

Table of Contents

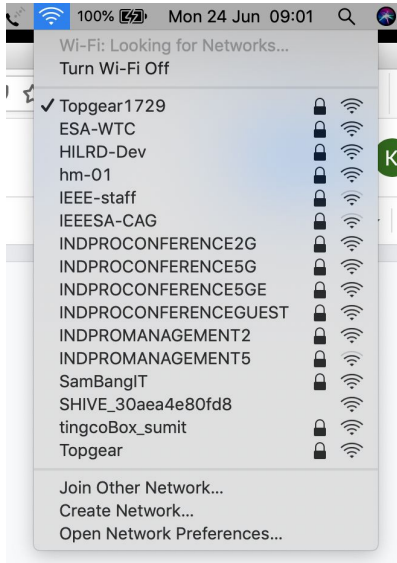
Hierarchy	2
Initial Provisioning	3
Get Mesh Config	5
Add User	6
Get User	7
Delete User	8
Get Thing	9
Set Thing	10
Change I2C address	11
Change WiFi credentials	12
OTA update	13

Hierarchy

Mesh [Root Node of mesh: SmartHive-GW]

- Controller 1 [ESP 32]
 - Port 1 [Lux, ELf, etc]
 - Thing 1 [Individual Smart Switches]
 - ...
 - Thing N
 - Port 2 [Lux, ELf, etc]
 - Thing 1 [Individual Smart Switches]
 - ...
 - Thing N
 - ...
 - Port M
- Controller 2 [Living Area]
 - Port 1 [Lux, ELf, etc]
 - Thing 1 [Individual Smart Switches]
 - ...
 - Thing N
 - Port 2 [Lux, ELf, etc]
 - Thing 1 [Individual Smart Switches]
 - ...
 - Thing N
 - ...
 - Port M
- Controller 3 [Bedroom 1]
 - Port 1 [Lux, ELf, etc]
 - Thing 1 [Individual Smart Switches]
 - ...
 - Thing N
 - Port 2 [Lux, ELf, etc]
 - Thing 1 [Individual Smart Switches]
 - ...
 - Thing N
 - ...
 - Port M

Initial Provisioning



192.168.4.1

Controller Id	MK6k6A_Y
Controller Name	30:ae:a4:e8:0f:d8
Controller Version	ki-1561119960
Controller Desc	Smart Hive Controller Unit

Router SSID	Topgear1729
Router Password	xiaqe2ustf
Mesh ID	1111
Mesh Password	qw easdzxc
RBAC Default	<input type="radio"/> Allow <input checked="" type="radio"/> Dis Allow

Super User 1	SmartHive00
Super User 2	
Super User 3	
Super User 4	

Update

HTTP POST to /conf

<http://SmartHive-GW.local/conf>

```
controller_id=MK6k6A_Y
&controller_name=30%3Aae%3Aa4%3Ae8%3A0f%3Ad8
&controller_version=ki-1561119960
&controller_description=Smart+Hive+Controller+Unit
&router_ssid=SmartHive
&router_password=qweasdzxc
&mesh_id=1111
&mesh_password=qweasdzxc
&rbac_default_allow=0
&su_1=SmartHive00
&su_2=
&su_3=
&su_4=
```

curl <http://192.168.4.1/conf>

```
-F "controller_id=MK6k6A_Y"
-F "controller_name=30%3Aae%3Aa4%3Ae8%3A0f%3Ad8"
-F "controller_version=ki-1561119960"
-F "controller_description=Smart+Hive+Controller+Unit"
-F "router_ssid=SmartHive"
-F "router_password=qweasdzxc"
-F "mesh_id=1111"
-F "mesh_password=qweasdzxc"
-F "rbac_default_allow=0"
-F "su_1=SmartHive00"
-F "su_2="
-F "su_3="
-F "su_4="
```

Get Mesh Config

```
curl -s http://SmartHive-GW.local/comm
-H "Content-Type: application/json"
-H "X-Auth-Token: SmartHive00" -H 'X-Dest-Nodes: ffffffff'
--data '{"command": "get_mesh_config"}'
{
  "30:ae:a4:e8:0f:d8": {
    "channel": 11,
    "children": [
      "3c:71:bf:4c:97:80"
    ],
    "description": "Operation Successful.",
    "elapsed_ms": 91,
    "firmware": "ki-1561119960",
    "heap": 123980,
    "layer": 1,
    "parent": "c8:d7:79:cf:59:68",
    "parent_rssi": -45,
    "self": "30:ae:a4:e8:0f:d8",
    "status": 1,
    "total_nodes": 2,
    "uptime_s": 1472,
    "x_auth_token": "SmartHive00"
  },
  "3c:71:bf:4c:97:80": {
    "channel": 11,
    "description": "Operation Successful.",
    "elapsed_ms": 1269895296,
    "firmware": "ki-1560936583",
    "heap": 156212,
    "layer": 2,
    "parent": "30:ae:a4:e8:0f:d9",
    "parent_rssi": -47,
    "self": "3c:71:bf:4c:97:80",
    "status": 1,
    "total_nodes": 2,
    "uptime_s": 57,
    "x_auth_token": "SmartHive00"
  },
  "elapsed_ms": 155
}
```

Add User

- Caller needs to be SU

```
curl -s http://SmartHive-GW.local/comm
-H "Content-Type: application/json"
-H "X-Auth-Token: SmartHive00"
-H 'X-Dest-Nodes: 3c71bf4c8520,3c71bf4c9780'
--data '{"command":"add_user", "auth_token": "SmartHive11",
"permission": "R", "port": "A", "mask": "P_-----"}'
```

Success:

```
{
  "30:ae:a4:e8:0f:d8": {
    "description": "Operation Successful.",
    "elapsed_ms": 105,
    "firmware": "ki-1561119960",
    "status": 1,
    "uptime_s": 589,
    "x_auth_token": "SmartHive00"
  },
  "elapsed_ms": 116
}
```

For dest Host with Mac Id: 3c:71:bf:4c:85:20

X-Dest-Nodes: 3c71bf4c8520 // Mac Addr without [:]

X-Dest-Nodes: ffffffff // All hosts

Permission: [R = read, W = Write]

Port: Lux, Elf, etc.

Mask: access flags for each thing base64 = 6 bits per symbol [0 .. 63]

[[ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789-_
_](#)]

0	001111	= 1 + 2 + 4 + 8	= 15 = P
1	111111	= 1 + 2 + 4 + 8 + 16 + 32	= 63 = _
2	111111	= 1 + 2 + 4 + 8 + 16 + 32	= 63 = _
3	111111	= 1 + 2 + 4 + 8 + 16 + 32	= 63 = _
4	111111	= 1 + 2 + 4 + 8 + 16 + 32	= 63 = _
5	111111	= 1 + 2 + 4 + 8 + 16 + 32	= 63 = _
6	111111	= 1 + 2 + 4 + 8 + 16 + 32	= 63 = _
7	111111	= 1 + 2 + 4 + 8 + 16 + 32	= 63 = _
8	111111	= 1 + 2 + 4 + 8 + 16 + 32	= 63 = _
9	111111	= 1 + 2 + 4 + 8 + 16 + 32	= 63 = _
10	111111	= 1 + 2 + 4 + 8 + 16 + 32	= 63 = _

Get User

- Caller needs to be SU

```
curl -s http://SmartHive-GW.local/comm
-H "Content-Type: application/json"
-H "X-Auth-Token: SmartHive00"
-H 'X-Dest-Nodes: 3c71bf4c8520,3c71bf4c9780'
--data '{"command":"get_user", "auth_token": "SmartHive11",
"permission": "R", "port": "A_-----"}'
```

Success:

```
{
  "30:ae:a4:e8:0f:d8": {
    "description": "Operation Successful.",
    "elapsed_ms": 98,
    "firmware": "ki-1561119960",
    "mask": "P_-----",
    "status": 1,
    "uptime_s": 474,
    "x_auth_token": "SmartHive00"
  },
  "elapsed_ms": 111
}
```

Delete User

- Caller needs to be SU

```
curl -s http://SmartHive-GW.local/comm  
-H "Content-Type: application/json"  
-H "X-Auth-Token: SmartHive00"  
-H 'X-Dest-Nodes: ffffffff'  
--data '{"command": "del_user", "auth_token": "SmartHive11",  
"permission": "R", "port": "A"}'
```

Success:

```
{  
  "30:ae:a4:e8:0f:d8": {  
    "description": "Operation Successful.",  
    "elapsed_ms": 96,  
    "firmware": "ki-1561119960",  
    "status": 1,  
    "uptime_s": 826,  
    "x_auth_token": "SmartHive00"  
  },  
  "elapsed_ms": 106  
}
```


Get Thing

```
curl -s http://SmartHive-GW.local/comm
  -H "Content-Type: application/json"
  -H "X-Auth-Token: SmartHive00"
  -H 'X-Dest-Nodes: ffffffff'
  --data '{"command":"get_thing", "port": "K", "addr": 0}'
```

Success:

```
{
  "30:ae:a4:e8:0f:d8": {
    "description": "Operation Successful.",
    "elapsed_ms": 121,
    "firmware": "ki-1561119960",
    "response": {
      "power": "18",
      "state": "0"
    },
    "status": 1,
    "uptime_s": 309,
    "x_auth_token": "SmartHive00"
  },
  "elapsed_ms": 127
}
```

Failure:

```
{
  "30:ae:a4:e8:0f:d8": {
    "description": "Port invalid or not responding.",
    "elapsed_ms": 86,
    "firmware": "ki-1561119960",
    "status": 0,
    "uptime_s": 235,
    "x_auth_token": "SmartHive00"
  },
  "elapsed_ms": 98
}
```

Set Thing

```
curl -s http://SmartHive-GW.local/comm  
  -H "Content-Type: application/json"  
  -H "X-Auth-Token: SmartHive00"  
  -H 'X-Dest-Nodes: ffffffff' --data '{"command":"set_thing", "port": "K", "addr": 1,  
"state":1}'
```

Success:

```
{  
  "30:ae:a4:e8:0f:d8": {  
    "description": "Operation Successful.",  
    "elapsed_ms": 140,  
    "firmware": "ki-1561119960",  
    "status": 1,  
    "uptime_s": 422,  
    "x_auth_token": "SmartHive00"  
  },  
  "elapsed_ms": 142  
}
```

Change I2C address

- Use when daisy chaining similar ports that have address collision

```
curl -s http://SmartHive-GW.local/comm  
-H "Content-Type: application/json"  
-H "X-Auth-Token: SmartHive00"  
-H 'X-Dest-Nodes: ffffffff'  
--data '{"command":"set_port_addr", "port": "K", "new_addr":"L"}'
```

Success:

```
{  
  "30:ae:a4:e8:0f:d8": {  
    "description": "Operation Successful.",  
    "elapsed_ms": 142,  
    "firmware": "ki-1561119960",  
    "status": 1,  
    "uptime_s": 1123,  
    "x_auth_token": "SmartHive00"  
  },  
  "elapsed_ms": 145  
}
```

Change WiFi credentials

- Caller needs to be SU

```
curl -s http://SmartHive-GW.local/comm  
-H "Content-Type: application/json"  
-H "X-Auth-Token: SmartHive00"  
-H 'X-Dest-Nodes: ffffffff'  
--data '{"command":"set_wifi", "ssid": "smarthive", "pass":  
"qweasdzxc"}'
```

OTA update

curl

http://smarthive-gw.local/upgr?name=ki_1561354482&url=http://192.168.0.104:8080/sh_worker.bin

Required param:

Name: name of the firmware

Url: source of Update firmware