# ANT-37000A / Spotbox1 API guide for firmware 1.1.1

## Contents



For further help and advice please contact Antrica on:
Email: support@antrica.com
Phone: +44 1628 626098 ext 3

# Document info

| Version | date | author | Comments |
|---|---|---|---|
| 1.0 | 30-Aug-22 | David M | Updated for firmware 1.1.1 |
| 1.1 | 31-Aug-22 | David M | Minor text changes |
| | | | |

# Extended Table of Contents

# 1. 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.

# 2. Changes

## 2.1. 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
```

# 3. POST /api/login

login to session

## 3.1. request (*json object*):

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

## 3.2. response (*json object*):

- ***result***: *OK*.

## 3.3. Example:

### 3.3.1. request:

```
POST /api/login
{"user": "admin", "password": "1234"}
```

### 3.3.2. response:

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

# 4. POST /api/logout

Logout of a session

## 4.1. response (*json object*):

- ***result***: *OK*.

## 4.2. Example:

### 4.2.1. request:

```
POST /api/logout
```

### 4.2.2. response:

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

# 5. GET /api/view

List of possible view modes

## 5.1. response (*json object*):

- ***result*** *(array)*
  - **Items**: One of: *['fullscreen', 'grid_1x1', 'grid_2x2', 'grid_3x3', 'grid_4x4']*.

## 5.2. Example:

### 5.2.1. request:

```
GET /api/view
```

### 5.2.2. response:

```
200 OK
Content-Type: application/json
{"result": ["grid_4x4", "fullscreen", "grid_3x3", "grid_1x1", "grid_2x2"]}
```

# 6. GET /api/views

Get list of cameras in each view

## 6.1. response (*json object*):

- ***result*** *(array)*
  - **Items** *(array)*
    - **Items** *(object)*
      - ***cnum*** *(integer)*
      - ***url*** *(string)*
      - ***profile*** *(integer)*

## 6.2. Example:

### 6.2.1. request:

```
GET /api/views
```

### 6.2.2. response:

```
200 OK
Content-Type: application/json
{"result": [[{"profile": null, "url": "rtsp://192.168.1.101", "cnum": 0}], [{"
profile": null, "url": "rtsp://192.168.1.102", "cnum": 1}], [], []]}
```

# 7. GET /api/view/mode

Get current view mode

## 7.1. response (*json object*):

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

## 7.2. Example:

### 7.2.1. request:

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

### 7.2.2. response:

```
200 OK
Content-Type: application/json
{"result": "grid_2x2"}
```

# 8. PUT /api/view/mode

Change current view mode

## 8.1. request (*json object*):

- ***mode***: One of: *['fullscreen', 'grid_1x1', 'grid_2x2', 'grid_3x3', 'grid_4x4']*.
- ***camera*** *(integer)*

## 8.2. response (*json object*):

- ***result***: *OK*.

## 8.3. Example:

### 8.3.1. request:

```
PUT /api/view/mode
{"mode": "grid_2x2"}
```

### 8.3.2. response:

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

# 9. GET /api/view/default

Get default view mode

## 9.1. response (*json object*):

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

## 9.2. Example:

### 9.2.1. request:

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

### 9.2.2. response:

```
200 OK
Content-Type: application/json
{"result": {"mode": "grid_1x1"}}
```

# 10.PUT /api/view/default

Set default view mode

## 10.1. request (*json object*):

- ***mode***: One of: *['fullscreen', 'grid_1x1', 'grid_2x2', 'grid_3x3', 'grid_4x4']*.
- ***camera*** *(integer)*

## 10.2. response (*json object*):

- ***result***: *OK*.

## 10.3. Example:

### 10.3.1. request:

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

### 10.3.2. response:

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

# 11.GET /api/view/status

Return current status of cameras displayed

## 11.1. response (*json object*):

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

## 11.2. Example:

### 11.2.1. request:

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

### 11.2.2. response:

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


```
{"result": [{"status": "PLAYING", "time_start": 1661797329, "status_exit": "NO
NE", "time_duration": 0, "uri": "rtsp://192.168.1.101", "time_update": 1661797329}
, {"status": "PLAYING", "time_start": 1661797329, "status_exit": "NONE", "time_dur
ation": 0, "uri": "rtsp://192.168.1.102", "time_update": 1661797329}, {"status": "
STOPPED", "time_start": 1661786211, "status_exit": "OK", "time_duration": 3631, "u
ri": "rtsp://192.168.1.103", "time_update": 1661797329}, null]}
```

# 12.GET /api/view/<v:int>

Get list of cameras playing on view *v*

## 12.1. response (*json object*):

- ***result*** *(array)*
  - **Items** *(object)*
    - ***cnum*** *(integer)*
    - ***url*** *(string)*
    - ***profile***
      - **One of**
        - *object*
          - ***uri*** *(string)*
          - ***name*** *(string)*
          - ***codec*** *(string)*
          - ***res_width*** *(string)*
          - ***res_height*** *(string)*
          - ***fps*** *(string)*
        - *null*

## 12.2. Example:

### 12.2.1. request:

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

### 12.2.2. response:

```
200 OK
Content-Type: application/json
{"result": [{"profile": null, "url": "rtsp://192.168.1.101", "cnum": 0}]}
```

# 13.PUT /api/view/<v:int>/play

Play list of cameras on specific view *v* .

## 13.1. request (*json object*):

- ***cameras*** *(array)*
  - **Items** *(integer)*: Minimum: *0*.
- ***period*** *(integer)*: Minimum: *0*.

## 13.2. response (*json object*):

- ***result***: *OK*.

## 13.3. Example:

### 13.3.1. request:

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

### 13.3.2. response:

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

# 14.GET /api/cameras

Get list of camera settings

## 14.1. response (*json object*):

- ***result*** *(array)*
  - **Items** *(object)*
    - ***grids*** *(object)*
      - ***fullscreen*** *(['integer', 'null'])*
      - ***grid_1x1*** *(['integer', 'null'])*
      - ***grid_2x2*** *(['integer', 'null'])*
      - ***grid_3x3*** *(['integer', 'null'])*
      - ***grid_4x4*** *(['integer', 'null'])*
    - ***label*** *(string)*
    - ***username*** *(string)*
    - ***password*** *(string)*
    - ***uri*** *(string)*
    - ***period*** *(integer)*
    - ***restart*** *(integer)*
    - ***lab_position*** *(integer)*
    - ***view*** *(['integer', 'null'])*: Minimum: *0*. Maximum: *15*.
    - ***props*** *(object)*
      - ***latency*** *(integer)*
      - ***timeout*** *(integer)*
      - ***protocols*** *(integer)*
      - ***drop_on_latency*** *(integer)*
      - ***debug*** *(integer)*
    - ***zoom*** *(integer)*
    - ***onvif*** *(boolean)*
    - ***profiles*** *(array)*
      - **Items** *(object)*
        - ***uri*** *(string)*
        - ***name*** *(string)*
        - ***codec*** *(string)*
        - ***res_width*** *(string)*
        - ***res_height*** *(string)*
        - ***fps*** *(string)*
    - ***idx*** *(integer)*: Minimum: *0*.

## 14.2. Example:

### 14.2.1. request:

```
GET /api/cameras
```

### 14.2.2. response:

**200 OK**
**Content-Type: application/json**

{"result": [{"username": "", "idx": 0, "grids": {"grid_1x1": 0, "grid_2x2": 0, "grid_4x4": null, "grid_3x3": null}, "uri": "rtsp://192.168.1.101", "profiles": [], "label": "C1", "period": 0, "lab_position": 0, "props": {"latency": 0, "audio": 0, "timeout": 0, "protocols": 0}, "password": "", "restart": 0, "zoom": 0}, {"username": "", "idx": 1, "grids": {"grid_1x1": null, "grid_2x2": 1, "grid_4x4": null, "grid_3x3": null}, "uri": "rtsp://192.168.1.102", "profiles": [], "label": "C2", "period": 0, "lab_position": 0, "props": {"latency": 0, "audio": 1, "timeout": 0, "protocols": 0}, "password": "", "restart": 0, "zoom": 0}, {"username": "", "idx": 2, "grids": {"grid_1x1": null, "grid_2x2": 2, "grid_4x4": null, "grid_3x3": null}, "uri": "rtsp://192.168.1.103", "profiles": [], "label": "C3", "period": 0, "lab_position": 0, "props": {"latency": 0, "audio": 1, "timeout": 0, "protocols": 0}, "password": "", "restart": 0, "zoom": 0}]}

# 15.PUT /api/cameras

Set new list of cameras to be displayed

## 15.1. request (*json object*):

- ***cameras*** *(array)*
  - **Items** *(object)*
    - ***grids*** *(object)*
      - ***fullscreen*** *(['integer', 'null'])*
      - ***grid_1x1*** *(['integer', 'null'])*
      - ***grid_2x2*** *(['integer', 'null'])*
      - ***grid_3x3*** *(['integer', 'null'])*
      - ***grid_4x4*** *(['integer', 'null'])*
    - ***label*** *(string)*
    - ***username*** *(string)*
    - ***password*** *(string)*
    - ***uri*** *(string)*
    - ***period*** *(integer)*
    - ***restart*** *(integer)*
    - ***lab_position*** *(integer)*
    - ***view*** *(['integer', 'null'])*: Minimum: *0*. Maximum: *15*.
    - ***props*** *(object)*
      - ***latency*** *(integer)*
      - ***timeout*** *(integer)*
      - ***protocols*** *(integer)*
      - ***drop_on_latency*** *(integer)*
      - ***debug*** *(integer)*
    - ***zoom*** *(integer)*
    - ***onvif*** *(boolean)*
    - ***profiles*** *(array)*
      - **Items** *(object)*
        - ***uri*** *(string)*
        - ***name*** *(string)*
        - ***codec*** *(string)*
        - ***res_width*** *(string)*
        - ***res_height*** *(string)*
        - ***fps*** *(string)*
    - ***idx*** *(integer)*: Minimum: *0*.

## 15.2. response (*json object*):

- ***result***: *OK*.

## 15.3. Example:

### 15.3.1. request:

```
PUT /api/cameras
{"cameras": [{"label": "C1", "grids": {"grid_1x1": 0, "grid_2x2": 0}, "props":
```

{"audio": 0}, "uri": "rtsp://192.168.1.101"}, {"label": "C2", "grids": {"grid_2x2"
: 1}, "props": {"audio": 1}, "uri": "rtsp://192.168.1.102"}]}

### 15.3.2. response:

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

# 16.DELETE /api/cameras

Delete existing list of cameras

## 16.1. response (*json object*):

- *result*: *0*.

## 16.2. Example:

### 16.2.1. request:

```
DELETE /api/cameras
```

### 16.2.2. response:

```
200 OK
Content-Type: application/json
{"result": 0}
```

# 17.PUT /api/cameras/restart

restart playing cameras in a view

## 17.1. response (*json object*):

- ***result***: *OK*.

## 17.2. Example:

### 17.2.1. request:

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

### 17.2.2. response:

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

# 18.PUT /api/camera

Append new camera to camera list, return camera number

## 18.1. request (*json object*):

- ***camera*** *(object)*
  - ***grids*** *(object)*
    - ***fullscreen*** *(['integer', 'null'])*
    - ***grid_1x1*** *(['integer', 'null'])*
    - ***grid_2x2*** *(['integer', 'null'])*
    - ***grid_3x3*** *(['integer', 'null'])*
    - ***grid_4x4*** *(['integer', 'null'])*
  - ***label*** *(string)*
  - ***username*** *(string)*
  - ***password*** *(string)*
  - ***uri*** *(string)*
  - ***period*** *(integer)*
  - ***restart*** *(integer)*
  - ***lab_position*** *(integer)*
  - ***view*** *(['integer', 'null'])*: Minimum: *0*. Maximum: *15*.
  - ***props*** *(object)*
    - ***latency*** *(integer)*
    - ***timeout*** *(integer)*
    - ***protocols*** *(integer)*
    - ***drop_on_latency*** *(integer)*
    - ***debug*** *(integer)*
  - ***zoom*** *(integer)*
  - ***onvif*** *(boolean)*
  - ***profiles*** *(array)*
    - **Items** *(object)*
      - ***uri*** *(string)*
      - ***name*** *(string)*
      - ***codec*** *(string)*
      - ***res_width*** *(string)*
      - ***res_height*** *(string)*
      - ***fps*** *(string)*
  - ***idx*** *(integer)*: Minimum: *0*.

## 18.2. response (*json object*):

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

## 18.3. Example:

### 18.3.1. request:

```
PUT /api/camera
{"camera": {"label": "C3", "grids": {"grid_2x2": 2}, "props": {"audio": 1}, "u
```

ri": "rtsp://192.168.1.103"}}

### 18.3.2. response:

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

# 19.GET /api/camera/<n:int>

Get camera definition for camera *n*

## 19.1. response (*json object*):

- ***result*** *(object)*
  - ***grids*** *(object)*
    - ***fullscreen*** *(['integer', 'null'])*
    - ***grid_1x1*** *(['integer', 'null'])*
    - ***grid_2x2*** *(['integer', 'null'])*
    - ***grid_3x3*** *(['integer', 'null'])*
    - ***grid_4x4*** *(['integer', 'null'])*
  - ***label*** *(string)*
  - ***username*** *(string)*
  - ***password*** *(string)*
  - ***uri*** *(string)*
  - ***period*** *(integer)*
  - ***restart*** *(integer)*
  - ***lab_position*** *(integer)*
  - ***view*** *(['integer', 'null'])*: Minimum: *0*. Maximum: *15*.
  - ***props*** *(object)*
    - ***latency*** *(integer)*
    - ***timeout*** *(integer)*
    - ***protocols*** *(integer)*
    - ***drop_on_latency*** *(integer)*
    - ***debug*** *(integer)*
  - ***zoom*** *(integer)*
  - ***onvif*** *(boolean)*
  - ***profiles*** *(array)*
    - **Items** *(object)*
      - ***uri*** *(string)*
      - ***name*** *(string)*
      - ***codec*** *(string)*
      - ***res_width*** *(string)*
      - ***res_height*** *(string)*
      - ***fps*** *(string)*
  - ***idx*** *(integer)*: Minimum: *0*.

## 19.2. Example:

### 19.2.1. request:

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

### 19.2.2. response:

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

{"result": {"username": "", "idx": 0, "grids": {"grid_1x1": 0, "grid_2x2": 0, "grid_4x4": null, "grid_3x3": null}, "uri": "rtsp://192.168.1.101", "profiles": [], "label": "C1", "period": 0, "lab_position": 0, "props": {"latency": 0, "audio": 0, "timeout": 0, "protocols": 0}, "password": "", "restart": 0, "zoom": 0}}

# 20.PUT /api/camera/<n:int>

Update camera definition for camera *n*

## 20.1. request (*json object*):

- ***camera*** *(object)*
  - ***grids*** *(object)*
    - ***fullscreen*** *(['integer', 'null'])*
    - ***grid_1x1*** *(['integer', 'null'])*
    - ***grid_2x2*** *(['integer', 'null'])*
    - ***grid_3x3*** *(['integer', 'null'])*
    - ***grid_4x4*** *(['integer', 'null'])*
  - ***label*** *(string)*
  - ***username*** *(string)*
  - ***password*** *(string)*
  - ***uri*** *(string)*
  - ***period*** *(integer)*
  - ***restart*** *(integer)*
  - ***lab_position*** *(integer)*
  - ***view*** *(['integer', 'null'])*: Minimum: *0*. Maximum: *15*.
  - ***props*** *(object)*
    - ***latency*** *(integer)*
    - ***timeout*** *(integer)*
    - ***protocols*** *(integer)*
    - ***drop_on_latency*** *(integer)*
    - ***debug*** *(integer)*
  - ***zoom*** *(integer)*
  - ***onvif*** *(boolean)*
  - ***profiles*** *(array)*
    - **Items** *(object)*
      - ***uri*** *(string)*
      - ***name*** *(string)*
      - ***codec*** *(string)*
      - ***res_width*** *(string)*
      - ***res_height*** *(string)*
      - ***fps*** *(string)*
  - ***idx*** *(integer)*: Minimum: *0*.

## 20.2. response (*json object*):

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

## 20.3. Example:

### 20.3.1. request:

```
PUT /api/camera/2
{"camera": {"label": "NEW LABEL", "grids": {"grid_2x2": 2}, "props": {"audio":
```

1}, "uri": "rtsp://192.168.1.103"}}

### 20.3.2. response:

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

# 21.DELETE /api/camera/<n:int>

Delete camera *n*

## 21.1. response (*json object*):

- ***result***
  - **One of**
    - *integer*
    - *null*

## 21.2. Example:

### 21.2.1. request:

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

### 21.2.2. response:

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

# 22.GET /api/display

Get current display resolution

## 22.1. response (*json object*):

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

## 22.2. Example:

### 22.2.1. request:

```
GET /api/display
```

### 22.2.2. response:

```
200 OK
Content-Type: application/json
{"result": "1920x1080-60"}
```

# 23.GET /api/display/modes

Get current resolutions supported by display

## 23.1. response (*json object*):

- ***result*** *(array)*
  - **Items** *(string)*

## 23.2. Example:

### 23.2.1. request:

```
GET /api/display/modes
```

### 23.2.2. response:

```
200 OK
Content-Type: application/json
{"result": ["640x480-59", "640x480-60", "720x576-50", "720x480-59", "720x480-60", "1280x720-50", "1280x720-59", "1280x720-60", "1920x1080-50", "1920x1080-59", "1920x1080-60", "3840x2160-25", "3840x2160-30"]}
```

# 24.GET /api/onvif/cameras

Search network for available onvif cameras

## 24.1. response (*json object*):

- ***result*** *(array)*
  - **Items** *(string)*

## 24.2. Example:

### 24.2.1. request:

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

### 24.2.2. response:

```
200 OK
Content-Type: application/json
{"result": ["192.168.86.10", "192.168.86.11"]}
```

# 25.GET /api/onvif/profiles

Query onvif cameras for available proifiles

## 25.1. params:

- ***uri*** *(string)*
- ***username*** *(string)*
- ***password*** *(string)*

## 25.2. response (*json object*):

- ***result*** *(object)*
  - ***uri*** *(string)*
  - ***name*** *(string)*
  - ***codec*** *(string)*
  - ***res_width*** *(string)*
  - ***res_height*** *(string)*
  - ***fps*** *(string)*

## 25.3. Example:

### 25.3.1. request:

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

### 25.3.2. response:

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


```
{"result": [{"codec": "H264", "name": "Multicast:mainStream", "fps": "15", "res_height": "1080", "uri": "rtsp://192.168.86.10:554/Streaming/Channels/101?transportmode=mcast&profile=Profile_1", "res_width": "1920"}, {"codec": "H264", "name": "Multicast:subStream", "fps": "20", "res_height": "480", "uri": "rtsp://192.168.86.10:554/Streaming/Channels/102?transportmode=mcast&profile=Profile_2", "res_width": "640"}, {"codec": "H264", "name": "Unicast:mainStream", "fps": "15", "res_height": "1080", "uri": "rtsp://192.168.86.10:554/Streaming/Channels/101?transportmode=unicast&profile=Profile_1", "res_width": "1920"}, {"codec": "H264", "name": "Unicast:subStream", "fps": "20", "res_height": "480", "uri": "rtsp://192.168.86.10:554/Streaming/Channels/102?transportmode=unicast&profile=Profile_2", "res_width": "640"}]}
```

# 26.GET /api/network/<iface>

Get network configuration for *iface*

## 26.1. response (*json object*):

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

## 26.2. Example:

### 26.2.1. request:

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

### 26.2.2. response:

```
200 OK
Content-Type: application/json
{"result": {"NTP": "80.86.38.193", "MAC": "00:16:2A:80:00:00", "Gateway": "192
.168.86.1", "Netmask": "255.255.255.0", "DNS": "192.168.0.1", "Address": "192.168.
86.8", "Method": "manual"}}
```

# 27.GET /api/system/info

Get sytem info

## 27.1. response (*json object*):

- ***result*** *(object)*
  - ***cpu*** *(object)*
    - ***temp*** *(number)*
  - ***date*** *(string)*
  - ***version*** *(object)*
    - ***SW_VERSION*** *(string)*
    - ***IMAGE_BUILD_DATE*** *(string)*
    - ***HW_REVISION*** *(string)*

## 27.2. Example:

### 27.2.1. request:

```
GET /api/system/info
```

### 27.2.2. response:

```
200 OK
Content-Type: application/json
{"result": {"date": "Mon Aug 29 18:22:17 2022", "version": {"SW_VERSION": "spo
tbox 1.1.1", "IMAGE_BUILD_DATE": "Fri Sep 25 11:23:18 BST 2020", "HW_REVISION": "n
ano2 1.0"}, "cpu": {"temp": "67"}}}
```

# 28.PUT /api/system/reboot

reboot device

## 28.1. response (*json object*):

- ***result***: *OK*.

## 28.2. Example:

### 28.2.1. request:

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

### 28.2.2. response:

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

# 29.PUT /api/system/restart

restart spotbox application

## 29.1. response (*json object*):

- ***result***: *OK*.

## 29.2. Example:

### 29.2.1. request:

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

### 29.2.2. response:

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

# 30.PUT /api/system/factoryreset

Reset device to factory defaults

## 30.1. response (*json object*):

- ***result***: *OK*.

## 30.2. Example:

### 30.2.1. request:

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

### 30.2.2. response:

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

# 31.PUT /api/system/user

Update login password for admin

## 31.1. request (*json object*):

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

## 31.2. response (*json object*):

- ***result***: *OK*.

## 31.3. Example:

### 31.3.1. request:

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

### 31.3.2. response:

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

# 32.GET /api/system/config

Get current system configuration

## 32.1. response (*json object*):

- ***result*** *(object)*
  - ***cameras*** *(array)*
    - **Items** *(object)*
      - ***grids*** *(object)*
        - ***fullscreen*** *(['integer', 'null'])*
        - ***grid_1x1*** *(['integer', 'null'])*
        - ***grid_2x2*** *(['integer', 'null'])*
        - ***grid_3x3*** *(['integer', 'null'])*
        - ***grid_4x4*** *(['integer', 'null'])*
      - ***label*** *(string)*
      - ***username*** *(string)*
      - ***password*** *(string)*
      - ***uri*** *(string)*
      - ***period*** *(integer)*
      - ***restart*** *(integer)*
      - ***lab_position*** *(integer)*
      - ***view*** *(['integer', 'null'])*: Minimum: *0*. Maximum: *15*.
      - ***props*** *(object)*
        - ***latency*** *(integer)*
        - ***timeout*** *(integer)*
        - ***protocols*** *(integer)*
        - ***drop_on_latency*** *(integer)*
        - ***debug*** *(integer)*
      - ***zoom*** *(integer)*
      - ***onvif*** *(boolean)*
      - ***profiles*** *(array)*
        - **Items** *(object)*
          - ***uri*** *(string)*
          - ***name*** *(string)*
          - ***codec*** *(string)*
          - ***res_width*** *(string)*
          - ***res_height*** *(string)*
          - ***fps*** *(string)*
      - ***idx*** *(integer)*: Minimum: *0*.
  - ***network*** *(object)*
    - ***eth0*** *(object)*
      - ***Method***: One of: *['manual', 'dhcp']*.
      - ***Netmask*** *(string)*
      - ***Address*** *(string)*
      - ***Gateway*** *(string)*

        - ***DNS*** *(string)*
        - ***NTP*** *(string)*
    - ***display*** *(string)*
    - ***default_view*** *(object)*
        - ***mode***: One of: *['fullscreen', 'grid_1x1', 'grid_2x2', 'grid_3x3', 'grid_4x4']*.
        - ***camera*** *(integer)*
    - ***grid_font*** *(object)*
        - ***fullscreen***: One of: *['s', 'm', 'l']*.
        - ***grid_1x1***: One of: *['s', 'm', 'l']*.
        - ***grid_2x2***: One of: *['s', 'm', 'l']*.
        - ***grid_3x3***: One of: *['s', 'm', 'l']*.
        - ***grid_4x4***: One of: *['s', 'm', 'l']*.
    - ***https_enable*** *(boolean)*
    - ***user_config*** *(object)*

## 32.2. Example:

### 32.2.1. request:

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

### 32.2.2. response:

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


```
{"result": {"default_view": {"mode": "grid_1x1"}, "network": {"lo0": {"Method"
: "static"}, "eth0": {"NTP": "80.86.38.193", "Gateway": "192.168.86.1", "Netmask":
"255.255.255.0", "DNS": "192.168.0.1", "Address": "192.168.86.8", "Method": "manua
l"}}, "cameras": [{"username": "", "idx": 0, "grids": {"grid_1x1": 0, "grid_2x2":
0, "grid_4x4": null, "grid_3x3": null}, "uri": "rtsp://192.168.1.101", "profiles":
[], "label": "C1", "period": 0, "lab_position": 0, "props": {"latency": 0, "audio"
: 0, "timeout": 0, "protocols": 0}, "password": "", "restart": 0, "zoom": 0}, {"us
ername": "", "idx": 1, "grids": {"grid_1x1": null, "grid_2x2": 1, "grid_4x4": null
, "grid_3x3": null}, "uri": "rtsp://192.168.1.102", "profiles": [], "label": "C2",
"period": 0, "lab_position": 0, "props": {"latency": 0, "audio": 1, "timeout": 0,
"protocols": 0}, "password": "", "restart": 0, "zoom": 0}], "display": "1920x1080-
60", "grid_font": {"grid_3x3": "l", "fullscreen": "s", "grid_4x4": "m", "grid_1x1"
: "s", "grid_2x2": "s"}, "https_enable": false, "user_config": {"location": "Brist
ol"}}}
```

# 33.PUT /api/system/config

Update current system configuration (requires restart to apply)

## 33.1. response (*json object*):

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

## 33.2. Example:

### 33.2.1. request:

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

### 33.2.2. response:

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

# 34. POST /api/system/config

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

## 34.1. request (Multipart-Encoded File):

## 34.2. response (*json object*):

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

## 34.3. Example:

### 34.3.1. request:

```
{ "files" : { "file": "<binary...data>"}}
```

### 34.3.2. response:

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

# 35.POST /api/system/ssl/key

Load new ssl key from file

## 35.1. request (Multipart-Encoded File):

## 35.2. response (*json object*):

- *result*: *OK*.

## 35.3. Example:

### 35.3.1. request:

```
{ "files" : { "file": "<binary...data>"}}
```

### 35.3.2. response:

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

# 36.POST /api/system/ssl/cert

Load new ssl certificate from file

## 36.1. request (Multipart-Encoded File):

## 36.2. response (*json object*):

- *result*: *OK*.

## 36.3. Example:

### 36.3.1. request:

```
{ "files" : { "file": "<binary...data>"}}
```

### 36.3.2. response:

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