LightAPI InterFace

User Guide



# Revison Record

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Revision Version** | **Description** | **Reviser** |
| 2017-4-14 | V1.00 | Initail version | Liu Yang |
| 2017-5-8 | V1.01 | 修改增加预置位接口-交换入参和返回值 | Liu Yang |
| 2017-8-10 | V2.00 | 合入LightAPI Interface二期接口； | Wang WeiJie |

# Catalog

[Revison Record 2](#_Toc498612567)

[Catalog 3](#_Toc498612568)

[Statement 5](#_Toc498612569)

[Introductions 6](#_Toc498612570)

[Supported HTTP Method 6](#_Toc498612571)

[URL Rule 6](#_Toc498612572)

[JSON 8](#_Toc498612573)

[Interface List 9](#_Toc498612574)

[System 9](#_Toc498612575)

[Json Block 9](#_Toc498612576)

[Time 27](#_Toc498612577)

[Reboot 30](#_Toc498612578)

[Logs 30](#_Toc498612579)

[Event 32](#_Toc498612580)

[DeviceInfo 37](#_Toc498612581)

[ChannelInfos 38](#_Toc498612582)

[Upgrade 38](#_Toc498612583)

[PTZ 42](#_Toc498612584)

[Json Block 42](#_Toc498612585)

[PTZCtrl 49](#_Toc498612586)

[Presets 49](#_Toc498612587)

[Capabilities 50](#_Toc498612588)

[AreaZoom 51](#_Toc498612589)

[Patrols 51](#_Toc498612590)

[Media 55](#_Toc498612591)

[Json Block 55](#_Toc498612592)

[LiveStreamURL 77](#_Toc498612593)

[Records 78](#_Toc498612594)

[Capabilities 83](#_Toc498612595)

[Video 84](#_Toc498612596)

[Audio 85](#_Toc498612597)

[OSD 85](#_Toc498612598)

[Snapshot 87](#_Toc498612599)

[Network 90](#_Toc498612600)

[Json Block 90](#_Toc498612601)

[Capabilities 95](#_Toc498612602)

[Interfaces 95](#_Toc498612603)

[DNS 97](#_Toc498612604)

[Image 98](#_Toc498612605)

[Json Block 98](#_Toc498612606)

[Capabilities 100](#_Toc498612607)

[Enhance 100](#_Toc498612608)

[Storage 101](#_Toc498612609)

[Json Block 101](#_Toc498612610)

[Capabilities 107](#_Toc498612611)

[Containers 107](#_Toc498612612)

[Schedule 109](#_Toc498612613)

[Alarm 110](#_Toc498612614)

[Json Block 110](#_Toc498612615)

[Capabilities 122](#_Toc498612616)

[MotionDetection 123](#_Toc498612617)

[VideoLoss 126](#_Toc498612618)

[TamperDetection 127](#_Toc498612619)

[IO 129](#_Toc498612620)

[Json Block 129](#_Toc498612621)

[InputSwitch 134](#_Toc498612622)

[OutputSwitch 137](#_Toc498612623)

# Statement

This document is the first draft and is for reference only. Contents of this document are subject to modification before official release. The products of our company may only support part of the interface in this document. Interface may changes caused by version update, no further notice. Our company should not assume any responsibility for any loss incurred.

# Introductions

The LightAPI is RESTful, it transmits data through the HTTP short connection.

**Important Notice!**

The API uses digest authentication.

## Supported HTTP Method

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Method** | **GET** | **PUT** | **POST** | **DELETE** |
| **Usage** | List members of resources. | Update resources. The request message must contain part or all members in resources. | Create resources. The Request message must contain one of the members in resources. | Delete one or all members of resources which support POST method. |

## URL Rule

/**LAPI**/**V1.0**[/**Channels**/<*ID>]*/<*service-name>*/<*resource-name>[*/<child-*resource-name>]*[/<*ID>*]][?<*key>=<value>*]

[ ]: means this part is an optional field.

<>: means this part is a parameter, it can be *string* or *unsigned long.*

*Channels: means the video input channel resources of device.*

<*ID>*:means the ID of video input channels, the *ID* is an unsigned long number.

<*service-name>*: we divide the services into a number of types, like “System” “Media” “Network” and so on.

*<resource-name>/*<*child*-*resource-name>*: *include physical resources and virtual resources.*

<*key>=<value>*: which follows the rule of HTTP URL parameter.

For example:

/LAPI/V1.0[/Channels/<ID>]/Network/Interfaces[/<ID>]

URL can be like: /LAPI/V1.0/Channels/1/Network/Interfaces/1

Or just like: /LAPI/V1.0/Network/Interfaces

**The format of common request is like**

PUT /LAPI/V1.0/System/Time/NTP HTTP/1.1

{

"Enabled":1,

"AddressType":0,

"Address":"24.56.178.140",

"Port":123,

"SynchronizeInterval":60

}

When method is PUT/POST, HTTP Body usually needs to include JSON Data.

Request must be sent to the HTTP/HTTPS port of the device.

**The format of common response is like**

HTTP/1.1 200 Ok

{

"Response":

{

"ResponseURL": "*URL*",

"ResponseCode": 0,

"SubResponseCode": 101,

"ResponseString": "Succeed",

"StatusCode": 0,

“Data”: *“null”* ***or*** *{…}*

}

}

"ResponseURL": on behalf of the URL carried by HTTP request message. When the request method is GET/PUT/DELETE, *URL* is the same as the URI in request. For POST request, it’s the *URL* of newly created object. Client can use it to get the new object directly.

"ResponseCode": on behalf of the processing System result. 0 means succeed.

"ResponseString": on behalf of the interpretation to the processing System result.

"Data": the info or data of the resource which is requested (GET). It’s JSON in most cases. With the same URL, PUT request data format is the same with the Data in GET response. The Data for each API are listed in Data item in tables below. If server work failed, “Data” contains “null”. If HTTP method is PUT/POST, Response “Data” contains “null”.

## JSON

For more information about JSON, please visit: [*www.json.org*](http://www.json.org). About string value of JSON, contain any Unicode character except*"* or *\* or *control character*.

# Interface List

## System

### Json Block

#### DeviceInfo

|  |  |
| --- | --- |
| DeviceInfo Json Block | {  "ID":,  "DeviceName":,  "DeviceType":,  "DeviceCode":,  "DeviceModel":,  "SerialNumber":,  "FirmwareVersion":,  "HardwareID":,  "UbootVersion":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Video input channel ID | 1 |
| DeviceName | M | string | Device name | “NVR-B200-E4” |
| DeviceType | C | unsigned long | 0：IPC  1：NVR  2：DVR  3：VMS  NVR surpported。 | 0 |
| DeviceCode | M | unsigned long | Device ID | 12 |
| DeviceModel | M | string | DeviceModel, length: 1~63 | “NVR-B200-E4” |
| SerialNumber | M | string | Device serial number, length: 0-25 | "210235C1MPF158000503" |
| FirmwareVersion | M | string | Firmware version, program version, length: 0-255 | "R2318P05" |
| HardwareID | M | string | Hardware ID, length: 0-255 | “A” |
| UbootVersion | M | string | UBOOT version, length: 0-255 | “V1.4” |

#### TimeCfgInfo

|  |  |
| --- | --- |
| TimeCfgInfo Json Block | {  "TimeZone": ,  "DeviceTime": ,  "DateFormat": ,  "HourFormat":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| TimeZone | M | string | String of Time zone, with range of [1,255]：  "GMT-12:00"  "GMT-11:00"  "GMT-10:00"  "GMT-09:00"  "GMT-08:00"  "GMT-07:00"  "GMT-06:00"  "GMT-05:00"  "GMT-04:30"  "GMT-04:00"  "GMT-03:30"  "GMT-03:00"  "GMT-02:00"  "GMT-01:00"  "GMT-00:00"  "GMT+01:00"  "GMT+02:00"  "GMT+03:00  "GMT+03:30"  "GMT+04:00"  "GMT+04:30"  "GMT+05:00"  "GMT+05:30"  "GMT+05:45"  "GMT+06:00"  "GMT+06:30"  "GMT+07:00"  "GMT+08:00"  "GMT+09:00"  "GMT+09:30"  "GMT+10:00"  "GMT+11:00"  "GMT+12:00"  "GMT+13:00" | "GMT+08:00" |
| DeviceTime | M | unsigned long | UTC, unit “second” | 1477104949 |
| DateFormat | C | unsigned long | Date formart：  0：YYYY-MM-DD  1：MM-DD-YYYY  2：DD-MM-YYYY  NVR required | 0 |
| HourFormat | C | unsigned long | Hour mode：  0：12h  1：24H  NVR required | 1 |

#### DeviceInfoList

|  |  |
| --- | --- |
| DeviceInfoList Json Block | {  “Nums”:,  “DeviceInfos”:[<[DeviceInfo](#_SysCapabilities)>]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Nums | M | unsigned long | Number of channel ID(s) of the connected device | 1 |
| DeviceInfos | C | array | Device info list. This node is optional. |  |
| DeviceInfo | C | Json Block | Device name. This node is optional. See DeviceInfo Json Block for details. |  |

#### DetailInfo

|  |  |
| --- | --- |
| DetailInfo Json Block | {  "ID": ,  "Name": ,  "Status": ,  "IsPoEPort":  "PoEStatus": ,  "StreamNums": ,  "DeviceType": ,  "AddressInfo": [AddressInfo](#_AddressInfo),  "AccessProtocol ": ,  "OffReason": ,  "RemoteIndex": ,  "Manufacturer": ,  "DeviceModel": ,  "GBID": ,  "AddType":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Channel ID | 1 |
| Name | M | string | Channel name, range of [0, 63] | "IP Camera 01" |
| Status | M | unsigned long | Channel status：  0: offline  1: online  2: idle | 0 |
| IsPoEPort | M | unsigned long | POE port or not：  0: no  1: yes  NVR required。 | 1 |
| PoEStatus | C | unsigned long | PoE status：  1: normal  2: lack of power  4: power over loaded  8: no power supply needed  Unneccessary when IsPoEPort =0 | 1 |
| StreamNums | C | unsigned long | Media stream limitation；  Absent if Status=2 | 3 |
| DeviceType | C | unsigned long | Type of connected device:  0: Fixed camera  1: PTZ camera  2: Fisheye camera (fisheye lens plus dewarping function)  3:Super wide angle lens camera (with fisheye lens or wide angle lens but without dewarping function)  Required for NVR.  Absent if Status=2. | 0 |
| AddressInfo | C | Json Block | Device address info.  Absent if Status=2. | See [Addressinfo](#_AddressInfo) Json Block for detail |
| AccessProtocol | C | unsigned long | Access protocol.  1: ONVIF  2: Private  4: RTSP  100: custom1  ~  131: custom32  Required for NVR.  Absent if Status=2. | 1 |
| OffReason | C | unsigned long | Device offline reason:  0: Offline, connecting  1: Online  2: Offline, incorrect username or password  3: Offline, network disconnected  4: Offline, failed to set live video stream  5: Offline, failed to start live video stream  6: Offline, stream stopped  7: Offline, timeout  8: IPCOffline, GB IPC waiting for registration message  9: IPCOffline, Stream transmission protocol modified for GB IPC  10: PoEOffline, abnormal PoE power supply  11: Offline, insufficient bandwidth  12: Offline, access denied for weak password  NVRRequired for NVR.  Absent if Status=2 | 1 |
| RemoteIndex | C | unsigned long | Remote channel ID (of multi-channel device).  NVRRequired for NVR.  Absent if Status=2. | 1 |
| Manufacturer | C | string | Manufacturer, lenth of string [0, 31]  NVR required  Absent if Status=2 | Company |
| DeviceModel | C | String | NVR Device model, lenth of string [0, 31], required for NVR.  Absent if Status=2. | IPC |
| GBID | C | String | [0, 31]GB resource ID, lenth of string: [0, 31].  optional |  |
| AddType | C | unsigned long | Add mode:  0: Manual  1: UPnP  NVRRequired for NVR.  Absent if Status=2. | 0 |

#### AddressInfo

|  |  |
| --- | --- |
| AddressInfo Json Block | {  "AddressType": ,  "Address": ,  "Port": ,  "AccessAddress" :  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| AddressType | M | unsigned long | IP address type:  0: IPv4  1: IPv6 | 0 |
| Address | M | string | Device IP address, lenth of string: [0,128].  Required for NVR. | 206.5.99.17 |
| Port | M | unsigned long | Device port.  Required for NVR. | 80 |
| AccessAddress | M | string | Device access address, lenth of string: [0, 63].  NVRRequired | http://206.5.99.17 |

#### SyncModeInfo

|  |  |
| --- | --- |
| SyncModeInfoJsonBlock | {  "Mode":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Mode | M | unsigned long | Time synchronization mode.  0:Accept synchronization from all servers  1:Synchronize with PC's system time  2:Synchronize with photo server  3:NTPSynchronize with NTP server  4:OnvifSynchronize with management server (not Onvif)  5:OnvifSynchronize with management server (Onvif) | 0 |

#### NTPServerInfo

|  |  |
| --- | --- |
| NTPServerInfoJsonBlock | {  “Enabled”: ,  “AddressType”: ,  “IPAddress”: ,  “DomainName”: ,  “Port”: ,  “SynchronizeInterval”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | M | boolean | Whether NTP server is enabled:  0:No  1:Yes | 0 |
| AddressType | M | unsigned long | Address type.  0:IPv4  1:IPv6 (currently not supported)  2:Domain name (supported by NVR and VMS) | 0 |
| IPAddress | C | string | IP address of NTP server, length range: [0, 64].  Mandatory if AddressType=0. | 0.0.0.0 |
| DomainName | C | string | Domain name of NTP server, length range: [0, 64].  Mandatory if AddressType=2. | pool.ntp.org |
| Port | C | unsigned long | NTP port, range: [1-65535].  IPCNot supported | 123 |
| SynchronizeInterval | M | unsigned long | Update interval.  Range for NVR and VMS: 5/10/15/30  1/2/3/6/12h11  5/10/15/30 minutes, 1/2/3/6/12h, 1 day, 1 week.  Range for IPC: 30-3600s.  All values must be converted into values in unit of second. | 10 |

#### DSTCfgInfo

|  |  |
| --- | --- |
| DSTCfgInfoJsonBlock | {  “Enabled”: ,  “BeginTime”:  {  “Month”: ,  “WeekNum”: ,  “WeekDay”: ,  “Hour”:  },  “EndTime”:  {  “Month”: ,  “WeekNum”: ,  “WeekDay”: ,  “Hour”:  },  “DSTBias”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | M | boolean | Whether DST is enabled.  0:No  1:Yes | 1 |
| BeginTime | M | Json Block | DST start time. | - |
| Month | M | unsigned long | Start month, complies with ISO8601.  1:Jan  2:Feb  3:Mar  4:Apr  5:May  6:Jun  7:Jul  8:Aug  9:Sep  10:Oct  11:Nov  12:Dec | 2 |
| WeekNum | M | unsigned long | Week of the start month:  1:1st week  2:2nd week  3:3rd week  4:4th week  5:Last week | 1 |
| WeekDay | M | unsigned long | Day of week *X* of the start month, complies with ISO8601:  1:Monday  2:Tuesday  3:Wednesday  4:Thursday  5:Friday  6:Saturday  7:Sunday | 6 |
| Hour | M | unsigned long | Start time, unit: hour (24H format), complies with ISO8601:  0:00:00  1:01:00  2:02:00  3:03:00  4:04:00  5:05:00  6:06:00  7:07:00  8:08:00  9:09:00  10:10:00  11:11:00  12:12:00  13:13:00  14:14:00  15:15:00  16:16:00  17:17:00  18:18:00  19:19:00  20:20:00  21:21:00  22:22:00  23:23:00 | 0 |
| EndTime | M | Json Block | DST end time. | - |
| DSTBias | M | unsigned long | 30, 60, 90, 120  DST bias, unit: min, values allowed: 30, 60, 90, 120. | 60 |

#### QueryInfo

|  |  |
| --- | --- |
| QueryInfoJsonBlock | {  " QryType": ,  " QryCondition": ,  " QryData":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| QryType | M | unsigned long | Query condition:  0: Username  1: Organization name  2: Device name  3: Channel name  4: UTC time, unit: sec  5: Service type (main type)  6: Operation type (sub type)  7: Operation object  8: Alarm type  9: Alarm source name  10: Alarm level  11: Whether alarm is acknowledged  12: User who acknowledged alarm  13: Time when alarm is acknowledged  14: IDAlarm device ID  15: IDAlarm channel ID  16: Alarm sub type  17: Server  For NVR, only 4, 5, and 6 are supported.  For VMS:  0, 4, 5, 6 and 7 are supported for operation logs;  4, 8, 9, 10, 11, 12, 13, 14, 15, 16 and 17 are supported for alarm logs. | 4 |
| QryCondition | M | unsigned long | Logic type of query condition:  0: Equal to  1: Greater than  2: Less than  3: Greater than or equal to  4: Less than or equal to  5: Not equal to  6: Fuzzy search  7: ININ set  8: Ascending order  9: Descending order  NVR  For NVR, only 0, 3 and 4 are supported.  VMS  For VMS, all values are supported. | 0 |
| QryData | C | string | Rvalue for query condition.  Alarm type is expressed as character string; a query allows up to 16 alarm types separated with comma(,); empty if QryCondition=6 or 7.  For NVR, only one type each time is allowed. | " DiskOffline" |

#### Log

|  |  |
| --- | --- |
| LogJsonBlock | {  "Time": ,  "MainType": ,  "SubType": ,  "ID" : ,  "LoginName": ,  "IP": ,  " DetailInfo": ,  "OperObject" ,  "Result":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Time | M | unsigned long | Log time in UTC, unit: sec. | 1476088399 |
| MainType | M | string | Main log type, expressed as enumerations. See [**Alarm&Operate Log Type**](#_Alarm&Operate_Log_Type). | "alarm" |
| SubType | M | string | Sub log type, expressed as enumerations. See [**Alarm&Operate Log Type**](#_Alarm&Operate_Log_Type) | "motionDetectionStart" |
| ID | C | unsigned long | Required if channel ID is involved in logs.  0 means device; other values (starts from 1) represent channel IDs.  Represents alarm input/output ID if alarm input/output configuration or alarm is involved.  NVR\*100+0  For NVR: channel ID\*100+alarm input ID. For example,  channel ID is 0 for the device itself, and  101 means ID of the first alarm input/output of the first channel.  For VMS, it is the alarm input ID on VMS. | 1 |
| LoginName | M | string | User, length range: [0, 64]. | Admin |
| IP | M | string | User IP, length range: [0, 64]. | 206.5.3.13 |
| DetailInfo | C | string | Detailed info, currently only shows PoE port status, length range: [0, 64].  Required for NVR. |  |
| OperObject | C | string | Operation object, length range: [0, 64].  Required for VMS. | bsdkabik |
| Result | C | unsigned long | Operation result. Required for VMS. | 1234 |

#### AlarmLog

|  |  |
| --- | --- |
| AlarmLogJsonBlock | {  “AlarmID”: ,  “AlarmType”:,  “AlarmSubType”:,  “AlarmLevel”:,  “ServerID”:,  “DevID”:,  “ChlID”:,  “AlarmSrc”:,  “AlarmTime”:,  “AlarmChecked”:,  “AlarmCheckUser”:,  “AlarmCheckTime”:,  “AlarmCheckDesc”:  }, |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| AlarmID | M | unsigned long | IDAlarm log ID. |  |
| AlarmType | M | string | Alarm type, expressed as enumerations. See [**Alarm&Operate Log Type**](#_Alarm&Operate_Log_Type) | "alarm" |
| AlarmSubType | O | string | Alarm sub type, expressed as enumerations.  [**Alarm&Operate Log Type**](#_Alarm&Operate_Log_Type) | "motionDetectionStart" |
| AlarmLevel | M | unsigned long | Alarm level:  0:Critical  1:Major  2:Minor  3:Warning  4:Message | 0 |
| ServerID | M | unsigned long | Server ID | 3 |
| DevID | M | unsigned long | Device ID | 1 |
| ChlID | M | unsigned long | Channel ID | 831 |
| AlarmSrc | M | string | Alarm source info, length range: [1, 128]. | “Camera1” |
| AlarmTime | M | unsigned long | Alarm time in UTC, unit: sec. |  |
| AlarmChecked | M | boolean | Whether alarm is acknowledged:  0 :No  1: Yes | 1 |
| AlarmCheckUser | M | string | User who acknowledged alarm, length range: [0, 64]. | ”admin” |
| AlarmCheckTime | M | unsigned long | UTC time when alarm is acknowledged, unit: sec. | 1476088399 |
| AlarmCheckDesc | M | string | Description.  Length range: [0, 516]. |  |

#### Alarm&Operate Log Type



#### AlarmInfo

|  |  |
| --- | --- |
| AlarmInfo  JsonBlock | {  "AlarmType": ,  "TimeStamp": ,  "AlarmSeq": ,  　 "AlarmSrcID": ,  "AlarmSrcType":,  　"AlarmSrcName":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| AlarmType | M | string | Alarm type, see [**Alarm&Operate Log Type**](#_Alarm&Operate_Log_Type) | “MotionAlarmOn” |
| TimeStamp | M | unsigned long | Alarm time. | 1489040894 |
| AlarmSeq | C | unsigned long | Alarm sequence number.  Used by VMS and IPC. | 327 |
| AlarmSrcID | C | unsigned long | Resource ID.  Used by VMS.  NVR requirements,  1.when AlarmSrcType is within 0-7, AlarmSrcID represents the index of storage disk;  2. when AlarmSrcType is 8, AlarmSrcID represents the channel ID;  3. when AlarmSrcType is within 9, AlarmSrcID represents the ID of alarm input channel;  4. when AlarmSrcType is within 10, AlarmSrcID is default 0, represents the system itself. | 860 |
| AlarmSrcType | C | unsigned long | Type of alarm source：  0: local hard disk;  1:HDD of expansion storage tank 1;  2:HDD ofexpansion storage tank 2;  3:HDD of storage array ;  4: HDD of NAS;  5: HDD of SAN;  6: HDD of eSATA;  7: SD card;  8:media channel;  9:alarm input channel;  10: system  Only for NVR | 2 |
| AlarmSrcName | C | string | Alarm source name.  String length range: [0, 128].  Used by VMS. | “206.9.252.7\_V\_6” |

#### ResInfo

|  |  |
| --- | --- |
| ResInfo  JsonBlock | {  "Reference": ,  "Action":,  "Num":,  "ResInfos": [{  "ResType": ,  "ResID": }, ……]  }  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Reference | M | string | Used by client to identify the URL of alarm event messages.  <SubscribersID>:IPportUsed to differentiate different subscription clients when a user logs in to multiple clients with the same IP and port. | "http://192.168.0.13:80/LAPI/V1.0/System/ALarmEvent/CommonAlarm/<SubscribersID>" |
| Action | M | unsigned long | Sub type of resource change events:  0:Add  1:Delete  2:Modify  3:Online  4:Offline  5: EmapAlarm report on e-map | 0 |
| Num | M | unsigned long | Number of changed resources. | 5 |
| ResInfos | M | array | Resource data change info. |  |
| ResType | M | unsigned long | Resource type:  0: User resource  1: Device resource  2: Channel resource  3: User permission resource  4: Sequence resource  5 Hot spot resouce  6: Hot zone resource  7: Alarm resource  8: Time template  9: System permission resource  10: Device permission resource  11: Organization resource  12: Alarm task resource  13: Master/slave resource  0xff:Invalid resource type. | 2 |
| ResID | M | unsigned long | Resource ID | 1206 |

#### VMSCommon

|  |  |
| --- | --- |
| VMSCommon Json Block | {  "OrgID": ,  "DevID": ,  "DevName": ,  "ChlIndex": ,  "Permission":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| OrgID | M | unsigned long | Organization ID. | 1 |
| DevID | M | unsigned long | IParent device ID. | 1 |
| DevName | M | string | Parent device name.  Length range: [1, 256]. | "123" |
| ChlIndex | M | unsigned long | Channel ID. | 1 |
| Permission | M | unsigned long | Channel permission. | 214 |

#### UpgradeStatusInfo

|  |  |
| --- | --- |
| UpgradeStatusInfo Json Block | {  "ID":  "Status" :,  "Percent":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Meaningless in /LAPI/V1.0/System/UpgradeStatus，0 as defalut；  To be as the device aID in /LAPI/V1.0/System/UpgradeStatuss?DevID=[<ID>,<ID>,......]  To be as the input video channel ID in /LAPI/V1.0/Channels/System/UpgradeStatuses | 1 |
| Status | M | unsigned long | Upgrade status, see as belows:  0：during Initialization 1：searching 2：downloading 3：transmission 4：upgrading 5：upgrade error 6：already the latest version 7：higher version 8：upgrade finished 9：fail to get the rate of upgrade progress  10：version not match  11：unable to run multiple upgrade progress  12：rate request timeout | 2 |
| Percent | C | unsigned long | Rate of progress，range of [0,100]。  Required when Status is 2/3/4 | 50 |

#### UpgradeStatusInfoList

|  |  |
| --- | --- |
| UpgradeStatusInfoListJsonBlock | {  "Nums":,  "UpgradeStatuses":[<[UpgradeStatusInfo](#_UpgradeStatusInfo)>,<[UpgradeStatusInfo](#_UpgradeStatusInfo)>,……]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Nums | M | unsigned long | numbers of IPC channel。 | 1 |
| UpgradeStatuses | C | array | List of status info, required when Nums is greater than 0 | - |
| UpgradeStatusInfo | C | Json Block | Status info, required when Numb is greater than 0 | See [UpgradeStatusInfo](#_UpgradeStatusInfo) Json Block for details |

### Time

#### /LAPI/V1.0/System/Time

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Time |
| **Description** | /Query device time info. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [TimeCfgInfo](#_TimeCfgInfo) |
| **Note** | NVR & IPC |
|  |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Time |
| **Description** | set device time info. |
| **Method** | PUT |
| **Input Data** | [TimeCfgInfo](#_DeviceInfoList) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
|  |  |

#### /LAPI/V1.0/System/Time/SyncMode

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Time/SyncMode |
| **Description** | Query time synchronization mode. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [SyncModeInfo](#_SyncModeInfo) |
| **Note** | IPC |
|  |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Time/SyncMode |
| **Description** | set time synchronization mode. |
| **Method** | PUT |
| **Input Data** | [SyncModeInfo](#_SyncModeInfo) |
| **Success Return Data** | None |
| **Note** | IPC |
|  |  |

#### /LAPI/V1.0/System/Time/NTP

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Time/NTP |
| **Description** | NTPQuery NTP server info. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | {  “Num”: ,  “NTPServerInfos”:[< [NTPServerInfo](#_NTPServerInfo)>…]  } |
| **Note** | IPC & NVR & VMS |
|  |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Time/NTP |
| **Description** | set NTP server info. |
| **Method** | PUT |
| **Input Data** | {  “Num”: ,  “NTPServerInfos”:[<[NTPServerInfo](#_NTPServerInfo)>…]  } |
| **Success Return Data** | None |
| **Note** | IPC & NVR & VMS |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | NTPNumber of NTP servers.  Max: 1 (currently only one NTP server is allowed.) | 1 |
| NTPServerInfos | M | array | NTP server list  Currently only one NTP server is supported. |  |
| NTPServerInfo | M | unsigned long | See [NTPServerInfo](#_NTPServerInfo) Json Block for details |  |

#### /LAPI/V1.0/System/Time/DST

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Time/DST |
| **Description** | Query DST info. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [DSTCfgInfo](#_DSTCfgInfo_1) |
| **Note** | IPC & NVR & VMS |
|  |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Time/DST |
| **Description** | set DST info. |
| **Method** | PUT |
| **Input Data** | [DSTCfgInfo](#_DSTCfgInfo_1) |
| **Success Return Data** | None |
| **Note** | IPC & NVR & VMS |
|  |  |

### Reboot

#### /LAPI/V1.0/System/Reboot

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Reboot |
| **Description** | Restart device. |
| **Method** | PUT |
| **Input Data** | None |
| **Success Return Data** | None |
| **Note** | IPC & NVR & VMS |
|  |  |

### Logs

#### /LAPI/V1.0/System/Logs

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Logs |
| **Description** | Query logs. |
| **Method** | POST |
| **Input Data** | {  "Num":  "QueryInfos": [<[QueryInfo](#_QueryInfo)>, <[QueryInfo](#_QueryInfo)>,...],  "Limit": ,  "Offset":  } |
| **Success Return Data** | {  "Total": ,  "Offset":  “Num”:,  "Logs":[<[Log](#_Log) >, <[Log](#_Log) >, ...]  } |
| **Note** | NVR & VMS |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num(Input Data) | M | unsigned long | Number of query conditions. | 2 |
| QueryInfos | C | array | Query condition info list. Optional if Num=0.  Max alarm types: 16. | - |
| QueryInfo | M | Json Block | Query condition info. | See [QueryInfo](#_QueryInfo) Json Block |
| Limit | M | unsigned long | Number of logs queried each time. Max: 200. | 100 |
| Offset | M | unsigned long | Start query from current sequence number. Sequence number starts from 0. | 0 |
| Num(Return Data) | M | unsigned long | Number of logs returned on current page. | 100 |
| Total | M | unsigned long | Total number of logs. | 2850 |
| Logs | C | array | Log info list. Optional if Num=0. | - |
| Log | M | Json Block | Log info ,see [Log](#_Log) Json Block | - |

#### /LAPI/V1.0/System/AlarmLogs

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/AlarmLogs |
| **Description** | Query VMS alarm logs. |
| **Method** | POST |
| **Input Data** | {  "Num":  "QueryInfos": [<[QueryInfo](#_QueryInfo)>, <[QueryInfo](#_QueryInfo)>,...],  "Limit": ,  "Offset":  } |
| **Success Return Data** | {  “Total”:,  “Offset”:,  “Num”:,  “AlarmLogs”:[<[AlarmLog](#_AlarmLog)>, <[AlarmLog](#_AlarmLog)>…]  } |
| **Note** | VMS |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num(Input Data） | M | unsigned long | Number of query conditions. | 2 |
| QueryInfos | C | array | Query condition info list. Optional if Num=0.  Max alarm types: 16. | - |
| QueryInfo | M | Json Block | Query condition info. | See [QueryInfo](#_QueryInfo) Json Block for details |
| Limit | M | unsigned long | Number of logs queried each time. Max: 200. | 100 |
| Offset | M | unsigned long | Start query from current sequence number. Sequence number starts from 0. | 0 |
| Num(Return Data) | M | unsigned long | Number of alarm logs returned on current page. | 100 |
| Total | M | unsigned long | Total number of alarm logs. | 2850 |
| AlarmLogs | C | array | Alarm log info list. Optional if Num=0. | - |
| AlarmLog | M | Json Block | See [AlarmLog](#_AlarmLog) Json Block for details | - |

### Event

#### /LAPI/V1.0/System/Event/Subscription

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Event/Subscription |
| **Description** | Subscribe to alarms. |
| **Method** | POST |
| **Input Data** | {  "AddressType": ,  "IPAddress": ,  "Port": ,  "Duration":  } |
| **Success Return Data** | {  “ID”:,  "Reference":,  "CurrentTime":,  "TerminationTime":  } |
| **Note** | IPC & NVR & VMS |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirment** | **Type** | **Description** | **Example** |
| AddressType | M | unsigned long | IP address type:  0: IPv4  1: IPv6  2:Domain name  3: IPv4 and IPv6  Currently only IPv4 is supported. | 0 |
| IPAddress | C | string | IP address of subscriber. When event happened, The device will create a TCP connect to this IP and push an event message. Each message will take a new connect  IPv4 address.  Length range: [0,64]. |  |
| Port | M | unsigned long | Port of subscriber. When event happened, IPC will connect to this port and send an event message  range: [1, 65535]. |  |
| Duration | M | unsigned long | The seconds from subscription start(POST succeeded) to finish. The subscription server will be terminated when timeout. IPC will never push message to the subscriber unless subscribe again. If termination time is not reached, subscriber can renew(PUT) subscription.  Subscription duration, unit: s, range: [30, 3600]. |  |
| ID | M | unsigned long | ReferenceURLID  The index of subscriber, same as the last segment of the URL for Reference (/0), used to identify subscription when subscription is refreshed or deleted. | 0 |
| Reference | M | string | When subscribe successed, IPC will return this string to subscriber. Each event message pushed from IPC will contain the same string. Subscriber can identify IPC by this string | "192.168.0.13:80/Subscription/Subscribers/0" |
| CurrentTime | M | unsigned long | Subscription start time  UTC time, seconds since 00:00 on Jan 1, 1970. | 1477104900 |
| TerminationTime | M | unsigned long | Subscription finish time  UTC time, seconds since 00:00 on Jan 1, 1970. | 1477104949 |

#### /LAPI/V1.0/System/Event/Subscription/<ID>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Event/Subscription/<ID> |
| **Description** | Renew subscription. |
| **Method** | PUT |
| **Input Data** | {  "Duration":  } |
| **Success Return Data** | {  "Reference":,  "CurrentTime":,  "TerminationTime":  } |
| **Note** | IPC & NVR & VMS |
|  |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Event/Subscription/<ID> |
| **Description** | Cancel subscription. |
| **Method** | DELETE |
| **Input Data** | None |
| **Success Return Data** | None |
| **Note** | IPC & NVR & VMS |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirment** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Alarm subscription ID, assigned by subscribed side. In the alarm subscription interface, a value is returned to the subscriber if subscription is successful. Required when refreshing or cancelling alarm subscription. | 0 |
| Duration | M | unsigned long | The seconds from subscription start(POST successed) to finish. The subscription server will be terminated when timeout. IPC will never push message to the subscriber unless subscribe again. If termination time is not reached, subscriber can renew(PUT) subscription.  Subscription duration, unit: s, range: [30, 3600]. | 300 |
| Reference | M | string | When subscribe successed, IPC will return this string to subscriber. Each event message pushed from IPC will contain the same string. Subscriber can identify IPC by this string | "192.168.0.13:80/Subscription/Subscribers/1" |
| CurrentTime | M | unsigned long | Subscription start time  UTC time, seconds since 00:00 on Jan 1, 1970. | 1477104900 |
| TerminationTime | M | unsigned long | Subscription finish time  UTC time, seconds since 00:00 on Jan 1, 1970. | 1477104949 |

#### /LAPI/V1.0/System/Event/Notification/Alarm

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Event/Notification/Alarm |
| **Description** | Push alarms. |
| **Method** | POST |
| **Input Data** | {  "Reference":,  "AlarmInfo": {  "AlarmType":,  "TimeStamp":,  "AlarmSeq":,  "AlarmSrcID": ,  "AlarmSrcType":,  "AlarmSrcName":  }  } |
| **Success Return Data** | None |
| **Note** | IPC & NVR & VMS |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirment** | **Type** | **Description** | **Example** |
| Reference | M | string | When subscribe successed, IPC will return this string to subscriber. Each event message pushed from IPC will contain the same string. Subscriber can identify IPC by this string | "192.168.0.13:80/Subscription/Subscribers/1" |
| AlarmInfo | M | Json Block | Format of alarm push message. | See [AlarmInfo](#_AlarmInfo_3) Json Block for details. |

#### /LAPI/V1.0/System/Event/Notification/ResChange

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Event/Notification/ResChange |
| **Description** | Push resource change events. |
| **Method** | POST |
| **Input Data** | [ResInfo](#_ResInfo) |
| **Success Return Data** | None |
| **Note** | VMS |
|  |  |

### DeviceInfo

#### /LAPI/V1.0/System/DeviceInfo

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/DeviceInfo |
| **Description** | get device information. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**DeviceInfo**](#_DeviceInfo) |
| **Notes** | 返回值DeviceInfo中的ID为0，暂时无任何意义。  the return value of ID from DeviceInfo is 0 and the API is temporarily meaningless  IPC & NVR |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/System/DeviceInfo

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/System/DeviceInfo |
| **Description** | It is used to get a particular VideoIn channel device information. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**DeviceInfo**](#_DeviceInfo) |
| **Notes** | NVR |
|  |  |

#### /LAPI/V1.0/Channels/System/DeviceInfos

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/System/DeviceInfos |
| **Description** | It is used to get all of VideoIn channels device information. |
| **Method** | GET |
| **Input Data** | None |
| **Data** | [**DeviceInfoList**](#_DeviceInfoList_1) |
| **Notes** | NVR |
|  |  |

### ChannelInfos

#### /LAPI/V1.0/Channels/System/ChannelDetailInfos

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/System/ChannelDetailInfos |
| **Description** | Query detailed info about all video channels. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | {  "Nums": ,  "DetailInfos": [<[DetaiInfo](#_ChannelsBasicInfo)>, <[DetaiInfo](#_ChannelsBasicInfo)>, <[DetaiInfo](#_ChannelsBasicInfo)>, …]  } |
| **Note** | NVR |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Ranger** | **Type** | **Description** | **Example** |
| Nums | M | unsigned long | Total number of video channels of device. | 3 |
| DetailInfos | C | Array | Video channel info list. Optional if Nums=0. | - |
| DetaiInfo | C | Json　Block | Detailed video channel info. | See [DetailInfo](#_ChannelsBasicInfo) Json Block |

### Upgrade

#### /LAPI/V1.0/System/Upgrade

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/Upgrade |
| **Description** | Excute the upgrade |
| **Method** | PUT |
| **Input Data** | {  "UpgradeType":  } |
| **Success Response Data** | {  "UpgradeTaskID":  } |
| **Note** | Upload firmware through /LAPI/V1.0/System/UploadFirmware when doing local upgrade. Query for status through /LAPI/V1.0/System/UpgradeStatus  IPC & NVR & VMS |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| UpgradeType | M | unsigned long | Type of upgrade:  1：cloud upgrade；  2：USB upgrade；  3：local upgrade | 3 |
| UpgradeTaskID | C | string | Mission ID, for vertification, required when NVR using local upgrade. | “2b188bd214ae1e2b0d96d5b7636d3bd0@1504663113” |

#### /LAPI/V1.0/System/UploadFirmware?UpgradeTaskID=<UpgradeTaskID>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/UploadFirmware?UpgradeTaskID=<UpgradeTaskID> |
| **Description** | Upload upgrade firmware |
| **Method** | POST |
| **Message body** | Accept-Encoding: gzip, deflate  Content-Length: 25665652  Content-Type: multipart/form-data; boundary=---------------------------5851473516029  -----------------------------5851473516029  Content-Disposition: form-data; name="UpgradeFile"; filename="Program.bin"  Content-Type: application/octet-stream  File content  -----------------------------5851473516029— |
| **Note** | Content-Dispostion row，Content-Type row and null string are also standard message body for http file transmission.  Query for status through /LAPI/V1.0/System/UpgradeStatus  IPC & NVR &VMS |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| UpgradeTaskID | C | string | Mission ID for vertification,  required when NVR using local upgrade. Returned through /LAPI/V1.0/System/Upgrade | “2b188bd214ae1e2b0d96d5b7636d3bd0@1504663113” |

#### /LAPI/V1.0/System/UpgradeStatus

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/System/UpgradeStatus |
| **Description** | Query for upgrade status |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [UpgradeStatusInfo](#_UpgradeStatusInfo) |
| **Note** | IPC & NVR |
|  |  |

#### /LAPI/V1.0/Channels/System/Upgrade

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/System/Upgrade |
| **Description** | Upgrade the specific video channel device |
| **Method** | PUT |
| **Input Data** | {  "Channels":[<ID>,<ID>,…],  "UpgradeType":  } |
| **Success Response Data** | {  "UpgradeTaskID":  } |
| **Note** | Upload firmware during Local upgrade through  /LAPI/V1.0/Channels/System/UploadFirmware?UpgradeTaskID=<UpgradeTaskID>  Query for upgrade status through  /LAPI/V1.0/Channels/System/UpgradeStatuses  NVR |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Channels | M | array | List of channel ID, only available when using local upgrade and required for specific channel. |  |
| ID | M | unsigned long | Input channel ID | 1 |
| UpgradeType | M | unsigned long | Type of upgrade:  1：cloud upgrade；  2：USB upgrade；  3：local upgrade | 1 |
| UpgradeTaskID | C | string | Mission ID, for vertification, required when NVR using local upgrade. | “956087319” |

#### /LAPI/V1.0/Channels/System/UploadFirmware?UpgradeTaskID=<UpgradeTaskID>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/System/UploadFirmware?UpgradeTaskID=<UpgradeTaskID> |
| **Description** | Upload firmware for specific channel |
| **Method** | POST |
| **Message body** | Accept-Encoding: gzip, deflate  Content-Length: 14231621  Content-Type: multipart/form-data; boundary=---------------------------178201537128716  -----------------------------178201537128716  Content-Disposition: form-data; name="UpgradeFile"; filename="IPC\_D1201-B5022P20D1702\_JPEG.zip"  Content-Type: application/octet-stream  File content  -----------------------------178201537128716— |
| **Note** | Content-Dispostion row，Content-Type row and null string are also standard message body for http file transmission.  Query for upgrade status through /LAPI/V1.0/Channels/System/UpgradeStatus  NVR |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| UpgradeTaskID | C | string | Mission ID for vertification,  required when NVR using local upgrade. Returned through /LAPI/V1.0/Channels/System/Upgrade | “956087319” |

#### /LAPI/V1.0/Channels/System/UpgradeStatuses

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/System/UpgradeStatuses |
| **Description** | Query for the upgrade status of input channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [UpgradeStatusInfoList](#_UpgradeStatusInfoList) |
| **Note** | NVR |
|  |  |

## PTZ

### Json Block

#### PTZCommandInfo

|  |  |
| --- | --- |
| PTZCommandInfo Json Block | {  "PTZCmd":,  "Para1":,  "Para2":,  "Para3":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| PTZCmd | M | unsigned long | PTZ operation command word:  0x0101: Stop decreasing iris  0x0102: Decrease iris  0x0103: Stop increasing iris  0x0104: Increase iris  0x0201: Stop focusing near  0x0202: Focus near  0x0203: Stop focusing far  0x0204: Focus far  0x0301: Stop zooming in  0x0302: Zoom in  0x0303: Stop zooming out  0x0304: Zoom out  0x0402: Turn up  0x0404: Turn down  0x0502: Turn right  0x0504: Turn left  0x0702: Turn upper left  0x0704: Turn lower left  0x0802: Turn upper right  0x0804: Turn lower right  0x0901: Stop  0x0A01: Wiper on  0x0A02: Wiper off  0x0B01: Light on  0x0B02: Light off  0x0C01: Heater on  0x0C02: Heater off  0x0D01: IR on  0x0D02: IR off | 1794（0x0702） |
| Para1 | C | unsigned long | PTZ command parameter 1, pan speed, range: [1,9]. Only for use by 0x4020x804, and pan speed is 0 when moving up or down. Optional if PTZCmd0x4020x804. | 2 |
| Para2 | C | unsigned long | PTZ command parameter 2, tilt speed, range: [1,9]. Only for use when PTZCmd is 0x4020x804, and tilt speed is 0 when moving left or right. Optional if PTZCmd0x4020x804. | 2 |
| Para3 | C | unsigned int | PTZ command parameter 3, not in use and fill in with 0. | 0 |

#### PresetInfoList

|  |  |
| --- | --- |
| PresetInfoList Json Block | {  "Nums": ,  "PresetInfos": [<[PresetInfo](#_PresetInfo)>,<[PresetInfo](#_PresetInfo)>,......]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Nums | M | unsigned long | Number of preset(s). Up to 255 presets can be returned. | 3 |
| PresetInfos | C | array | Preset list. Optional if Nums=0. |  |
| PresetInfo | C | Json Block | Preset info. See [PresetInfo](#_PresetInfo) Json Block for details. This node is optional if the value of Nums is 0. |  |

#### PresetInfo

|  |  |
| --- | --- |
| PresetInfo Json Block | {  "ID":,  "Name":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Preset ID, which is generated at the device side when a preset is created. The preset ID is numbered in sequence from 1 and the maximum is 255. | 1 |
| Name | M | string | Preset name. String length range: [1,64] (the value obtained by back end devices is the default value; configuration of preset name is currently not supported) | "Preset1" |

#### PTZCapabilityInfo

|  |  |
| --- | --- |
| PTZCapInfo Json Block | {  "IsSupportPTZ": ,  "IsSupportRecordPatrol":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| IsSupportPTZ | M | boolean | Whether supports PTZ:  0: No  1: Yes | 0 |
| IsSupportRecordPatrol | M | boolean | Whether supports recorded patrol:  0: No  1: Yes | 1 |
|  | | | | |

#### PTZZoomAreaInfo

|  |  |
| --- | --- |
| PTZZoomAreaInfoList Json Block | {  "CenterPointX":,  "CenterPointY":,  "LengthX":,  "LengthY":,  "Width": ,  "Height":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| CenterPointX | M | unsigned long | Horizontal coordinate of center point of zoom area, which does not exceed Width. | 500 |
| CenterPointY | M | unsigned long | HeightVertical coordinate of center point of zoom area, which does not exceed Height. | 500 |
| LengthX | M | unsigned long | Length of zoom area. | 0 |
| LengthY | M | unsigned long | Width of zoom area. | 0 |
| Width | M | unsigned long | Actual length of playing window. | 1300 |
| Height | M | unsigned long | Actual width of playing window. | 1300 |

#### PatrolInfoList

|  |  |
| --- | --- |
| PatrolInfoList Json Block | {  "Num": ,  "PatrolInfos": [<[PatrolInfo](#_PatrolInfo_1)>, …]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | Number of patrol routes configured. | 4 |
| PatrolInfos | C | array | Patrol route info. |  |

#### PatrolInfo

|  |  |
| --- | --- |
| PatrolInfo Json Block | {  "ID":,  "Name": ,  "Num": ,  "Actions": [<[Action](#_Action)>, …]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Patrol route ID, starts from 0. Set using the obtained route ID. | 0 |
| Name | C | string | Patrol route name.  IPCOnly used by IPC and VMS. |  |
| Num | M | unsigned long | Number of actions in PTZ operation. | 2 |
| Actions | M | array | See [Action](#_Action) Json Block for details. |  |

#### Action

|  |  |
| --- | --- |
| Action Json Block | {  "ID":,  "Type":,  "Para1: ,  "Para2: ,  "Para3: ,  "PresetID":,  "Duration":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | C | unsigned long | Starts from 0, max: (Num- 1) in [PatrolInfo](#_PatrolInfo_1).  Supported by NVR and VMS. | 0 |
| Type | M | unsigned long | Action type:  0x0302: Zoom in  0x0304: Zoom out  0x0402: Turn up  0x0404: Turn down  0x0502: Turn right  0x0504: Turn left  0x0602: Turn to preset  0x0702: Turn upper left  0x0704:  0x0802: Turn upper right  0x0804: Turn lower right  0x0901: Stop  NVR and VMS only support 0x0602. | 0 |
| Para1 | C | unsigned long | PTZ command parameter 1, pan speed, range: [1,9], only for use by 0x4020x804, and pan speed is 0 when moving up or down.  Optional if PTZCmd0x4020x804.  Preset speed uses this field. | 5 |
| Para2 | C | unsigned long | PTZ command parameter 2, tilt speed, range: [1,9], only for use by 0x402-0x804, and tilt speed is 0 when moving left or right.  Mandatory if PTZCmd is 0x402 or 0x404. When PTZCmd is 0x702-0x804, Para1 will be used; Para2 is null. | 5 |
| Para3 | C | unsigned long | Type0x03020x0304  Lens zoom ratio\*100  Mandatory if Type= 0x0302 or 0x0304. | 2 |
| PresetID | C | unsigned long | Preset ID.  Optional if Type0x0602. | 1 |
| Duration | C | unsigned long | Duration of action when Type0x0602. 0 means a lasting action, range: [0,1800000], unit: ms.  As NVR and VMS only support precision to seconds, an integer number of seconds must be set for NVR and VMS and then converted into milliseconds (ms). | 10 |

#### RecordedPatrolInfoList

|  |  |
| --- | --- |
| RecordedPatrolInfoList  Json Block | {  "Num": ,  "RecordedPatrolInfos": [<[RecordedPatrolInfo](#_RecordedPatrolInfo_1)>]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | Number of recorded patrol routes. | 1 |
| RecordedPatrolInfos | C | array | Info about recorded patrol route. |  |

#### RecordedPatrolInfo

|  |  |
| --- | --- |
| RecordedPatrolInfo Json Block | {  "ID": ,  "Name":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | ID of recorded patrol route, set using the obtained ID. | 1 |
| Name | C | string | Name of recorded patrol route.  Supported by VMS. | “PatrolInfo1” |

### PTZCtrl

#### /LAPI/V1.0/Channels/<ID>/PTZ/PTZCtrl

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/PTZCtrl |
| **Description** | It is used control PTZ of the specified channel. |
| **Method** | PUT |
| **Input Data** | [PTZCommandInfo](#_PTZCommandInfo) |
| **Success Return Data** | None |
| **Note** | For IPC，ID is fixed to 0  IPC & NVR & VMS |
|  |  |

### Presets

#### /LAPI/V1.0/Channels/<ID>/PTZ/Presets

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Presets |
| **Description** | It is used to get preset list info of the specified video input channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [PresetInfoList](#_PresetInfoList) |
| **Note** | For IPC，ID is fixed to 0  IPC & NVR & VMS |
|  |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Presets |
| **Description** | It is used to add a preset for the specified video input channel. |
| **Method** | POST |
| **Input Data** | [PresetInfo](#_PresetInfo) |
| **Success Return Data** | None |
| **Note** | For IPC，ID is fixed to 0  IPC & VMS |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/PTZ/Presets/<ID>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Presets/<ID> |
| **Description** | It is used to modify the specified preset for the specified video input channel. |
| **Method** | PUT |
| **Input Data** | [PresetInfo](#_PresetInfo) |
| **Success Return Data** | None |
| **Note** | For IPC，ID of channel is fixed to 0  IPC & NVR |
|  |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Presets/<ID> |
| **Description** | It is used to delete a preset for the specified video input channel. |
| **Method** | DELETE |
| **Input Data** | None |
| **Success Return Data** | None |
| **Note** | For IPC，ID of channel is fixed to 0  IPC & NVR & VMS |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/PTZ/Presets/<ID>/Goto

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Presets/<ID>/Goto |
| **Description** | It is used to steer the PTZ unit of the specified channel to the specified preset. |
| **Method** | PUT |
| **Input Data** | None |
| **Success Return Data** | None |
| **Note** | For IPC，ID of channel is fixed to 0  IPC & NVR & VMS |
|  |  |

### Capabilities

#### /LAPI/V1.0/Channels/<ID>/PTZ/Capabilities

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Capabilities |
| **Description** | Query PTZ configuration capability. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [PTZCapabilityInfo](#_PTZCapabilityInfo) |
| **Note** | IPC & NVR |
|  |  |

### AreaZoom

#### /LAPI/V1.0/Channels/<ID>/PTZ/AreaZoomIn

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/AreaZoomIn |
| **Description** | Drag mouse to zoom in for specified channel. |
| **Method** | PUT |
| **Input Data** | [PTZZoomAreaInfo](#_PTZZoomAreaInfo) |
| **Success Return Data** | None |
| **Note** | IPC & NVR & VMS |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/PTZ/AreaZoomOut

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/ AreaZoomOut |
| **Description** | Drag mouse to zoom out for specified channel. |
| **Method** | PUT |
| **Input Data** | [PTZZoomAreaInfo](#_PTZZoomAreaInfo) |
| **Success Return Data** | None |
| **Note** | IPC & NVR & VMS |
|  |  |

### Patrols

#### /LAPI/V1.0/Channels/<ID>/PTZ/Patrols/<ID>/Start

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Patrols/<ID>/Start |
| **Description** | Start patrol with specified route of specified channel. |
| **Method** | PUT |
| **Input Data** | None |
| **Success Return Data** | None |
| **Note** | ID here is the patrol ID  IPC & NVR & VMS |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/PTZ/Patrols/<ID>/Stop

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Patrols/<ID>/Stop |
| **Description** | Stop patrol with specified route of specified channel. |
| **Method** | PUT |
| **Input Data** | None |
| **Success Return Data** | None |
| **Note** | ID here is the patrol ID  IPC & NVR & VMS |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/PTZ/Patrols/<ID>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Patrols/<ID> |
| **Description** | Get info about specified patrol route of specified channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [PatrolInfo](#_PatrolInfo_1) |
| **Note** | IPC & NVR & VMS。 |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Patrols/<ID> |
| **Description** | Modify info about specified patrol route of specified channel. |
| **Method** | PUT |
| **Input Data** | [PatrolInfo](#_PatrolInfo_1) |
| **Success Return Data** | None |
| **Note** | IPC & NVR & VMS |
|  |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Patrols/<ID> |
| **Description** | Delete info about specified patrol route of specified channel. |
| **Method** | DELETE |
| **Input Data** | None |
| **Success Return Data** | None |
| **Note** | ID here is the patrol ID  IPC & VMS |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/PTZ/Patrols

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Patrols |
| **Description** | Get info about all preset patrol routes of specified channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [PatrolInfoList](#_PatrolInfoList) |
| **Note** | IPC & NVR & VMS |
|  |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Patrols |
| **Description** | Modify info about preset patrol routes of specified channel. |
| **Method** | PUT |
| **Input Data** | [PatrolInfoList](#_PatrolInfoList) |
| **Success Return Data** | None |
| **Note** | NVR |
|  |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/Patrols |
| **Description** | Add info about preset patrol routes of specified channel. |
| **Method** | POST |
| **Input Data** | [PatrolInfo](#_PatrolInfo_1) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/PTZ/RecordedPatrols/<ID>/Start

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/RecordedPatrols/<ID>/Start |
| **Description** | Start recorded patrol with specified route for specified channel. |
| **Method** | PUT |
| **Input Data** | None |
| **Success Return Data** | None |
| **Note** | ID here is the recorded patrol ID  IPC & NVR & VMS |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/PTZ/RecordedPatrols/<ID>/Stop

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/RecordedPatrols/<ID>/Stop |
| **Description** | Stop recorded patrol with specified route for specified channel. |
| **Method** | PUT |
| **Input Data** | None |
| **Success Return Data** | None |
| **Note** | ID here is the recorded patrol ID  IPC & NVR & VMS |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/PTZ/RecordedPatrols/<ID>/StartRecord

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/ RecordedPatrols /<ID>/StartRecord |
| **Description** | Start recording patrol route for specified channel. |
| **Method** | PUT |
| **Input Data** | None |
| **Success Return Data** | None |
| **Note** | ID here is the recorded patrol ID  IPC & NVR & VMS |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/PTZ/RecordedPatrols/<ID>/StopRecord

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/RecordedPatrols/<ID>/StopRecord |
| **Description** | Stop recording patrol route for specified channel. |
| **Method** | PUT |
| **Input Data** | None |
| **Success Return Data** | None |
| **Note** | ID here is the recorded patrol ID  IPC & NVR & VMS |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/PTZ/RecordedPatrols/<ID>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/RecordedPatrols/<ID> |
| **Description** | Delete info about specified recorded patrol of specified channel. |
| **Method** | DELETE |
| **Input Data** | None |
| **Success Return Data** | None |
| **Note** | ID here is the recorded patrol ID  IPC & VMS |
|  |  |

#### /LAPI/V1.0/Channels/<ID>/PTZ/RecordedPatrols

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/PTZ/RecordedPatrols |
| **Description** | Get info about recorded patrol route of specified channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [RecordedPatrolInfoList](#_RecordedPatrolInfoList) |
| **Note** | IPC & NVR |
|  |  |

## Media

### Json Block

#### RecordList

|  |  |
| --- | --- |
| RecordList Json Block | {  "Total": ,  "Offset": ,  “Nums”: ,  “RecordInfos”: [<[RecordInfo](#_RecordInfo)>, [<[RecordInfo](#_RecordInfo)>, …]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Total | C | unsigned long | If query on the paged search, system will return the total number of recoding found IPC supported | 100 |
| Offset | C | unsigned long | 本次查询从第几个开始查询。  Limit和Offset参数用于分页查询机制，同时使用或者同时缺省。  缺省表示：将返回最多100条查询到的录像记录。  Shows the start number of this search.  Limit and Offser should use or be null in the same time when they are used in the paged search.  Null means to return maxinum 100 records.  IPC supported | 10 |
| Nums | M | unsigned long | Number of recordings found. | 4 |
| RecordInfos | C | Array | Nums0Recording info list. Optional if Nums=0. |  |
| RecordInfo | M | Json Block |  | Refer to [RecordInfo](#_RecordInfo) Json Block |

#### RecordInfo

|  |  |
| --- | --- |
| RecordInfo | {  “FileName”:“Abc123”,  “RecordType”: 4,  “Begin”: ,  “End”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Ranger** | **Type** | **Description** | **Example** |
| FileName | M | string | Recording file name, range: [0, 64]. | Abc123 |
| RecordType | M | unsigned long | Type of recording to query, handled by the order of digit (digits from left to right: 0-31). Each digit represents a recording type. The value will be converted into ULong data type after the corresponding digit is assigned a value.  0:Manual recording  1:(NVR)Common alarm (currently not supported by NVR)  2:Motion detection  3:Alarm input  4:Video loss  5:Audio detection  6:(NVR)Transaction info (currently not supported by NVR)  7:Scheduled recording  8:Face detection  9:Cross line detection  10:Intrusion detection | Query motion detection recording:00000000000000000000000000000100 = 4，Fill in with 4  Query motion detection and face detection recording.  00000000000000000000000100000100 = 260  Fill in with 260. |
| Begin | M | unsigned long | Start time of recording file in UTC, unit: **sec**. | 1454428800 |
| End | M | unsigned long | End time of recording file in UTC, unit: **sec**. | 1454435960 |

#### RecordChannelInfoList

|  |  |
| --- | --- |
| RecordChannelInfoList Json Block | {  "Num": ,  "ChannelInfoList": [<[ChanelInfo](#_VideotapeInfo) >, [<ChanelInfo >, …]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | Input video channel number | 3 |
| ChannelInfoList | C | array | Input video channel number list, required when Num > 0 | - |
| ChanelInfo | C | Json Block | Info of video channel, require when Num > 0 | See [ChanelInfo](#_VideotapeInfo) Json Block |

#### ChanelInfo

|  |  |
| --- | --- |
| ChanelInfo Json Block | {  "ChannelID": ,  "StreamID": ,  "Begin": ,  "End":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ChannelID | M | unsigned long | ID of record video input channel under inquiry | 1 |
| StreamID | M | unsigned long | Record stream ID under inquiry,  0: main stream；  1: second stream；  2: third stream. | 0 |
| Begin | M | unsigned long | Start time of recording file in UTC, unit: **sec**. | 1454428800 |
| End | M | unsigned long | End time of recording file in UTC, unit: **sec**. | 1454435960 |

#### RecordFileList

|  |  |
| --- | --- |
| RecordFileList Json Block | {  "Num": ,  “RecordFileInfoList”: [<[RecordFileInfo](#_RecordFileInfo) >, [<RecordFileInfo >, …]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | Inquiried record file number | 4 |
| RecordFileInfos | C | array | Record file info list, required when Num is 0 | - |
| RecordFileInfo | C | Json Block | Record file info, required when Num is 0 | See [RecordFileInfo](#_RecordFileInfo) Json Block |

#### RecordFileInfo

|  |  |
| --- | --- |
| RecordFileInfo Json Block | {  “ChannelID”: ,  “StreamID”: ,  “Begin”: ,  “End”: ,  “FileSizeForRecord”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ChannelID | M | unsigned long | ID of video record channel under inquiry | 1 |
| StreamID | M | unsigned long | Record stream ID under inquiry,  0: main stream；  1: second stream；  2: third stream. | 0 |
| Begin | M | unsigned long | Start time of recording file in UTC, unit: **sec**. | 1454428800 |
| End | M | unsigned long | End time of recording file in UTC, unit: **sec**. | 1454435960 |
| FileSizeForRecord | M | unsigned long | Size fo recording file , unit: KB | 1024 |

#### VideoCapabilityInfo

|  |  |
| --- | --- |
| **VideoCapabilityInfo Json Block** | {  “IsSupportCfg”: ,  “IsSupportSmoothLevel”: ,  “IsSupportImageFormat”: ,  “EncodeFormatNum“: ,  “EncodeFormatList”: ,  “MinIFrameInterval”: ,  “MaxIFrameInterval”: ,  “StreamCapabilityNum”: ,  “StreamCapabilityList”:[<[StreamCapability](#_StreamCapability)>, <[StreamCapability](#_StreamCapability)> …]  “VideoModeNum”: ,  “VideoModeInfoList”: [<[VideoModeInfo](#_VideoModeInfo)>, <[VideoModeInfo](#_VideoModeInfo)> … ]  "GOPTypeNum":,  "GOPTypeList ":[]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| IsSupportCfg | M | boolean | Capability to configure  0: disabled  1: enabled | 1 |
| IsSupportSmoothLevel | M | boolean | Caability of video smooth  0: disabled  1: enabled | 0 |
| IsSupportImageFormat | M | boolean | Capability of image format  0: disabled  1: enabled | 0 |
| EncodeFormatNum | M | unsigned long | Encode format number |  |
| EncodeFormatList | C | array | Encode format list  0：MJPEG  1：H.264  2：H.265  3：JPEG | [1,2] |
| MinIFrameInterval | M | unsigned long | Minimum GOP | 5 |
| MaxIFrameInterval | M | unsigned long | Maximum GOP | 250 |
| StreamCapabilityNum | M | unsigned long | Stream number capability |  |
| StreamCapabilityList | C | array | Stream capability list |  |
| StreamCapability | M | Json Block | Stream capability info, see [StreamCapability](#_StreamCapability) Json Block |  |
| VideoModeNum | M | unsigned long | Number of video mode supported | 3 |
| VideoModeInfoList | C | Json Block | Video mode list |  |
| VideoModeInfo | M | Json Block | Video mode info, see [VideoModeInfo](#_VideoModeInfo_1) Json Block |  |
| GOPTypeNum | M | unsigned long | Number of GOP type |  |
| GOPTypeList | C | array | GOP type list,  0: IP  1: IBP  2: IBBP  3: I  Optional when GOPTypeNum is 0 | [0, 3] |

#### StreamCapability

|  |  |
| --- | --- |
| **StreamCapability Json Block** | {  “ID”: ,  “ResolutionNum”: ,  “ResolutionCapabilityList”:[<[ResolutionCapability](#_ResolutionCapability)>, <[ResolutionCapability](#_ResolutionCapability)>…]  "MaxFrameRate":,  "MaxMJPEGFrameRate":  “SmartEncode”:<[SmartEncodeInfo](#_SmartEncodeInfo)>  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Stream ID:  0Main stream  1Sub stream  2Third stream | 3 |
| ResolutionNum | M | unsigned long | Number of supported resolutions. | 4 |
| ResolutionCapabilityList | M | array | Supported resolutions.  Relations of resolution:  Main stream>= Sub stream>= Third stream |  |
| ResolutionCapability | M | Json Block | Detailed resolution info.  See [ResolutionCapability](#_ResolutionCapability) Json Block |  |
| MaxFrameRate | C | unsigned long | Maximum frame rate of current stream.  Supported by IPC. | 25 |
| MaxMJPEGFrameRate | C | unsigned long | MJPEGMaximum frame rate of current MJPEG stream.  Supported by IPC. | 5 |
| SmartEncode | M | Json Block | Extended image encoding format.  See [SmartEncodeInfo](#_SmartEncodeInfo) Json Block |  |

#### VideoModeInfo

|  |  |
| --- | --- |
| **VideoModeInfo Json Block** | {  “Resolution”: {  “Width”: ,  “Height”:  },  “FrameRate”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Resolution | M | Json Block | image resolution |  |
| Width | M | unsigned long | Width of image | 1920 |
| Height | M | unsigned long | Height of image | 1080 |
| FrameRate | M | unsigned long | Maximum frame rate of current stream. | 25 |

#### ResolutionCapability

|  |  |
| --- | --- |
| **ResolutionCapability Json Block** | {  “Width”: ,  “Height”: ,  “MinBitRate”: ,  “MaxBitRate”: ,  “DefaultBitRate”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Width | M | unsigned long | Image width. | 0 |
| Height | M | unsigned long | Image height. | 1 |
| MinBitRate | M | unsigned long | Minimum bit rate. | 128 |
| MaxBitRate | M | unsigned long | Maximum bit rate. | 16384 |
| DefaultBitRate | C | unsigned long | Default bit rate.  upported by NVR. | 2048 |

#### SmartEncodeInfo

|  |  |
| --- | --- |
| **SmartEncodeInfo Json Block** | {  “H264SmartEncodeModeNum”: ,  “H264SmartEncodeModeList”: ,  “H265SmartEncodeModeNum”: ,  “H265SmartEncodeModeList”: ,  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| H264SmartEncodeModeNum | M | unsigned long | Number of smart H.264 encoding formats supported. | 2 |
| H264SmartEncodeModeList | M | array | Extended H.264 encoding format:  0.Off  1.Basic(basic smart encoding mode)  2.Advanced(advanced smart encoding mode) | [1,2] |
| H265SmartEncodeModeNum | M | unsigned long | Number of smart H.265 encoding formats supported. | 2 |
| H265SmartEncodeModeList | M | array | Extended H.265 encoding formats:  0.off  1.Basic (basic smart encoding format)  2.Advanced (advanced smart encoding format) | [1,2] |

#### AudioCapabilityInfo

|  |  |
| --- | --- |
| **AudioCapabiliInfo**  **Json Block** | {  "AudioInNum": ,  "AudioInModeInfoList": [<[AudioModeInfo](#_AudioModeInfo)>, ...],  "AudioInEncodeFormatNum":,  "AudioInEncodeFormatInfoList": [<[EncodeFormatInfo](#_EncodeFormatInfo_1)>,...]  "SerialInNum": ,  "SerialInModeInfoList":[<[AudioModeInfo](#_AudioModeInfo)>, ...]  "SerialInEncodeFormatNum": ,  "SerialInEncodeFormatInfoList":[<[EncodeFormatInfo](#_EncodeFormatInfo)>, ...]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| AudioInNum | M | unsigned long | Number of audio inputs supported. | 2 |
| AudioInModeInfoList | C | array | List of audio input modes. | --- |
| [AudioModeInfo](#_AudioModeInfo) | M | Json Block | Audio input mode. See [AudioModeInfo](#_AudioModeInfo) Json Block for details. | --- |
| AudioInEncodeFormatNum | M | unsigned long | Number of supported encoding formats for audio input. | 3 |
| AudioInEncodeFormatInfoList | C | array | List of supported encoding formats for audio input. | --- |
| EncodeFormatInfo | M | Json Block | Info about supported encoding formats. See [EncodeFormatInfo](#_EncodeFormatInfo) Json Block for details. | --- |
| SerialInNum | M | unsigned long | Number of serial port inputs supported.  Used to connect external digital sound pickup. | 1 |
| SerialInModeInfoList | C | array | List of serial port input modes. | --- |
| SerialInEncodeFormatNum | M | unsigned long | Number of supported encoding formats for serial port input. |  |
| SerialInEncodeFormatInfoList | M | array | List of supported encoding formats for serial port input. See [EncodeFormatInfo](#_EncodeFormatInfo) Json Block for details. |  |

#### EncodeFormatInfo

|  |  |
| --- | --- |
| **EncodeFormatInfo**  **Json Block** | {  "Type": ,  "Num",  "SampleList":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Type | M | unsigned long | Audio encoding format types:  0 : MPEG1  1 : G.711A  2 : G.711U  3 : ADPCM  4 : OGG  5 : Auto  6 : AAC-LC | 2 |
| Num | M | unsigned long | Number of sampling rates supported. | 3 |
| SampleList | M | array | Sampling rate list:  0 : 8KHz  1 : 16KHZ  2 : 22.05KHZ  3 : 24KHZ  4 : 32KHZ  5 : 44.1KHZ  6 : 48KHZ  7 : 96KHZ | [1, 2, 6] |

#### AudioModeInfo

|  |  |
| --- | --- |
| **AudioModeInfo**  **Json Block** | {  "Channel": ,  "ModeNum",  "ModeList":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Channel | M | unsigned long | Audio channel ID. | 2 |
| ModeNum | M | unsigned long | Number of audio modes. | 2 |
| ModeList | C | array | Supported audio input channel modes. Optional if ModeNum=0.  0: LINE  1: MIC  2: RS485 digital sound pickup. | [0, 1] |

#### VideoStreamInfoList

|  |  |
| --- | --- |
| VideoStreamInfoList Json Block | {  "Num":,  "VideoStreamInfos":[<[VideoStreamInfo](#_VideoStreamInfo)>,<[VideoStreamInfo](#_VideoStreamInfo)>,……]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | Number of video streams. | 1 |
| VideoStreamInfos | C | array | Video stream info list. Optional if Num=0. | - |
| VideoStreamInfo | M | Json Block | Video stream info. See [VideoStreamInfo](#_VideoStreamInfo) Json Block for details. | - |

#### VideoStreamInfo

|  |  |
| --- | --- |
| VideoStreamInfo Json Block | {  "ID":,  "MainStreamType":,  "Enabled":,  "VideoEncodeInfo":<[VideoEncodeInfo](#_VideoEncodeInfo_1)>  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Video stream ID, starts from 0.  0:Main stream  1:Sub stream  2:Third stream | 0 |
| MainStreamType | C | unsigned long | 0 :Main scheduled stream  1: Main event stream  Effective if ID=0.  Only used by NVR. | 0 |
| Enabled | M | boolean | Whether encoding is enabled for video stream:  0:No  1:Yes | 1 |
| VideoEncodeInfo | M | Json Block | Video encoding parameter info. See for [VideoEncodeInfo](#_VideoEncodeInfo_1) Json Block details. | - |

#### VideoEncodeInfo

|  |  |
| --- | --- |
| VideoEncodeInfo Json Block | {  "EncodeFormat":,  "Resolution": {  "Width":,  "Height":  },  "BitRate":,  "BitRateType":,  "FrameRate":,  "GOPType":  "IFrameInterval":,  "ImageQuality":,  "SmoothLevel":,  "SVCMode":,  "SmartEncodeMode":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| EncodeFormat | M | unsigned long | Video encode format info  0：MJPEG  1：H.264  2：H.265 | 2 |
| Resolution | M | Json Block | Resolution |  |
| Width | M | unsigned long | Width | 1920 |
| Height | M | unsigned long | Height | 1080 |
| BitRate | M | unsigned long | Bit rate | 4096 |
| BitRateType | M | unsigned long | Bit rate type  0：CBR  1：VBR | 0 |
| FrameRate | M | unsigned long | Frame rate | 25 |
| GOPType | C | unsigned long | GOP type  0：IP  1：IBP  2：IBBP  3：I  IPC supported | 0 |
| IFrameInterval | M | unsigned long | GOP，range is determined by the capability | 50 |
| ImageQuality | C | unsigned long | Image quality, range: [1, 9] (9 for best).  Effective only when BitrateType=VBR. Can be absent when ineffective. | 9 |
| SmoothLevel | M | unsigned long | Smooth level, range: [1,9] (1 for lowest). | 5 |
| SVCMode | C | boolean | SVC configuration  0：off  1：on  IPC supported | 0 |
| SmartEncodeMode | M | unsigned long | Smart encoding mode:  0：off  1：Basic(basic smart encoding mode  2：Advanced(advanced encoding mode) | 2 |

#### AudioInputCfg

|  |  |
| --- | --- |
| **AudioInputCfg Json Block** | {  "IsMute":,  "Type":,  "EncodeFormat ":,  "SampleRate":,  "InputGain":,  "NoiseReduction": {  "Enabled":,  "Mode":,  "Strength":  }  "AudioInputNum":  "AudioInputList": [<[AudioInput](#_AudioInput)>,<[AudioInput](#_AudioInput)>,......],  "SerialInputNum":  "SerialIInputList": [<[AudioInput](#_AudioInput)>......],  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| IsMute | M | boolean | Whether sound is mute:  0:No  1:Yes | 0 |
| Type | M | unsigned long | Collection type:  0:Audio input  1:Serial port |  |
| EncodeFormat | M | unsigned long | Audio encoding format. See definitions in [EncodeFormatInfo](#_EncodeFormatInfo). | 2 |
| SampleRate | M | unsigned long | Audio sampling rate. See definition in [EncodeFormatInfo](#_EncodeFormatInfo). | 0 |
| InputGain | M | unsigned long | Audio input gain, range: [0,255]. | 128 |
| NoiseReduction | M | Json Block | Noise reduction info. | 1 |
| Enabled | M | unsigned long | Whether noise reduction is enabled:  0: No  1: Yes | 0 |
| Mode | O | unsigned long | Reserved, currently not in use. |  |
| Strength | O | unsigned long | Reserved, currently not in use. | 1 |
| AudioInputNum | M | unsigned long | Number of audio inputs. |  |
| AudioInputList | C | array | Audio input collection info. Mandatory if Type=0. |  |
| AudioInput | M | Json Block | Audio input channel info.  see [AudioInput](#_AudioInput) Json Block for details. |  |
| SerialIInputNum | M | unsigned long | Number of serial port inputs. | 1 |
| SerialInputList | C | array | Audio collection through serial port. Mandatory if Type=1. |  |

#### AudioInput

|  |  |
| --- | --- |
| **AudioInput Json Block** | {  "ID":,  "Enabled":,  "Mode":,  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Audio channel ID, starts from 1. | 1 |
| Enabled | M | unsigned long | Whether audio input channel is enabled. | 1 |
| Mode | M | unsigned long | Audio input channel modes:  0: LINE  1: MIC  2:RS485 digital sound pickup. | 0 |

#### OSDCapabilities

|  |  |
| --- | --- |
| [OSDCap](#_AlarmCap)abilities Json Block | {  "IsSupportCfg": ,  "SupportedOSDContentTypeNum": ,  "SupportedOSDContentTypeList": ,  "IsSupportFontSizeCfg": ,  "IsSupportFontColorCfg": ,  "MaxAreaNum": ,  "MaxOSDNum": ,  "MaxPerAreaOSDNum": ,  "SupportedTimeFormatNum": ,  "SupportedTimeFormatList": ,  "SupportedDateFormatNum": ,  "SupportedDateFormatList":  } |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | | **Example** |
| IsSupportCfg | M | boolean | Whether configuration is supported:  0No  1Yes | | 0 |
| SupportedOSDContentTypeNum | M | unsigned long | Number of supported OSD content types: | | 6 |
| SupportedOSDContentTypeList | M | array | Configurable OSD content types:  0 : Not use  1 : Custom  2 : Time&date  3 : PTZ controller  4 : PTZ coordinates  5 : Patrol info  6 : Zoom info  7 : Preset info  8 : Alarm info  9 : Encoding info  10 : OSDSerial port OSD  11 : PTZ position info  12 : Channel name  13 : Debug OSD  14 : People counting OSD  15 : Network interface OSD  16 : Time  17 : Date  18 : Ultra smart OSD  19 : Battery OSD  20 : Rolling caption OSD  21: LOGO OSD  22: Vehicle flow (LPR)  23: Non-vehicle flow (LPR)  24: Pedestrian flow (LPR)  25: Number of INFOOSD types | | [1,2,3,10,16,17] |
| IsSupportFontSizeCfg | M | boolean | | Whether OSD size can be configured. | 0 |
| IsSupportFontColorCfg | M | boolean | Whether OSD color can be configured. | | 0 |
| MaxAreaNum | M | unsigned long | Maximum number of OSD areas. | | 8 |
| MaxOSDNum | M | unsigned long | Maximum number of OSDs. | | 8 |
| MaxPerAreaOSDNum | M | unsigned long | Maximum number of OSDs in each area. | | 8 |
| SupportedTimeFormatNum | M | unsigned long | Number of time OSD formats. | | 6 |
| SupportedTimeFormatList | M | array | Time OSD formats:  h/H=12/24 Hour; tt=A.M. or P.M.; mm=Minute; ss=Second; xxx=millisecond  0: HH:mm:ss()  1: hh:mm:ss t.t.()  2: hh:mm:ss tt()  3: tt hh:mm:ss(())  4: hh:mm:ss(())  5: HH:mm:ss.xxx()  6: hh:mm:ss.xxx tt() | | 7 |
| SupportedDateFormatNum | M | unsigned long | Number of date OSD formats. | | 6 |
| SupportedDateFormatList | M | unsigned long | Date OSD formats:  MMMMJuneddddMon  0-5  0: yyyy-MM-dd  1: MM-dd-yyyy  2: yyyyMMdd  3: MMddyyyy  4: yyyyMMdd X  5: MMddyyyy X  100-108  100: dd/MM/yyyy  101: MM/dd/yyyy  102: dd MMMM, yyyy  103: MMMM dd, yyyy  104: dddd, dd MMMM, yyyy  105: dddd, MMMM dd, yyyy  106: yyyy/MM/dd  107: yyyy, MMMM dd  108: dddd, yy, MM dd | | [0,1,2,3,4,5] |

#### OSDContentStyle

|  |  |
| --- | --- |
| OSDContentStyle Json Block | {  "FontStyle":,  "FontSize":,  "Color":,  "DateFormat":,  "TimeFormat":,  "FontAlignList":,  "Margin":  } |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | | **Example** |
| FontStyle | C | unsigned long | Font style  0 : Background  1 : Stroke  2 : Hollow  3 : Normal  IPCSupported by IPC. | | 3 |
| FontSize | M | unsigned long | | Font size.  0 : X-large  1 : Large  2 : Medium  3 : Small | 1 |
| Color | M | unsigned long | Color (decimal)  A RGB color is calculated based on the combination of relative brightness of three primary colors: red, gree and blue, with value range 0-255 for each primary color.  Formula: (65536\*Red + 256\*Green + Blue). To apply color #665432, apply 65536 \* 102 (66 in decimal) + 256 \* 84(54 in decimal) + 50(32 in decimal) = 6706226(#665432 in decimal).  To apply RGB(102, 84, 50), apply 65536 \* 102 + 256 \* 84 + 50 = 6706226. | |  |
| DateFormat | M | unsigned long | Date format (see [OSDCapabilities](#_OSDCapabilities) for corresponding enumerations). | | 7 |
| TimeFormat | M | unsigned long | Time format (see [OSDCapabilities](#_OSDCapabilities) for corresponding enumerations). | | 2 |
| FontAlignList | C | array | Font alignment in area,  0 : Left aligned  1 : Right aligned  Supported by IPC | | [0,1,1,0,0,0,0,0] |
| Margin | C | unsigned long | Number of character(s) in the margin:  0:None  1:character width  2:character width  Supported by IPC. | | 2 |

#### OSDContent

|  |  |
| --- | --- |
| **OSDContent Json Block** | {  "Num": ,  "ContentList":[<[OSDContentInfo](#_OSDContentInfo)> ,< [OSDContentInfo](#_OSDContentInfo)>,......} |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | Number of OSD contents. | 8 |
| ContentList | C | array | OSD content info list.  Optional if Num=0. | - |
| OSDContentInfo | M | Json Block | OSD content info.  Note: For NVR, channel name always uses the 1st OSD, date&time always uses the 2nd; people counting (if supported, determined by querying capabilities) always uses the 3rd; and other scene names always use the remaining OSDs respectively.  See [OSDContentInfo](#_OSDContentInfo) Json Block for details. | - |

#### OSDContentInfo

|  |  |
| --- | --- |
| **OSDContentInfo Json Block** | {  "ID": ,  "Enabled": ,  "Num": ,  "ContentInfo": [{  "ContentType": ,  "Value":  },......]  "Area": {  "TopLeft": {  "X": ,  "Y":  }  }  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | OSD ID, range: [0,7]. | 0 |
| Enabled | M | boolean | Whether OSD is enabled:  0: No  1: Yes | 1 |
| Num | M | unsigned long | Number of OSDs in an area, subject to MaxPerAreaOSDNum in [OSDCapabilities](#_OSDCapabilities).  For NVR, one OSD in each area. | 2 |
| ContentInfo | M | array | OSD content info. | - |
| ContentType | M | unsigned long | OSD content types:  0 : Not use  1 : Custom  2 : Time&date  3 : PTZ controller  4 : PTZ coordinates  5 : Patrol info  6 : Zoom info  7 : Preset info  8 : Alarm info  9 : Encoding info  10 : OSDSerial port OSD  11 : PTZ position info  12 : Channel name  13 : Debug OSD  14 : People counting OSD  15 :Network interface OSD  16 : Time  17 : Date  18 : Ultra smart OSD  19 : Battery OSD  20:Rolling caption OSD  21 :LOGO OSD  22 Vehicle flow (LPR)  23 Non-vehicle flow (LPR)  24: Pedestrian flow (LPR)  25: Number of INFOOSD types | 1 |
| Value | M | string | OSD text info. | “OSD文本” |
| Area | M | Json Block | OSD area info. | - |
| TopLeft | M | Json Block | Top left coordinates of area. |  |
| X | M | unsigned long | Horizontal coordinate of OSD, range: [0,9999]. | 200 |
| Y | M | unsigned long | Vertical coordinate of OSD, range: [0,9999]. | 200 |

#### OSDList

|  |  |
| --- | --- |
| **OSDList Json Block** | {  "ContentStyle": <[OSDContentStyle](#_OSDContentStyle)> ,  "Num": ,  "ContentList":[<[OSDContentInfo](#_OSDContentInfo)> ,< [OSDContentInfo](#_OSDContentInfo)>,......} |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ContentStyle | M | Json Block | OSD style. See [OSDContentStyle](#_OSDContentStyle) Json Block for details. | - |
| Num | M | unsigned long | Number of OSD areas. | 8 |
| ContentList | C | array | Area and OSD info list. See [OSDContentInfo](#_OSDContentInfo) Json for details. Optional if Num=0. | - |

#### SnapshotURLInfo

|  |  |
| --- | --- |
| **SnapshotURLInfo Json Block** | {  "Num":,  "PicInfos":[<[PicInfo](#_PicInfo)>,......]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | Number of URL | 2 |
| PicInfos | C | array | Alarm pictures，optional when Num is 0 | - |
| PicInfo | M | Json block | Alarm picture info | See [PicInfo](#_PicInfo) Json Block |

#### PicInfo

|  |  |
| --- | --- |
| **PicInfo Json Block** | {  "URL":,  "Name":,  "Size":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| URL | M | string | Alarm picture URL，string lenth: [1,128]。 | http://206.5.3.32:80/LAPI/V1.0/Channels/1/Alarm/Snapshot |
| Name | M | string | Picture name，string lenth: [1,64]。 | ubs\_19\_0\_0\_1\_0\_1\_409\_2795 |
| Size | M | unsigned long | Picture size, unit: byte | 64164 |

#### SnapshotCapabilities

|  |  |
| --- | --- |
| SnapshotCapabilities Json Block | {  “SupportSnapshotCfg”:,  "PictureMaxSize": ,  “ResolutionNum”: ,  “ResolutionCapabilityList”:[<[SnapshotResolutionCapability](#_SnapshotResolutionCapability)>, <[SnapshotResolutionCapability](#_SnapshotResolutionCapability)>…]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| SupportSnapshotCfg | M | boolean | Whether snapshot is configurable:  0: No  1: Yes | 1 |
| PictureMaxSize | M | unsigned long | Max snapshot size allowed. | 800 |
| ResolutionNum | M | unsigned long | Number of resolutions supported. | 4 |
| ResolutionCapabilityList | C | array | Snapshot resolutions supported; not needed when ResolutionNum=0. |  |
| SnapshotResolutionCapability | M | Json Block | Detailed snapshot resolution info.  See [SnapshotResolutionCapability](#_SnapshotResolutionCapability) Json Block for details. |  |

#### SnapshotResolutionCapability

|  |  |
| --- | --- |
| SnapshotResolutionCapability Json Block | {  “Width”: ,  “Height”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Width | M | unsigned long | Image width | 1920 |
| Height | M | unsigned long | Image height | 1680 |

#### SnapshotInfo

|  |  |
| --- | --- |
| SnapshotInfo Json Block | {  “Enabled”:,  “Resolution”:{  “Width”:,  “Height”:  },  “PictureMaxSize”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Param | Requirement | Type | Description | Example |
| Enabled | M | Boolean | Whether snapshot is enabled.  0: Disabled  1: Enabled | 0 |
| Resolution | M | Json block | Snapshot resolution |  |
| Width | M | unsigned long | Snapshot width  Unit: pixel  See snapshot capability for supported snapshot resolutions. | 1920 |
| Height | M | unsigned long | Snapshot height.  Unit: pixel  See snapshot capability for supported snapshot resolutions. | 1080 |
| PictureMaxSize | M | unsigned long | Max snapshot size.  Unit: KB  This value cannot exceed the max snapshot size in snapshot capability. | 10 |

### LiveStreamURL

#### /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/<ID>/LiveStreamURL?TransType=<TransType>&TransProtocol=<TransProtocol>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/<ID>/LiveStreamURL?TransType=<TransType>&TransProtocol=<TransProtocol> | | | |
| **Description** | It is used to get the live video URL of specified stream of the specified video channel. | | | |
| **Method** | GET | | | |
| **Success Return Data** | {  "LoginName":,  "PIN":,  "URL":  } | | | |
| **Note** | Caller should get the media stream through RTSP after they get the URL  For IPC , Channel ID is 0 in fix.  IPC & NVR & VMS supported | | | |
| **Status** |  | | | |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Streams/<ID> | M | unsigned long | Video stream ID, which is numbered in sequence from 0. | 0 |
| TransType | O | unsigned long | Stream transmission type:  0:transmit  1:direct connection  VMS supported, 0 for default | 0 |
| TransProtocol | C | unsigned long | Transmission protocol:  0：TCP  1：UDP  VMS supported, corporate with TransType | 1 |
| LoginName | C | string | Device username encrypted using private algorithm; decryption library is available.  Required if VMS Transtype=1. | sdfsdf |
| PIN | C | string | Device password encrypted using private algorithm; decryption library is available.  Required if Transtype=1. | sdfsdfjskdsdfsdfjskdsdfsdfjs |
| URL | M | string | Live video URL corresponding to the specified video stream. | rtsp://192.168.0.13:554/media/video1 |

### Records

#### /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/<ID>/Records?Begin=<Begin>&End=<End>&Limit=<Limit>&Offset=<Offset>&Types=<Types>&RelationOfTypes=<RelationOfTypes>&Position=<Position>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/<ID>/Records?Begin=<Begin>&End=<End>&Limit=<Limit>&Offset=<Offset>&Types=<Types>&RelationOfTypes=<RelationOfTypes >&Position=<Position> |
| **Description** | Query info about recording file of a video stream of specified video channel during a specified time. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [RecordList](#_RecordList) |
| **Note** | IPC & NVR & VMS |
| **Status** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Ranger** | **Type** | **Description** | **Example** |
| Begin | M | unsigned long | Start time of recording query in UTC, unit: sec | 1454428800 |
| End | M | unsigned long | End time of recording query in UTC, unit: sec | 1454435960 |
| Limit | C | unsigned long | Number of recordings queried [0, 100]  Only supported by IPC. | 100 |
| Offset | C | unsigned long | The recording from which the query starts.  Only supported by IPC. | 100 |
| Types | O | unsigned long | Recording type to query, described by the order of bit (left to right: 0-31), each represents a recording type. A bit with 1 means the corresponding recording type is effective.  0: Manual recording (this type is expressed by an all-0 value because it cannot be queried together with other recording types)  BIT0: Common alarm (not supported by NVR)  BIT1: Motion detection  BIT2: Alarm input  BIT3: Video loss  BIT4: Audio detection  BIT5: Transaction info (not supported by NVR)  BIT6: Scheduled recording  BIT7: Face detection  BIT8: Cross line detection  BIT9: Intrusion detection  All recording types will be queried if this field is absent.  Only supported by NVR. | Query motion detection recording: 00000000000000000000000000000100 = 4. Fill in with 4.  Query motion detection and face detection recording:  00000000000000000000000100000100 = 260. Fill in with 260. |
| RelationOfTypes | C | unsigned long | Relation between recording types:  0: AND  1: OR  Not needed when Types is absent or with one recording type.  Required in the case of two or more recording types.  Only supported by NVR. | 1 |
| Position | O | unsigned long | Location of recording to query:  1: Local recording  2: Recording on other devices  Only supported by VMS. | 1 |

#### /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/RecordURL？Begin=<Begin>&End=<End>&Types=<Types>&RelationOfTypes=<RelationOfTypes>&Position=<Position>&SessionID=<SessionID>&TransType=<TransType>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/RecordURL?Begin=<Begin>&End=<End>&Types=<Types>&RelationOfTypes=<RelationOfTypes>&Position=<Position>&SessionID=<SessionID>&TransType=<TransType> |
| **Description** | Query the RTSP URL of specific time, record type and channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | {  “URL”: ””  } |
| **Note** | The URL that returned by this interface is only used for play and download stream that is based on RTSP  IPC & NVR & VMS |
| **Status** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Begin | O | unsigned long | Recording start time in UTC, unit: sec.  End time is required if start time exists. | 1454428800 |
| End | O | unsigned long | Recording end time in UTC, unit: sec.  Start time is required if end time exists. | 1454435960 |
| Types | O | unsigned long | Type of recording to query, described by order of BIT (from left to right 0-31), each BIT represents a recording type. BIT=1 means valid recording type.  0: Manual recording (this type cannot be mixed with other recording types and therefore is represented with all-zeros value.)  BIT0: Common alarm (currently not supported by NVR)  BIT1:Motion detection  BIT2:Alarm input  BIT3:Video loss  BIT4:Audio detection  BIT5: Transaction info (currently not supported by NVR)  BIT6:(Scheduled recording)  BIT7:(Face detection)  BIT8:(Cross line detection)  BIT9:(Intrusion detection)  All types of recordings will be queried if this field is absent. | Query motion detection recording:  00000000000000000000000000000100 = 4 Fill in with 4.  Query motion detection and face detection recording:  00000000000000000000000100000100 = 260 Fill in with 260. |
| RelationOfTypes | O | unsigned long | Relation of recording types:  0:AND  1:OR  Optional if Types is absent or there is only one recording type.  Mandatory if there are two or more recording types. | 1 |
| Position | O | unsigned long | Location of recording to play:  1:Local recording  2:Recoding on other device  VMS  Currently used by VMS only. | 1 |
| SessionID | O | unsigned long | Session ID used to forward playback streams, starts from 1.  Currently only used by VMS. | 1 |
| TransType | O | unsigned long | 0:VMS forwards  1:IPC/NVR connect directly | 0 |
| URL | M | string | RTSP URL of recording, range: [0,128] | rtsp://206.5.3.13:554/c2/replay/ |

#### /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/Records/DailyDistribution?Year=<Year>&Month=<Month>&Position=<Position>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/Records/DailyDistribution?Year=<Year>&Month=<Month>&Position=<Position> |
| **Description** | Query the recording info of specific month and channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | {  "Nums": ,  "DailyStatuses": [<Status>, <Status>, <Status>, <Status>…]  } |
| **Note** | NVR |
| **Status** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Ranger** | **Type** | **Description** | **Example** |
| Year | M | unsigned long | Year | 2016 |
| Month | M | unsigned long | Month, starts from 1. | 9 |
| Position | O | unsigned long | Location of recording to query:  1: Local recording.  2: Recording on other device.  Currently used by VMS only. | 1 |
| Nums | M | unsigned long | Actual days of a month. | 30 |
| DailyStatuses | C | Array | Daily recording status list. Optional if Nums=0. |  |
| Status | M | unsigned long | Recording status:  0:No recording  1:Event recording  2:Normal recording  Event recording has a higher priority level than normal recording. | 1 |

#### /LAPI/V1.0/Channels/Media/Video/Streams/RecordFileSize

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/Media/Video/Streams/RecordFileSize |
| **Description** | Query video record file size of specified time zone. |
| **Method** | POST |
| **Input Data** | [RecordChannelInfoList](#_RecordChannelInfoList) |
| **Success Return Data** | [RecordFileList](#_FileRecordList_1) |
| **Note** | NVR |
| **Status** |  |

### Talk

#### /LAPI/V1.0/Channels/<ID>/Media/Talk

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Talk | | | |
| **Description** | Get URL of audio for two-way audio for specified video channel. | | | |
| **Method** | GET | | | |
| **Success Return Data** | {  "URL":  } | | | |
| **Note** | After getting the URL, the caller uses RTSP to get audio stream for two-way audio.  For IPC, the ID is fixed to 0 in the URL.  IPC & VMS. | | | |
| **Status** |  | | | |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| URL | M | string | URL of live video stream corresponding to the specified audio stream. | rtsp://192.160.1.22:554/media/video1 |

### Capabilities

#### /LAPI/V1.0/Channels/<ID>/Media/Video/Capabilities

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Video/Capabilities |
| **Description** | Query video capability of specified video channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**VideoCapabilityInfo**](#_VideoCapabilityInfo) |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Media/Audio/Capabilities

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Audio/Capabilities |
| **Description** | Query audio capability of specified video channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**AudioCapabilityInfo**](#_AudioCapabilityInfo) |
| **Note** | IPC |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Media/Snapshot/Capabilities

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Snapshot/Capabilities |
| **Description** | Query snapshot capability of specified video channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**SnapshotCapabilities**](#_SnapshotCapabilities) |
| **Note** | IPC |
| **Status** |  |

### Video

#### Streams

##### /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/DetailInfos

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/DetailInfos |
| **Description** | Query info about all video streams of specified video channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**VideoStreamInfoList**](#_VideoStreamInfoList) |
| **Note** | IPC & NVR & VMS |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/DetailInfos |
| **Description** | Set info about all video streams of specified video channel. |
| **Method** | PUT |
| **Input Data** | [**VideoStreamInfoList**](#_VideoStreamInfoList) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### Mode

##### /LAPI/V1.0/Channels/<ID>/Media/Video/Mode

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Video/Mode |
| **Description** | Query image capturing mode. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**VideoModeInfo**](#_VideoModeInfo) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Video/Mode |
| **Description** | Set image capturing mode. |
| **Method** | PUT |
| **Input Data** | [**VideoModeInfo**](#_VideoModeInfo) |
| **Success Return Data** | None |
| **Note** | NVR & IPC |
| **Status** |  |

### Audio

#### /LAPI/V1.0/Media/Audio/Input

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Media/Audio/Input |
| **Description** | Get audio input parameters. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [AudioInputCfg](#_AudioInputCfg) |
| **Note** | IPC |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Media/Audio/Input |
| **Description** | Set audio input parameters. |
| **Method** | PUT |
| **Input Data** | [AudioInputCfg](#_AudioInputCfg) |
| **Success Return Data** | None |
| **Note** | IPC |
| **Status** |  |

### OSD

#### /LAPI/V1.0/Channels/<ID>/Media/OSDs/Capabilities

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/OSDs/Capabilities |
| **Description** | Query OSD configuration capability. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [OSDCapabilities](#_OSDCapabilities) |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Media/OSDs/ContentStyle

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/OSDs/ContentStyle |
| **Description** | Query OSD content style. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**OSDContentStyle**](#_OSDContentStyle) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/OSDs/ContentStyle |
| **Description** | Set OSD content style. |
| **Method** | PUT |
| **Input Data** | [**OSDContentStyle**](#_OSDContentStyle) |
| **Success Return Data** | None |
| **Note** | IPC & NVR支持。 |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Media/OSDs/Contents

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/OSDs/Contents |
| **Description** | Query OSD content. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**OSDContent**](#_OSDContent) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/OSDs/Contents |
| **Description** | SetOSD content. |
| **Method** | PUT |
| **Input Data** | [**OSDContent**](#_OSDContent) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Media/OSDs

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/OSDs |
| **Description** | Query all OSD info. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**OSDList**](#_OSDList) |
| **Note** | NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/OSDs |
| **Description** | Set all OSD info. |
| **Method** | PUT |
| **Input Data** | [**OSDList**](#_OSDList) |
| **Success Return Data** | None |
| **Note** | NVR |
| **Status** |  |

### Snapshot

#### /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/<ID>/Snapshot

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Video/Streams/<ID>/Snapshot |
| **Description** | Snapshot (not in preview). |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | Content-Length: 123131  Content-Type: image/jpeg  Accept-Ranges: bytes  图片数据 |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Channels/<ID> | M | unsigned long | ID: video channel ID | 1 |
| Streams/<ID> | M | unsigned long | ID: stream ID  0：Main stream  1：Sub stream  2：Third stream | 0 |

#### /LAPI/V1.0/Channels/<ID>/Alarm/SnapshotURL?Type=<Type>&Time=<Time>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/SnapshotURL?Type=<Type>&Time=<Time> |
| **Description** | Query URL of alarm picture |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [SnapshotURLInfo](#_SnapshotURLInfo) |
| **Note** | NVR & VMS |
| **Status** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Video input channel ID | 1 |
| Type | M | string | Alarm type:  "MotionAlarmOn": start of motion alarm  "VideoLossAlarmOn": Start of video loss alarm  "AbnormalAudioOn": start of abnormal audio alarm | "MotionAlarmOn" |
| Time | M | unsigned long | UTC time that alarm occurred, starts from seconds of 1, 1st, 1970, 0 o’clock | 1476088399 |

#### /LAPI/V1.0/IO/InputSwitches/<ID>/Alarm/SnapshotURL?Time=<Time>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/InputSwitches/<ID>/Alarm/SnapshotURL?Time=<Time> |
| **Description** | Query URL of Alarm picture that is linked by I/O input |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [SnapshotURLInfo](#_SnapshotURLInfo) |
| **Note** | NVR & VMS |
| **Status** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | ID of alarm input  NVR: the parameter is the made of channel ID\*100+alarm input number, the channel ID of the device itself is 0;  VMS: the parameter is the ID number in the configuration | 1 |
| Time | M | unsigned long | UTC time that alarm occurred, starts from seconds of 1, 1st, 1970, 0 o’clock | 1476088399 |

#### /LAPI/V1.0/Channels/<ID>/Alarm/Snapshot?Name=<Name>&Size=<Size>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/Snapshot?Name=<Name>&Size=<Size> |
| **Description** | Query snapshot picture |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | Content-Length: 44  Content-type: image/jpeg  Accept-Ranges: bytes  图片数据 |
| **Note** | Name and Size are returned by the interface of query alarm picture URL  NVR & VMS |
| **Status** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Channel ID of channel with snapshot picture | 1 |
| Name | M | string | Name of the picture, length of [1,64] | “aaa” |
| Size | M | unsigned long | Size of picture, unit: byte | 61440 |

#### /LAPI/V1.0/Channels/<ID>/Media/Snapshot/Info

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Snapshot/Info |
| **Description** | Query snapshot parameters of specified video channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [SnapshotInfo](#_SnapshotInfo) |
| **Note** | IPC |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Media/Snapshot/Info |
| **Description** | Set snapshot parameters of specified channel. |
| **Method** | PUT |
| **Input Data** | [SnapshotInfo](#_SnapshotInfo) |
| **Success Return Data** | None |
| **Note** | IPC |
| **Status** |  |

## Network

### Json Block

#### NICCapabilities

|  |  |
| --- | --- |
| **NICCapabilities Json Block** | {  "WorkModeNum": ,  "WorkModeList":,  "NICNum":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| WorkModeNum | C | unsigned long | the number of work mode of the NIC  NVR &VMS supported | 3 |
| WorkModeList | C | array | work mode list  0: multiple address set  1: network load balancing  2: network fault tolerant | [0,1,2] |
| NICNum | M | unsigned long | Number of NIC | 4 |

#### NetworkInterfaceInfoList

|  |  |
| --- | --- |
| **NetworkInterfaceInfoList Json Block** | {  "Num": ,  "NetworkInterfaceList":[<[NetworkInterfaceInfo](#_NetworkInterfaceInfo)>,<[NetworkInterfaceInfo](#_NetworkInterfaceInfo)>,…]  "DefaultRouteNIC": ,  "WorkMode":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Param | Requirement | Type | Description | Example |
| Num | M | unsigned long | Number of NIC | 1 |
| NetworkInterfaceList | C | array | NIC info list, available only when Num is 0 |  |
| NetworkInterfaceInfo | M | Json Block | detailed info of each NIC | See [NetworkInterfaceInfo](#_NetworkInterfaceInfo) Json Block |
| DefaultRouteNIC | C | unsigned long | NIC ID for the default router, required when there are multiple NICs.  NVR &VMS supported | 1 |
| WorkMode | C | unsigned long | work mode list  0: multiple address set  1: network load balancing  2: network fault tolerant  Required when there are multiple NICs.  NVR &VMS support | 0 |

#### NetworkInterfaceInfo

|  |  |
| --- | --- |
| **NetworkInterfaceInfo Json Block** | {  "ID":,  "Name":,  "WorkMode":,  "IsInnerNIC":,  "InnerNICIPAddress":,  "InnerNICNetmask":,  "InnerNICName":,  "MTU":,  "MAC":,  "NegotiationMode":,  "IPv4": {  "IPGetType":,  "PPPoE": {  "LoginName":,  "PIN":  },  "AddressNum":,  "AddressList": [  {  "Address":,  "Netmask":,  "Gateway":  }]  },  "IPv6": {  "IPGetType":,  "AddressNum":,  "AddressList": [  {  "PrefixLenth":,  "Address":,  "Gateway":  }]  }  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | NIC ID,read only  For VMS &IPC, start from :1  For NVR,it is a kind of source code and can be acquired through method etho | 1 |
| Name | C | string | name of NIC, read only  NVR &IPC supported | "eth0" |
| WorkMode | C | unsigned long | woke mode list  0: multiple address set  1: network load balancing  2: network fault tolerant  VMS required | 0 |
| IsInnerNIC | M | boolean | Estimate if it is the inner NIC when dealing with the IP address configuration of the NVR POE port. Optional  0 no  1 yes | false |
| InnerNICIPAddress | C | string | IP address of the Inner NIC, required when IsInnerNIC is 1 | “172.16.0.1” |
| InnerNICNetmask | C | string | Netmask of the Inner NIC. Read only | “255.255.0.0” |
| InnerNICName | C | string | Name of Inner NIC; Read only | “eth0” |
| MTU | M | unsigned long | Lenth of MTU, [576,1500] | 1500 |
| MAC | M | string | MAC address, read only.  Lenth of string [0,48] | "48:EA:63:17:1F:40" |
| NegotiationMode | C | unsigned long | Work mode of Network port  0:  :auto-negotiation  1: 10M FULL  2: 10M HALF  3: 100M FULL  4: 100M HALF  5: 1000M FULL  6: 10M AUTO  7: 100M AUTO  8: 1000M AUTO  IPC supported only | 0 |
| IPv4 | O | Json Block | IPv4 info, optional when device does not support IPv4 |  |
| IPGetType | C | unsigned long | method to get IP address, 3 when in IPv4:  0 for static  1 for PPPOE  2 for DHCP  PPPoE is not applicable to NVR.  IPv6  3 when IPv6:  0: for manual  1: for DHCP  2: for route advertisement  IPC supports only manual | 0 |
| PPPoE | C | Json Block | required when IPGetType is 1.  IPC &NVR supported |  |
| LoginName | C | string | PPPoE account | "TestUser" |
| PIN | C | string | PPPoE password(MD5) | "21232f297a57a5a74389" |
| AddressNum | C | unsigned long | number of IP address | 1 |
| AddressList | C | array | list of IP address |  |
| Address | C | string | IP address, length of string [0,64] | "203.5.1.82"或"2001:0:0:301::115" |
| Netmask | C | string | netmask, with length of string [0,64] | "255.255.255.0" |
| Gateway | C | string | default gateway, length of string [0,64] | "203.5.1.1"或者"2001:0:0:301::1" |
| IPv6 | C | Json Block | IPv6 info, optional when device doesn’t support IPv6 |  |
| PrefixLenth | C | unsigned long | lenth of network prefix, [3,127] | 64 |

#### DNSInfo

|  |  |
| --- | --- |
| DNSList Json Block | {  "Num": ,  "DNSList": [<[DNSAddress](#_DNSAddress)>, <[DNSAddress](#_DNSAddress)>…]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | number of DNS address | 2 |
| DNSList | C | array | List of DNS, alternate the first as the chief by default. Only supports two DNS address | “8.8.4.4” |
| DNSAddress | M | Json Block | DNS address, see [DNSAddress](#_DNSAddress) Json Block |  |

#### DNSAddress

|  |  |
| --- | --- |
| DNSAddress Json Block | {  "AddressType": ,  "IPAddress": ,  "IPv6Address":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| AddressType | M | unsigned long | type of IP address  0: IPv4  1: IPv6  3: for both IPv4 &IPv6  IPv4 supported only now | 0 |
| IPAddress | C | string | IPv4 address, lenth [0,64]  Required when AddressType is 0 or 3 | 206.5.99.17 |
| IPv6Address | C | string | IPv6 address, lenth [0,64]  Required when AddressType is 1 or 3 |  |

### Capabilities

#### /LAPI/V1.0/Network/Capabilities

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Network/Capabilities |
| **Description** | Query the capability of NIC |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [NICCapabilities](#_NICCapabilities) |
| **Notes** | NVR |
| **Status** |  |

### Interfaces

#### /LAPI/V1.0/Network/Interfaces

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Network/Interfaces |
| **Description** | Query the configuration of NICs |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**NetworkInterfaceList**](#_NetworkInterfaceInfoList) |
| **Note** | IPC & NVR & VMS |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Network/Interfaces |
| **Description** | Set the configuration of NICs |
| **Method** | PUT |
| **Input Data** | [**NetworkInterfaceList**](#_NetworkInterfaceInfoList) |
| **Success Return Data** | None |
| **Note** | IPC & NVR & VMS |
| **Status** |  |

#### /LAPI/V1.0/Network/Interfaces/<ID>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Network/Interfaces/<ID> |
| **Description** | Query the configuration of appointed NIC |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**NetworkInterfaceInfo**](#_NetworkInterfaceInfo) |
| **Note** | NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Network/Interfaces/<ID> |
| **Description** | Set the configuration of appointed NIC |
| **Method** | PUT |
| **Input Data** | [**NetworkInterfaceInfo**](#_NetworkInterfaceInfo) |
| **Success Return Data** | **None** |
| **Note** | NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Network/Interfaces

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Network/Interfaces |
| **Description** | Query and set the configuration of NICs with specified channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**NetworkInterfaceList**](#_NetworkInterfaceInfoList) |
| **Note** | NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Network/Interfaces |
| **Description** | Set the configuration of NICs with specified channel |
| **Method** | PUT |
| **Input Data** | [**NetworkInterfaceList**](#_NetworkInterfaceInfoList) |
| **Success Return Data** | **None** |
| **Note** | NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Network/Interfaces/<ID>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Network/Interfaces/<ID> |
| **Description** | Query the configuration of appointed NIC with specified channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**NetworkInterfaceInfo**](#_NetworkInterfaceInfo) |
| **Note** | NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Network/Interfaces/<ID> |
| **Description** | Set the configuration of appointed NIC with specified channel |
| **Method** | PUT |
| **Input Data** | [**NetworkInterfaceInfo**](#_NetworkInterfaceInfo) |
| **Success Return Data** | None |
| **Note** | NVR |
| **Status** |  |

### DNS

#### /LAPI/V1.0/Network/DNS

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Network/DNS |
| **Description** | Query the DNS address |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**DNSInfo**](#_DNSInfo) |
| **Note** | IPC & NVR & VMS |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Network/DNS |
| **Description** | Set the DNS address |
| **Method** | PUT |
| **Input Data** | [**DNSInfo**](#_DNSInfo) |
| **Success Return Data** | None |
| **Note** | IPC & NVR & VMS |
| **Status** |  |

## Image

### Json Block

#### EnhanceCapabilitiesInfo

|  |  |
| --- | --- |
| EnhanceCapabilitiesInfo  Json Block | {  “ImageRotationModeNum”: ,  “ImageRotationModeList”: ,  “IsSupportSharpness”:,  “IsSupport2DNoiseReduce”:,  “IsSupport3DNoiseReduce”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ImageRotationModeNum | M | unsigned long | Number of image rotation modes supported. | 3 |
| ImageRotationModeList | C | array | Image rotation mode list:  0: Normal  1: Flip Vertical  2: Flip Horizontal  3: 180°  4: 90° Clockwise  5: 90° Anti-clockwise  Not needed when ImageRotationModeNum=0. | [1,3,5] |
| IsSupportSharpness | M | boolean | Whether supports sharpness:  0: No  1: Yes | 1 |
| IsSupport2DNoiseReduce | M | boolean | Whether supports 2D noise reduction:  0: No  1: Yes | 0 |
| IsSupport3DNoiseReduce | M | boolean | Whether supports 3D noise reduction:  0: No  1: Yes  Supported by IPC and VMS. | 0 |

#### FocusInfoCapabilitiesInfo

|  |  |
| --- | --- |
| FocusInfoCapabilitiesInfo Json Block | {  “SupportFocusCfg”:,  “FocusModeNum”:,  “FocusModeList”:,  “FocusSceneNum”:,  “FocusSceneList”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| SupportFocusCfg | M | boolean | Whether focus is configurable:  0: No  1: Yes | 1 |
| FocusModeNum | M | unsigned long | Number of focus modes supported. |  |
| FocusModeList | C | array | Focus mode list:  0: Auto  1: Manual  2: One-click focus  3: One-click focus (IR)  4: One-click focus (lock)  One-click focus: trigger a one-time automatic focus when operating the device (such as rotating, zooming or calling a preset) on the PTZ.  One-click focus (IR): suitable for low-light environment such as at night or in a dark room.  One-click focus (lock): Perform an automatic focus following a zoom; other actions cannot trigger focus.  Not needed when FocusModeNum=0. | [1,2,3,4] |
| FocusSceneNum | M | unsigned long | Number of focus scenes supported. |  |
| FocusSceneList | C | array | Focus scenes supported:  0: Normal  1: Long distance  2: Indoor  Normal: Suitable for common scenes such as road, park.  Long distance: suitable for remote road scenes, for example, when device installation height is over 30m.  Indoor: suitable for indoor scenes.  Not needed when FocusSceneNum=0. | [1,2] |

#### LampInfo

|  |  |
| --- | --- |
| LampInfo Json Block | {  “LampType”: ,  “LampCtrlModeListNum”: ,  “LampCtrlModeList”: ,  “SupportNearLampCfg”:,  “SupportMiddleLampCfg”:,  “SupportFarLampCfg”: ,  “SupportSuperFarLampCfg”:,  “SupportLaserAngleCfg”:,  “LaserAngleMin”: ,  “LaserAngleMax”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| LampType | M | unsigned long | Lamp type:  0: No lamp  1: White light  2: Infrared  3: Laser  4: Housing IR illuminator | 1 |
| LampCtrlModeListNum | M | unsigned long | Number of control modes supported. |  |
| LampCtrlModeList | C | Array | Control modes supported:  0: Global Mode  1: Overexposure Restrain  2: Preset-Road  3: Manual  4: Preset-Park  5: Auto Sensitivity Control  6: Indoor  7: Manual-Always On  8: Video detection mode  Global Mode and Overexposure Restrain: Lamp on/off is triggered by day/night mode; illumination intensity is calculated based on AE algorithm.  Preset-Road Mode or Preset-Park Mode: Lamp on/off is triggered by day/night mode; illumination intensity is calculated in real time based on the current zoom ratio, PTZ angle, and preset table files.    Manual Mode: Lamp on/off is triggered by day/night mode; illumination intensity is determined by the value set on screen.  Manual-Always On: Lamp is turned on by force; not triggered by day/night mode.  Video Detection Mode: Triggered by day/night mode and smart virtual iris status; the lamp is turned on only when day/night mode is night and smart virtual iris is on; illumination intensity is determined by the value set on screen.  Not needed when LampCtrlModeListNum=0. | 0 |
| SupportNearLampCfg | M | boolean | Whether near-illumination level is configurable:  0: No  1: Yes | 0 |
| SupportMiddleLampCfg | M | boolean | Whether mid-illumination level is configurable:  0: No  1: Yes | 0 |
| SupportFarLampCfg | M | boolean | Whether far-illumination level is configurable:  0: No  1: Yes | 0 |
| SupportSuperFarLampCfg | M | boolean | Whether super far illumination level is configurable:  0: No  1: Yes | 1 |
| SupportLaserAngleCfg | M | boolean | Whether laser angle is configurable:  0: No  1: Yes | 0 |
| LaserAngleMin | C | unsigned long | Minimum laser angle; not needed when SupportLaserAngleCfg=0. |  |
| LaserAngleMax | C | unsigned long | Maximum laser angle; not needed when SupportLaserAngleCfg=0. |  |

#### LampCtrlCapabilitiesInfo

|  |  |
| --- | --- |
| LampCtrlCapabilitiesInfo Json Block | {  “LampNum”:  “LampInfos”: [<[LampInfo](#_LightInfo)>,<[LampInfo](#_LightInfo)>,……]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| LampNum | M | unsigned long | Number of lamp types supported.  0 means the device does not support lamp. |  |
| LampInfos | C | array | Lamp info list; not needed when LampNum=0. |  |
| LampInfo | M | Json Block | Lamp info. | See [LampInfo](#_LightInfo) Json Block for details. |

#### ExposureCapabilitiesInfo

|  |  |
| --- | --- |
| ExposureCapabilitiesInfo Json Block | {  “SupportCompensation”:,  “ExposureModeNum”:,  “ExposureModeList”:,  “SupportIrisCfg”:,  “IrisRangeNum”:,  “IrisRangeList”:,  “MinGain”:,  “MaxGain”:,  “SupportHLCCfg”:,  “SupportMeteringCfg”:,  “MeteringModeNum”:,  “MeteringModeList”:,  “Shutter”: <[**ShutterCapabilitiesInfo**](#_ShutterCapabilitiesInfo_1)>,  “WideDynamic”<[WideDynamicCapabilitiesInfo](#_WideDynamicCapabilitiesInfo)>,  “DayNight”: <[**DayNightCapabilitiesInfo**](#_DayNightCapabilitiesInfo)>  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| SupportCompensationCfg | M | boolean | Whether exposure compensation is configurable:  0: No  1: Yes | 1 |
| ExposureModeNum | M | unsigned long | Number of exposure modes supported. |  |
| ExposureModeList | C | array | Exposure modes supported:  0: Auto  1: Custom  2: Shutter priority  3: Iris priority  4: Gain priority  5: Indoor 50Hz  6: Indoor 60Hz  7: Manual  8: Low motion blur  Not needed when ExposureModeNum=0. | [0,1,3,4,6,7] |
| SupportIrisCfg | M | boolean | Whether iris is configurable:  0: No  1: Yes | 1 |
| IrisRangeNum | M | unsigned long | Number of iris values supported. |  |
| IrisRangeList | C | array | Iris values supported:  160, 200, 240, 280, 340, 400, 480, 560, 680, 800, 960, 1100,1400, 1600, 2200  Not needed when IrisRangeNum=0. | [160, 200, 240] |
| MinGain | M | unsigned long | Minimum gain supported. | 1 |
| MaxGain | M | unsigned long | Maximum gain supported. | 100 |
| SupportHLCCfg | M | boolean | Whether HLC sensitivity is configurable:  0: No  1: Yes | 1 |
| SupportMeteringCfg | M | boolean | Whether metering control is configurable:  0: No  1: Yes | 1 |
| MeteringModeNum | M | unsigned long | Number of metering control modes supported. |  |
| MeteringModeList | C | array | Metering control modes:  0: Center-weighted average metering  1: Evaluative Metering  2: Highlight Compensation  3: Vehicle Metering  4: Face Metering  5: Spot Metering  Not needed when MeteringModeNum=0. | [0,1,2,3,4,5] |
| Shutter | M | Json block | Shutter | See [**ShutterCapabilitiesInfo**](#_ShutterCapabilitiesInfo) **Json Block** for details. |
| WideDynamic | M | Json block | WDR | See [WideDynamicCapabilitiesInfo](#_WideDynamicCapabilitiesInfo) **Json Block** for details. |
| DayNight | M | Json block | Day/night mode | See [**DayNightInfo**](#_DayNightInfo) **Json Block** for details. |

#### [ImageCapabilities](#_VideoInfo)

|  |  |
| --- | --- |
| ImageCapabilities Json Block | {  “IsSupportCfg”:,  “SupportIlluminationCfg”,  “Enhance”: <[EnhanceCapabilitiesInfo](#_EnhanceCapabilitiesInfo)>,  “Focus”:<[FocusCapabilitiesInfo](#_FocusCapabilitiesInfo)>,  “LampCtrl”:< [LampCtrlCapabilitiesInfo](#_LedCtrlCapabilitiesInfo)>,  “Exposure”: <[ExposureCapabilitiesInfo](#_ExposureCapabilitiesInfo)>  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| IsSupportCfg | M | boolean | Whether image parameters are configurable:  0: No  1: Yes | 1 |
| SupportIlluminationCfg | M | boolean | Whether can obtain relative illumination level:  0: No  1: Yes  Only supported by IPC. | 1 |
| Enhance | M | Json block | Image enhancement parameter. | See [EnhanceCapabilitiesInfo](#_ImageCapabilities) **Json Block** for details. |
| Focus | M | Json block | Focus capability parameter.  Only supported by IPC. | See [FocusCapabilitiesInfo](#_FocusCapabilitiesInfo) **Json Block** for details. |
| LampCtrl | M | Json block | Lamp capability parameter.  Only supported by IPC. | See [LampCtrlCapabilitiesInfo](#_LedCtrlCapabilitiesInfo) **Json Block** for details. |
| Exposure | M | Json block | Exposure capability parameter.  Only supported by IPC. | See [ExposureCapabilitiesInfo](#_ExposureCapabilitiesInfo) **Json Block** for details. |
| 备注：本接口其他参数暂时不对外发布。 | | | | |

#### DayNightCapabilitiesInfo

|  |  |
| --- | --- |
| DayNightCapabilitiesInfo Json Block | {  “SupportDayNightCfg”:,  “DayNightModeNum”:,  “DayNightModeList”:,  “SupportSensibilityCfg”:,  “SupportSwitchingCfg”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| SupportDayNightCfg | M | boolean | Whether day/night mode is configurable:  0: No  1: Yes | 1 |
| DayNightModeNum | M | unsigned long | Number of day/night mode types supported. |  |
| DayNightModeList | C | array | Day/night mode types:  0: Auto  1: Day-Color  2: Night-Black/white  3: Trigger Color to Black  Not needed when DayLightModeNum=0. | [0,1,2] |
| SupportSensibilityCfg | M | boolean | Whether day/night mode sensitivity is configurable:  0: No  1: Yes | 1 |
| SupportSwitchingCfg | M | boolean | Whether day/night mode switching time is configurable:  0: No  1: Yes | 1 |

#### WideDynamicCapabilitiesInfo

|  |  |
| --- | --- |
| WideDynamicCapabilitiesInfo Json Block | {  “SupportWideDynamicCfg”:,  “WideDynamicModeNum”:,  “WideDynamicModeList”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| SupportWideDynamicCfg | M | boolean | Whether WDR is configurable:  0: No  1: Yes | 1 |
| WideDynamicModeNum | M | unsigned long | Number of WDR modes supported. |  |
| WideDynamicModeList | C | array | WDR modes:  0: Off  1: On  2: Auto  Not needed when WideDynamicModeNum=0. | [0,1] |

#### ShutterCapabilitiesInfo

|  |  |
| --- | --- |
| ShutterCapabilities Json Block | {  “SupportShutterTimeCfg”:  “ShtterTimeUnit”:  “ShutterTimeRangeNum”:  “ShutterTimeRangeList”:  “SupportSlowShutterCfg”:,  “SlowShutterTimeRangeNum”:,  “SlowShutterTimeRangeList”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| SupportShutterTimeCfg | M | boolean | Whether shutter time is configurable.  0: No  1: Yes | 1 |
| ShtterTimeUnit | M | unsigned long | Shutter time unit:  0: Microsecond  1: Second | 1 |
| ShutterTimeRangeNum | M | unsigned long | Number of shutter time values supported. |  |
| ShutterTimeRangeList | C | array | Shutter time values supported:  0: Auto  1: 1/1  2: 1/2  3: 1/3  4: 1/4  6: 1/6  8: 1/8  10: 1/10  12: 1/12  15: 1/15  20: 1/20  25: 1/25  30: 1/30  50: 1/50  60: 1/60  100: 1/100  120: 1/120  150: 1/150  180: 1/180  200: 1/200  250: 1/250  500: 1/500  1000: 1/1000  2000: 1/2000  4000: 1/4000  8000: 1/8000  50000: 1/50000  100000: 1/100000  Not needed when ShutterTimeRangeNum=0. | [0,4,8,10] |
| SupportSlowShutterCfg | M | boolean | Whether slow shutter is configurable:  0: No  1: Yes | 1 |
| SlowShutterTimeRangeNum | M | unsigned long | Number of slow shutter time values supported. |  |
| SlowShutterTimeRangeList | C | array | Slow shutter time values:  0: Auto  1: 1/1  2: 1/2  3: 1/3  4: 1/4  6: 1/6  8: 1/8  10: 1/10  12: 1/12  15: 1/15  20: 1/20  25: 1/25  Not needed when SlowShutterTimeRangeNum=0. | [0,4,8,10] |

#### ImageEnhanceInfo

|  |  |
| --- | --- |
| **ImageEnhanceInfo Json Block** | {  “Brightness”: ,  “Contrast”:,  “Saturation”:,  “Sharpness”:,  “ImageRotation”:,  “2DNoiseReduce”:,  “3DNoiseReduce”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Brightness | M | unsigned long | Brightness, range[0,255] | 128 |
| Contrast | M | unsigned long | Contrast, range[0,255] | 128 |
| Saturation | M | unsigned long | Saturation, range[0,255] | 128 |
| Sharpness | C | unsigned long | Enhanced configuration Sharpness, range[0,255]  Required when using GET. When using PUT for deploy, the segment NeedSetEnhanceInfo should be 1. | 128 |
| ImageRotation | C | unsigned long | Enhanced configuration for image rotation.  Required when using GET. When using PUT for deploy, the segment NeedSetEnhanceInfo should be 1. | 1 |
| 2DNoiseReduce | C | unsigned long | Enhanced configuration 2D noise reduce, range[0,255]  Required when using GET. When using PUT for deploy, the segment NeedSetEnhanceInfo should be 1. | 128 |
| 3DNoiseReduce | C | unsigned long | 3D noise reduce, range[0,255] IPC supported | 128 |

#### FocusInfo

|  |  |
| --- | --- |
| FocusInfo Json Block | {  "Mode": ,  "Scene":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Mode | M | unsigned long | Focus mode (see [FocusInfoCapabilitiesInfo](#_FocusCapabilitiesInfo) for enumerations of supported modes). | 1 |
| Scene | C | unsigned long | Focus scene (see [FocusInfoCapabilitiesInfo](#_FocusCapabilitiesInfo) for enumerations of supported scenes).  Required when supported by image capability. | 5 |

#### LampCtrlInfo

|  |  |
| --- | --- |
| LampCtrlInfo Json Block | {  "Enabled": ,  "Type": ,  "Mode": ,  "NearLevel": ,  "MiddleLevel": ,  "FarLevel": ,  "SuperFarLevel ": ,  "LaserAngle":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | M | Boolean | Lamp enablement:  0: Disable  1: Enable | 1 |
| Type | M | unsigned long | Lamp type (see [LampCtrlCapabilitiesInfo](#_LightCtrlCapabilitiesInfo) for enumerations of supported modes). | 1 |
| Mode | M | unsigned long | Lamp control modes (see [LampCtrlCapabilitiesInfo](#_LedCtrlCapabilitiesInfo) for enumerations of supported modes). | 1 |
| NearLevel | C | unsigned long | Near-illumination level  Effective when IR control mode is Manual or Manual-Always On.  Required when supported by lamp capability.  Range: [0-1000]  Higher level means higher illumination intensity. | 5 |
| MiddleLevel | C | unsigned long | Mid-illumination level.  Effective when IR control mode is Manual or Manual-Always On.  Required when supported by lamp capability.  Range: [0-1000]  Higher level means higher illumination intensity. | 5 |
| FarLevel | C | unsigned long | Far-illumination level  Effective when IR control mode is Manual or Manual-Always On.  Required when supported by lamp capability.  Range: [0-1000].  Higher level means higher illumination intensity. | 5 |
| SuperFarLevel | C | unsigned long | Super far illumination level  Effective when IR control mode is Manual or Manual-Always On.  Required when supported by lamp capability.  Range: [0-1000]  Higher level means higher illumination intensity. | 5 |
| LaserAngle | C | unsigned long | Laser angle  Effective when lamp type is laser, and IR control mode is Manual or Manual-Always On.  Required when supported by lamp capability.  See lamp capability for the range.  A smaller angle means more energy of higher intensity; a greater angle means scattered energy. | 1 |

#### IrisInfo

|  |  |
| --- | --- |
| IrisInfo Json Block | {  “Iris”:,  “MinIris”:,  “MaxIris”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Iris | C | unsigned long | Iris  Not needed in Iris Priority or Manual Exposure mode.  Required when supported by image capability. See image capability for enumerations. | 160 |
| MinIris | C | unsigned long | Minimum iris.  Effective when in custom exposure mode. See iris capability for enumerations; must not be greater than the maximum iris.  Required when supported by image capability. | 160 |
| MaxIris | C | unsigned long | Maximum iris.  Effective when in custom exposure mode. See iris capability for enumerations; must not be smaller than the minimum iris.  Required when supported by image capability. | 200 |

#### ShutterInfo

|  |  |
| --- | --- |
| ShutterInfo Json Block | {  “Shutter”:,  “MinShutter”:,  “MaxShutter”:,  “IsEnableSlowShutter”:,  “SlowestShutter”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Shutter | C | unsigned long | Shutter time.  Effective when in shutter priority, manual exposure, and low motion blur mode.  Required when supported by image capability. See image capability for enumerations and unit. | 0 |
| MinShutter | C | unsigned long | Minimum shutter time.  Effective when in custom exposure mode. See shutter capability for enumerations; must not be greater than the maximum shutter time.  Required when supported by image capability. | 0 |
| MaxShutter | C | unsigned long | Maximum shutter time.  Effective when in custom exposure mode. See shutter capability for enumerations; must not be smaller than the minimum shutter time.  Required when supported by image capability. | 30 |
| IsEnableSlowShutter | C | boolean | Slow shutter enablement.  Effective when not in shutter priority mode  0: Disable  1: Enable  Required when supported by image capability. | 1 |
| SlowestShutter | C | unsigned long | Effective when slow shutter is enabled.  Slowest shutter.  Required when supported by image capability. See image capability for enumerations. | 20 |

#### GainInfo

|  |  |
| --- | --- |
| GainInfo Json Block | {  “Gain”:,  “MinGain”:,  “MaxGain”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Gain | C | unsigned long | Gain (unit: db)  Effective when in manual exposure mode.  Required when supported by image capability.  See image capability for the range. | 0 |
| MinGain | C | unsigned long | Minimum gain.  Effective when in custom exposure mode; must not be greater than the maximum gain.  Required when supported by image capability; see image capability for the range. | 0 |
| MaxGain | C | unsigned long | Maximum gain.  Effective when in custom exposure mode; must not be smaller than the minimum gain.  Required when supported by image capability; see image capability for the range. | 30 |

#### WideDynamicInfo

|  |  |
| --- | --- |
| WideDynamicInfo Json Block | {  “Mode”:,  “Level”:,  “OpenSensitivity”:,  “CloseSensitivity”:,  “AntiFlicker”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Mode | M | unsigned long | See [WideDynamicCapabilitiesInfo](#_WideDynamicCapabilitiesInfo) for enumerations. | 1 |
| Level | M | unsigned long | WDR level.  Effective when WDR is enabled and in one of these exposure modes: auto, custom, shutter priority, indoor 50Hz, indoor 60Hz, and low motion blur.  Range: [1, 9] | 1 |
| OpenSensitivity | M | unsigned long | Sensitivity for turning on WDR.  Effective when in auto WDR mode.  Range: [1, 9] | 1 |
| CloseSensitivity | M | unsigned long | Sensitivity for turning off WDR.  Effective when in auto WDR mode.  Range: [1, 9] | 1 |
| AntiFlicker | M | boolean | WDR anti-flicker:  0: Disable  1: Enable  When enabled, this function can reduce flickers in the image. | 1 |

#### DayNightInfo

|  |  |
| --- | --- |
| DayNightInfo Json Block | {  “Mode”:,  “Sensitivity”:,  “Time”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Mode | C | unsigned long | Day/night mode type.  See [DayNightCapabilitiesInfo](#_DayNightInfo) for enumerations.  Required when supported by image capability. | 2 |
| Sensitivity | C | unsigned long | Day/night mode sensitivity.  Effective when day/night mode is auto.  Range: [0, 9]  Required when supported by image capability. | 4 |
| Time | C | unsigned long | Day/night mode switching time.  Effective when day/night mode is auto.  Range: [3, 120]  Unit: second  Required when supported by image capability. | 3 |

#### MeteringInfo

|  |  |
| --- | --- |
| MeteringInfo Json Block | {  "Mode": ,  “RefBrightness”:  “HoldTime”:  "Area": {  "TopLeft": {  "X":,  "Y":  },  " BottomRight": {  "X":,  "Y":  }  }  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Mode | C | unsigned long | Metering control mode.  Effective when not in manual exposure mode.  See [ExposureCapabilitiesInfo](#_ExposureCapabilitiesInfo) for enumerations.  Required when supported by image capability. | 1 |
| RefBrightness | C | unsigned long | Face brightness.  Effective when in face metering mode.  Range: [0, 100] | 1 |
| HoldTime | C | unsigned long | Minimum duration.  Effective when in face metering mode.  Unit: min  Range: [0, 60] | 1 |
| MeteringArea | C | Json block | Metering area.  Effective when in evaluative metering or spot metering mode. |  |
| TopLeft | M | Json block | Coordinates of top left corner | (25, 25) |
| X | M | unsigned long | X-axis of top left corner (scale)  Evaluative metering range: [0, 100] | 25 |
| Y | M | unsigned long | Y-axis of top left corner (scale)  Evaluative metering range: [0, 100] | 25 |
| BottomRight | M | Json block | Coordinates of bottom right corner | (75, 75) |
| X | M | unsigned long | X-axis of bottom right corner (scale):  Evaluative metering range: [0, 100].  Must not be smaller than X-axis of top left corner. | 75 |
| Y | M | unsigned long | Y-axis of bottom right corner (scale):  Evaluative metering range: [0, 100]  Must not be smaller than Y-axis of top left corner. | 75 |

#### ExposureInfo

|  |  |
| --- | --- |
| ExposureInfo Json Block | {  “Mode”: ,  “CompensationLevel”:,  “HLCSensitivity”:,  "IrisInfo": [IrisInfo](#_IrisInfo),  "ShutterInfo": [ShutterInfo](#_ShutterInfo),  "GainInfo": [GainInfo](#_GainInfo),  "WideDynamic": [WideDynamicInfo](#_WideDynamicInfo_1),  "Metering": [MeteringInfo](#_MeteringInfo),  "DayNight": [DayNightInfo](#_DayNightInfo_1)  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Mode | M | unsigned long | Exposure mode.  See [ExposureCapabilitiesInfo](#_ExposureCapabilitiesInfo) for enumerations. | 0 |
| CompensationLevel | C | unsigned long | Exposure compensation level.  Effective when not in manual exposure mode.  Range: [-100,100]  Required when supported by image capability. | 0 |
| HLCSensitivity | C | unsigned long | HLC sensitivity.  Effective when the scene is road or park.  Range: [1, 9].  Required when supported by image capability. |  |
| IrisInfo | C | Json Block | Iris info.  Required when supported by image capability. |  |
| ShutterInfo | C | Json Block | Shutter info.  Required when supported by image capability. |  |
| GainInfo | M | Json Block | Gain info. |  |
| WideDynamic | C | Json Block | WDR info.  See [WideDynamicInfo](#_WideDynamicInfo_1) for details. Required when supported by image capability. |  |
| Metering | C | Json Block | Metering info.  See [MeteringInfo](#_MeteringInfo) for details.  Effective when the scene is not road or park.  Required when supported by image capability. |  |
| DayNight | C | Json Block | Day/night mode info.  See [DayNightInfo](#_DayNightInfo_1) for details.  Required when supported by image capability. |  |

#### IlluminationInfo

|  |  |
| --- | --- |
| IlluminationInfo Json Block | {  “Value”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Value | C | unsigned long | Relative illumination level, which is normalized and divided into [0-255] levels.  A higher level means greater overall brightness for the scene.  Required when supported by image capability. | 1 |

### Capabilities

#### /LAPI/V1.0/Channels/<ID>/Image/Capabilities

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Image/Capabilities |
| **Description** | Query the image parameter to the appointed channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**ImageCapabilities**](#_ImageCapabilities_1) |
| **Note** | IPC & NVR |
| **Status** |  |

### Enhance

#### /LAPI/V1.0/Channels/<ID>/Image/Enhance

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Image/Enhance |
| **Description** | Query the image enhanced parameter of the appointed channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [ImageEnhanceInfo](#_ImageEnhanceInfo_1) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Image/Enhance |
| **Description** | Set the image enhanced parameter of the appointed channel |
| **Method** | PUT |
| **Input Data** | [ImageEnhanceInfo](#_ImageEnhanceInfo_1) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

### Focus

#### /LAPI/V1.0/Channels/<ID>/Image/FocusInfo

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Image/FocusInfo |
| **Description** | Query focus parameter for the specified video channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**FocusInfo**](#_FocusInfo) |
| **Note** | IPC |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Image/FocusInfo |
| **Description** | Set focus parameter for the specified video channel. |
| **Method** | PUT |
| **Input Data** | [**FocusInfo**](#_FocusInfo) |
| **Success Return Data** | None |
| **Note** | IPC |
| **Status** |  |

### LampCtrl

#### /LAPI/V1.0/Channels/<ID>/Image/LampCtrl

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Image/LampCtrl |
| **Description** | Query lamp parameter of the specified video channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**LampCtrlInfo**](#_LightCtrlInfo) |
| **Note** | IPC |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Image/LampCtrl |
| **Description** | Set lamp parameter of the specified video channel. |
| **Method** | PUT |
| **Input Data** | [**LampCtrlInfo**](#_IrCtrlInfo) |
| **Success Return Data** | None |
| **Note** | IPC |
| **Status** |  |

### Exposure

#### /LAPI/V1.0/Channels/<ID>/Image/Advanced/Exposure

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Image/Advanced/Exposure |
| **Description** | Query image exposure parameter. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [ExposureInfo](#_ImageExposureInfo) |
| **Note** | IPC |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Image/Advanced/Exposure |
| **Description** | Set image exposure parameter. |
| **Method** | PUT |
| **Input Data** | [ExposureInfo](#_ImageExposureInfo) |
| **Success Return Data** | None |
| **Note** | IPC |
| **Status** |  |

### Illumination

#### /LAPI/V1.0/Channels/<ID>/Image/Illumination

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Image/Illumination |
| **Description** | Query relative illumination level. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [IlluminationInfo](#_ILLuminationInfo) |
| **Note** | IPC |
| **Status** |  |

## Storage

### Json Block

#### StorageCapabilities

|  |  |
| --- | --- |
| **StorageCapabilities Json Block** | {  "IsSupportRedundancy": ,  "IsSupporteSATA": ,  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| IsSupportRedundancy | M | boolean | if support redundancy  0: no  1:yes | 1 |
| IsSupporteSATA | M | boolean | if support eSATA  0: no  1:yes | 0 |

#### ContainerInfoList

|  |  |
| --- | --- |
| **ContainerInfoList Json Block** | {  "LocalHDDNum" : ,  "LocalHDDList": [<[ContainerInfo](#_ContainerInfo)>, <[ContainerInfo](#_ContainerInfo)>,……],  "SDNum" : ,  "SDList": [<[ContainerInfo](#_ContainerInfo)>, <[ContainerInfo](#_ContainerInfo)>,……],  "ArrayNum" : ,  "ArrayList":[<[ContainerInfo](#_ContainerInfo)>, <[ContainerInfo](#_ContainerInfo)>,……],  "ExtendCabinet1HDDNum" : ,  "ExtendCabinet1HDDList": [<[ContainerInfo](#_ContainerInfo)>, <[ContainerInfo](#_ContainerInfo)>,……],  "ExtendCabinet2HDDNum" : ,  "ExtendCabinetHDDList": [<[ContainerInfo](#_ContainerInfo)>, <[ContainerInfo](#_ContainerInfo)>,……],  "NASNum" : ,  "NASList": [<[ExContainerInfo](#_ExContainerInfo)>, <[ExContainerInfo](#_ExContainerInfo)>,……],  "eSATANum" : ,  "eSATAList": [<[ExContainerInfo](#_ExContainerInfo)>, <[ExContainerInfo](#_ExContainerInfo)>,……],  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| LocalHDDNum | M | unsigned long | the number of local harddisk | 4 |
| LocalHDDList | C | array | info list of local harddisk, optional when segment LocalHDDNum is 0 | - |
| SDNum | M | unsigned long | the number of SD card. NVR &VMS are not supported. Reserved parameter. | 0 |
| SDList | C | array | the info list of SD card. Optional when segment SDNum is 0. | - |
| ArrayNum | M | unsigned long | the number of storage array. | 1 |
| ArrayList | C | array | the info list of storage array. Optional when segment ArrayNum is 0. | - |
| ExtendCabinet1HDDNum | M | unsigned long | the number of harddisk in the extend cabinet 1 | 2 |
| ExtendCabinet1HDDList | C | array | the info list of the extend cabinet 1, optional when segment ExtendCabinetHDD1Num is 0. | - |
| ExtendCabinet2HDDNum | M | unsigned long | the number of harddisk in the extend cabinet 2 | 2 |
| ExtendCabinet2HDDList | C | array | the info list of the extend cabinet 2, optional when segment ExtendCabinetHDD1Num is 0. | - |
| ContainerInfo | C | Json Block | info of container, refer to [ContainerInfo](#_ContainerInfo) Json Block | - |
| NASNum | M | unsigned long | number of NAS  NVR supported | 2 |
| NASList | C | array | the info list of NAS. Optional when segment NASNum is 0. | - |
| eSATANum | M | unsigned long | number of eSATA  NVR supported | 2 |
| eSATAList | C | array | the info list of eSATA. Optional when segment eSATANum is 0. |  |
| ExContainerInfo | M | Json Block | info of extend container, refer to [ExContainerInfo](#_ExContainerInfo) Json Block. | - |

#### ContainerInfo

|  |  |
| --- | --- |
| **ContainerInfo Json Block** | {  "ID": ,  "RemainCapacity": ,  "TotalCapacity": ,  "Manufacturer": ,  "Status": ,  "Property": ,  "FormatProgress": ,  "GroupID": ,  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | container ID | 1 |
| RemainCapacity | M | unsigned long | remaining capacity of the container(MB) | 1907730 |
| TotalCapacity | M | unsigned long | total capacity of the container(MB) | 1907730 |
| Manufacturer | O | string | name of manufacturer, length of string [1,64] | “1,64]符串长度er000VX000-9YW1CV12: |
| Status | M | unsigned long | status of container  0: no hard disk/idle  1: unformatted  2: formatting  3: normal  4: dormant  5: abnormal  6: switching  7: discharged(NAS only, NVR supported) | 3 |
| Property | C | unsigned long | property of hard disk  0: read and write  1: read  2: backup  Optional when segment Status is 0.  NVR supported. | 0 |
| FormatProgress | O | unsigned long | Format progress in percentage. Required when segment Status is 2. | 0 |
| GroupID | O | unsigned long | Group ID | 1 |

#### ExContainerInfo

|  |  |
| --- | --- |
| **ExContainerInfo Json Block** | {  "ID": ,  "RemainCapacity": ,  "TotalCapacity": ,  "AddressType": ,  "IPAddress": ,  "Path": ,  "UsageType": ,  "Status": ,  "Property": ,  "FormatProgress": ,  "GroupID": } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | extend container ID | 1 |
| RemainCapacity | M | unsigned long | remain capacity of the extend container(MB) | 1907730 |
| TotalCapacity | M | unsigned long | total capacity of the extend container(MB) | 1907730 |
| Status | M | unsigned long | status of extend hard disk  0: no harddisk/leisure  1: non-formatting  2: formatting  3: normal  4: dormant  5: abnormal  6: switching  7: unload(NAS only, Unicorn not supported) | 3 |
| Property | M | unsigned long | property of harddisk  0: read and write  1: read  2: backup | 0 |
| FormatProgress | C | unsigned long | format progress in percentage. Required when segment Status is 2. | 0 |
| AddressType | C | unsigned long | IP type, IPv4 default  0:IPv4  1:IPv6  2: domain name  3: IPv4 &IPv6  Only IPv4 supported at present.  Required when using NAS. | 0 |
| IPAddress | C | string | IP address, length of string [0, 64].  Required when using NAS. | “192.168.100.30” |
| Path | C | string | path of NAS server, length of string [1,256] | “/volume2/123456” |
| UsageType | M | unsigned long | usage  0: record /snapshot  1: backup | 0 |
| GroupID | C | unsigned long | Group ID, 1 as default | 1 |

#### RecordScheduleInfo

|  |  |
| --- | --- |
| RecordScheduleInfo Json Block | {  "Enabled": ,  "IsRedundantStorage": ,  "RecordRule": {  "PreRecordTime": ,  "PostRecordTime": ,  },  "WeekPlan":<[WeekPlanInfo](#_WeekPlanInfo_1)>  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | M | boolean | whether plan enabled  0: no  1: yes | 1 |
| IsRedundantStorage | C | unsigned long | whether redundant plan enabled  0: no  1: yes  If there’s no device for redundant, parameter is invalid. | 1 |
| RecordRule | M | Json Block | rule of record plan | - |
| PreRecordTime | M | unsigned long | Pre-record time, second as unit, range: 0,5,10,20,30,60 | 5 |
| PostRecordTime | M | unsigned long | Post-record time, second as unit, range: 0,5,10,20,30,60 | 10 |
| WeekPlan | M | Json Block | list of plan time | - |
| WeekPlanInfo | M | Json Block | info of plan time, refer to [WeekPlanInfo](#_WeekPlanInfo_1) Json Block | - |

#### PictureScheduleInfo

|  |  |
| --- | --- |
| **PictureScheduleInfo Json Block** | {  "Enabled": ,  "IsRedundantStorage": ,  "WeekPlan":<[WeekPlanInfo](#_WeekPlanInfo_1)>  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | M | boolean | whether plan enabled  0: no  1: yes | 1 |
| IsRedundantStorage | C | unsigned long | whether redundant snap enabled  0: no  1: yes  If there’s no device for redundant, parameter is invalid.  NVR supported | 1 |
| WeekPlan | M | Json Block | list of plan time | - |
| WeekPlanInfo | M | Json Block | info of plan time, refer to [WeekPlanInfo](#_WeekPlanInfo_1) Json Block | - |

### Capabilities

#### /LAPI/V1.0/Storage/Capabilities

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Storage/Capabilities |
| **Description** | Query the capabilities of storage |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [StorageCapabilities](#_StorageCapabilities) |
| **Notes** | NVR |
| **Status** |  |

### Containers

#### /LAPI/V1.0/Storage/Containers/DetailInfos

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Storage/Containers/DetailInfos |
| **Description** | Query the info list of storage container |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**ContainerInfoList**](#_ContainerInfoList) |
| **Note** | NVR & VMS |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Storage/Containers/DetailInfos |
| **Description** | set the configuration of storage container |
| **Method** | PUT |
| **Input Data** | {  "Num":,  "ContainerInfoList": [{  "ID": ,  "Type": ,  "Property": ,  "UsageType": ,  "GroupID":  }, {  "ID": ,  "Type": ,  "Property": ,  "UsageType": ,  "GroupID":  }, ……]  } |
| **Success Return Data** | None |
| **Note** | NVR & VMS |
| **Status** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | number of storage container to be modified | 3 |
| ContainerInfoList | M | array | info list of storage container | - |
| ID | M | unsigned long | Container ID, read only | 1 |
| Type | M | unsigned long | type of container, read only  0:LocalHDD  1:ExtendCabinet1HDD  2:ExtendCabinet2HDD  3:RAID  4:NAS  5:SAN  6:eSATA  7:SD | 0 |
| Property | M | unsigned long | property of container  0: read and write  1: ready only  2: backup(only when device supports redundant)  NVR supported | 0 |
| GroupID | O | unsigned long | Group ID  NVR supported | 1 |
| UsageType | O | unsigned long | usage  0: record /snap  1: backup(only NAS and eSATA)  Invalid when using SD card. | 0 |

### Schedule

#### /LAPI/V1.0/Channels/<ID>/Storage/Schedule/Record

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Storage/Schedule/Record |
| **Description** | Query the record plan for an appointed channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [**RecordScheduleInfo**](#_RecordScheduleInfo) |
| **Note** | NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Storage/Schedule/Record |
| **Description** | Set the record plan for an appointed channel |
| **Method** | PUT |
| **Input Data** | [**RecordScheduleInfo**](#_RecordScheduleInfo) |
| **Success Return Data** | None |
| **Note** | NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Storage/Schedule/Picture

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Storage/Schedule/Picture |
| **Description** | Query the capture plan for an appointed channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [PictureScheduleInfo](#_PictureScheduleInfo) |
| **Note** | NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Storage/Schedule/Picture |
| **Description** | Set the capture plan for an appointed channel |
| **Method** | PUT |
| **Input Data** | [PictureScheduleInfo](#_PictureScheduleInfo) |
| **Success Return Data** | None |
| **Note** | NVR |
| **Status** |  |

## Alarm

### Json Block

#### AlarmCapabilities

|  |  |
| --- | --- |
| **AlarmCapabilities Json Block** | {  "MotionDetection": {  "IsSupportCfg": ,  "RectangleAreaNum": ,  "BlockWidth": ,  "BlockHeight":  },  "TamperDetection": {  "IsSupportCfg": ,  "TamperRectangleAreaNum":  },  "AudioDetection": {  "SupportCfg": ,  }  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| MotionDetection | M | Json Block | capability of motion detection | - |
| IsSupportCfg | M | boolean | whether support configure  0:not support  1:support | 1 |
| RectangleAreaNum | C | unsigned long | the number of support rectangle detection area.  Optional when not support rectangle area  IPC supportive |  |
| BlockWidth | M | unsigned long | maximum block width | 22 |
| BlockHeight | M | unsigned long | maximum block height | 18 |
| TamperDetection | M | Json Block | capability to detect a tamper | - |
| IsSupportCfg | M | boolean | whether support configure to linked equipment  0:not support  1:support | 1 |
| TamperRectangleAreaNum | M | unsigned long | the number of available tamper detection area, now this number is fixed to 1 | 1 |
| AudioDetection | M | Json Block | Audio detection capability |  |
| SupportCfg | M | boolean | Whether is configurable:  0: No  1: Yes | 1 |

#### LinkageActionList

|  |  |
| --- | --- |
| **LinkageActionList Json Block** | {  "Num": ,  "Actions":[< [LinkageActionInfo](#_LinkageActionInfo_1)>,…]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | number of linkage | 8 |
| Actions | C | array | linkage action information list, this node is not needed when Num is 0 | - |
| LinkageActionInfo | M | Json Block | for more details on linkage action, please refer to [LinkageActionInfo](#_LinkageActionInfo_1) Json Block | - |

#### LinkageActionInfo

|  |  |
| --- | --- |
| **LinkageActionInfo Json Block** | {  "ActID": ,  "ActParam":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ActID | M | unsigned long | 0 link to NVR preview, NVR supportive  1 link to buzzer, IPC currently not supportive  2 link to E-Mail, NVR s supportive  3 link to storage  4 link to PTZ preset  5 link to output switch value  6 link to image capture  7 link to alarm message box, NVR supportive  8 link to central storage  9 link to launch local storage  10 link to pause local storage  11link to image capture and upload to Ftp  12link to image capture and upload to E-Mail  13 link to image capture and upload them to E-mail and Ftp at the same time  14 link to smart image capture and upload  15 link to image capture (small image) of human face and upload  16 link to alarm upload | 1 |
| ActParam | C | Json Block | IPC and NVR have different link actions, for more details on corresponding relation between ActID and ActParam, please refer to list below |  |

|  |  |  |  |
| --- | --- | --- | --- |
| ActID | Description | IPC ActParam | NVR ActParam |
| 0 | link to NVR preview, NVR supportive | Not Support | [ChannelActParamInfo](#_ChannelActParamInfo) |
| 1 | link to buzzer, IPC currently not supportive | Not Support | [EnabledActParamInfo](#_EnabledActParamInfo) |
| 2 | link to E-Mail, NVR supporative | Not Support | [EnabledActParamInfo](#_EnabledActParamInfo) |
| 3 | link to storage | Not Support | [ChannelActParamInfo](#_ChannelActParamInfo) |
| 4 | link to PTZ preset | [PresetActParamInfo](#_PresetActParamInfo_1) | [PresetActParamInfo](#_PresetActParamInfo_1) |
| 5 | link to alarm out | [OutputSwitchActParamInfo](#_OutputSwitchActParamInfo_1) | [OutputSwitchActParamInfo](#_OutputSwitchActParamInfo_1) |
| 6 | link to image capture | Null | [ChannelActParamInfo](#_ChannelActParamInfo) |
| 7 | link to alarm message box, NVR supportive | Not Support | [EnabledActParamInfo](#_EnabledActParamInfo) |
| 8 | link to central storage | Null | Not Support |
| 9 | link to launch local storage | Null | Not Support |
| 10 | link to pause local storage | Null | Not Support |
| 11 | link to image captured upload to Ftp | Null | Not Support |
| 12 | link to image captured upload to E-Mail | Null | Not Support |
| 13 | link to image captue and upload them to E-mail and Ftp at the same time | Null | Not Support |
| 14 | link to smart image capture and uplod | Null | Not Support |
| 15 | link to image capture (small image) of human face and upload | Null | Not Support |
| 16 | link to alarm upload | Null | Not Support |

Different alarm recourses link to different linkage actions, for more details on different linkage actions by different product, please refer to the list below



#### EnabledActParamInfo

|  |  |
| --- | --- |
| **EnabledActParamInfo Json Block** | {  “Enabled”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | M | boolean | 1:linkage enabled  0:linkage disabled | 1 |

#### ChannelActParamInfo

|  |  |
| --- | --- |
| **ChannelActParamInfo Json Block** | {  “Num”:,  “IDs”: [<ID>,<ID>…]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | number of video channel |  |
| IDs | C | array | video channel ID |  |
| ID | C | unsigned long | serial number of video channel | 1 |

#### PresetActParamInfo

|  |  |
| --- | --- |
| **PresetActParamInfo Json Block** | {  “Num“:,  “ChannelPresetList”: [<[ChannelPreset](#_ChannelPreset)>, <[ChannelPreset](#_ChannelPreset)>…]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | number of linkage action | 2 |
| ChannelPresetList | C | array | link to preset information list  this field is not needed when Num is 0 |  |
| ChannelPreset | C | Json block | serial number of channel and preset | See [ChannelPreset](#_ChannelPreset) Json block |

#### ChannelPreset

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ChannelPreset Json Block** | | {  "ChannelID":,  "PresetID"  } | | | |
| **Param** | **Requirement** | | **Type** | **Description** | **Example** |
| ChannelID | C | | unsigned long | video input channel ID  this field is not needed for IPC | 1 |
| PresetID | M | | unsigned long | preset ID | 1 |

#### OutputSwitchActParamInfo

|  |  |
| --- | --- |
| **OutputSwitchActParamInfo Json Block** | {  “Num”:,  “IDs”:[<ID>,<ID>…]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | number of linked alarm out | M |
| IDs | C | array | list of linked alarm out | C |
| ID | C | unsigned long | Alarm out channel ID  for NVR, the value is video channel ID\*100+alarm output serial number,  Equipment channel ID is 0,  for VMS, the value is serial number of equipment itself | 1 |

#### DayPlanInfo

|  |  |
| --- | --- |
| **DayPlanInfo Json Block** | {  "ID": ,  "Num": ,  "TimeSectionInfos": [<[TimeSectionInfo](#_TimeSectionInfo_1)>, …]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Day index  1:Monday; 2:Tuesday;  3:Wednesday; 4:Thursday;  5:Friday; 6:Saturday;  7:Sunday; 8:Holiday; | 1 |
| Num | M | unsigned long | Number of plan everyday  up to 8 plans for NVR  up to 4 plans for IPC | 8 |
| TimeSectionInfos | C | array | arming configuring information,  time cross is not allowed at any plan in one day  This field is not needed when Num is 0 |  |
| TimeSectionInfo | C | Json Block | for more details on arming configuring, please refer to [[TimeSectionInfo](#_TimeSectionInfo_1)](#_TimeSectionInfo) Json Block；  This field is not needed when Num is 0 |  |

#### WeekPlanInfo

|  |  |
| --- | --- |
| **WeekPlanInfo Json Block** | {  "Enabled": ,  "Num": ,  "Days": [<[DayPlanInfo](#_DayPlanInfo_1)>, …]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | C | boolean | whether enabled arming plan  0:disabled  1:enabled  Only IPC support this field  Enabled by default | 1 |
| Num | M | unsigned long | number of plan days  up to 8 for NVR (7 days a week and holiday)  up to 7 for IPC (7 days a week) | 8 |
| Days | M | array | arming plan of everyday in one week |  |
| DayPlanInfo | M | Json Block | for more details on arming plan of everyday, please refer to [[DayPlanInfo](#_DayPlanInfo_1)](#_DayPlanInfo) Json Block |  |

#### TimeSectionInfo

|  |  |
| --- | --- |
| **TimeSectionInfo Json Block** | {  "Begin": ,  "End": ,  "ArmingType":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Begin | M | string | begin time, format hh:mm:ss  Length range [0,31]  NVR not support seconds | 00:00:00 |
| End | M | string | end time, format hh:mm:ss  Length range [0,31]  NVR not support seconds | 24:00:00 |
| ArmingType | C | unsigned long | arming type  0:schedule  1:motion  2:alarm  3:motion and alarm  4:motion or alarm  5:not planned  10:event  Alarm plan only support schedule type  Recording/image capture plan support all types  IPC only support schedule type, schedule by default |  |

#### VideoLossRuleInfo

|  |  |
| --- | --- |
| **VideoLossRuleInfo Json Blcok** | {  "Enabled":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | M | boolean | whether enabled video loss  0:disabled  1:enabled | 1 |

#### TamperDetectionRuleInfo

|  |  |
| --- | --- |
| **TamperDetectionRuleInfo Json Blcok** | {  "Enabled": ,  " Sensitivity ": ,  "Duration":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | M | boolean | whether enabled tamper detection  0:disabled  1:enabled | 1 |
| Sensitivity | M | unsigned long | sensitivity  Range [1,100] | 50 |
| Duration | C | unsigned long | duration (by seconds),  Lengths range [0,10]  Needed for IPC | 1 |

#### MotionDetectionAreaType

|  |  |
| --- | --- |
| MotionDetectionAreaType Json Block | {  "Type":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Type | M | unsigned long | type of motion detection area, currently have two types  0:rectangle area  1:gird area | 1 |

#### MotionDetectionRectangleAreaInfo

|  |  |
| --- | --- |
| MotionDetectionRectangleAreaInfo Json Block | {  "ID": ,  "Enabled": ,  "Sensitivity": ,  "Duration": ,  "TargetSize": ,  "Area": {  "TopLeft": {  "X": 0,  "Y": 0  },  "BottomRight": {  "X": 10,  "Y": 10  }  }  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Rectangle area ID, start from 0, the maximum number of areas can be accessed through /LAPI/V1.0/Channels/<ID>/Alarm/Capabilities | 1 |
| Enabled | M | boolean | Motion detection rectangle area enabled sign  0:disabled  1:enabled | 1 |
| Sensitivity | M | unsigned long | Sensitivity of motion detection, the larger the number is, the more sensitive the capability is.  range [1,100] | 50 |
| Duration | C | unsigned long | duration time, range [1,100]  needed for IPC | 5 |
| TargetSize | C | unsigned long | target size, range [1,100]  needed for IPC | 1 |
| RectangleArea | M | Json Block | rectangle area information | - |
| TopLeft | M | Json Block | top-left coordinate information of rectangle area | - |
| X | M | unsigned long | Horizontal coordinate,range is 0 to 9999 | 0 |
| Y | M | unsigned long | Vertical coordinate, range is 0 to 9999 | 0 |
| BottomRight | M | Json Block | bottom-right coordinate information of rectangle area | - |
| X | M | unsigned long | Horizontal coordinate, range is 0 to 9999 | 10 |
| Y | M | Unsigned long | Vertical coordinate, range is 0 to 9999 | 10 |

#### MotionDetectionRectangleAreaInfoList

|  |  |
| --- | --- |
| MotionDetectionRectangleAreaInfoList Json Block | {  "Num": ,  "RectangleAreas": [ <[MotionDetectionRectangleAreaInfo](#_MotionDetectionRectangleAreaInfo)>, ……]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | number of rectangle area | 1 |
| RectangleAreas | C | array | rectangle area information list  This node is optional when Num is 0 | - |
| MotionDetectionRectangleAreaInfo | M | Json Block | for more details on rectangle area, please refer to [MotionDetectionRectangleAreaInfo](#_MotionDetectionRectangleAreaInfo) Json Block | - |

#### MotionDetectionGridAreaInfo

|  |  |
| --- | --- |
| MotionDetectionGridAreaInfo Json Block | {  "Enabled": ,  "Sensitivity":,  "Grid":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | M | boolean | Motion detection grid area enabled sign  0:disabled  1:enabled | 1 |
| Sensitivity | M | unsigned long | Sensitivity of motion detection, the larger the number is, the more sensitive the capability is.  Range [1,100] | 50 |
| Grid | M | string | detection area, length range [0,256]  1. the image is divided into 22\*18 gridding, from top-left corner to bottom-right corner (from left to right, from top to bottom), stack up to396 gridding, and mark them accordingly.  Use binary system (8-digit, maximum 255) to present 8 griddings motion detection areas selection status: for 1, the gridding is selected; for 0, the gridding is not selected (from lower to higher positions). all gridding are converted to 50 digital group combined with 8-digit bit.  2.the digital group is compressed by PackBits computing first, then get the compressed string from Base64 coding.  for PackBits compression and decompression computing please refer to below: | "DWAAAYAABj/5mP/mAAAY/gAgP/4A//77//vv/++//74ABvgBG+AAb4ABvh/2+H/YAf8A" |

#### AudioDetectInfo

|  |  |
| --- | --- |
| AudioDetectInfo Json Block | {  "Enabled": ,  "DetectType":,  "DiffValue": ,  "Threshold":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | M | boolean | Whether audio detection is enabled:  0: Disabled  1: Enabled | 1 |
| DetectType | M | unsigned long | Audio detection trigger type:  0: Sudden Rise  1: Sudden Fall  2: Sudden Change  3: Threshold | 0 |
| DiffValue | C | unsigned long | Difference value; use when Audio Detection is Sudden Rise, Sudden Fall and Sudden Change; range: [0,400].  Alarm is triggered when Sudden Rise, Sudden Fall or Sudden Change exceeds the difference value. | 100 |
| Threshold | C | unsigned long | Threshold value; use when Audio Detection is Threshold; range: [0,400].  Alarm is triggered when sound volume exceeds the threshold value. | 100 |

### Capabilities

#### /LAPI/V1.0/Channels/<ID>/Alarm/Capabilities

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/Capabilities |
| **Description** | Query channel alarm capabilities information |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [AlarmCapabilities](#_AlarmCapabilities) |
| **Note** | IPC & NVR |
| **Status** |  |

### AudioDetect

#### /LAPI/V1.0/Channels/<ID>/Alarm/AudioDetection/Rule

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/AudioDetection/Rule |
| **Description** | Get audio detection configuration. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [AudioDetectInfo](#_AudioDetectInfo) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/AudioDetection/Rule |
| **Description** | Configure audio detection. |
| **Method** | PUT |
| **Input Data** | [AudioDetectInfo](#_AudioDetectInfo) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Alarm/AudioDetection/LinkageActions

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/AudioDetection/LinkageActions |
| **Description** | Query parameters of audio detection triggered action. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [LinkageActionList](#_LinkageActionList) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/AudioDetection/LinkageActions |
| **Description** | Configure parameters of audio detection triggered action. |
| **Method** | PUT |
| **Input Data** | [LinkageActionList](#_LinkageActionList) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Alarm/AudioDetection/WeekPlan

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/AudioDetection/WeekPlan |
| **Description** | Query arming schedule of audio detection. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/AudioDetection/WeekPlan |
| **Description** | Configure arming schedule of audio detection. |
| **Method** | PUT |
| **Input Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

### MotionDetection

#### /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/AreaType

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/AreaType |
| **Description** | Query area type of motion detection for specific video channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [MotionDetectionAreaType](#_MotionDetectionAreaType) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/AreaType |
| **Description** | Set area type of motion detection for specific video channel |
| **Method** | PUT |
| **Input Data** | [MotionDetectionAreaType](#_MotionDetectionAreaType) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/Areas/Rectangle

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/Areas/Rectangle |
| **Description** | Query all rectangle area information of motion detection for certain video channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [MotionDetectionRectangleAreaInfoList](#_MotionDetectionRectangleAreaInfoLis) |
| **Note** | IPC |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/Areas/Rectangle |
| **Description** | Set all rectangle area information of motion detection for certain video channel |
| **Method** | PUT |
| **Input Data** | [MotionDetectionRectangleAreaInfoList](#_MotionDetectionRectangleAreaInfoLis) |
| **Success Return Data** | None |
| **Note** | IPC |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/Areas/Rectangle/<ID>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/Areas/Rectangle/<ID> |
| **Description** | Query specific rectangle area information of motion detection for certain video input channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [MotionDetectionRectangleAreaInfo](#_MotionDetectionRectangleAreaInfo) |
| **Note** | IPC |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/Areas/Rectangle/<ID> |
| **Description** | Set specific rectangle area information of motion detection for certain video input channel |
| **Method** | PUT |
| **Input Data** | [MotionDetectionRectangleAreaInfo](#_MotionDetectionRectangleAreaInfo) |
| **Success Return Data** | None |
| **Note** | IPC |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/Areas/Grid

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/Areas/Grid |
| **Description** | Query all grid area information of motion detection for certain video input channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [MotionDetectionGridAreaInfo](#_MotionDetectionGridAreaInfo_1) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/Areas/Grid |
| **Description** | Set all grid area information of motion detection for certain video input channel |
| **Method** | PUT |
| **Input Data** | [MotionDetectionGridAreaInfo](#_MotionDetectionGridAreaInfo_1) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/LinkageActions

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/LinkageActions |
| **Description** | Query linkage action of motion detection |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [LinkageActionList](#_LinkageActionList) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/LinkageActions |
| **Description** | Set linkage action of motion detection |
| **Method** | PUT |
| **Input Data** | [LinkageActionList](#_LinkageActionList) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/WeekPlan

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/WeekPlan |
| **Description** | Query arming plan of motion detection |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/MotionDetection/WeekPlan |
| **Description** | Set arming plan of motion detection |
| **Method** | PUT |
| **Input Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

### VideoLoss

#### /LAPI/V1.0/Channels/<ID>/Alarm/VideoLoss/Rule

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/VideoLoss/Rule |
| **Description** | Query video loss configuring information for certain video input channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [VideoLossRuleInfo](#_VideoLossRuleInfo_1) |
| **Note** | NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/VideoLoss/Rule |
| **Description** | Set video loss configuring information for certain video input channel |
| **Method** | PUT |
| **Input Data** | [VideoLossRuleInfo](#_VideoLossRuleInfo_1) |
| **Success Return Data** | None |
| **Note** | NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Alarm/VideoLoss/LinkageActions

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/VideoLoss/LinkageActions |
| **Description** | Query linkage action of video loss |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [LinkageActionList](#_LinkageActionList) |
| **Note** | NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/VideoLoss/LinkageActions |
| **Description** | Set linkage action of video loss |
| **Method** | PUT |
| **Input Data** | [LinkageActionList](#_LinkageActionList) |
| **Success Return Data** | None |
| **Note** | NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Alarm/VideoLoss/WeekPlan

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/VideoLoss/WeekPlan |
| **Description** | Query arming plan of video loss |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Note** | NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/VideoLoss/WeekPlan |
| **Description** | Set arming plan of video loss |
| **Method** | PUT |
| **Input Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Success Return Data** | None |
| **Note** | NVR |
| **Status** |  |

### TamperDetection

#### /LAPI/V1.0/Channels/<ID>/Alarm/TamperDetection/Rule

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/TamperDetection/Rule |
| **Description** | Query the configuration of tamper detection for certain video input channel |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [TamperDetectionRuleInfo](#_TamperDetectionRuleInfo) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/TamperDetection/Rule |
| **Description** | Set the configuration of tamper detection for certain video input channel |
| **Method** | PUT |
| **Input Data** | [TamperDetectionRuleInfo](#_TamperDetectionRuleInfo) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Alarm/TamperDetection/LinkageActions

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/TamperDetection/LinkageActions |
| **Description** | Query linkage action of tamper detection |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [LinkageActionList](#_LinkageActionList) |
| **Note** | IPC & NVR支持。 |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/TamperDetection/LinkageActions |
| **Description** | Set linkage action of tamper detection |
| **Method** | PUT |
| **Input Data** | [LinkageActionList](#_LinkageActionList) |
| **Success Return Data** | None |
| **Note** | The linkage action of occlusion detection doesn’t support snapshot and alarm frame  IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Alarm/TamperDetection/WeekPlan

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/TamperDetection/WeekPlan |
| **Description** | Query arming plan of tamper detection |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Alarm/TamperDetection/WeekPlan |
| **Description** | Set arming plan of tamper detection |
| **Method** | PUT |
| **Input Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

## IO

### Json Block

#### InputSwitchBasciInfo

|  |  |
| --- | --- |
| InputSwitchDetailInfo  **Json Block** | {  "ID": ,  "Name": ,  "Status": ,  "GBID": ,  “VMSCommon”: ,  "RunMode": ,  "Enabled":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Alarm in channel ID  for NVR, the value is video channel ID\*100+alarm output serial number,  Equipment channel ID is 0,  for VMS, the value is serial number of equipment itself | 1 |
| Name | C | string | channel name  String length range [1,64]  NVR not supportive | "IP Camera 01" |
| Status | C | unsigned long | channel status  0:offline  1:online  for VMS use | 0 |
| GBID | C | string | GB resourses ID  string length range [0,20]  VMS & IPC supportive | "34000000002000000010" |
| VMSCommon | C | Json Block | VMS common information | 具体见[VMSCommon Json Block](#_VMSCommon) |
| RunMode | M | unsigned long | alarm type  1: Normally open  2: Normally closed | 1 |
| Enabled | M | boolean | whether enabled alarm  0:disabled  1:enabled | 1 |

#### InputSwitchBasicInfoList

|  |  |
| --- | --- |
| InputSwitchBasicInfoList Json Block | {  "Num": ,  "BasicInfos":[<[InputSwitchBasicInfo](#_InputSwitchBasciInfo)>,……]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | number of alarm input channel | 4 |
| BasicInfos | C | array | alarm input channel detailed information list  This node is optional when Num is 0 | - |
| InputSwitchBasicInfo | M | Json Block | for more details on alarm input channel, please refer to [InputSwitchBasicInfo](#_InputSwitchBasciInfo) Json Block | - |

#### OutputSwitchBasicInfo

|  |  |
| --- | --- |
| OutputSwitchBasicInfo Json Block | {  "ID": ,  "Name": ,  "Status": ,  “VMSCommon”: ,  "RunMode": ,  "Duration":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | alarm outchannel ID  for NVR, the value is video channel ID\*100+alarm output serial number,  Equipment channel ID is 0,  for VMS, the value is serial number of equipment itself | 1 |
| Name | C | string | channel name  String length range [1,64]  NVR not supportive | "O\_AlarmOutput\_2" |
| Status | C | unsigned long | channel status  0:offline  1:online  needed forVMS | 0 |
| VMSCommon | C | Json Block | VMS common information | See [VMSCommon Json Block](#_VMSCommon) |
| RunMode | M | unsigned long | alarm type  1:always on  2:always off | 1 |
| Duration | M | unsigned long | duration time  for NVR, VMS and conventional camera, range [5,3600], unit: second  for License Plate Capture Camera, range [500,3600000], unit: millisecond  for Smart camera, range [1,3600], unit: second  All ranges list above should convert to millisecond | 6 |

#### OutputSwitchBasicInfoList

|  |  |
| --- | --- |
| OutputSwitchBasicInfoList Json Block | {  "Num": ,  "BasicInfos":[<[OutputSwitchBasicInfo](#_OutputSwitchBasicInfo)>,……]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | number of alarm output channel | 4 |
| BasicInfos | C | array | alarm output channel detailed information list  This node is optional when Num is 0 | - |
| OutputSwitchBasicInfo | M | Json Block | for more details on alarm output channel, please refer to [OutputSwitchBasicInfo](#_OutputSwitchBasicInfo) Json Block | - |

#### SetOutputSwitchManualAlarmList

|  |  |
| --- | --- |
| SetOutputSwitchManualAlarmList  Json Block | {  "Num": ,  "IDs": [<ID>, …],  "AlarmAction":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | channel number of alarm out with change of alarm status | 4 |
| IDs | C | array | alarm out ID list  This node is optional when Num is 0 | - |
| ID | M | unsigned long | alarm out ID  for NVR, the value is video channel ID\*100+alarm output serial number,  Equipment channel ID is 0,  for VMS, the value is serial number of equipment itself | 1 |
| AlarmAction | M | unsigned long | alarm action  0:alarm disabled  1:alarm enabled | 1 |

#### OutputSwitchStatus

|  |  |
| --- | --- |
| OutputSwitchStatus Json Block | {  “ID”: ,  “Status”:  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Param | Requirement | Type | Description | Example |
| ID | M | unsigned  long | the value is channel ID\*100+alarm output serial number,  Equipment channel ID is 0,  e.g. : the first ID of switch value of IPC:0\*100+1=1 | 1 |
| Status | M | unsigned  long | switch value output status  0:off, closed circuit  1:on, open circuit | 0 |

#### OutputSwitchStatusList

|  |  |
| --- | --- |
| OutputSwitchStatusList Json Block | {  "Nums":,  "Statuses":[<[OutputSwitchStatus](#_OutputSwitchStatus_1)>, <[OutputSwitchStatus](#_OutputSwitchStatus_1)>...]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Param | Requirement | Type | Description | Example |
| Nums | M | unsigned long | number of alarm out | 1 |
| Statuses | C | array | list of alarm out, this field is optional when Num is 0 | - |
| OutputSwitchStatus | M | Json Block | for more details on output switch value, please refer to [OutputSwitchStatus](#_OutputSwitchStatus_1) Json Block | - |

#### OutputSwitchAlarmStatusList

|  |  |
| --- | --- |
| OutputSwitchAlarmStatusList Json Block | {  "Num": ,  "AlarmStatusList": [<[AlarmStatusInfo](#_AlarmStatusInfo)>, …]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Ranger** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | channel number of equipment alarm output | 1 |
| AlarmStatusList | C | array | alarm status list of alarm out port  this field is optional when Num is 0 | - |
| AlarmStatusInfo | C | Json Block | for more details on output switch value alarm status list, please refer to [AlarmStatusInfo](#_AlarmStatusInfo) Json Block | - |

#### AlarmStatusInfo

|  |  |
| --- | --- |
| AlarmStatusInfo Json Block | {  "ID": ,  "AlarmStatus":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Ranger** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | alarm out ID  for NVR, the value is video channel ID\*100+alarm output serial number,  Equipment channel ID is 0,  for VMS, the value is serial number of equipment itself | 1 |
| AlarmStatus | M | unsigned long | alarm status of alarm out  0:alarm disabled  1:alarm enabled | 1 |

### InputSwitch

#### /LAPI/V1.0/IO/InputSwitches/BasicInfos?[DevID=<DevID>]&[OrgID=<OrgID>]

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/InputSwitches/BasicInfos?[DevID=<DevID>]&[OrgID=<OrgID>] |
| **Description** | Query multiple alarm input channels detailed information list |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [InputSwitchBasicInfoList](#_InputSwitchBasicInfoList) |
| **Note** | IPC & NVR & VMS |
| **Status** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| DevID | O | unsigned long | device id  blank means get all equipments channels  for VMS use | 21 |
| OrgID | O | unsigned long | organization ID  blank means get channels under all organizations  for VMS use | 55 |

#### /LAPI/V1.0/IO/InputSwitches/<ID>/BasicInfos

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/InputSwitches/<ID>/BasicInfos |
| **Description** | Query specific alarm input channel detailed information |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [InputSwitchBasicInfo](#_InputSwitchBasciInfo) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/InputSwitches/<ID>/BasicInfos |
| **Description** | Set specific alarm input channel detailed information |
| **Method** | PUT |
| **Input Data** | [InputSwitchBasicInfo](#_InputSwitchBasciInfo) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | alarm input channel ID  for NVR, the value is video channel ID\*100+alarm input serial number,  Equipment channel ID is 0,  for VMS, the value is serial number of alarm input VMS has itself | 108 |

#### /LAPI/V1.0/IO/InputSwitches/<ID>/LinkageActions

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/InputSwitches/<ID>/LinkageActions |
| **Description** | Query the linkage action of alarm input port |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [LinkageActionList](#_LinkageActionList) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/InputSwitches/<ID>/LinkageActions |
| **Description** | Set the linkage action of alarm input port |
| **Method** | PUT |
| **Input Data** | [LinkageActionList](#_LinkageActionList) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/IO/InputSwitches/<ID>/WeekPlan

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/InputSwitches/<ID>/WeekPlan |
| **Description** | Query the arming plan of alarm input port |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/InputSwitches/<ID>/WeekPlan |
| **Description** | Set the arming plan of alarm input port |
| **Method** | PUT |
| **Input Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

### OutputSwitch

#### /LAPI/V1.0/IO/OutputSwitches/BasicInfos?[DevID=<DevID>]&[OrgID=<OrgID>]

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/OutputSwitches/BasicInfos?[DevID=<DevID>]&[OrgID=<OrgID>] |
| **Description** | Query multiple alarm output channels detailed information list |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [OutputSwitchBasicInfoList](#_OutputSwitchBasicInfoList) |
| **Note** | IPC & NVR & VMS |
| **Status** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| DevID | O | unsigned long | device id  blank means get all equipments channels  for VMS use | 21 |
| OrgID | O | unsigned long | organization ID  blank means get channels under all organizations  for VMS use | 55 |

#### /LAPI/V1.0/IO/OutputSwitches/<ID>/BasicInfos

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/OutputSwitches/<ID>/BasicInfos |
| **Description** | Query alarm output channel detailed information |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [OutputSwitchBasicInfo](#_OutputSwitchBasicInfo) |
| **Note** | IPC & NVR & VMS |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/OutputSwitches/<ID>/BasicInfos |
| **Description** | Set alarm output channel detailed information |
| **Method** | PUT |
| **Input Data** | [OutputSwitchBasicInfo](#_OutputSwitchBasicInfo) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | alarm output channel ID | 99 |

#### /LAPI/V1.0/IO/OutputSwitches/<ID>/WeekPlan

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/OutputSwitches/<ID>/WeekPlan |
| **Description** | Query arming plan of equipment itself or accessed equipment for output switch value |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/OutputSwitches/<ID>/WeekPlan |
| **Description** | Set arming plan of equipment itself or accessed equipment for output switch value |
| **Method** | PUT |
| **Input Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/IO/OutputSwitches/ManualAlarm

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/OutputSwitches/ManualAlarm |
| **Description** | used to manual launch or pause output switch value alarm |
| **Method** | PUT |
| **Input Data** | [SetOutputSwitchManualAlarmList](#_SetOutputSwitchManualAlarmList) |
| **Success Return Data** | None |
| **Note** | NVR & VMS |
| **Status** |  |

#### /LAPI/V1.0/IO/OutputSwitches/AlarmStatus

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/IO/OutputSwitches/AlarmStatus |
| **Description** | Query logical alarm status of output switch value |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [OutputSwitchAlarmStatusList](#_OutputSwitchAlarmStatusList) |
| **Note** | IPC & NVR & VMS |
| **Status** |  |

## Smart

### Json Block

#### IntrusionDetectionCapInfo

|  |  |
| --- | --- |
| IntrusionDetectionCapInfo Json Block | {  " SupportCfg": ,  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| SupportCfg | M | boolean | Whether supports intrusion detection:  0: No  1: Yes | 0 |

#### CrossLineDetectionCapInfo

|  |  |
| --- | --- |
| CrossLineDetectionCapInfo Json Block | {  "SupportCfg":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| SupportCfg | M | boolean | Whether supports cross line detection:  0: No  1: Yes | 0 |

#### SmartCapabilities

|  |  |
| --- | --- |
| SmartCapInfo Json Block | {  "CrossLineDetection": <[CrossLineDetectionCapInfo](#_CrossLineDetectionCapInfo)>,  "IntrusionDetection": <[IntrusionDetectionCapInfo](#_IntrusionDetectionCapInfo)> ,  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| IntrusionDetection | M | Json Block | Intrusion detection capability. See [IntrusionDetectionCapInfo](#_IntrusionDetectionCapInfo) Json Block for details. |  |
| CrossLineDetection | M | Json Block | Cross line detection capability. See [CrossLineDetectionCapInfo Json](#_CrossLineDetectionCapInfo) Block for details. |  |

#### SmartWorkingStatusInfo

|  |  |
| --- | --- |
| SmartWorkingStatusInfo Json Block | {  "EnableNum":,  "EnableIDList": [<ID>,…],  "DisableNum":,  "DisableIDList": [<ID>,…]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| EnableNum | M | unsigned long | Number of enabled smart functions among all.  Remarks:  EnableNums+DisableNums=Total number of smart functions | 6 |
| EnableIDList | M | array | ID array of enabled smart functions. | [100,101,102,103,104,105] |
| DisableNum | M | unsigned long | Number of disabled smart functions among all.  Remarks:  EnableNums+DisableNums=Total number of smart functions | 11 |
| DisableIDList | M | array | ID array of disabled smart functions | [0, 1, 2, 3, 4, 200, 201, 202, 203, 204, 205] |
| ID | M | unsigned long | Smart function ID:  0: Face detection  1: People counting  2: Skynet LPR/road monitoring  3: Track  4: Chain calculation (I)/motion detection  100: Cross line detection  101: Intrusion detection  102: Enter area  103: Leave area  104: Loitering detection  105: Quick moving (100-105 are defined as SMART1)  200: People gathering  201: Parking detection;  202: Object left behind  203: Object removed  204: Defocus detection  205: Scene change  206: 206: Heat map (200-206 are defined as SMART2) | 1 |

#### CrossLineDetectionRuleInfo

|  |  |
| --- | --- |
| CrossLineDetectionRuleInfoJson Blcok | {  "Enabled":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | M | boolean | Whether cross line detection is enabled  0: Disabled  1: Enabled | 1 |

#### CrossLineDetectionLineInfo

|  |  |
| --- | --- |
| CrossLineDetectionLineInfo Json Block | {  "ID": ,  "Enabled": ,  "Sensitivity": ,  "Direction": ,  "StartPoint": {  "X": ,  "Y":  },  "EndPoint": {  "X": ,  "Y":  }  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Detection area index, starting from 0; up to 4 areas are supported. | 1 |
| Enabled | M | boolean | Whether detection area is enabled  0: Disabled  1: Enabled | 1 |
| Sensitivity | M | unsigned long | Sensitivity, range: [1,100] | 1 |
| Direction | M | unsigned long | Direction  0: A < - > B, bi-directional  1: B -> A, from B to A  2: A -> B, from A to B | 1 |
| StartPoint | M | Json Block | Coordinate of start point of detection line |  |
| EndPoint | M | Json Block | Coordinate of end point of detection line |  |
| X | M | unsigned long | X-axis coordinate, range: [0,10000] | 5000 |
| Y | M | unsigned long | Y-axis coordinate, range: [0,10000] | 5000 |

#### CrossLineDetectionAreaInfoList

|  |  |
| --- | --- |
| CrossLineDetectionAreaInfoList Json Block | {  "Num": ,  "LineInfoList": [<[CrossLineDetectionLineInfo](#_CrossLineDetectionLineInfo)>,…]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | Max number of areas that user can draw; the default is 4 for cross line detection | 4 |
| LineInfoList | C | array | Detection area parameter list; not needed when Num=0. |  |
| CrossLineDetectionLineInfo | M | Json Block | Detection area parameter info.  See [CrossLineDetectionLineInfo](#_CrossLineDetectionLineInfo) Json Block for details. |  |

#### IntrusionDetectionRuleInfo

|  |  |
| --- | --- |
| IntrusionDetectionRuleInfoJson Blcok | {  "Enabled":  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Enabled | M | boolean | Whether intrusion detection is enabled:  0: Disabled  1: Enabled | 1 |

#### IntrusionDetectionAreaInfoList

|  |  |
| --- | --- |
| IntrusionDetectionAreaInfoList Json Block | {  "Num": ,  "PolygonInfoList": [<[IntrusionDetectionPolygonInfo](#_IntrusionDetectionPolygonInfo)>,…]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| Num | M | unsigned long | Max number of areas that user can draw; the default is 4 for intrusion detection | 4 |
| PolygonInfoList | C | array | Detection area parameter list, not needed when Num=0. |  |
| IntrusionDetectionPolygonInfo | M | Json Block | Detection area parameter info.  See [IntrusionDetectionPolygonInfo](#_IntrusionDetectionPolygonInfo) Json Block for details. |  |

#### IntrusionDetectionPolygonInfo

|  |  |
| --- | --- |
| IntrusionDetectionPolygonInfo Json Block | {  "ID": ,  "Enabled": ,  "Sensitivity": ,  "Percentage": ,  "TimeThreshold": ,  "PointNum": ,  "PointList": [  {  "X": ,  "Y":  },  …  ]  } |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Param** | **Requirement** | **Type** | **Description** | **Example** |
| ID | M | unsigned long | Detection area index, starting from 0; up to 4 areas are supported. | 1 |
| Enabled | M | boolean | Whether detection area is enabled  0: Disabled  1: Enabled | 1 |
| Sensitivity | M | unsigned long | Sensitivity, range: [1,100] | 1 |
| Percentage | M | unsigned long | Percentage, range: [1,100] | 1 |
| TimeThreshold | M | unsigned long | Time threshold value, range: [1,10] | 1 |
| PointNum | M | unsigned long | Number of points of polygon detection area; range: [0,6]. | 4 |
| PointList | C | array | Coordinates of points of detection area; not needed when PointNum=0. |  |
| X | M | unsigned long | X-axis coordinates, range: [0, 10000] | 5000 |
| Y | M | unsigned long | Y-axis coordinates, range: [0, 10000] | 5000 |

### Capabilities

#### /LAPI/V1.0/Channels/<ID>/Smart/Capabilities

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/Capabilities |
| **Description** | Query channel's smart capability. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [SmartCapabilities](#_SmartCapInfo) |
| **Note** | IPC & NVR |
| **Status** |  |

### CrossLineDetection

#### /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/Rule

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/Rule |
| **Description** | Get cross line detection configuration for specified video channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [CrossLineDetectionRuleInfo](#_CrossLineDetectionRuleInfo) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/Rule |
| **Description** | Set cross line detection configuration for specified video channel. |
| **Method** | PUT |
| **Input Data** | [CrossLineDetectionRuleInfo](#_CrossLineDetectionRuleInfo) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/Areas

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/Areas |
| **Description** | Get all cross line detection areas for specified video channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [CrossLineDetectionAreaInfoList](#_CrossLineDetectionAreaInfoList) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/Areas |
| **Description** | Set all cross line detection areas for specified video channel. |
| **Method** | PUT |
| **Input Data** | [CrossLineDetectionAreaInfoList](#_CrossLineDetectionAreaInfoList) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/Areas/<ID>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/Areas/<ID> |
| **Description** | Get specified cross line detection area for specified video channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [CrossLineDetectionLineInfo](#_CrossLineDetectionLineInfo) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/Areas/<ID> |
| **Description** | Set specified cross line detection area for specified video channel. |
| **Method** | PUT |
| **Input Data** | [CrossLineDetectionLineInfo](#_CrossLineDetectionLineInfo) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/LinkageActions

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/LinkageActions |
| **Description** | Query triggered actions of cross line detection. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [LinkageActionList](#_LinkageActionList) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/LinkageActions |
| **Description** | Set triggered action for cross line detection. |
| **Method** | PUT |
| **Input Data** | [LinkageActionList](#_LinkageActionList) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/WeekPlan

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/WeekPlan |
| **Description** | Query arming schedule for cross line detection. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/CrossLineDetection/WeekPlan |
| **Description** | Set arming schedule for cross line detection. |
| **Method** | PUT |
| **Input Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

### IntrusionDetection

#### /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/Rule

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/Rule |
| **Description** | Get intrusion detection configuration for specified video channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [IntrusionDetectionRuleInfo](#_IntrusionDetectionRuleInfo) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/Rule |
| **Description** | Set intrusion detection configuration for specified video channel. |
| **Method** | PUT |
| **Input Data** | [IntrusionDetectionRuleInfo](#_IntrusionDetectionRuleInfo) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/Areas

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/Areas |
| **Description** | Get all intrusion detection areas for specified video channel. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [IntrusionDetectionAreaInfoList](#_IntrusionDetectionAreaInfoList) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/Areas |
| **Description** | Set all intrusion detection areas for specified video channel. |
| **Method** | PUT |
| **Input Data** | [IntrusionDetectionAreaInfoList](#_IntrusionDetectionAreaInfoList) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/Areas/<ID>

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/Areas/<ID> |
| **Description** | Get intrusion detection info. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [IntrusionDetectionPolygonInfo](#_IntrusionDetectionPolygonInfo) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/Areas/<ID> |
| **Description** | Set intrusion detection info. |
| **Method** | PUT |
| **Input Data** | [IntrusionDetectionPolygonInfo](#_IntrusionDetectionPolygonInfo) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/LinkageActions

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/LinkageActions |
| **Description** | Query triggered action of intrusion detection. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [LinkageActionList](#_LinkageActionList) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/LinkageActions |
| **Description** | Set triggered action for intrusion detection. |
| **Method** | PUT |
| **Input Data** | [LinkageActionList](#_LinkageActionList) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |

#### /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/WeekPlan

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/WeekPlan |
| **Description** | Query arming schedule of intrusion detection. |
| **Method** | GET |
| **Input Data** | None |
| **Success Return Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Note** | IPC & NVR |
| **Status** |  |

|  |  |
| --- | --- |
| **URL** | /LAPI/V1.0/Channels/<ID>/Smart/IntrusionDetection/WeekPlan |
| **Description** | Set arming schedule for intrusion detection. |
| **Method** | PUT |
| **Input Data** | [WeekPlanInfo](#_WeekPlanInfo_1) |
| **Success Return Data** | None |
| **Note** | IPC & NVR |
| **Status** |  |