	Channel Statistics data JSON show as follow Table-17.8.3.7

Table-17.8.3.6 (Groups Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
GroupId		int	Group Id
Num		Object array	Group Statistics data JSON show as follow Table-17.8.3.7

Table-17.8.3.7 (Num Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
EnterCnt		int	Number of entrants
ExitCnt		int	Number of departures
StayCnt		int	Number of Stays

Example:

Request message:

POST /API/AI/Scenario/CC/Statistics/Set

```
{
  "version": "1.0",
  "data": {
    "Channels" : [0,3],
    "Groups": [],
    "Date" : "2021-01-14",
    "ReportType" : "Week",
    "ChnObjType" : [1,1],
    "GrpObjType" : []
  }
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "Channels": [
      {
        "ChnId": 0,
        "Num": [
          {
            "EnterCnt": 0,
            "ExitCnt": 0,

```

```

        "StayCnt": 0
    },
    {
        "EnterCnt": 0,
        "ExitCnt": 0,
        "StayCnt": 0
    },
    {
        "EnterCnt": 0,
        "ExitCnt": 0,
        "StayCnt": 0
    },
    {
        "EnterCnt": 11,
        "ExitCnt": 3,
        "StayCnt": 8
    },
    {
        "EnterCnt": 16,
        "ExitCnt": 2,
        "StayCnt": 14
    },
    {
        "EnterCnt": 0,
        "ExitCnt": 0,
        "StayCnt": 0
    },
    {
        "EnterCnt": 0,
        "ExitCnt": 0,
        "StayCnt": 0
    }
]
},
{
    "ChnId": 3,
    "Num": [
        {
            "EnterCnt": 0,
            "ExitCnt": 0,
            "StayCnt": 0
        },
        {
            "EnterCnt": 0,

```



```

        "ExitCnt": 0,
        "StayCnt": 0
    },
    {
        "EnterCnt": 0,
        "ExitCnt": 0,
        "StayCnt": 0
    },
    {
        "EnterCnt": 0,
        "ExitCnt": 0,
        "StayCnt": 0
    },
    {
        "EnterCnt": 1,
        "ExitCnt": 0,
        "StayCnt": 1
    },
    {
        "EnterCnt": 0,
        "ExitCnt": 0,
        "StayCnt": 0
    },
    {
        "EnterCnt": 0,
        "ExitCnt": 0,
        "StayCnt": 0
    }
    ]
}

```

17.8.4 Config

GET	
URL	POST /API/AI/Scenario/CC/Config/Get
Description	It is used to get the CC Scenario config parameters
Request Body	See Table -17.8.4.1

Successful Response	Config Information JSON (show as follow Table-17.1.4.2)
---------------------	---

SET	
URL	POST /API/AI/Scenario/CC/Config/Set
Description	It is used to set the CC Scenario parameters
Request Body	Channel Information JSON (show as follow Table-17.1.4.2)
Successful Response	The successful result response that described in 2.5

Table-17.8.4.1 (ChannelRequest JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel	"CH1"...CH1x" "IP_CH1"...IP_CH1x" "WIFI_CH1"...WIFI_CH1x" The number of channels depends on the capabilities of the device.	string array	Each array bit represents a channel with a string. DVR/NVR need; IPC only use CH1

Table-17.8.4.2 (Config Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
adSwitch		bool	Ad mode Switch
ad_displayMode		bool	Display Mode Switch
ad_seqTime		int	Ad swquence time
channel_info		JSON object	Channel Information JSON show as follow Table-17.8.4.3
group_info		JSON object	Group Information JSON show as follow Table-17.8.4.4

Table-17.8.4.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
channel_switch		bool	Control whether CC scenario function is enabled
channel_group		int	Which group the channel belong to. "-1" stands for "the channel(s) is not in the group"
channel_capacity		int	Capacity in the channel
chn_set_enable		bool	Channel configurable, it's linked to the 2 nd generation CC function.
chn_buzzer	"0","10","20","40","60"	string	Buzzer time
chn_alarm_out	"Local->1"... "Local->x"	array	Alarm output channel

	<p>“IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2”</p> <p>The number of channels depends on the capabilities of the device.</p>		<p>Each array bit represents aalarm output channel with a string. Camera: Local->1: Enable, Empty: Disable</p>
chn_latch_time	"10","20","40", "60"	string	Latch time for Alarm output

Table-17.8.4.4 (Group Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
group_switch		bool	Control whether CC scenario function is enabled
group_capacity		int	Capacity in one group
start_time	8	string	Start time
end_time	8	string	End time
alarm_type	"Unuse","Person","Vehicle"	string	Alarm Type: 1: Person, 2: Vehicle
grp_buzzer	"0","10","20","40", "60"	string	Buzzer Time
grp_alarm_out	<p>“Local->1”... ”Local->x” “IP_CH1->1”... “IP_CH1->2”... ” IP_CHx->1” ” IP_CHx->2”</p> <p>The number of channels depends on the capabilities of the device.</p>	array	<p>Alarm output Channel</p> <p>Each array bit represents aalarm output channel with a string. Camera: Local->1: Enable, Empty: Disable</p>
grp_latch_time	"10","20","40", "60"	string	Latch time for Alarm output

Example:

Request message:

POST /API/AI/Scenario/CC/Config/Set

{

```

"version": "1.0",
"data": {
  "adSwitch": false,
  "ad_displayMode": false,
  "ad_seqTime": 1,
  "channel_info": {
    "CH1": {
      "channel_switch": false,
      "channel_group": -1,
      "channel_capacity": 11,
      "chn_set_enable": false,
      "chn_buzzer": "0",
      "chn_alarm_out": [],
      "chn_latch_time": "20"
    },
    "CH8": {
      "channel_switch": false,
      "channel_group": 4,
      "channel_capacity": 33,
      "chn_set_enable": false,
      "chn_buzzer": "0",
      "chn_alarm_out": [],
      "chn_latch_time": "10"
    }
  },
  "group_info": {
    "Group1": {
      "group_switch": false,
      "group_capacity": 30,
      "start_time": "00:00:00",
      "end_time": "23:59:59",
      "alarm_type": "Person",
      "grp_buzzer": "0",
      "grp_alarm_out": [],
      "grp_latch_time": "10"
    },
    "Group8": {
      "group_switch": false,
      "group_capacity": 10,
      "start_time": "00:00:00",
      "end_time": "23:59:29",
      "alarm_type": "Person",
      "grp_buzzer": "0",
      "grp_alarm_out": [],

```

```

        "grp_latch_time": "10"
    }
}
}

```

Response message:

```

{
  "result": "success",
  "data": {}
}

```

17.8.5 Image Manage

SET	
URL	POST /API/AI/Scenario/CC/Config/ImageManage
Description	It is used to set the CC Scenario Image
Request Body	Request Information JSON (show as follow Table-17.8.5.1)
Successful Response	Response Information JSON (show as follow Table-17.8.5.2)

Table -17.8.5.1 (Request Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
operate	"getImageList", "getImageData", "AddImage", "DeleteImage"	string	Operation Type: Get Image list Get Image data (Get one image every time) Add Image Delete Image
image_name		string	Image name you added or got
image_data	"base64(imgData)" 2 * 1024 *1024	string	Image data you added and support JPG, PNG and BMP format.
image_list		array	Delete image list

Table -17.8.5.2 (Response Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
max_count		int	Maximum image counts supported
image_name		string	Image name, Get image name after reponse successfully
image_data	"base64(imgData)"	string	Get iamge data
image_list		array	Get image list

Example:

Request message:

POST /API/AI/Scenario/CC/Config/ImageManage

```
{
  "version": "1.0",
  "data": {"operate": "GetImageList"}
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "max_count": 16,
    "image_list": [
      "c278.png",
      "c236.png",
      "c263.png"
    ]
  }
}
```

Request message:

POST /API/AI/Scenario/CC/Config/ImageManage

```
{
  "result": "success",
  "data": {
    "operate": "AddImage",
    "image_name": "eeeeee.jpg",
    "image_data": " base64(imgData)"
  }
}
```

Response message:

```
{
  "result": "success",
  "data": {}
}
```

Request message:

POST /API/AI/Scenario/CC/Config/ImageManage

```
{
  "version": "1.0",
  "data": {
    "operate": "DeleteImage",
    "image_list": [
      "c278.png",
      "c236.png",

```

```

        "c263.png"
    ]
}

```

Response message:

```

{
  "result": "success",
  "data": {
    "c278.png": "Delete success!",
    "c236.png": "Delete success!",
    "c263.png": "Delete success!"
  }
}

```

Request message:

POST /API/AI/Scenario/CC/Config/ImageManage

```

{
  "version": "1.0",
  "data": {
    "operate": "GetImageData",
    "image_name": "c70.png"
  }
}

```

Response message:

```

{
  "result": "success",
  "data": {
    "image_name": "c70.png",
    "image_data": "base64(imgData)"
  }
}

```

17.x AI Operation Result Type(NVR-Specific)

Table-17.x.x(AI_OPT_RESULT_TYPE)

Result Type	COMMENT
AORT_SUCCESS = 0	Success
AORT_NO_DB = -1	No database
AORT_DB_EXEC_FAILED = -2	Database execute failed
AORT_CALC_FEATURE_FAILED = -3	Failed to extract feature
AORT_CANCELED = -4	Canceled
AORT_NO_DISK = -5	No disk
AORT_DISK_ERROR = -6	Disk error
AORT_EXIST = -7	Existed
AORT_GROUP_INVALID = -8	Group is invalid

AORT_NOT_EXIST = -9	Not Exist
AORT_MORE_FILE_EXIST = -10	File has already existed
AORT_SEARCH_ERROR = -11	Searching error
AORT_OVER_MAX_COUNT = -12	Over the max count
AORT_UPDATING_FEATURE = -13	Updating the feature
AORT_NO_USABLE_IPC = -14	No usable an IPC of calculating the feature count
AORT_INVALID_PARAM = -15	Invalid Parameter
AORT_INVALID_FORMAT = -16	Invalid Format
AORT_INVALID_RES = -17	Invalid resolution
AORT_INVALID_MEM = -18	Document memory is over
AORT_CREAT_FAILED = -19	Failed to create
AORT_MD5_NOT_MATCH = -20	Not match with MD5
AORT_POS_ERROR = -21	Position Error
AORT_SIZE_ERROR = -22	Size Error
AORT_NOT_READY = -23	Not ready
AORT_INVALID_DB = -24	Invalid Database

18 Extended Functionality

18.1 IPCVoice Prompt

18.1.1 Description

GET	
URL	POST /API/Extended/IPCVoicePrompts/Get
Description	It is used to get the IPC voice prompt config parameters
Request Body	See Table -18.1.2.1
Successful Response	Parameter Information JSON (show as follow Table-18.1.2.2)

SET	
URL	POST /API/Extended/IPCVoicePrompts/Set
Description	It is used to operate the voice prompt function of IPC
Request Body	See Table -18.1.2.1
Successful Response	The successful result response that described in 2.5

18.1.2 Syntax

Table-18.1.2.1 (Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
command	"GetAudioFilesList", "GetAudioIndex", "ConfigureAudioIndex", "PlayAudio", "StopPlaying"	string	Operating Command: Get Audio File List、 Get Audio Index from IP Channel、 Configure Audio index from IP Channel、 Play Audio from IPC Stop Playing audio
channel	"IP_CH1"... "IP_CHx" The number of channels depends on the capabilities of the device.	string array	Be used for the specific IP channel (PlayAudio, StopPlaying)
channel_info		JSON object	Channel Information JSON show as follow Table-18.1.2.3 (ConfigureAudioIndex)

Table-18.1.2.2 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
audio_list		string array	Audio List
channel_info		JSON object	Channel Information JSON show as follow Table- 18.1.2.3

Table-18.1.2.3 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
IP_CH1		Json Object	JSON show as follow Table-18.1.2.4
...		Json Object	
IP_CHx		Json Object	

Table-18.1.2.4 (Channel Information JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
state	"Offline", "Online"	string	The state of online IP channel
audio_index		int	Audio document index

18.1.3 Example

18.1.3.1 RequestGet Audio Files List

Request message:

```
POST/API/Extended/IPCVoicePrompts/Ge
{
  "data": {
    "command": "GetAudioFilesList "
  }
}
```

Response message:

HTTP/1.1 200 OK

```
{
  "result": "success",
  "data": {
    "audio_list": [
      "5_13227.mp3", //Each items of this list was separated to two parts by “_”, the front part is the only
one number that was distributed to the audio file by system.
      "6_9528.mp3",
      "7_AlienBoi.mp3"
    ]
  }
}
```

18.1.3.2 RequestGet Audio Index List

POST API/Extended/IPCVoicePrompts/Get

```
{
  "data": {
    "command": "GetAudioIndex"
  }
}
```

Response message:

```
{
  "result": "success",
  "data": {
    "channel_info": {
      "IP_CH1": {
        "status": "Online",
        "audio_index": 2
      },

```

```

        "IP_CH2": {
            "status": "Online",
            "audio_index": 3
        }
    }
}

```

18.1.3.3 Request Configure Audio Index

POST API/Extended/IPCVoicePrompts/Set

```

{
  "data": {
    "command": "ConfigureAudioIndex",
    "channel_info": {
      "IP_CH1": {
        "audio_index": 2
      },
      "IP_CH2": {
        "audio_index": 3
      }
    }
  }
}

```

Response message:

```

{
  "result": "success",
  "data": {
    "channel_info": {
      "IP_CH1": {
        "status": "Online",
        "audio_index": 2
      },
      "IP_CH2": {
        "status": "Online",
        "audio_index": 3
      }
    }
  }
}

```

18.1.3.4 Request Player Stop Audio

POST API/Extended/IPCVoicePrompts/Set

```
{
  "data": {
    "command": "PlayAudio", // "StopPlaying"
    "channel": [
      "CH1", "CH2"
    ]
  }
}
```

Response message:

```
{
  "result": "success",
  "data": {}
}
```

19 Push

19.1.1 Description

GET	
URL	POST /API/Push/GetToken
Description	Rspush get AccessToken
Request Body	See Table -19.1.1.1
Successful Response	Parameter Information JSON (show as follow Table -19.1.1.2)

GET	
URL	POST /API/Push/Query
Description	Rspush Query rspush parameter
Request Body	See Table -19.1.1.3
Successful Response	Parameter Information JSON (show as follow Table -19.1.1.4)

SET	
URL	POST /API/Push/Subscribe
Description	Rspush Subscribe
Request Body	See Table -19.1.1.10
Successful Response	The successful result response that described in 2.5

SET	
URL	POST /API/Push/Unsubscribe
Description	Rspush Unsubscribe
Request Body	See Table -19.1.1.1
Successful Response	The successful result response that described in 2.5

GET	
URL	POST /API/Push/QueryDefault
Description	Rspush QueryDefault
Request Body	None
Successful Response	Parameter Information JSON (show as follow Table -19.1.1.12)

GET	
URL	POST /API/PushSubscribe/Get
Description	Tutkpush Subscribe Parameter Get
Request Body	None
Successful Response	Parameter Information JSON (show as follow Table -19.1.1.13)

SET	
URL	POST /API/PushSubscribe/Set
Description	Tutkpush Subscribe Parameter Set
Request Body	Parameter Information JSON (show as follow Table -19.1.1.13)
Successful Response	The successful result response that described in 2.5

19.1.2 Syntax

Table -19.1.1.1 (Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Token		string	

Table -19.1.1.2

KEY	VALUE		COMMENT
	RANGE	TYPE	
AccessToken		string	

Table -19.1.1.3 (Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Token		string	Optional Field, The request returns only the default parameter without this field.

			The default parameter with this field is returned with the parameter
--	--	--	--

Table -19.1.1.4

KEY	VALUE		COMMENT
	RANGE	TYPE	
Default		JSON object	Default, Please refer to Table -19.1.1.5
Filter		JSON object	Subscribe parameter, Please refer to Table -19.1.1.8

Table -19.1.1.5 (Default JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
StorageError	"True"、"False"	string	
StorageFull	"True"、"False"	string	
StorageUnformatted	"True"、"False"	string	
StorageNull	"True"、"False"	string	
StorageReadOnly	"True"、"False"	string	
Motion	"True"、"False"	string	
IOAlarm	"True"、"False"	string	
PIRAlarm	"True"、"False"	string	
Intellect	"True"、"False"	string	
AiHuman	"True"、"False"	string	
AiVehicle	"True"、"False"	string	
VideoLoss	"True"、"False"	string	
AiFaceDetection		JSON object	Please refer to Table -19.1.1.6
LCD	"True"、"False"	string	
PID	"True"、"False"	string	
PD&VD(PD)	"True"、"False"	string	
FD	"True"、"False"	string	
AD	"True"、"False"	string	
CC	"True"、"False"	string	
CD	"True"、"False"	string	
QD	"True"、"False"	string	
RSD	"True"、"False"	string	
LPD	"True"、"False"	string	
SOD	"True"、"False"	string	
VT	"True"、"False"	string	
SD	"True"、"False"	string	

Table -19.1.1.6 (AiFaceDetection JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Group		Object array	Please refer to Table -19.1.1.7

Table -19.1.1.7 (Group Array JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Name		string	
AutoSubscribe	"True"、"False"		

Table -19.1.1.8 (Filter JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Intellect		Object	Please refer to Table -19.1.1.9
IOAlarm		Object	Please refer to Table -19.1.1.9
Motion		Object	Please refer to Table -19.1.1.9
PIRAlarm		Object	Please refer to Table -19.1.1.9
VideoLoss		Object	Please refer to Table -19.1.1.9
StorageError		Object	
StorageFull		Object	
StorageNull		Object	
StorageReadOnly		Object	
StorageUnformatted		Object	
LCD		Object	
PID		Object	
PD&VD(PD)		Object	
FD		Object	
AD		Object	
CC		Object	
CD		Object	
QD		Object	
RSD		Object	
LPD		Object	
SOD		Object	
VT		Object	
SD		Object	

Table -19.1.1.9 (Filter JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	

Channel	0~Maximum channel number	Int array	
---------	--------------------------	-----------	--

Table -19.1.1.10 (Request JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
Filter		JSON object	Subscribe parameter, Please refer to Table -19.1.1.8
Mobile		JSON object	Subscribe parameter, Please refer to Table -19.1.1.11
Notification		JSON object	Subscribe parameter, Please refer to Table -19.1.1.18

Table -19.1.1.11 (Mobile JSON)

KEY	VALUE		COMMENT
	RANGE	TYPE	
AppID		String	
Language		String	
PushChannel		String	
Token		String	

Table -19.1.1.12

KEY	VALUE		COMMENT
	RANGE	TYPE	
Types		JSON object	Default Parameter, Please refer to Table -19.1.1.5

Table -19.1.1.13

KEY	VALUE		COMMENT
	RANGE	TYPE	
HddAlarm		JSON object	Please refer to Table -19.1.1.14
IOAlarm		JSON object	Please refer to Table -19.1.1.15
LowPower		JSON object	Please refer to Table -19.1.1.15
MotionAlarm		JSON object	Please refer to Table -19.1.1.15
PIRAlarm		JSON object	Please refer to Table -19.1.1.15
SmartAlarm		JSON object	Please refer to Table -19.1.1.15
VideoLoss		JSON object	Please refer to Table -19.1.1.15
HumanVehicle		JSON object	Please refer to Table -19.1.1.15
FaceAlarm		JSON object	Please refer to Table -19.1.1.16
LCDAlarm		JSON object	Please refer to Table -19.1.1.15
PIDAlarm		JSON object	Please refer to Table -19.1.1.15
SODAlarm		JSON object	Please refer to Table -19.1.1.15
PDAlarm		JSON object	Please refer to Table -19.1.1.15

FDAAlarm		JSON object	Please refer to Table -19.1.1.15
CCAlarm		JSON object	Please refer to Table -19.1.1.15
ADAlarm		JSON object	Please refer to Table -19.1.1.15
CDAAlarm		JSON object	Please refer to Table -19.1.1.15
QDAAlarm		JSON object	Please refer to Table -19.1.1.15
LPDAAlarm		JSON object	Please refer to Table -19.1.1.15
RSDAlarm		JSON object	Please refer to Table -19.1.1.15
VTAlarm		JSON object	Please refer to Table -19.1.1.15
SDAlarm		JSON object	Please refer to Table -19.1.1.15

Table -19.1.1.14

KEY	VALUE		COMMENT
	RANGE	TYPE	
Enabled		Int	
Type		Int	

Table -19.1.1.15

KEY	VALUE		COMMENT
	RANGE	TYPE	
ChnFlags		Int array	

Table -19.1.1.16

KEY	VALUE		COMMENT
	RANGE	TYPE	
Group		object array	Please refer to Table -19.1.1.17

Table -19.1.1.17

KEY	VALUE		COMMENT
	RANGE	TYPE	
Id		int	
Name		string	
ChnFlags		Int array	

Table -19.1.1.18

KEY	VALUE		COMMENT
	RANGE	TYPE	
notification_int erval		int	Time interval of notification
notification_int erval_max		int	Notification of displaying the maximum subscribing time for app
notification_int erval_min		Int	Notification of displaying the minimum subscribing time for app


```
"StorageReadOnly": "True",
"Motion": "True",
"IOAlarm": "True",
"PIRAlarm": "True",
"LCD": "True",
"PID": "True",
"PD&VD": "True",
"FD": "True",
"AD": "True",
"CC": "True",
"CD": "True",
"QD": "True",
"RSD": "True",
"LPD": "True",
"SOD": "True",
"VT": "True",
"SD": "True",
"AIfaceDetection": {
  "Group": [
    {
      "Name": "Allow List",
      "AutoSubscribe": "True"
    },
    {
      "Name": "Block List",
      "AutoSubscribe": "True"
    },
    {
      "Name": "Stranger",
      "AutoSubscribe": "True"
    },
    {
      "Name": "Group 1",
      "AutoSubscribe": "True"
    },
    {
      "Name": "Group 2",
      "AutoSubscribe": "True"
    },
    {
      "Name": "Group 3",
      "AutoSubscribe": "True"
    }
  ]
}
```

```
    },
    "AiHuman": "True",
    "AiVehicle": "True",
    "VideoLoss": "True"
  },
  "Filter": {
    "Intellect": {
      "Channel": [
        0,
        1,
        2,
        3,
        4,
        5,
        6,
        7,
        8,
        9
      ]
    },
    "IOAlarm": {
      "Channel": [
        0,
        1,
        2,
        3,
        4,
        5,
        6,
        7,
        8,
        9
      ]
    },
    "Motion": {
      "Channel": [
        0,
        1,
        2,
        3,
        4,
        5,
        6,
        7,

```

```
        8,  
        9  
    ]  
},  
"PIRAAlarm": {  
    "Channel": [  
        0,  
        1,  
        2,  
        3,  
        4,  
        5,  
        6,  
        7,  
        8,  
        9  
    ]  
},  
"StorageError": {},  
"StorageFull": {},  
"StorageNull": {},  
"StorageReadOnly": {},  
"StorageUnformatted": {},  
"VideoLoss": {  
    "Channel": [  
        0,  
        1,  
        2,  
        3,  
        4,  
        5,  
        6,  
        7,  
        8,  
        9  
    ]  
}  
}  
}
```

Request message:

POST/API/Push/Subscribe

{

```
"data": {
  "Filter": {
    "LCD": {
      "Channel": [
        0,
        1,
        2,
        3,
        4,
        5,
        6,
        7,
        8,
        9
      ]
    },
    "PID": {
      "Channel": [
        0,
        1,
        2,
        3,
        4,
        5,
        6,
        7,
        8,
        9
      ]
    },
    "PD&VD": {
      "Channel": [
        0,
        1,
        2,
        3,
        4,
        5,
        6,
        7,
        8,
        9
      ]
    }
  },
  "PD&VD": {
    "Channel": [
      0,
      1,
      2,
      3,
      4,
      5,
      6,
      7,
      8,
      9
    ]
  }
},
```

```
"FD": {
  "Channel": [
    0,
    1,
    2,
    3,
    4,
    5,
    6,
    7,
    8,
    9
  ]
},
"AD": {
  "Channel": [
    0,
    1,
    2,
    3,
    4,
    5,
    6,
    7,
    8,
    9
  ]
},
"CC": {
  "Channel": [
    0,
    1,
    2,
    3,
    4,
    5,
    6,
    7,
    8,
    9
  ]
},
"CD": {
  "Channel": [
```

```
    0,
    1,
    2,
    3,
    4,
    5,
    6,
    7,
    8,
    9
  ]
},
"QD": {
  "Channel": [
    0,
    1,
    2,
    3,
    4,
    5,
    6,
    7,
    8,
    9
  ]
},
"RSD": {
  "Channel": [
    0,
    1,
    2,
    3,
    4,
    5,
    6,
    7,
    8,
    9
  ]
},
"LPD": {
  "Channel": [
    0,
    1,
```



```
        2,
        3,
        4,
        5,
        6,
        7,
        8,
        9
    ]
},
"SOD": {
    "Channel": [
        0,
        1,
        2,
        3,
        4,
        5,
        6,
        7,
        8,
        9
    ]
},
"VT": {
    "Channel": [
        0,
        1,
        2,
        3,
        4,
        5,
        6,
        7,
        8,
        9
    ]
},
"SD": {
    "Channel": [
        0,
        1,
        2,
        3,
```

```
        4,  
        5,  
        6,  
        7,  
        8,  
        9  
    ]  
},  
"IOAlarm": {  
    "Channel": [  
        0,  
        1,  
        2,  
        3,  
        4,  
        5,  
        6,  
        7,  
        8,  
        9  
    ]  
},  
"Motion": {  
    "Channel": [  
        0,  
        1,  
        2,  
        3,  
        4,  
        5,  
        6,  
        7,  
        8,  
        9  
    ]  
},  
"PIRAAlarm": {  
    "Channel": [  
        0,  
        1,  
        2,  
        3,  
        4,  
        5,
```

```

        6,
        7,
        8,
        9
    ]
},
"StorageError": {},
"StorageFull": {},
"StorageNull": {},
"StorageReadOnly": {},
"StorageUnformatted": {},
"VideoLoss": {
    "Channel": [
        0,
        1,
        2,
        3,
        4,
        5,
        6,
        7,
        8,
        9
    ]
}
},
"Mobile": {
    "AppID": "com.RXCamView.push",
    "Language": "zh-Hans",
    "PushChannel": "APNS",
    "Token":
"eyJhbGciOiJIUzI1NiIsImtpZCI6ImJzdHM4Mjg1NWl4MmNmNDk0YWw5OWNiZGM4OTQ2YTQ0YWYxNyJ9.eyJhdWQiOiJlbnZGVsIl0sIlgtc3ViIjpw7IIRva2VuIjoizjA2MjE0YzFkOTM0OGRIZTEyYXUxMzIxM2M5YTM4ZDBiNjJjOWZmZDM5ZDFjMwI2ZjY0ODUxMTdkMwYxODdiOSIsIlVVSUQiOiI2ZTMzMjJjMy01MjFmLTQ0OWItYjk0Yy00MjE5ZGJiOTIwMmMifX0.ec_DrzO6AYidvJytmKADN9iW4sy3LqHBMJj9QEVaySquqlby43Oe5UvtqrU0y0t6o8cno6ypX9v4vzp5QGRbZw"
    }
}
}
}

```

Response message:

HTTP/1.1 200 OK

```
{
  "result": "success",
```

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

Request message:

POST/API/Push/Unsubscribe

```
{  
  "data": {  
    "Token": "f06214c1d9348dee11a513213c9a38d0b62c9ffd32d1c1b6f6485117d1f187b9"  
  }  
}
```

Response message:

HTTP/1.1 200 OK

```
{  
  "result": "success",  
  "data": {}  
}
```

Request message:

POST/API/Push/QueryDefault

null

Response message:

HTTP/1.1 200 OK

```
{  
  "result": "success",  
  "data": {  
    "Types": {  
      "StorageError": "True",  
      "StorageFull": "True",  
      "StorageUnformatted": "True",  
      "StorageNull": "True",  
      "StorageReadOnly": "True",  
      "Motion": "True",  
      "IOAlarm": "True",  
      "PIRAlarm": "True",  
      "LCD": "True",  
      "PID": "True",  
      "PD&VD": "True",  
      "FD": "True",  
      "AD": "True",  
      "CC": "True",  
      "CD": "True",  
      "QD": "True",  
      "RSD": "True",  
      "LPD": "True",  
    }  
  }  
}
```

```
        "SOD": "True",
        "VT": "True",
        "SD": "True",
        "VideoLoss": "True"
    }
}
}
```

Request message:

POST/API/PushSubscribe/Get

```
{
  "data": {
    "app_support_ai_notification_subscribe":true
  }
}
```

Response message:

HTTP/1.1 200 OK

```
{
  "result": "success",
  "data": {
    "HddAlarm": {
      "Enabled": 0,
      "Type": 0
    },
    "IOAlarm": {
      "ChnFlags": [
        255,
        255
      ]
    },
    "LowPower": {
      "ChnFlags": [
        255,
        0
      ]
    },
    "MotionAlarm": {
      "ChnFlags": [
        255,
        255
      ]
    },
    "PIRAlarm": {
      "ChnFlags": [
```

```
        255,  
        255  
    ]  
},  
"LCDAlarm": {  
    "ChnFlags": [  
        255,  
        255  
    ]  
},  
"PIDAlarm": {  
    "ChnFlags": [  
        255,  
        255  
    ]  
},  
"SODAlarm": {  
    "ChnFlags": [  
        255,  
        255  
    ]  
},  
"PDAlarm": {  
    "ChnFlags": [  
        255,  
        255  
    ]  
},  
"FDAlarm": {  
    "ChnFlags": [  
        255,  
        255  
    ]  
},  
"CCAlarm": {  
    "ChnFlags": [  
        255,  
        255  
    ]  
},  
"ADAlarm": {  
    "ChnFlags": [  
        255,  
        255  
    ]  
}
```

```
    ]
  },
  "CDAlarm": {
    "ChnFlags": [
      255,
      255
    ]
  },
  "QDAlarm": {
    "ChnFlags": [
      255,
      255
    ]
  },
  "LPDAlarm": {
    "ChnFlags": [
      255,
      255
    ]
  },
  "RSDAlarm": {
    "ChnFlags": [
      255,
      255
    ]
  },
  "VTAlarm": {
    "ChnFlags": [
      255,
      255
    ]
  },
  "SDAlarm": {
    "ChnFlags": [
      255,
      255
    ]
  },
  "VideoLoss": {
    "ChnFlags": [
      255,
      255
    ]
  },
}
```

```
"HumanVehicle": {
  "ChnFlags": [
    255,
    255
  ]
},
"FaceAlarm": {
  "Group": [
    {
      "Id": 2,
      "Name": "Allow List",
      "ChnFlags": [
        0,
        0
      ]
    },
    {
      "Id": 3,
      "Name": "Block List",
      "ChnFlags": [
        0,
        0
      ]
    },
    {
      "Id": 4,
      "Name": "Stranger",
      "ChnFlags": [
        0,
        0
      ]
    },
    {
      "Id": 5,
      "Name": "Group 1",
      "ChnFlags": [
        0,
        0
      ]
    },
    {
      "Id": 6,
      "Name": "Group 2",
      "ChnFlags": [
```



```
    "ChnFlags": [
      255,
      255
    ]
  },
  "LCDAlarm": {
    "ChnFlags": [
      255,
      255
    ]
  },
  "PIDAlarm": {
    "ChnFlags": [
      255,
      255
    ]
  },
  "SODAlarm": {
    "ChnFlags": [
      255,
      255
    ]
  },
  "PDAlarm": {
    "ChnFlags": [
      255,
      255
    ]
  },
  "FDAlarm": {
    "ChnFlags": [
      255,
      255
    ]
  },
  "CCAlarm": {
    "ChnFlags": [
      255,
      255
    ]
  },
  "ADAlarm": {
    "ChnFlags": [
      255,
```

```
        255
    ]
},
"CDAlarm": {
    "ChnFlags": [
        255,
        255
    ]
},
"QDAlarm": {
    "ChnFlags": [
        255,
        255
    ]
},
"LPDAlarm": {
    "ChnFlags": [
        255,
        255
    ]
},
"RSDAlarm": {
    "ChnFlags": [
        255,
        255
    ]
},
"VTAlarm": {
    "ChnFlags": [
        255,
        255
    ]
},
"SDAlarm": {
    "ChnFlags": [
        255,
        255
    ]
},
"VideoLoss": {
    "ChnFlags": [
        255,
        255
    ]
}
```

```
},
"HumanVehicle": {
  "ChnFlags": [
    255,
    255
  ]
},
"FaceAlarm": {
  "Group": [
    {
      "Id": 2,
      "Name": "Allow List",
      "ChnFlags": [
        0,
        0
      ]
    },
    {
      "Id": 3,
      "Name": "Block List",
      "ChnFlags": [
        0,
        0
      ]
    },
    {
      "Id": 4,
      "Name": "Stranger",
      "ChnFlags": [
        0,
        0
      ]
    },
    {
      "Id": 5,
      "Name": "Group 1",
      "ChnFlags": [
        0,
        0
      ]
    },
    {
      "Id": 6,
      "Name": "Group 2",
```


pu_error_code_unsupport_error	Non-supported request
pu_error_code_parameter_error	Parameter error
pu_error_code_partly_support	Success in setting up the parameter partly
pu_error_code_device_busy	Device Busy
pu_error_code_operate_flash_error	FLASH error
pu_error_code_result_unsupported_dest	Unsupported Corresponding service or APP
pu_error_code_result_passwd_weak	Weak grade of password
pu_error_code_not_support_ability	Non-support ability parameter
pu_error_code_update	System is updating
pu_error_code_reboot	System is rebooting
pu_error_code_restore	Default configuration is recovering
pu_error_code_recover_oprate_forbidden	Recovering operations of the default configuration and forbidden configuration
pu_error_code_roi_error_by_mjpeg	Failed to set up MJPEG under ROI
pu_error_code_roi_error_by_nosubstream	Only mainstream, failed to set up substream ROI
pu_error_code_tz_error_by_conflict	Conflict happen when setting up the trip zone detection
pu_error_code_oc_error_by_conflict	Conflict happen when setting up the object counting
pu_error_code_md_error_by_conflict	Conflict happen when setting up the motion detection
pu_error_code_tw_error_by_conflict	Conflict happen when setting up the trip wire detection
pu_error_code_area_error_by_conflict	Conflict happen when setting up the area detection(Perimeter 、 Goods left 、 Goods remove)
pu_error_code_vd_error_by_conflict	Conflict happen when setting up video interference
pu_error_code_igt_enable_conflict	Intelligent-switch has been opened, failed to set up Illuminance enhancement and shelter
pu_error_code_illumination_enable_conflict	Illumination switch enabling, failed to set up intelligence-switch and shelter
pu_error_code_videoblock_enable_conflict	Video masking enabled, set smart switch, failed to find enhancement
pu_error_code_igt_fail_by_switchoff	Smart switch not enabled, failed to set up smart service
pu_error_code_tw_error_by_over_num	Set up the number of tripwires over the maximum number
pu_error_code_area_error_by_over_num	Set up the number of zones over the maximum number 4
pu_error_code_area_error_by_over_total_num	The number of set areas exceeds the maximum

	total by 10
pu_error_code_area_error_by_zero_num	Set the number of zones to 0
pu_error_code_area_point_error_by_over_num	Set the number of region vertices over the maximum number
pu_error_code_area_abn_error_by_conflict	Set area check item removed, legacy enablement turned on with conflict
pu_error_code_area_rmv_error_by_conflict	Set up area check items left behind, removal enablement is turned on with conflict
pu_error_code_vim_global_para_error	Failure to set up the global parameters for intelligent analysis algorithm
pu_error_code_area_notsurport_error	Setting the intelligent analysis area is invalid
pu_error_code_privacy_mask_area_beyond	Privacy protected area over total area
pu_error_code_media_adapt_bw_mutex	Bandwidth adaptation conflicts with encoding type or code rate type
pu_error_code_wd_frame_conflict	Wide dynamic manual or automatic, acquisition frames cannot be set to 50, 60
pu_error_code_frame_wd_conflict	Capture frames of 50, 60, wide dynamic cannot be set manually or automatically
pu_error_code_conflict_videoblock_or_igt_is_enable	Video masking or behavioural analysis enabled, set illumination enhancement failed
pu_error_code_invalid_igt_area_id	Intelligent analysis area does not exist
pu_error_code_invalid_igt_line_id	Intelligent analysis lines do not exist
pu_error_code_invalid_privacy_id	Privacy protected areas do not exist
pu_error_code_set_drc_backlight_conflict	Backlight compensation enable conflicts with DRC enable
pu_error_code_privacymask_region_intersect	Overlapping privacy protection zones
pu_error_code_conflict_with_mosaic	Operation conflicts with mosaic
pu_error_code_conflict_with_osd	Operation conflicts with OSD
pu_error_code_conflict_with_pic	Operation conflicts with image overlay
pu_error_code_video_enc_close	Video encoder not open
pu_error_code_isp_para_conflict	ISP Conflicting parameters
pu_error_code_wb_high_temp_lower	The lower limit of colour temperature is greater than and equal to the upper limit of colour temperature in white balance auto-tracking mode
pu_error_code_isp_conflict_with_framerate	WDR/backlight compensation is mutually exclusive with full frame rate
pu_error_code_framerate_conflict_with_isp	50 fps/60 fps and WDR/backlight compensation are mutually exclusive
pu_error_code_osd_string_len_err	OSD length of character error
pu_error_code_audio_detect_mic_not_support	Audio anomaly detection is not supported when the audio input is in mic-in mode
pu_error_code_cif_conflict	CIF cannot be set due to mutual exclusivity

		with the corridor
pu_error_code_corridor_conflict		The corridor cannot be set up as it is mutually exclusive with CIF
pu_error_isp_ae_manual_conflict_wd		Manual exposure and wide dynamic are mutually exclusive
pu_error_isp_ae_manual_conflict_flicker		Manual exposure is mutually exclusive with the background frequency
pu_error_isp_ae_manual_conflict_hlc		Manual exposure and glare suppression
pu_error_isp_ae_shutpri_conflict_wd		Exposure modes shutter priority and wide dynamic are mutually exclusive
pu_error_isp_ae_shutpri_conflict_flicker		Exposure mode shutter priority and background frequency mutually exclusive
pu_error_isp_ae_shutpri_conflict_hlc		Exposure modes shutter priority and glare suppression
pu_error_isp_backlight_conflict_wd		Backlight compensation is mutually exclusive with wide dynamic
pu_error_isp_backlight_conflict_hlc		Backlight compensation and glare suppression are mutually exclusive
pu_error_isp_wd_conflict_hlc		Wide dynamic and glare suppression are mutually exclusive
pu_error_isp_defog_conflict_wd		Fogging mode and wide dynamic mutually exclusive
pu_error_isp_hlc_conflict_framerate		Strong light suppression and 50 fps 60 fps mutually exclusive
pu_error_isp_wd_conflict_framerate		Wide Dynamic and 50 fps 60 fps mutually exclusive
pu_error_isp_ae_manual_conflict_backlight		Manual exposure and backlight compensation
pu_error_isp_hlc_conflict_slow_shutter		Strong light suppression and slow shutter mutually exclusive
pu_error_isp_wd_conflict_slow_shutter		Wide dynamic and slow shutter are mutually exclusive
pu_error_multi_stream_conflict_face_dt		Multiple code streams and face detection mutually exclusive
pu_error_code_request_overtime		Request timeout
pu_error_result_need_face_alg_upgrade		Need to upgrade face algorithm package
pu_error_result_need_vhd_alg_upgrade		Need to upgrade the machine non-human algorithm package
pu_error_code_custom_package_undefined		Code stream package, ISP package not defined
pu_error_code_osdi_area_not_exist		Area does not exist, wrong index value
pu_error_code_osdi_area_too_large		Horizontal width of area over 180 degrees
pu_error_code_osdi_area_invalid_verdegree		Area coordinate points do not conform to the bottom-left, top-right rule
pu_error_code_osdi_index_already_exist		New area index number illegal, index already

	exists
pu_error_code_osdi_hor_or_ver_equal	Equal horizontal or vertical coordinates of area coordinate points
pu_error_code_osdi_index_invalid	Area index number illegal, less than 1
pu_error_code_osdi_index_equal	Duplicate regional index numbers
pu_error_code_itgt_imrs_err_vertex_illegal	Point coordinates do not make sense
pu_error_code_itgt_imrs_err_resolution_unsupport	Resolution not supported
pu_error_code_itgt_imrs_err_param_err	Parameter error
pu_error_code_itgt_imrs_err_not_config	Configuration error
pu_error_code_itgt_imrs_err_param_set	Parameter setting error
pu_result_tgt_function_mutual_lrpr	Please turn off the number plate recognition function first
pu_result_tgt_function_mutual_fd	Please turn off the face detection function first
pu_result_tgt_function_mutual_cd	Please turn off the parking detection function first
pu_result_tgt_function_mutual_va	Please turn off the behaviour analysis function first
pu_result_tgt_function_mutual_hc	Please turn off the overline statistics first
pu_result_tgt_function_mutual_at	Please turn off the auto-tracking function first
pu_result_tgt_function_mutual_vhd	Please turn off the non-person detection function first
pu_result_tgt_function_mutual_its	Please turn off the Smart Traffic function first
pu_result_tgt_function_mutual_qd	Please turn off the queue detection function first
pu_result_tgt_function_mutual_cdd	Please turn off the crowd density function first
pu_result_tgt_function_mutual_iball	Please turn off the Smart Traffic - Illegal Ball function first
pu_result_tgt_function_mutual_vhdface	Please turn off the function of multi-use by a device first
pu_result_itgt_mode_notsupport_alg	The algorithm is not supported in the current mode
pu_result_tgt_vhd_detection_err_wrong_video_mode	Object Classification is not supported by the machine non-human, turn off corridor mode first
pu_result_tgt_function_mutual_beh	Please turn off Enhanced Mode behaviour analysis first
pu_result_tgt_function_mutual_campus_trafficvent	Please close the entrance/exit operation first
pu_result_tgt_function_mutual_md	Please turn off the motion detection function first
pu_result_tgt_function_mutual_od	Please turn off the object shelter function first
pu_result_tgt_function_mutual_sc	Please turn off the scene change function first

pu_result_tgt_function_mutual_ood	Please turn off the virtual focus detection function first
pu_result_tgt_function_mutual_ca	Please turn off the people gathering function first
pu_result_tgt_function_mutual_hm	Please turn off the heat map function first
pu_result_tgt_function_mutual_trafficevent_cam pus	Please close all vehicle incident services first
pu_result_tgt_function_mutual_traffic_statistic_ campus	Please turn off the traffic counting service first
pu_result_tgt_function_mutual_vd	Please turn off video quality diagnostics first
pu_result_tgt_function_mutual_stabilizer	Please turn off the electronic stabilisation service first
pu_result_tgt_function_mutual_face_rec	Please turn off the face recognition function first
pu_result_tgt_function_mutual_fc	Please turn off high density faces first
pu_result_itgt_err_facerec_enable_fail	Face recognition enablement failure for non-existent algorithm package or version mismatch
pu_result_tgt_switch_mode_err_wrong_video_m ode	The switched mode does not support corridor mode
pu_result_tgt_switch_mode_err_enable_alg	The switched mode algorithm failed to enable, check the model first

A.2 Secondary Certification Development Guide

Secondary authentication scenarios such as formatting a hard drive, rebooting a device, etc. The following is an example of a device reboot.

Calling sequence	Interface name	Description
Obtain pre-login device information (optional)	/API/Login/Range	<p>1. Default HTTPS protocol. Please ignore SSL certificate when using https to access, the request URL is NVR800 IP, port is 443 by default, the actual port is based on the NVR800 port configuration.</p> <p>2. This interface is used before login, no need to use cookies and token authentication.</p>

Calling sequence	Interface name	Description
1、 User login	/API/Web/Login	<p>1. Default HTTPS protocol. Please ignore the SSL certificate when using https access, the request URL is NVR800 IP, port is 443 by default, the actual port is based on the NVR800 port configuration.</p> <p>2. The cookie and token returned by this interface are used for subsequent authentication of other interfaces.</p>
2、 User Heartbeat	/API/Login/Heartbeat (/API/SystemConfig/General/Set This interface allows you to configure the web session timeout)	<p>1、 After login, the default cookie timeout is 30 seconds and the heartbeat interface must be called to keep the session alive.</p> <p>2. When calling the interface during the debugging phase, it is recommended to use the keep_alive field to maintain 210 seconds of session retention time. In addition, the user are also controlled by a web session timeout after logging in. The session timeout defaults to 5 minutes and is recommended to be changed to one day for longer logins (the maximum session timeout is 1440 minutes).</p>
3、 Obtaining the public key	/API/Maintenance/TransKey/Get	1. Obtain the public key for encrypting the password on the board side.
4、 Equipment reboot	/API/Maintenance/DeviceReboot/Set	1. Transmit the cipher text to the board end, which decrypts it to obtain

Calling sequence	Interface name	Description
		the password for secondary authentication, and reboots the device after successful authentication.

A.3 Set Password Development Guide

The following is an illustration of setting up an email account

Calling sequence	Interface name	Description
Obtain pre-login device information (optional)	/API/Login/Range	<p>1. Default HTTPS protocol. Please ignore SSL certificate when using https access, the request URL is NVR800 IP, port is 443 by default, the actual port is based on the NVR800 port configuration.</p> <p>2. This interface is used before login, no need to use cookies and token authentication.</p>
1、 User login	/API/Web/Login	<p>1. Default HTTPS protocol. Please ignore the SSL certificate when using https access, the request URL is NVR800 IP, port is 443 by default, the actual port is based on the NVR800 port configuration.</p> <p>2. The cookie and token returned by this interface are used for subsequent authentication of other interfaces.</p>
2、 User Heartbeat	/API/Login/Heartbeat (/API/SystemConfig/General/Set This interface allows you to configure the web session timeout)	<p>1、 After login, the default cookie timeout is 30 seconds and the heartbeat interface must be</p>

Calling sequence	Interface name	Description
		<p>called to keep the session alive.</p> <p>2. When calling the interface during the debugging phase, it is recommended to use the keep_alive field to maintain 210 seconds of session retention time. In addition, the user is also controlled by the web session timeout after login, the default session timeout is 5 minutes, if you want to login for a long time, it is recommended to modify it to one day (the maximum session timeout is 1440 minutes).</p>
3、 Obtaining the public key	/API/Maintenance/TransKey/Get	1. Obtain the public key for encrypting the password on the board side.
4、 Set up an email account	/API/NetworkConfig/Email/Set	1. The client will send the public key and cipher text generated by itself to the board end, which will decrypt the cipher and save it.

FQA

Q: Why is it not feasible to develop web applications and send requests to the device via HTTP/HTTPS API?

A: As the cross-domain resource sharing strategy of the device server (CORS <https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>), any cross-domain request cannot be requested on the browser, only and Only the same-origin policy (Same-origin policy https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin_policy) web program on the device server can access the device normally. It is recommended to use other methods to send HTTP/HTTPS API requests instead of web applications.