

ANT-37500

API Version 3.5.x

Table of Contents

Introduction	6
Changes	7
V.1.1.1	7
V.2.0.x.....	7
V.3.1.x.....	7
V.3.3.x.....	7
PUT /api/camera	8
request (<i>json object</i>):.....	8
DELETE /api/camera/<idx:int>	9
response (<i>json object</i>):.....	9
GET /api/camera/<idx:int>	10
response (<i>json object</i>):.....	10
PUT /api/camera/<idx:int>	11
request (<i>json object</i>):.....	11
GET /api/camera/discover/<idx:int>	12
response (<i>json object</i>):.....	12
DELETE /api/cameras	13
response (<i>json object</i>):.....	13
GET /api/cameras	14
response (<i>json object</i>):.....	14
PUT /api/cameras	15
request (<i>json object</i>):.....	15
PUT /api/cameras/restart.....	16
response (<i>json object</i>):.....	16
GET /api/display.....	17
response (<i>json object</i>):.....	17
GET /api/display/modes.....	18
response (<i>json object</i>):.....	18
GET /api/network/<iface>	19
response (<i>json object</i>):.....	19

PUT /api/network/<iface>	20
request (<i>json object</i>):.....	20
PUT /api/network/restart/<iface>	21
response (<i>json object</i>):.....	21
GET /api/network/wifi/scan.....	22
response (<i>json object</i>):.....	22
GET /api/onvif/cameras.....	23
response (<i>json object</i>):.....	23
GET /api/onvif/profiles	24
response (<i>json object</i>):.....	24
POST /api/login.....	25
request (<i>json object</i>):.....	25
POST /api/logout.....	26
response (<i>json object</i>):.....	26
GET /api/system/config	27
response (<i>json object</i>):.....	27
POST /api/system/config.....	28
request (<i>Multipart-Encoded JSON File</i>):.....	28
PUT /api/system/config	29
request (<i>json object</i>):.....	29
DELETE /api/system/config/<v>	30
response (<i>json object</i>):.....	30
GET /api/system/config/<v>	31
response (<i>json object</i>):.....	31
PUT /api/system/factoryreset.....	32
response (<i>json object</i>):.....	32
GET /api/system/info.....	33
response (<i>json object</i>):.....	33
PUT /api/system/reboot.....	34
response (<i>json object</i>):.....	34
PUT /api/system/restart.....	35
response (<i>json object</i>):.....	35
DELETE /api/system/ssl.....	36
response (<i>json object</i>):.....	36

POST /api/system/ssl/cert	37
request (<i>Multipart-Encoded File</i>):.....	37
response (<i>json object</i>):.....	37
POST /api/system/ssl/key	38
request (<i>Multipart-Encoded File</i>):.....	38
response (<i>json object</i>):.....	38
PUT /api/system/user.....	39
request (<i>json object</i>):.....	39
PUT /api/system/config/views	40
GET /api/view.....	41
response (<i>json object</i>):.....	41
GET /api/view/<v:int>	42
response (<i>json object</i>):.....	42
PUT /api/view/<v:int>/play.....	43
request (<i>json object</i>):.....	43
GET /api/view/default.....	44
response (<i>json object</i>):.....	44
PUT /api/view/default.....	45
request (<i>json object</i>):.....	45
GET /api/view/mode.....	46
response (<i>json object</i>):.....	46
PUT /api/view/mode.....	47
request (<i>json object</i>):.....	47
GET /api/view/status.....	48
response (<i>json object</i>):.....	48
GET /api/views.....	49
response (<i>json object</i>):.....	49
Definitions.....	50
CameraPropsT.....	50
CameraT.....	50
CaptureMode.....	50
ConfigT	50
DefaultViewT	51
GridFontE.....	51

GridModes.....	51
MediaItemT	51
NetAgentParam.....	51
NetIpv4.....	51
NetSecurity	51
NetStateT.....	52
NetworkInterfaceT	52
ProfileT.....	52
REQ_camera_add	52
REQ_camera_put.....	53
REQ_cameras_put.....	53
REQ_login	53
REQ_net_config.....	53
REQ_system_config.....	53
REQ_update_password.....	53
REQ_vendor_storage	53
REQ_view_mode.....	53
REQ_view_play	53
RES_Network.....	54
RES_camera	54
RES_camera_discover	54
RES_cameras	54
RES_config.....	54
RES_config_v.....	54
RES_del_camera	54
RES_int.....	54
RES_media_list.....	54
RES_network_scan	55
RES_ok	55
RES_profiles.....	55
RES_strings	55
RES_system_info	55
RES_vendor_storage.....	55
RES_view.....	55

RES_view_default.....	56
RES_view_mode	56
RES_view_playing.....	56
RES_view_status.....	56
RES_views.....	56
RES_zero	56
ResCameraDiscoverT.....	56
ResError.....	57
ResScanT.....	57
ResSystemInfoT	57
ResViewPlayingT.....	57
ResViewStatusT	57
ResViewT.....	58
RtspEncProps.....	58
RtspStreamAddress.....	58
RtspStreamT.....	58
RtspSvrT	58
UserT	59
ViewCamT	59
ViewListT.....	59
ViewSequenceModeT	59
ViewSequenceT.....	59
ViewStatusT.....	60
ViewT	60
WebUIT	60
cameras	60
media_list.....	60
user_config.....	60

Introduction

The Spotbox provides an API for managing the device through HTTP requests. The API methods are implemented as a set of http routes.

Each method take parameters as part of the URL or as a JSON object in the body of the request.

All calls will return a JSON object with one of two properties:

```
{"result": <result object> } // Successful result
```

or

```
{"error": "ERROR_CODE", "msg": "message"} // Error result
```

The API is protected with a basic authorisation mechanism which requires the “/api/login” method to be called before any other methods are used.

On successful login a cookie is returned which must be provided for subsequent calls to be successful. The cookie will be removed by the “/api/logout” method.

Changes

V.1.1.1

The following routes have been added:

```
GET /api/display/modes
POST /api/system/config
POST /api/system/ssl/key
POST /api/system/ssl/cert
```

V.2.0.x

The following routes have been added:

```
DELETE /api/system/ssl
```

V.3.1.x

The following routes have been added:

```
PUT /api/network/<iface>
PUT /api/network/restart/<iface>
GET /api/network/wifi/scan
PUT /api/network/wifi/connect/<iface>
```

Also to support WIFI the format of the 'network' config value has been changed. The old network config will be accepted and be transformed to the new format.

V.3.3.x

The following routes have been added:

```
GET /api/camera/discover/<idx>
GET /api/system/config/<item>
PUT /api/system/config/<item>
DELETE /api/system/config/<item>
```

The following config entry has been added:

```
/api/system/config/views
```

The following config entry has been modified:

```
/api/system/config/cameras
```

PUT /api/camera

Append new camera to camera list, return camera number

request (json object):

response (json object): ## Example: ##### request:

```
PUT /api/camera
{"camera": {"label": "C3", "tag": "C3", "uri":
"rtsp://192.168.86.13:554/media/stream.sdp?profile=Profile100", "username":
"admin", "password": "**password**"}}
```

response:

```
200 OK
Content-Type: application/json
{"result":2}
```

DELETE /api/camera/<idx:int>

Delete camera *idx*

response (json object):

Example: #### request:

```
DELETE /api/camera/2
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result":2}
```

GET /api/camera/<idx:int>

Get camera definition for camera *idx*

response (json object):

Example: ##### request:

```
GET /api/camera/0
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result":{"tag":"C1","uri":"rtsp://192.168.86.10:554/Streaming/Channels/101",  
,"label":"C1","devinfo":"rtsp:","idx":0,"lab_position":0,"password":  
"***password***","restart":0,"username":"user","zoom":0,"props":{"audio":0,"lat  
ency":0,"protocols":0,"timeout":0},"profiles":[]}}
```

PUT /api/camera/<idx:int>

Update camera definition for camera *idx*

request (json object):

response (json object): ## Example: ##### request:

```
PUT /api/camera/2
{"camera": {"label": "NEW LABEL", "tag": "new"}}
```

response:

```
200 OK
Content-Type: application/json
{"result":2}
```

GET /api/camera/discover/<idx:int>

Probe camera stream returning its profiles if onvif or simplified profile for other sources

response (json object):

Example: ##### request:

```
GET /api/camera/discover/0
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result":{"devinfo":"rtsp:","profiles":[{"name":"video","codec":"H264","res_width":1920,"res_height":1080,"fps":"15","uri":"rtsp://192.168.86.10:554/Streaming/Channels/101"}]}}
```

DELETE /api/cameras

Delete existing list of cameras

response (json object):

Example: #### request:

```
DELETE /api/cameras
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result":0}
```

GET /api/cameras

Get list of camera settings

response (json object):

Example: #### request:

```
GET /api/cameras
```

response:

200 OK

Content-Type: application/json

```
{"result":[{"tag":"C1","uri":"rtsp://192.168.86.10:554/Streaming/Channels/101", "label":"C1", "devinfo":"rtsp:", "idx":0, "lab_position":0, "password":
***password***, "restart":0, "username":"user", "zoom":0, "props":{"audio":0, "lat
ency":0, "protocols":0, "timeout":0}, "profiles":[]}, {"tag":"C2", "uri":"rtsp://1
92.168.86.11:554/Streaming/Channels/101", "label":"C2", "devinfo":"rtsp:", "idx"
:1, "lab_position":0, "password":
***password***, "restart":0, "username":"user", "zoom":0, "props":{"audio":0, "lat
ency":0, "protocols":0, "timeout":0}, "profiles":[]}, {"tag":"C3", "uri":"rtsp://1
92.168.86.13:554/media/stream.sdp?profile=Profile100", "label":"C3", "devinfo":
"rtsp:", "idx":2, "lab_position":0, "password":
***password***, "restart":0, "username":"admin", "zoom":0, "props":{"audio":0, "la
tency":0, "protocols":0, "timeout":0}, "profiles":[]}]}
```

PUT /api/cameras

Set new list of cameras to be displayed

request (json object):

response (json object): ## Example: ##### request:

```
PUT /api/cameras
{"cameras": [{"tag": "C1", "label": "C1", "uri":
"rtsp://192.168.86.10:554/Streaming/Channels/101", "username": "user",
"password": "**password**", "props": {"audio": 0}}, {"label": "C2", "tag":
"C2", "uri": "rtsp://192.168.86.11:554/Streaming/Channels/101", "username":
"user", "password": "**password**"}]}
```

response:

```
200 OK
Content-Type: application/json
{"result": "OK"}
```

PUT /api/cameras/restart

restart playing cameras in a view

response (json object):

Example: #### request:

```
PUT /api/cameras/restart
```

response:

200 OK

Content-Type: application/json

```
{"result": "OK"}
```

GET /api/display

Get current display resolution

response (json object):

Example: ##### request:

GET /api/display

response:

200 OK

Content-Type: application/json

{"result": "3840x2160@60"}

GET /api/display/modes

Get current resolutions supported by display

response (json object):

Example: #### request:

GET /api/display/modes

response:

200 OK

Content-Type: application/json

```
{"result":["640x480@73","640x480@75","640x480@60","640x480@67","720x400@70","720x576@50","720x480@60","800x600@60","800x600@72","800x600@75","800x600@56","832x624@75","1024x768@60","1024x768@75","1024x768@70","1280x960@60","1280x720@60","1280x1024@75","1280x720@50","1280x1024@60","1440x900@60","1680x1050@60","1920x1080@50","1920x2160@60","1920x1080@60","2560x1440@60","3840x2160@50","3840x2160@30","3840x2160@60","3840x2160@24","3840x2160@25"]}
```

GET /api/network/<iface>

Get network configuration for *iface*

response (json object):

Example: #### request:

```
GET /api/network/eth0
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result":{"Type":"Wired","Device":"","Available":true,"Enabled":true,"AgentParams":[],"Connect":"","DNS":"192.168.86.1","IPv4":{"Address":"192.168.86.222","Gateway":"192.168.86.1","Method":"static","Netmask":"255.255.0.0"},"NTP":"192.168.86.1","State":"ready","Address":"CE:4F:33:B3:4B:02"}}
```

PUT /api/network/<iface>

Set network configuration for *iface*

request (json object):

response (json object): ## Example: ##### request:

```
PUT /api/network/wlan0
{"config": {"Enabled": true, "Connect": "", "IPv4": {"Method": "dhcp"},
"AgentParams": [{"ssid": "pumplane-g", "passphrase": "**password**"]}}
```

response:

```
200 OK
Content-Type: application/json
{"result": "OK"}
```

PUT /api/network/restart/<iface>

Restart network interface *iface*

response (json object):

Example: #### request:

```
PUT /api/network/restart/wlan0
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result": "OK"}
```

GET /api/network/wifi/scan

Return available wifi networks

response (json object):

Example: ##### request:

```
GET /api/network/wifi/scan
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result":{"wlan0":[{"Name":"pumplane-  
g","State":"idle","Strength":90,"Security":["psk"]},{"Name":"pumplane","State  
":"idle","Strength":57,"Security":["psk"]}]}
```

GET /api/onvif/cameras

Search network for available onvif cameras

response (json object):

Example: #### request:

```
GET /api/onvif/cameras
```

response:

200 OK

Content-Type: application/json

```
{"result":["192.168.86.10","192.168.86.13:80","192.168.86.222:8000"]}
```

GET /api/onvif/profiles

Query onvif cameras for available profiles

response (json object):

Example: ##### request:

```
GET /api/onvif/profiles?uri=192.168.86.10&username=user&password=a1b2c3d4
```

response:

200 OK

Content-Type: application/json

```
{"result":[{"name":"Multicast:mainStream","codec":"H264","res_width":"1920","res_height":"1080","fps":"15.0","uri":"rtsp://192.168.86.10:554/Streaming/Channels/101?transportmode=mcast&profile=Profile_1"}, {"name":"Multicast:subStream","codec":"H265","res_width":"640","res_height":"480","fps":"10.0","uri":"rtsp://192.168.86.10:554/Streaming/Channels/102?transportmode=mcast&profile=Profile_2"}, {"name":"Unicast:mainStream","codec":"H264","res_width":"1920","res_height":"1080","fps":"15.0","uri":"rtsp://192.168.86.10:554/Streaming/Channels/101?transportmode=unicast&profile=Profile_1"}, {"name":"Unicast:subStream","codec":"H265","res_width":"640","res_height":"480","fps":"10.0","uri":"rtsp://192.168.86.10:554/Streaming/Channels/102?transportmode=unicast&profile=Profile_2"}]}
```

POST /api/login

login to session

request (json object):

response (json object): ## Example: #### request:

```
POST /api/login  
{"user": "admin", "password": "**password**"}
```

response:

```
200 OK  
Content-Type: application/json  
{"result": "OK"}
```

POST /api/logout

Logout of a session

response (json object):

Example: #### request:

POST /api/logout

response:

200 OK

Content-Type: application/json

{"result": "OK"}

GET /api/system/config

Get current system configuration

response (json object):

Example: #### request:

```
GET /api/system/config
```

response:

200 OK

Content-Type: application/json

```
{
  "result": {
    "cameras": [
      {
        "tag": "C1",
        "uri": "rtsp://192.168.86.10:554/Streaming/Channels/101",
        "label": "C1",
        "devinfo": "rtsp",
        "idx": 0,
        "lab_position": 0,
        "password": "****password****",
        "restart": 0,
        "username": "user",
        "zoom": 0,
        "props": {
          "audio": 0,
          "latency": 0,
          "protocols": 0,
          "timeout": 0,
          "profiles": []
        },
        {
          "tag": "C2",
          "uri": "rtsp://192.168.86.11:554/Streaming/Channels/101",
          "label": "C2",
          "devinfo": "rtsp",
          "idx": 1,
          "lab_position": 0,
          "password": "****password****",
          "restart": 0,
          "username": "user",
          "zoom": 0,
          "props": {
            "audio": 0,
            "latency": 0,
            "protocols": 0,
            "timeout": 0,
            "profiles": []
          },
          "network": {
            "eth0": {
              "Type": "Wired",
              "Device": "",
              "Available": true,
              "Enabled": true,
              "AgentParams": [],
              "Connect": "",
              "DNS": "192.168.86.1",
              "IPv4": {
                "Address": "192.168.86.222",
                "Gateway": "192.168.86.1",
                "Method": "static",
                "Netmask": "255.255.0.0",
                "NTP": "192.168.86.1",
                "State": "ready",
                "Address": "CE:4F:33:B3:4B:02"
              },
              "wlan0": {
                "Type": "Wired",
                "Device": "",
                "Available": true,
                "Enabled": false,
                "AgentParams": [],
                "Connect": "",
                "DNS": "",
                "IPv4": {
                  "Address": "192.168.0.222",
                  "Gateway": "192.168.0.1",
                  "Method": "dhcp",
                  "Netmask": "255.255.255.0",
                  "NTP": "",
                  "State": "offline",
                  "Address": ""
                },
                "display": "3840x2160@60",
                "default_view": {
                  "camera": 0,
                  "name": "",
                  "mode": "grid_2x2"
                },
                "grid_font": {
                  "fullscreen": "s",
                  "grid_1x1": "s",
                  "grid_2x2": "m",
                  "grid_3x3": "s",
                  "grid_4x4": "s"
                },
                "user_config": {
                  "location": "Bristol",
                  "web_ui": {
                    "host": "0.0.0.0",
                    "http_port": 80,
                    "https_enable": false,
                    "https_port": 443,
                    "iface": "eth0"
                  },
                  "rtsp_svr": {
                    "src": "kms",
                    "available": false,
                    "enabled": true,
                    "onvif": true,
                    "password": "****password****",
                    "port": 554,
                    "streams": [
                      {
                        "enc_props": {
                          "bps": 6000000,
                          "gop": 40,
                          "height": 1080,
                          "width": 1920,
                          "profile": "baseline",
                          "rc_mode": "vbr"
                        },
                        "codec": "h264",
                        "framerate": 20,
                        "name": "video",
                        "src": "",
                        "stream_address": {
                          "address": "224.3.0.0",
                          "port": 5000,
                          "ttl": 16
                        },
                        "transport": "ANY"
                      }
                    ],
                    "username": "user",
                    "views": {
                      "grid_1x1": {
                        "grid_layout": "grid_1x1",
                        "views": [
                          {
                            "cameras": [
                              {
                                "profile": 0,
                                "tag": "C*"
                              }
                            ],
                            "period": 0,
                            "period": 60
                          }
                        ],
                        "view_sequence": [
                          {
                            "name": "test",
                            "sequence": [
                              {
                                "mode": "grid_1x1",
                                "period": 20
                              },
                              {
                                "mode": "grid_2x2",
                                "period": 60
                              }
                            ],
                            "enabled": true,
                            "at": ""
                          }
                        ],
                        "display_capture": 480,
                        "dante": {
                          "available": true,
                          "rx": {
                            "enabled": true,
                            "parameters": {
                              "rxVideoChannels": [
                                {
                                  "subtypes": ["H264"]
                                }
                              ]
                            },
                          },
                          "tx": {
                            "enabled": false,
                            "parameters": {
                              "txVideoChannels": [
                                {
                                  "subtypes": ["H264"]
                                }
                              ]
                            },
                          },
                          "audio": {
                            "enabled": true,
                            "parameters": {}
                          }
                        }
                      }
                    ]
                  }
                }
              }
            }
          }
        }
      }
    ]
  }
}
```

POST /api/system/config

Update system configuration from a file (requires restart to apply)

request (*Multipart-Encoded JSON File*):

response (*json object*): ## Example: ##### request:

POST /api/system/config

PUT /api/system/config

Update current system configuration (requires restart to apply)

request (json object):

response (json object): ## Example: ##### request:

```
PUT /api/system/config
{"config": {"user_config": {"location": "Bristol"}}
```

response:

```
200 OK
Content-Type: application/json
{"result": "OK"}
```

DELETE /api/system/config/<v>

Reset config value to factory default value (requires restart to apply)

response (json object):

Example: #### request:

```
DELETE /api/system/config/views
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result": "OK"}
```

GET /api/system/config/<v>

Get specific config entry

response (json object):

Example: #### request:

GET /api/system/config/views

response:

200 OK

Content-Type: application/json

```
{"result":{"grid_1x1":{"grid_layout":"grid_1x1","views":[{"cameras":[{"profile":0,"tag":"C*","period":0}], "period":60}]}}
```

PUT /api/system/factoryreset

Reset device to factory defaults

response (json object):

Example: #### request:

```
PUT /api/system/factoryreset
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result": "OK"}
```

GET /api/system/info

Get sytem info

response (json object):

Example: #### request:

GET /api/system/info

response:

200 OK

Content-Type: application/json

```
{"result":{"version":{"SW_VERSION":"spotbox
3.5.6","IMAGE_BUILD_DATE":"Mon 23 Oct 12:23:57 BST
2023","HW_REVISION":"spotbox
ANT37500"},"sn":"","sys":{"cpu_temp":47.777,"gpu_temp":48.888,"boot_time":"23
-10-2023 11:58:19","cpu_freq":1416.0,"mem":51.0,"cpu_times":20.1},"date":"Tue
Oct 24 13:50:56 2023"}}
```

PUT /api/system/reboot

reboot device

response (json object):

Example: ##### request:

```
PUT /api/system/reboot
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result": "OK"}
```

PUT /api/system/restart

restart spotbox application

response (json object):

Example: ##### request:

```
PUT /api/system/restart
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result": "OK"}
```

DELETE /api/system/ssl

Delete current ssl certificate and key

response (json object):

Example: #### request:

```
DELETE /api/system/ssl
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result": "OK"}
```

POST /api/system/ssl/cert

Load new ssl certificate from file

request (*Multipart-Encoded File*):

response (*json object*):

Example: ##### request:

```
POST /api/system/ssl/cert
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result": "OK"}
```

POST /api/system/ssl/key

Load new ssl key from file

request (Multipart-Encoded File):

response (json object):

Example: #### request:

POST /api/system/ssl/key

response:

200 OK

Content-Type: application/json

{"result": "OK"}

PUT /api/system/user

Update login password for admin

request (json object):

response (json object): ## Example: ##### request:

```
PUT /api/system/user
{"user": {"username": "admin", "password": "**password**"}}
```

response:

```
200 OK
Content-Type: application/json
{"result": "OK"}
```

PUT /api/system/config/views

Configure cameras to display in grid

request:

```
PUT /api/system/config/views
{"config": {"grid_1x1": {"grid_layout": "grid_1x1", "views": [{"period":
60, "cameras": [{"tag": "C*"}]}]}}
```

response:

```
200 OK
Content-Type: application/json
{"result": "OK"}
```

GET /api/view

List of possible view modes

response (json object):

Example: #### request:

```
GET /api/view
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result":["grid_1x1","grid_2x2","grid_3x3","grid_4x4"]}
```

GET /api/view/<v:int>

Get list of cameras playing on view v

response (json object):

Example: ##### request:

```
GET /api/view/0
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result":[{"idx":0,"tag":"C1","url":"rtsp://user:a1b2c3d4@192.168.86.10:554/Streaming/Channels/101","stream_profile":{"name":"","codec":"","res_width":0,"res_height":0,"fps":0},"profile":0,"period":60},{"idx":1,"tag":"C2","url":"rtsp://192.168.86.11:554/Streaming/Channels/101"}, {"idx":2,"tag":"C3","url":"rtsp://192.168.86.13:554/media/stream.sdp?profile=Profile100"}]}
```

PUT /api/view/<v:int>/play

Play list of cameras on specific view v .

request (*json object*):

response (*json object*): ## Example: ##### request:

```
PUT /api/view/0/play
{"cameras": [0, 1], "period": 10}
```

response:

```
200 OK
Content-Type: application/json
{"result":"OK"}
```

GET /api/view/default

Get default view mode

response (json object):

Example: #### request:

```
GET /api/view/default
```

response:

200 OK

Content-Type: application/json

```
{"result":{"camera":0,"name":"","mode":"grid_2x2"}}
```

PUT /api/view/default

Set default view mode

request (json object):

response (json object): ## Example: ##### request:

```
PUT /api/view/default  
{"mode": "grid_1x1"}
```

response:

```
200 OK  
Content-Type: application/json  
{"result":"OK"}
```

GET /api/view/mode

Get current view mode

response (json object):

Example: #### request:

```
GET /api/view/mode
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result": "grid_1x1"}
```

PUT /api/view/mode

Change current view mode

request (json object):

response (json object): ## Example: ##### request:

```
PUT /api/view/mode  
{"mode": "grid_1x1"}
```

response:

```
200 OK  
Content-Type: application/json  
{"result":"OK"}
```

GET /api/view/status

Return current status of cameras displayed

response (json object):

Example: ##### request:

```
GET /api/view/status
```

response:

```
200 OK
```

```
Content-Type: application/json
```

```
{"result":[{"status":4,"status_exit":0,"time_start":1698155477,"time_duration":4,"time_update":1698155481,"uri":"rtsp://192.168.86.10:554/Streaming/Channels/101"}]}
```

GET /api/views

Get list of cameras in each view

response (json object):

Example: ##### request:

GET /api/views

response:

200 OK

Content-Type: application/json

```
{"result":[[{"idx":0,"tag":"C1","url":"rtsp://user:a1b2c3d4@192.168.86.10:554/Streaming/Channels/101","stream_profile":{"name":"","codec":"","res_width":0,"res_height":0,"fps":0},"profile":0,"period":60},{"idx":1,"tag":"C2","url":"rtsp://192.168.86.11:554/Streaming/Channels/101"}, {"idx":2,"tag":"C3","url":"rtsp://192.168.86.13:554/media/stream.sdp?profile=Profile100"}]]}
```

Definitions

CameraPropsT

- *object*
 - **audio** (*integer*)
 - **latency** (*integer*)
 - **protocols** (*integer*)
 - **timeout** (*integer*)

CameraT

- *object*: Properties for Camera.
 - **tag** (*string*)
 - **uri** (*string*)
 - **label** (*string*)
 - **devinfo** (*string*)
 - **idx** (*integer*)
 - **lab_position** (*integer*): One of: [0, 1, 2, 3, 99].
 - **password** (*string*)
 - **restart** (*integer*)
 - **username** (*string*)
 - **zoom** (*integer*)
 - **props**:
 - **profiles** (*array*)
 - **Items**:

CaptureMode

- : One of: [480, 640, 1280].

ConfigT

- *object*
 - **cameras** (*array*)
 - **Items**:
 - **views** (*object*): Can contain additional properties.
 - **Additional Properties**:
 - **view_sequence** (*array*)
 - **Items**:
 - **network** (*object*): Can contain additional properties.
 - **Additional Properties**:
 - **display** (*string*)
 - **default_view**:
 - **grid_font** (*object*): Can contain additional properties.

- **Additional Properties** (*string*): One of: ['l', 'm', 's'].
- **https_enable** (*boolean*)
- **user_config** (*object*)
- **rtsp_svr**:
- **display_capture** (*integer*): One of: [480, 640, 1280].

DefaultViewT

- *object*
 - **camera** (*integer*)
 - **name** (*string*)
 - **mode** (*string*)

GridFontE

- : One of: ['l', 'm', 's'].

GridModes

- : One of: ['grid_1x1', 'grid_2x2', 'grid_3x3', 'grid_4x4'].

MediaItemT

- *object*
 - **name** (*string*)
 - **path** (*string*)
 - **uri** (*string*)

NetAgentParam

- *object*
 - **identity** (*string*)
 - **name** (*string*)
 - **passphrase** (*string*)
 - **ssid** (*string*)
 - **wpspin** (*string*)

NetIpv4

- *object*
 - **Address** (*string*)
 - **Gateway** (*string*)
 - **Method** (*string*): One of: ['dhcp', 'manual', 'static'].
 - **Netmask** (*string*)

NetSecurity

- : One of: ['none', 'wep', 'psk', 'ieee8021x', 'wps'].

NetStateT

- : One of: ['association', 'configuration', 'disconnect', 'failure', 'idle', 'offline', 'online', 'ready'].

NetworkInterfaceT

- *object*
 - **Type** (*string*): One of: ['Loopback', 'WiFi', 'Wired'].
 - **Device** (*string*)
 - **Available** (*boolean*)
 - **Enabled** (*boolean*)
 - **AgentParams** (*array*)
 - **Items:**
 - **Connect** (*string*)
 - **DNS** (*string*)
 - **IPv4:**
 - **NTP** (*string*)
 - **State** (*string*): One of: ['association', 'configuration', 'disconnect', 'failure', 'idle', 'offline', 'online', 'ready'].
 - **Address** (*string*)

ProfileT

- *object*
 - **enabled** (*boolean*)
 - **name** (*string*)
 - **codec** (*string*)
 - **uri** (*string*)
 - **fps**
 - **Any of**
 - *integer*
 - *string*
 - **res_height**
 - **Any of**
 - *integer*
 - *string*
 - **res_width**
 - **Any of**
 - *integer*
 - *string*

REQ_camera_add

- *object*

- camera:

REQ_camera_put

- *object*
 - camera:

REQ_cameras_put

- *object*
 - cameras (*array*)
 - **Items:**

REQ_login

- *object*
 - user (*string*)
 - password (*string*)

REQ_net_config

- *object*
 - config:

REQ_system_config

- *object*
 - config:

REQ_update_password

- *object*
 - user:

REQ_vendor_storage

- *object*
 - value (*string*)

REQ_view_mode

- *object*
 - mode (*string*)
 - camera (*integer*)

REQ_view_play

- *object*
 - cameras (*array*)
 - **Items** (*integer*)
 - period (*integer*)

RES_Network

- *object*
 - **error:**
 - **result:**

RES_camera

- *object*
 - **error:**
 - **result:**

RES_camera_discover

- *object*
 - **error:**
 - **result:**

RES_cameras

- *object*
 - **error:**
 - **result (array)**
 - **Items:**

RES_config

- *object*
 - **error:**
 - **result:**

RES_config_v

- *object*
 - **error:**
 - **result (object)**

RES_del_camera

- *object*
 - **error:**
 - **result (integer)**

RES_int

- *object*
 - **result (integer)**

RES_media_list

- *object*
 - **error:**

- **result** (*array*)
 - **Items:**

RES_network_scan

- *object*
 - **error:**
 - **result** (*object*): Can contain additional properties.
 - **Additional Properties** (*array*)
 - **Items:**

RES_ok

- *object*
 - **result** (*string*)

RES_profiles

- *object*
 - **error:**
 - **result** (*array*)
 - **Items:**

RES_strings

- *object*
 - **error:**
 - **result** (*array*)
 - **Items** (*string*)

RES_system_info

- *object*
 - **error:**
 - **result:**

RES_vendor_storage

- *object*
 - **error:**
 - **result** (*string*)

RES_view

- *object*
 - **error:**
 - **result** (*array*)
 - **Items** (*string*)

RES_view_default

- *object*
 - **error:**
 - **result:**

RES_view_mode

- *object*
 - **error:**
 - **result** (*string*)

RES_view_playing

- *object*
 - **error:**
 - **result** (*array*)
 - **Items** (*array*)
 - **Items:**

RES_view_status

- *object*
 - **error:**
 - **result** (*array*)
 - **Items** (*array*)
 - **Items:**

RES_views

- *object*
 - **error:**
 - **result** (*array*)
 - **Items** (*array*)
 - **Items:**

RES_zero

- *object*
 - **error:**
 - **result** (*integer*)

ResCameraDiscoverT

- *object*
 - **devinfo** (*string*)
 - **profiles** (*array*)
 - **Items:**

ResError

- *object*
 - **error** (*string*)
 - **msg** (*string*)

ResScanT

- *object*
 - **Name** (*string*)
 - **State** (*string*): One of: ['association', 'configuration', 'disconnect', 'failure', 'idle', 'offline', 'online', 'ready'].
 - **Strength** (*integer*)
 - **Security** (*array*)
 - **Items** (*string*): One of: ['none', 'wep', 'psk', 'ieee8021x', 'wps'].

ResSystemInfoT

- *object*
 - **cpu** (*object*): Can contain additional properties.
 - **Additional Properties**
 - **Any of**
 - *number*
 - *string*
 - **date** (*string*)
 - **version** (*object*): Can contain additional properties.
 - **Additional Properties** (*string*)

ResViewPlayingT

- *object*
 - **idx** (*integer*)
 - **tag** (*string*)
 - **url** (*string*)
 - **profile** (*integer*)
 - **period** (*integer*)

ResViewStatusT

- *object*
 - **status** (*integer*)
 - **status_exit** (*integer*)
 - **time_start** (*number*)
 - **time_duration** (*number*)
 - **time_update** (*number*)
 - **uri** (*string*)

ResViewT

- *object*
 - **idx** (*integer*)
 - **tag** (*string*)
 - **url** (*string*)
 - **profile** (*integer*)
 - **stream_profile:**
 - **period** (*integer*)
 - **time_remaining** (*integer*)
 - **status** (*integer*)

RtspEncProps

- *object*
 - **bps** (*integer*)
 - **gop** (*integer*)
 - **height** (*integer*)
 - **width** (*integer*)
 - **profile** (*string*): One of: ['baseline', 'high', 'main'].
 - **rc_mode** (*string*): One of: ['cbr', 'vbr'].

RtspStreamAddress

- *object*
 - **address** (*string*)
 - **port** (*integer*)
 - **ttl** (*integer*)

RtspStreamT

- *object*
 - **enc_props:**
 - **codec** (*string*): One of: ['h264', 'h265'].
 - **framerate** (*integer*)
 - **name** (*string*)
 - **src** (*string*)
 - **stream_address:**
 - **transport** (*string*): One of: ['ANY', 'MCAST', 'TCP', 'UDP', 'UNICAST'].

RtspSvrT

- *object*
 - **src** (*string*)
 - **available** (*boolean*)
 - **enabled** (*boolean*)

- **onvif** (*boolean*)
- **password** (*string*)
- **port** (*integer*)
- **streams** (*array*)
 - **Items:**
- **username** (*string*)

UserT

- *object*
 - **username** (*string*)
 - **password** (*string*)

ViewCamT

- *object*
 - **profile** (*integer*)
 - **tag** (*string*)
 - **period**
 - **Any of**
 - *integer*
 - *string*

ViewListT

- *object*
 - **cameras** (*array*)
 - **Items:**
 - **period**
 - **Any of**
 - *integer*
 - *string*

ViewSequenceModeT

- *object*
 - **mode** (*string*)
 - **period**
 - **Any of**
 - *integer*
 - *string*

ViewSequenceT

- *object*
 - **name** (*string*)
 - **sequence** (*array*)

- **Items:**
 - **enabled** (*boolean*)
 - **at** (*string*)

ViewStatusT

- *object*
 - **status** (*string*)
 - **time_start** (*integer*)
 - **status_exit** (*string*)
 - **time_duration** (*integer*)
 - **uri** (*string*)
 - **time_update** (*integer*)

ViewT

- *object*
 - **grid_layout** (*string*)
 - **views** (*array*)
 - **Items:**

WebUIT

- *object*
 - **host** (*string*)
 - **http_port** (*integer*)
 - **https_enable** (*boolean*)
 - **https_port** (*integer*)
 - **iface** (*string*): One of: ['eth0', 'wlan0'].

cameras

- *array*
 - **Items:**

media_list

- *array*
 - **Items:**

user_config

- *object*