

Welcome to use Device HTTP API!

The Device HTTP API is a programming interface for this model network decoding and software development. Aimed at simple HTTP request/responding mechanism, Device HTTP wants to realize various functions controlling for device. Software developers make use of Device HTTP API to meet the purpose of remote manage and device control.

Each function of API corresponds to one HTTP request path. In the current version, the path format is: `http://[:]/api/[module name]/[method].json`

This file will list each API's corresponding filename and functions description. Meanwhile, in each API description file, there will be a test interface. You could write your request parameters according to parameter description, executing test, testifying API function online, very convenient.

API requests follow the standard HTTP protocol, and developers can use any tools and software development library that according to HTTP standard (For example Web browsers, JavaScript library, C/C++ HTTP protocol file, JAVA, .NET, etc.) The submission of parameters might be POST or GET, or both, depending on different API interfaces request, please refer to API instruction for details. API's response is in JSON format. For the meaning of each parameter in the JSON object, please refer to each API instruction.

We maintain the conformity of API interfaces, but there may some differences due to products features difference leading different API interfaces. So please distinguish and deal with these minor differences according to official guidance.,

1. Video Source

Module name: capture

Basic URL: `/api/codec/capture/`

1.1 Get video source information

API URL

`/api/codec/capture/GetInfo.json`

Request

Method:**GET/POST**Parameters: NONE

Response

Example:

```
{
  "result": "ok",
  "data": [
    {
      "rotate": 0,
      "videoSource": "hdmi",
    }
  ]
}
```

```

    "sourceSize": "1280x720",
    "sourceFrameRate": 60,
    "bMirror": true,
    "bFlip": false,
    "contrast": 128,
    "saturation": 128,
    "brightness": 128
  }
]
}

```

Data Field Description:

Field	Value	Instruction
rotate	[Integer]	0/90/180/270 rotation angle
videoSource	[String]	hdmi/sdi video source interface
sourceSize	[String]	1920x1080 video source resolution
rotate	[Integer]	60 video source framerate
bMirror	[Boolean]	true/false turn on horizontal flip (mirroring) or not
bFlip	[Boolean]	true/false vertical flip (mirroring) or not
contrast	[Integer]	contrast ratio, range value: [0,255]
saturation	[Integer]	saturation, range value: [0,255]
brightness	[Integer]	brightness, range value: [0,255]

1.2 Set rotation, mirroring

API URL

/api/codec/capture/UpdateDynamic.json

Request

Method: **GET/POST** Parameters: NONE

```

{
  cfg: {
    rotate = 90,
    bMirror = true/false,
    bFlip = true/false,
  }
}

```

Response

Example:

```
{
  result: "ok",
  data: true
}
```

Data**Field Description: **

Field	Value	Instruction
rotate	[Integer]	0/90/180/270 rotation angle
bMirror	[Boolean]	true/false horizontal flip is enabled or not
bFlip	[Boolean]	true/fas vertical flip is enabled or not

1.3 Set color adjustment

API URL

/api/codec/capture/UpdateColor.json

Request

Method:**GET/POST**Parameters: NONE

```
{
  "chnId" : 1,
  "brightness" : 128,
  "contrast": 128,
  "saturation": 128
}
```

Response

Example:

```
{
  result: "ok",
  data: true
}
```

2. Audio

Module name: audio

Basic URL: /api/codec/audio/

2.1 Get audio source information

API URL

/api/codec/audio/GetInfo.json

Request

Method:**GET/POST**Parameters: NONE

Response

Example:

```
{
  "result": "ok",
  "data": {
    "source": {
      "audioSource": "hdmi", //Audio source information, from hdmi or linein
    },
    "audioEncode": [
      {
        "name" : "Channel-1",
        "resample": "fastest",
        "sampling": 48000,
        "channels": 2,
        "aacType": "AAC",
        "bitrate": 64000,
      },
      {
        "name" : "Channel-1",
        "g711Law": "PCMU",
        "resample": "fastest",
        "aacType": "G711",
        "bitrate": 64000
      }
    ]
  }
}
```

Data Field Description:

Field	Value	Instruction
resample	[String]	fastest/common fastest fast/fair sound quality, common high sound quality/higher CPU consumption
source.audioSource	[String]	hdmi/Line-IN hdmi indicates source is hdmi audio, Line-IN indicates that the source is analog audio
aacType	[String]	AAC/G711 indicates the audio encoding format
channels	[Integer]	1/2 indicates the number of sound channels: 1 indicates mono 2 indicates stereo
bitrate	[Integer]	64000 indicates 64Kbps encoding bit rate
sampling	[Integer]	48000 indicates 48KHz sampling rate
g711Law	[String]	PCMU/PCMA indicates G711 type

2.2 Update audio settings

API URL

/api/codec/audio/Update.json

Request

Method:**GET/POST**Parameters:

```
{
  "chnId" : 1,
  "new" : {
    "name" : "Channel-1",
    "sampling": 8000,
    "channels": 2,
    "aacType": "G711",
    "bitrate": 64000
  }
}
```

The chnId here, is the array subscript when getting the audio encoding channel (the array subscript starts counting from 1)

Response

Example:

```
{
  result: "ok",
  data: true
}
```

Request Data**Field Description: **

Field	Value	Description
chnId	[integer]	Indicates the channel number, just fill in 1 in the current version
new	[Object]	Indicates a newly set data object that supports local updates, i.e., only a part of the key-value is passed to update it

2.3 Add audio channel

API URL

** /api/codec/audio/AddAudioEncode.json**

Request

Method:**GET/POST**Parameters:

```
{
  "cfg" : {
    "name" : "Channel-1",
    "sampling": 8000, //Sampling rate
    "channels": 2, // Stereo, mono
    "aacType": "G711", // Encoding format
    "bitrate": 64000 //Bitrate
  }
}
```

Response

Example:

```
{
  result: "ok",
  data: true
}
```

2.4 Delete audio encoding channel

API URL

** /api/codec/audio/RemoveAudioEncode.json**

Request

Method:**GET/POST**Parameters:

```
{
  "chnId" : 1
}
```

Response

Example:

```
{
  result: "ok",
  data: true
}
```

3. Video Encoding

Module name: codec

Basic URL: /api/codec/codec/

3.1 Get audio source information

API URL

/api/codec/codec/GetInfo.json

Request

Method:**GET/POST**

Parameters: [stream]

Optional parameters stream: "main"/"sub" main indicates the main stream, sub indicates sub-stream.

Response

Example:

```
{
  "result": "ok",
  "data": {
    "ve": {
      "u32Profile": 1,
      "u32PicHeight": 720,
      "u32Picwidth": 1280
    },
    "type": "H264",
    "rc": {
      "u32BitRate": 6000,
      "mode": "CBR",
      "u32MinQp": 10,
      "u32MinIQp": 29,
      "u32Gop": 60,
      "fr32DstFrmRate": 60,
      "u32MaxQp": 48,
      "u32MaxBitRate": 6000,
    },
    "grey": false
  }
}
```

DataField Description: ****

Field	Value	Instruction
type	[String]	H264/H265 encoding method
grey	[Boolean]	true/false turn on/off ash
ve.u32Profile	[Integer]	0/1/2 0:Baseline 1:Main Profile 2: High Profile
ve.u32PicHeight	[Integer]	Encoding image height
ve.u32PicWidth	[Integer]	Encoding image width
rc.mode	[String]	CBR/VBR code rate control method
rc.u32GOP	[Integer]	GOP size
rc.fr32DstFrmRate	[String]	Encoding framerate will not actually be higher than the source frame rate
rc.u32BitRate	[Integer]	6000 6000K encoding code rate, applicable to CBR mode
rc.u32MaxBitRate	[Integer]	20000 maximum bit rate, applicable to VBR mode
rc.u32MinIqp	[Integer]	29 IQP value, which must be between rc.u32MaxQp and rc.u32MinQp
rc.u32MaxQp	[Integer]	48 maximum QP value
rc.u32MinQp	[Integer]	10 minimum QP value

3.2 Update video encoding settings

API URL

/api/codec/codec/Update.json

Request

Method:**GET/POST**Parameters:

```
{
  "stream" : "main",
  "new" : {
    "ve": {
      "u32Profile": 1,
      "u32PicHeight": 720,
      "u32Picwidth": 1280
    },
    "type": "H264",
    "rc": {
      "u32BitRate": 6000,
      "mode": "CBR",

```



```

        "u32MinQp": 10,
        "u32MinIQp": 29,
        "u32Gop": 60,
        "fr32DstFrmRate": 60,
        "u32MaxQp": 48,
        "u32MaxBitRate": 6000,
    },
    "grey": false
}
}

```

Response

Example:

```
{result: "ok", data: true}
```

Request Data**Field Description: **

Field	Value	Instruction
stream	[String]	Encoding stream, main stream and sub stream respectively
new	[Object]	A newly set data object that supports partial updates, only a part of the key-value is passed to update it

3.3 Get real-time encoding code rate

API URL

/api/codec/codec/GetRealBitrate.json

Request

Method:**GET/POST**Parameters:

```

{
  "stream" : "main",
}

```

Response

Example:

```
{"result": "ok", "data": 56 // Real-time code rate in kb}
```

Request Data**Field Description: **

Field	Value	Instruction
stream	[String]	Encoding stream, main stream and sub stream respectively

3.4 Cropping information acquisition

API URL

/api/codec/codec/GetCrop.json

Request

Method:**GET/POST**Parameters: NONE

Response

Example:

```
{
  "result": "ok", "data": { "enable": false, "s32Y": 25, "u32width": 50,
    "u32Height": 50, "s32X": 25 }}
```

Data**Field Description: **

Field	Value	Instruction
enable	[Boolean]	true/false crop function is enabled or not
s32Y	[Integer]	[0,100] vertical offset percentage
s32X	[Integer]	[0,100] vertical offset percentage
u32Width	[Integer]	[0,100] cropping width percentage
u32Height	[Integer]	[0,100] cropping height percentage

3.5 Set Cropping

API URL

/api/codec/codec/SetCrop.json

Request

Method:**GET/POST**Parameters: NONE

```
{ "cfg": { "enable": false, "s32Y": 25, "u32width": 50, "u32Height": 50,
  "s32X": 25 }}
```

Response

Example:

```
{"result": "ok"}
```

Data**Field Description: **

Field	Value	Instruction
enable	[Boolean]	true/false crop function is enabled or not
s32Y	[Integer]	[0,100] vertical offset percentage
s32X	[Integer]	[0,100] lateral offset percentage
u32Width	[Integer]	[0,100] cropping width percentage
u32Height	[Integer]	[0,100] cropping height percentage

4. Streaming Service

Module name: streamer **Basic URL:** /api/codec/streamer/

4.1 Get stream information

API URL

/api/codec/streamer/GetInfo.json

Request

Method:**GET/POST**Parameters:

```
{"stream" : "main",}
```

Response

Example:

```

{"result":"ok"
,
"data":
{"main":
[{"enable":true,"NDIname":"Chan1","status":"start","addressUrl":"","discovery_server":"","netmask":"255.255.0.0","netprefix":"239.255.0.0","connection":"tcp","group":"","type":"ndi_hx","ttl":"1","name":"NDI-HX","bindAudio":1},
{"enable":true,"multicast_addr":"224.0.0.1","httpTunnelPort":8554,"session":"ch01","multicast_video_port":3101,"auth":false,"addressUrl":"rtsp://\/%HOST%:554/c/h01","type":"rtsp","multicast_audio_port":3102,"multicast_ttl":127,"status":"start","ssm":0,"no_adts":0,"bindAudio":1,"name":"RTSP","port":554},
{"payloadSize":1316,"srt_stream_id":"","passphrase":"","status":"start","bandwidth":5,"addressUrl":"srt://\/%HOST%:8083","connectionMode":"Listener","address":"","enable":true,"latency":125,"listenerPort":8083,"type":"SRT","bindAudio":2,"name":"SRT","encryption":0}], "sub":{}}
}

```

DataField Description: ****

Field	Value	Instruction
main	[Object]	Stream information of main stream
sub	[Object]	Stream information of sub-stream
type	[String]	stream type
enable	[Boolean]	turn on/off
name	[String]	stream name
status	[String]	Stream status. start means it is streaming stop means streaming is paused error means there is stream error destroy steam has been destroyed

For information on each of the different stream parameters, please refer to the appendix.

4.2 Add Streaming Services

API URL

/api/codec/streamer/AddStreamer.json

Request

Method:**GET/POST**Parameters:

```

{"stream" : "main","cfg": {"type": "rtsp","enable": true/false,"name" : "Name1",}}

```

Response

```
{"result": "ok"}
```

4.3 Delete Streaming Service

API URL

/api/codec/streamer/DeleteStreamer.json

Request

Method:**GET/POST**Parameters:

```
{"stream": "main", "id": 2}
```

Response

```
{"result": "ok"}
```

Request Data**Field Description: **

Field	Value	Instruction
stream	[String]	Code stream main main stream sub sub-stream
id	[Integer]	Stream id, get list is array subscript

The array subscript starts counting from 1.

4.4 Modify Streaming Service

API URL

/api/codec/streamer/UpdateStreamer.json

Request

Method:**GET/POST**Parameters:

```
{"stream": "main", "id" : 2, "cfg" : {"enable" : true, "name" : "Name2"...}}
```

Modifying the type of the stream is not supported.

Response

```
{"result": "ok"}
```

5. Network Module

Module name: network **Basic URL:** /api/network/network/

5.1 Get wired connection status

API URL

/api/network/network/GetLinkDetail.json

Request

Method: **GET/POST** Parameters: NONE

Response

```
{
  "result": "ok",
  "data": {
    "netmask": "255.255.255.0",
    "method": "static",
    "mac": "0E:62:80:BB:60:7D",
    "dns": "8.8.8.8",
    "status": "up",
    "gw": "192.168.0.1",
    "aliases": {
      "enable": true,
      "netmask": "255.255.255.0",
      "gw": "",
      "address": "192.168.1.131",
    },
    "address": "192.168.0.132"
  }
}
```

** Data Field Description **

Field	Value	Instruction
netmask	[String]	subnet mask
method	[String]	address acquisition method static manual assignment dhcp automatic acquisition
mac	[String]	MAC address
dns	[String]	DNS address
status	[String]	Ethernet status up connected down unconnected
gw	[String]	Gateway
address	[String]	IP address
aliases.enable	[Boolean]	Maintenance address on or off
aliases.address	[String]	Maintenance address IP
aliases.gw	[String]	Maintain address gateway
aliases.netmask	[String]	Maintenance address subnet mask

5.2 Set up a wired connection

API URL

/api/network/network/ChangeConnection.json

Request

Method:**GET/POST**Parameters:

```
{ "cfg": { "netmask": "255.255.255.0", "method": "static", "mac": "0E:62:80:BB:60:7D", "dns": "8.8.8.8", "status": "up", "gw": "192.168.0.1", "aliases": { "enable": true, "netmask": "255.255.255.0", "gw": "", "address": "192.168.1.131" }, "address": "192.168.0.132" }
```

Response

```
{ "result": "ok" }
```

6. Advanced Settings

Module name: webDaemon **Basic URL:** /api/systemctrl/webDaemon/

6.1 Get web port information

API URL

/api/systemctrl/webDaemon/GetInfo.json

Request

Method:**GET/POST**Parameters: NONE

Response

```
{ "result": "ok", "data": { "https_port": 443, "port": 80 }}
```

Data Field Description

Field	Value	Instruction
https_port	[Integer]	HTTPS port
port	[Integer]	HTTP port

6.2 Set the web port

API URL

/api/systemctrl/webDaemon/UpdateNginx.json

Request

Method:**GET/POST**Parameters:

```
{"port": 80, "https_port": 443}
```

Both parameters of this interface are required.

Response

```
{ "result": "ok", }
```

7. OSD overlay

Module name: osdBasic URL: /api/codec/osd/

7.1 Get overlay information

API URL

```
/api/codec/osd/GetInfo.json
```

Request

Method:**GET/POST**Parameters: [chnlId]

```
{"chnlId": 1, // 1 indicates the main stream, 2 indicates sub-stream}
```

Response

```
{ "result": "ok", "data": { "config": { "enable": true, }, "items": [ [1]
= { "enable": true, "position": "center", "color": 1, "y": 0,
"style": "normal", "text": "This is the 132 overlay item", "fileID": "",
"outline": 0, "type": "text", "outlineColor": 0, "x": 0, "size": 48
}, ... ] }}
```

Data Field Description

Field	Value	Instruction
config.enable	[Boolean]	Character overlay is on/off
enable	[Boolean]	Overlay is on/off
type	[String]	Overlay item type: time : '%Y/%m/%d %H:%M:%S' time of format date : '%Y/%m/%d' time of format clock : '%H:%M:%S' time of format image : image text : text
position	[String]	Overlay position: left-top : top left corner top-center : top centered right-top : top right corner center : main center left-bottom : lower left corner bottom-center : centered below right-bottom : lower right corner custom : customization
size	[Integer]	text size, 0, 1, 2 adaptive font size, the rest is the corresponding px size: 0 : automatic - large font 1 : automatic - medium font 2 : auto-smaller font For example 99px
style	[String]	normal normal font bold bold font
outline	[Integer]	Text outer border px value
color	[Integer]	Text color: 0 : Black 1 : White 2 : Red 3 : Blue 4 : Green 5 : Yellow 6 : Purple 7 : Cyan
outlineColor	[Integer]	outer border text color: 0 : Black 1 : White 2 : Red 3 : Blue 4 : Green 5 : Yellow 6 : Purple 7 : Cyan
text	[String]	Text content
x	[Integer]	X-axis coordinates when indicating a custom position
y	[Integer]	Y-axis coordinates when indicating a custom position
fileID	[Integer]	overlay image ID, this ID must exist in the image list

7.2 Update the whole settings of the code stream

API URL

/api/codec/osd/UpdateChannel.json

Request

Method:**GET/POST**Parameters:

```
{"chnId" : 1, // 1 indicates the main stream, 2 indicates sub-stream"cfg": {"enable": true}}
```

Response

```
{ "result": "ok", }
```

7.3 Update overlay items

API URL

/api/codec/osd/UpdateChannel.json

Request

Method:**GET/POST**Parameters:

```
{"chnId" : 1, "idx" : 1, "cfg": {"enable": true,"size" : 1,...}}
```

Response

```
{ "result": "ok", }
```

Request Data Field Description

Field	Value	Instruction
chnId	[Integer]	1 : main stream 2 : sub-stream
idx	[Integer]	The id of this overlay item is the subscript of the array in items, and the subscript starts counting from 1
cfg	[Object]	overlay item attribute, with the same structure as above

7.4 New overlay items

API URL

/api/codec/osd/AddOSDItem.json

Request

Method:**GET/POST**Parameters:

```
{"chnId" : 1, "cfg": {"enable": true,"size" : 1,...}}
```

Response

```
{ "result": "ok", }
```

** Request Data Field Description **

Field	Value	Instruction
chnId	[Integer]	1 : main stream 2 : sub-stream
cfg	[Object]	Overlay item attribute, with the same structure as above

7.5 Delete Overlay

API URL

/api/codec/osd/DelOSDItem.json

Request

Method: **GET/POST** Parameters:

```
{"chnId" : 1, "idx" : 1, }
```

Response

```
{ "result": "ok", }
```

** Request Data Field Description **

Field	Value	Instruction
chnId	[Integer]	1 : main stream 2 : sub-stream
idx	[Integer]	Overlay item id

8. Image Management

Module name: picBasic URL: /api/pic/

8.1 Get a list of images

API URL

/api/pic/get.json

Request

Method:**GET/POST**Parameters: NONE

Response

```
{ "result": "ok", "data": [ { "size_h": 3648, "id": 1, "url":  
"/img/3821563.jpg", "size_w": 4576, "name": "1", "rgba":  
"/data/storage/img/3821563.jpg.rgb" } ]}
```

Data Field Description

Field	Value	Instruction
id	[Integer]	image id
url	[Integer]	Image url, use the device address to joint this url to get the image, for example: 192.168.1.168/img/1.jpg
size_h	[Integer]	image height
size_w	[Integer]	image width
name	[Integer]	image Name

8.2 Delete image

API URL

/api/pic/remove.json

Request

Method:**GET/POST**Parameters:

```
{"id" : 1}
```

Response

```
{ "result": "ok",}
```

Request Data Field Description

Field	Value	Instruction
id	[Integer]	image id

9.PTZ

9.1 PTZ Information Access

API URL

/api/onvif/PTZ/GetInfo.json

Request

Method:**GET/POST**Parameters:

Response

```
{ "result": "ok", "data": { "deviceType": "serial", "options": {  
  "zoom_speed": 0.5, "iris_speed": 0.5, "focus_speed": 0.5, "pan_speed": 0.5,  
  "protocol": "null", "tilt_speed": 0.5, "address": 1 }, "device": {  
  "rtscts": 0, "device": "null", "databits": 8, "xonxoff": 0, "stopbits":  
  1, "parity": "none", "baudrate": 9600 }, "socket_svc": { "addr":  
  "192.168.0.1", "protocol": "TCP", "port": 10232 } }}
```

Request Data Field Description

Field	Value	Instruction
deviceType	[String]	Selected PTZ type "serial": serial port "net": network PTZ
options.pan_speed	[Integer]	default translation speed, range value: [0,1]
options.iris_speed	[Integer]	default aperture speed, range value: [0,1]
options.focus_speed	[Integer]	default focus speed, range value: [0,1]
options.zoom_speed	[Integer]	default scaling speed, range value: [0,1]
options.tilt_speed	[Integer]	default drape speed, range value: [0,1]
options.protocol	[String]	PTZ protocol, null means none
options.address	[Integer]	PTZ address
device.rtscts	[Integer]	RTS/CTS switch 0 means off, 1 means on
device.xonxoff	[Integer]	XON/XOFF switch 0 means off, 1 means on
device.databits	[Integer]	data bits
device.stopbits	[Integer]	stop bit
device.device	[String]	selected serial device, null is none
device.parity	[String]	checksum, none
device.baudrate	[Integer]	baud rate
socket_svc.protocol	[String]	network PTZ protocol type: TCP/UDP
socket_svc.addr	[String]	network PTZ address
socket_svc.port	[Integer]	network PTZ port

9.2 Update PTZ settings

API URL

/api/onvif/PTZ/Update.json

Request

Method:**GET/POST**Parameters:

```
{ "deviceType": "serial", "options": { "zoom_speed": 0.5, "iris_speed": 0.5, "focus_speed": 0.5, "pan_speed": 0.5, "protocol": "null", "tilt_speed": 0.5, "address": 1 }, "device": { "rtscts": 0, "device": "null", "databits": 8, "xonxoff": 0, "stopbits": 1, "parity": "none", "baudrate": 9600 }, "socket_svc": { "addr": "192.168.0.1", "protocol": "TCP", "port": 10232 } }
```

Response

```
{ data: true result: "ok"}
```

9.3 PTZ serial port list

API URL

/api/onvif/PTZ/GetUSBSerials.json

Request

Method:**GET/POST**Parameters:

Response

```
{ "result": "ok", "data": { [1] = { "device" : "ttyUSB0", //device  
"driver" : "GENERIC" / "FTDI" / "PL230X" // drive},}}
```

Request Data Field Description

Field	Value	Instruction
device	[String]	device
driver	[String]	drive

9.4 PTZ Control Panel

API URL

/api/onvif/PTZ/PTZControl.json

Request

Method:**GET/POST**Parameters:

```
{"action": "left-up"}
```

Response

```
{ "result": "ok", "data": true}
```

Request Data Field Description

Field	Value	Instruction
action	[String]	action: "left-up":Move to the top left corner"up":Move up"right-up":Move to the upper right corner"left":Move to the left"home":Return to center position"right":Move to the right"left-down":Move to the lower left corner"down":Move down"right-down":Move to the lower right corner

10. NAS Management

10.1 Get NAS list

API URL

/api/nas_manager/nas/GetList.json

Request

Method:**GET/POST**Parameters:

Response

```
{ "result": "ok", "data": [ { "choose" : true/false, "host": "192.168.0.21",
"name": "test", "mount": "/media/nas/1", "options": "", "status":
"mounted", "id": "1", "volumn": "/mnt/disk-data/", "type": "nfs" } ] }
```

Data Field Description

Field	Value	Instruction
choose	[Boolean]	NAS is selected or not
host	[String]	mount ip
name	[String]	mount name
mount	[String]	current device directory
options	[String]	mount option
status	[String]	mount status "mounted" : connected "unmount" : unconnected"mounting" : connecting
id	[String]	ID
volumn	[String]	mount directory
type	[String]	type

10.2 New NAS

API URL

/api/nas_manager/nas/Add.json

Request

Method:**GET/POST**Parameters:

```
{id = "",name = "",type = "",host = "",volumn = "",options = "",}
```

Response

```
{ "result": "ok", "data": true}
```

10.3 Delete NAS

API URL

/api/nas_manager/nas/Remove.json

Request

Method:**GET/POST**Parameters:

```
{id = "1",}
```

Response

```
{ "result": "ok", "data": true}
```

10.4 Checked/unchecked NAS

API URL

/api/nas_manager/nas/ChooseNas.json

Request

Method:**GET/POST**Parameters:

```
{"id" : 1,"choose":true/false}
```

Response

```
{ "result": "ok", "data": true}
```

11. Recording Settings

11.1 Get recording settings

API URL

/api/record/record/GetInfo.json

Request

Method:**GET/POST**Parameters:

Response

```
{ "result": "ok", "data": { "choose": true, "disk_policy": "overwrite",  
"disk_type": "largest", "limit_type": "size-loop", "start": false,  
"limit_size": 4000000, "auto_record": true, "limit_time": 3600, "bindAudio":  
1, "file_prefix": "REC", "format": "ts" }}
```

Data Field Description

Field	Value	Instruction
choose	[Boolean]	USB disk is checked for recording or not
disk_policy	[String]	Storage policy "overwrite": overwrite old video files "stop": stop
disk_type	[String]	USB drive selection policy "largest": the largest partition "first": the first partition
limit_type	[String]	file restriction method "size-loop": size limit and automatically cut multiple files "size": size limit for single file "time-loop": duration limit and automatically cut multiple files "time": duration limit for single file "none": unlimited duration and size
start	[String]	current recording is on/off
limit_size	[String]	file size limit (in KB)
auto_record	[String]	turn on/off automatic recording
limit_time	[String]	duration limit for file (in seconds)
bindAudio	[Integer]	selected audio channel ID
file_prefix	[String]	file name prefix
format	[String]	file format

11.2 Set recording settings

API URL

/api/record/record/Update.json

Request

Method:**GET/POST**Parameters:

```
{ "cfg": { "choose": true, "disk_policy": "overwrite", "disk_type": "largest", "limit_type": "size-loop", "start": false, "limit_size": 4000000, "auto_record": true, "limit_time": 3600, "bindAudio": 1, "file_prefix": "REC", "format": "ts" } }
```

Response

```
{ "result": "ok", "data": true }
```

11.3 Get disk list

API URL

/api/record/record/GetInfo.json

Request

Method:**GET/POST**Parameters:

Response

```
{ "result": "ok", "data": { "nas": [ { "TOTAL": 1006903, "PERCENT": 7, "MOUNT": "/media/nas/1", "VOLNAME": "test", "TYPE": "nas", "FREE": 888429, "USED": 67258, "FILESYS": "nfs", "key": "/media/nas/1", "status" : "mounted", "choose" : true/false, } ], "usb": [ { "TOTAL": 29510, "PERCENT": 5, "MOUNT": "/media/usb/sda1", "VOLNAME": "", "TYPE": "usb", "FREE": 28062, "USED": 1447, "FILESYS": "ntfs", "key": "sda1", } ] }}
```

Data Field Description

Field	Value	Instruction
nas.TOTAL	[Integer]	total space (MB)
nas.PERCENT	[Integer]	percentage used
nas.MOUNT	[Integer]	mount directory
nas.VOLNAME	[String]	mount name
nas.TYPE	[String]	mount type
nas.FREE	[Integer]	remaining space
nas.USED	[Integer]	space used
nas.FILESYS	[String]	disk format
nas.key	[String]	parameters that should be taken when requesting a list of files
nas.status	[String]	pending status
nas.choose	[Boolean]	selected
nas.host	[String]	mount destination address
nas.id	[Integer]	disk id

| usb.TOTAL | **[Integer]** | total space (MB) || usb.PERCENT | **[Integer]** |percentage used ||
usb.MOUNT | **[String]** | mount directory || usb.VOLNAME | **[String]** | mount name || usb.TYPE
| **[String]** | mount type || usb.FREE | **[Integer]** | remaining space || usb.USED | **[Integer]** |
space used || usb.FILESYS | **[String]** | disk format || usb.key | **[String]** | parameters that should
be taken when requesting a list of files |

11.4 Get file list

API URL

/api/record/record/GetInfo.json

Request

Method:**GET/POST**Parameters:

```
{"device" : "sda1"}
```

Response

```
{ { startTime = 1631784815, endTime = 1631784847, stream = "main", size = 27505.0, name = "REC-20210916173333.ts", path = "/media/usb/sda1/RECORD/M-20210916_174132_0000/REC- 20210916173333.ts", current = true, }, { startTime = 1631785292, endTime = 1631787125, stream = "main", size = 1389873.0, name = "REC-20210916174132.ts" }...}
```

Data Field Description

Field	Value	Instruction
startTime	[Integer]	video start time
endTime	[Integer]	video end time
stream	[Integer]	code stream
size	[Integer]	file size in KB
name	[Integer]	file name
path	[Integer]	file path for downloading and deleting
current	[Integer]	video is being recorded or not

11.5 Delete video files

API URL

/api/record/record/RemoveFile.json

Request

Method:**GET/POST**Parameters:

```
{path = "/media/usb/sda1/RECORD/M-20210916_174132_0000/REC- 20210916173333.ts"}
```

Response

```
{ "result": "ok", "data": true}
```

Appendix: Field Descriptions of Streams in Streaming Service

rtsp

Field	Value	Instruction
session	[String]	session id
port	[Integer]	port
httpTunnelPort	[Integer]	HttpTunnel port
ssm	[Integer]	multicast is enabled or not 0 Indicates off 1 : Indicates on
multicast_addr	[String]	multicast address
multicast_ttl	[Integer]	multicast TTL
multicast_video_port	[Integer]	multicast video port
multicast_audio_port	[Integer]	multicast audio port

rtmp

Field	Value	Instruction
address	[String]	push address
user	[String]	user name
pass	[String]	password
connTimeout	[String]	timeout connection seconds
connIntv	[String]	reconnection interval

srt

Field	Value	Instruction
address	[String]	Stream address
connectionMode	[String]	listener mode Listener : Listener mode Caller : Caller mode
encryption	[Integer/String]	Encryption method : 0 none, not encrypted AES-128 : AES-128 encryption AES-192 : AES-192 encryption AES-256 : AES-256 encryption
passphrase	[String]	AES Key
latency	[Integer]	Transmission delay
bandwidth	[Integer]	Bandwidth Overhead
payloadSize	[Integer]	Load size
listenerPort	[Integer]	Port
srt_stream_id	[String]	SRT stream ID

TS-UDP

Field	Value	Instruction
address	[String]	Push Address
port	[Integer]	Target port
ttl	[String]	Multicast ttl
ts_pts_pcr_delay	[String]	pcr delay
ts_service_name	[String]	service name
ts_null_multiple	[String]	TS Encapsulated code rate: 2000000 2mb/s
ts_pcr_period	[String]	pcr period
ts_service_provider	[String]	service provider
ts_transport_stream_id	[String]	Stream ID
ts_start_pid	[String]	STREAM START ID
ts_pmt_start_pid	[String]	PMT START PID
ts_tables_version	[String]	Tables version

hls

Field	Value	Instruction
session	[String]	session id
segmentTime	[Integer]	slice length

ndi_hx

Field	Value	Instruction
ttl	[Integer]	TTL
NDIname	[String]	channel name
group	[String]	group name
netmask	[String]	subnet mask
connection	[String]	connection: multicast : multicast tcp default
netprefix	[String]	multicast address
discovery_server	[String]	server address