

**< RESTful API for WISE-6610 >**  
**Document v1.0.3**

**Modification History**

<b>Revision</b>	<b>Date</b>	<b>Author</b>	<b>Comments</b>
1.0.0	2020/4/8	David Yang	
1.0.1	2020/4/17	David Yang	Fixed string error
1.0.2	2020/4/22	David Yang	Fixed restart format on Put function
1.0.3	2020/6/16	David Yang	Adding packet lose information

< RESTful API for WISE-6610 > Document .....	1
Authorization .....	4
RESTful API .....	4
/api/applications .....	6
/api/gateways.....	6
/api/networks .....	8
/api/profiles .....	11
/api/devices.....	14
/api/nodes.....	16
/api/ignored_nodes .....	18
/api/txframes .....	19
/api/rxframes .....	21
/api/handlers.....	22
/api/connectors.....	23
/api/events.....	25
/api/wise6610 .....	27
/api/lorawan_radio .....	28
/api/packet_forward .....	29
/api/MQTT_broker .....	30
/api/MQTT_Bridge .....	30
/api/network_server .....	31
/api/adv_application.....	32
/api/advantech/node .....	33
/api/advantech/node/ABC .....	33
/api/ping_req .....	37
/api/ping_res.....	38
/api/port_list.....	38
/api/ethernet .....	41
/api/reboot.....	42

# Authorization

RESTful API Server is using “Basic Auth” to authenticate , default username and password are “root”, And support HTTP(81 port) and HTTPS(8445 port)

## RESTful API

Resource	Methods	Explanation
/api/applications	GET	Supported LoRaWAN applications
/api/gateways	GET, POST	LoRaWAN gateways
/api/gateways/123	GET, PUT, POST, DELETE	Gateway with MAC=123
/api/networks	GET, POST	Network and regional settings
/api/networks/ABC	GET, PUT, DELETE	Network ABC
/api/profiles	GET, POST	Device profiles
/api/profiles/ABC	GET, PUT, DELETE	Device profile ABC
/api/devices	GET, POST	Devices commissioned for over-the-air activation (OTAA)
/api/devices/123	GET, PUT, DELETE	Commissioned device with DevEUI=123
/api/nodes	GET, POST	Active network nodes, both ABP and activated OTAA
/api/nodes/123	GET, PUT, DELETE	Active network node with DevAddr=123

<b>Resource</b>	<b>Methods</b>	<b>Explanation</b>
/api/ignored_nodes	GET, POST	Nodes ignored by the server
/api/ignored_nodes/123	GET, PUT, DELETE	Ignored node with DevAddr=123
/api/txframes	GET,POST	Frames queued for transmission
/api/rxframes	GET	Recent received frames
/api/handlers	GET	Backend handlers
/api/handlers/ABC	GET, DELETE	Backend handler for the Group ABC
/api/connectors	GET	Backend connectors
/api/connectors/ABC	GET, DELETE	Backend connector ABC
/api/events	GET	Recent errors and warnings
/api/wise6610"	GET	Get IP model name and Temperature
/api/lorawan_radio	GET, PUT	Get or put LoRaWAN Radio setting.
/api/packet_forward	GET, PUT	Get or put packet forward setting.
/api/MQTT_broker	GET, PUT	Get or put MQTT broker setting.
/api/MQTT_Bridge	GET, PUT	Get or put MQTT Bridge setting.
/api/network_server	GET, PUT	Get or put network server setting.
/api/adv_application	GET, PUT	Get or put application server setting.
api/advantech/node	GET	Get Advantech's node list

Resource	Methods	Explanation
api/advantech/node/ABC	GET	Get ABC detail data
api/ping_req	PUT	Trigger ping test.
api/ping_res	GET	Get ping test result.
api/port_list	GET	Get device port list.
api/ethernet	GET,PUT	Get and Set device ethernet setting.
api/reboot	PUT	Reboot.

## /api/applications

Request :

```
GET /api/application HTTP/1.1
```

Response:

```
HTTP/1.1 200 OK
```

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

```
[
  {
    "name": "WISE6610_Handler"
  }
]
```

## /api/gateways

Request : Get gateway list

```
GET /api/gateways HTTP/1.1
```

Response:

```
HTTP/1.1 200 OK
```

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

```
[
  {
    "delays": [],
    "dwell": [],

```

```
"health_alerts": [],
"health_decay": 0,
"ip_address": {
  "ip": "127.0.0.1",
  "port": 58334,
  "ver": 1
},
"last_alive": "2019-03-13T11:31:55Z",
"last_report": "2019-03-13T11:31:52Z",
"mac": "000A14FFFEDEF0A1",
"tx_rfch": 0
}
]
```

Request : Post gateways network server

**POST** /api/gateways HTTP/1.1

**{"mac": "000A14FFFEDEF0A2", "tx\_rfch": 0}**

Response:

HTTP/1.1 204 No Content

Request : update gateways information tp network server

**PUT** /api/gateways/000A14FFFEDEF0A2 HTTP/1.1

**{"mac": "000A14FFFEDEF0A2", "tx\_rfch": 1}**

Response:

HTTP/1.1 204 No Content

Request : Get 000A14FFFEDEF0A1 gateway infomation

**GET** /api/gateways/000A14FFFEDEF0A1 HTTP/1.1

Response:

HTTP/1.1 200 OK

**Content-Type: application/json**

```
{
  "delays": [],
  "dwell": [],
  "health_alerts": [],
  "health_decay": 0,
  "ip_address": {
    "ip": "127.0.0.1",
    "port": 58334,
    "ver": 1
  },
  "last_alive": "2019-03-13T11:37:01Z",
```

```
"last_report": "2019-03-13T11:36:52Z",
"mac": "000A14FFFEDEF0A1",
"tx_rfch": 0
}
```

```
Request : Delete 000A14FFFEDEF0A1 gateway
DELETE /api/gateways/000A14FFFEDEF0A1 HTTP/1.1
Response:
HTTP/1.1 204 No Content
```

## /api/networks

```
Request : Get networks list
GET /api/networks HTTP/1.1
Response:
HTTP/1.1 200 OK
Content-Type: application/json
```

```
[
  {
    "gw_power": 16,
    "init_chans": "0-2",
    "join1_delay": 5,
    "join2_delay": 6,
    "max_datr": 5,
    "max_eirp": 16,
    "max_power": 0,
    "min_power": 7,
    "name": "EU868",
    "netid": "000000",
    "region": "EU868",
    "rx1_delay": 1,
    "rx2_delay": 2,
    "rxwin_init": {
      "rx1_dr_offset": 0,
      "rx2_dr": 0,
      "rx2_freq": 869.525
    },
    "tx_codr": "4/5"
  },
  {
    "gw_power": 30,
```



```
"init_chans": "0-71",
"join1_delay": 5,
"join2_delay": 6,
"max_datr": 6,
"max_eirp": 30,
"max_power": 0,
"min_power": 10,
"name": "AU915",
"netid": "000000",
"region": "AU915",
"rx1_delay": 1,
"rx2_delay": 2,
"rxwin_init": {
    "rx1_dr_offset": 0,
    "rx2_dr": 8,
    "rx2_freq": 923.3
},
"tx_codr": "4/5"
```

```
},
```

```
{
```

```
"gw_power": 16,
"init_chans": "0-15",
"join1_delay": 5,
"join2_delay": 6,
"max_datr": 5,
"max_eirp": 16,
"max_power": 0,
"min_power": 7,
"name": "AS923",
"netid": "000000",
"region": "AS923",
"rx1_delay": 1,
"rx2_delay": 2,
"rxwin_init": {
    "rx1_dr_offset": 0,
    "rx2_dr": 2,
    "rx2_freq": 923.2
},
"tx_codr": "4/5"
```

```
},
```

```
{
```

```
"gw_power": 26,
```

```
"init_chans": "0-71",
"join1_delay": 5,
"join2_delay": 6,
"max_datr": 4,
"max_eirp": 30,
"max_power": 0,
"min_power": 10,
"name": "US902",
"netid": "000000",
"region": "US902",
"rx1_delay": 1,
"rx2_delay": 2,
"rxwin_init": {
    "rx1_dr_offset": 0,
    "rx2_dr": 8,
    "rx2_freq": 923.3
},
"tx_codr": "4/5"
}
]
```

Request : Post networks profile

POST /api/networks HTTP/1.1

```
{
    "gw_power": 16,
    "init_chans": "0-2",
    "join1_delay": 5,
    "join2_delay": 6,
    "max_datr": 5,
    "max_eirp": 16,
    "max_power": 0,
    "min_power": 7,
    "name": "EU868",
    "netid": "000000",
    "region": "EU868",
    "rx1_delay": 1,
    "rx2_delay": 2,
    "rxwin_init": {
        "rx1_dr_offset": 0,
        "rx2_dr": 0,
        "rx2_freq": 869.525
    },
}
```

```
"tx_codr": "4/5"  
}
```

Response:

HTTP/1.1 204 No Content

Request : Get single network profile

GET /api/networks/EU868 HTTP/1.1

Request : Delete single network profile

DELETE /api/networks/EU868 HTTP/1.1

Request : Update single network profile

PUT /api/networks/EU868 HTTP/1.1

```
{  
  "gw_power": 16,  
  "init_chans": "0-2",  
  "join1_delay": 5,  
  "join2_delay": 6,  
  "max_datr": 5,  
  "max_eirp": 16,  
  "max_power": 0,  
  "min_power": 7,  
  "name": "EU868TT",  
  "netid": "000000",  
  "region": "EU868",  
  "rx1_delay": 1,  
  "rx2_delay": 2,  
  "rxwin_init": {  
    "rx1_dr_offset": 0,  
    "rx2_dr": 0,  
    "rx2_freq": 869.525  
  },  
  "tx_codr": "4/5"  
}
```

## /api/profiles

Request : Get profile list

GET /api/profiles HTTP/1.1

Response:

HTTP/1.1 200 OK

Content-Type: application/json

```
[
  {
    "adr_mode": 0,
    "adr_set": {
      "chans": null,
      "datr": null,
      "power": null
    },
    "app": "WISE6610_Handler",
    "can_join": true,
    "fcnt_check": 1,
    "name": "US902_WISE6610_Handler",
    "network": "US902",
    "request_devstat": true,
    "rxwin_set": {
      "rx1_dr_offset": null,
      "rx2_dr": null,
      "rx2_freq": null
    },
    "txwin": 0
  },
  {
    "adr_mode": 0,
    "adr_set": {
      "chans": null,
      "datr": null,
      "power": null
    },
    "app": "WISE6610_Handler",
    "can_join": true,
    "fcnt_check": 1,
    "name": "AS923_WISE6610_Handler",
    "network": "AS923",
    "request_devstat": true,
    "rxwin_set": {
      "rx1_dr_offset": null,
      "rx2_dr": null,
      "rx2_freq": null
    },
    "txwin": 0
  }
]
```

```
},
{
  "adr_mode": 0,
  "adr_set": {
    "chans": null,
    "datr": null,
    "power": null
  },
  "app": "WISE6610_Handler",
  "can_join": true,
  "fcnt_check": 1,
  "name": "EU868_WISE6610_Handler",
  "network": "EU868",
  "request_devstat": true,
  "rxwin_set": {
    "rx1_dr_offset": null,
    "rx2_dr": null,
    "rx2_freq": null
  },
  "txwin": 0
},
```

```
{
  "adr_mode": 0,
  "adr_set": {
    "chans": null,
    "datr": null,
    "power": null
  },
  "app": "WISE6610_Handler",
  "can_join": true,
  "fcnt_check": 1,
  "name": "AU915_WISE6610_Handler",
  "network": "AU915",
  "request_devstat": true,
  "rxwin_set": {
    "rx1_dr_offset": null,
    "rx2_dr": null,
    "rx2_freq": null
  },
  "txwin": 0
},
```

```
]
```

Request : Post profile

**POST** /api/profiles HTTP/1.1

```
{
  "adr_mode": 0,
  "adr_set": {
    "chans": null,
    "datr": null,
    "power": null
  },
  "app": "WISE6610_Handler",
  "can_join": true,
  "fcnt_check": 1,
  "name": "US902_WISE6610_Handler_TT",
  "network": "US902",
  "request_devstat": true,
  "rxwin_set": {
    "rx1_dr_offset": null,
    "rx2_dr": null,
    "rx2_freq": null
  },
  "txwin": 0
}
```

Response:

HTTP/1.1 204 No Content

Request : Get single profile

**GET** /api/profiles/{profile\_name} HTTP/1.1

Example **GET** /api/profiles/US902\_WISE6610\_Handler HTTP/1.1

Request : Delete single profile

**DELETE** /api/profiles/{profile\_name} HTTP/1.1

Example **DELETE** /api/profiles/US902\_WISE6610\_Handler HTTP/1.1

Request : Update single profile

**PUT** /api/profiles/{profile\_name} HTTP/1.1

Example **PUT** /api/profiles/US902\_WISE6610\_Handler HTTP/1.1

## /api/devices

Request : Get device(OTAA) list

GET /api/devices HTTP/1.1

Response:

HTTP/1.1 200 OK

Content-Type: application/json

```
[
  {
    "appeui": "0018B243454E5333",
    "appkey": "0BECF77455192D4A8692A6FAEFE00FB3",
    "deveui": "0018B22000000274",
    "last_join": "2019-02-26T09:23:26Z",
    "node": "01C665E0",
    "profile": "US902_WISE6610_Handler"
  },
  {
    "appeui": "0018B243454E5333",
    "appkey": "0BECF77455192D4A8692A6FAEFE00FB3",
    "deveui": "0018B22000000273",
    "last_join": "2019-02-26T09:23:26Z",
    "node": "01C665E0",
    "profile": "US902_WISE6610_Handler"
  }
]
```

Request : Add device(OTAA) to network server

POST /api/profiles HTTP/1.1

```
{
  "appeui": "0018B243454E5337",
  "appkey": "0BECF77455192D4A8692A6FAEFE00FB3",
  "deveui": "0018B22000000276",
  "profile": "US902_WISE6610_Handler"
}
```

Response:

HTTP/1.1 204 No Content

Request : Get single device(OTAA) information

GET /api/devices/{deveui} HTTP/1.1

Example GET /api/profiles/0018B22000000274 HTTP/1.1

Response:

HTTP/1.1 200 OK

```
{
  "appeui": "0018B243454E5333",
  "appkey": "0BECF77455192D4A8692A6FAEFE00FB3",
```

```
"deveui": "0018B22000000274",
"last_join": "2019-02-26T09:23:26Z",
"node": "01C665E0",
"profile": "US902_WISE6610_Handler"
}
```

Request : Delete single device(OTAA)

**DELETE** /api/profiles/{deveui} HTTP/1.1

Example **DELETE** /api/profiles/0018B22000000274 HTTP/1.1

Request : Updata single device(OTAA)

**PUT** /api/profiles/{deveui} HTTP/1.1

Example **PUT** /api/profiles/0018B22000000274 HTTP/1.1

```
{
  "appeui": "0018B243454E5333",
  "appkey": "0BECF77455192D4A8692A6FAEFE00FB3",
  "deveui": "0018B22000000274",
  "profile": "US902_WISE6610_Handler"
}
```

## /api/nodes

Request : Get nodes list

**GET** /api/nodes HTTP/1.1

Response:

HTTP/1.1 200 OK

**Content-Type:** application/json

```
[
  {
    "appskey": "A46847D184323C21C992D8F9EF4B7CE9",
    "devaddr": "F7002F7E",
    "nwkskey": "965F6942F29C9EBE5747E25F07DA5114",
    "profile": "US902_WISE6610_Handler"
  },
  {
    "adr_failed": [],
    "adr_flag": 0,
    "adr_use": {
      "chans": "0-71",
      "datr": 0,
      "power": 0
    }
  }
]
```



```
},
"appskey": "1AEAAAD0D9C2F3D9BF7EAD2E0AAE3458",
"devaddr": "01E4CC66",
"devstat": [],
"fcntdown": 0,
"gateways": [
```

...

...

Request : Add node to network server

POST /api/nodes HTTP/1.1

```
{
  "appskey": "A46847D184323C21C992D8F9EF4B7CEA",
  "devaddr": "F7002F7A",
  "nwkskey": "965F6942F29C9EBE5747E25F07DA5114",
  "profile": "US902_WISE6610_Handler"
}
```

Response:

HTTP/1.1 200 OK

Request : Get single node information

GET /api/nodes/{devaddr} HTTP/1.1

Example GET /api/nodes/27002F71 HTTP/1.1

Response:

HTTP/1.1 200 OK

```
{
  "adr_flag": 1,
  "adr_set": {
    "chans": null,
    "datr": null,
    "power": null
  },
  "adr_use": {
    "chans": "0-71",
    "datr": 0,
    "power": 0
  },
  "appargs": "Advantech",
  "appskey": "A46847D184323C21C992D8F9EF4B7CE9",
  "average_qs": {
    "rssi": -36.10275684383821,
    "snr": 8.26184517896646
  }
}
```

```
},  
  "devaddr": "27002F71",  
...  
...  
...
```

Request : Delete single node

**DELETE** /api/nodes/{devaddr} HTTP/1.1

Example **DELETE** /api/nodes/27002F71 HTTP/1.1

Request : Update node information

**PUT** /api/nodes/{devaddr} HTTP/1.1

Example **PUT** /api/nodes/27002F71 HTTP/1.1

```
{  
  "appskey": "A46847D184323C21C992D8F9EF4B7CEA",  
  "devaddr": "F7002F7A",  
  "nwkskey": "965F6942F29C9EBE5747E25F07DA5114",  
  "profile": "US902_WISE6610_Handler"  
}
```

## /api/ignored\_nodes

Request : Get ignored\_nodes list

**GET** /api/ignored\_nodes HTTP/1.1

Response:

HTTP/1.1 200 OK

**Content-Type:** application/json

```
[  
  {  
    "devaddr": "FE4E939E",  
    "mask": "FFFFFFFF"  
  }  
]
```

Request : Add ignored\_nodes

**POST** /api/ignored\_nodes HTTP/1.1

```
{  
  "devaddr": "FE4E939A",  
  "mask": "FFFFFFFF"  
}
```

Response:

HTTP/1.1 204 No Content

Request : Get single ignored node information

GET /api/ignored\_nodes/{devaddr} HTTP/1.1

Example GET /api/ignored\_nodes/FE4E939A HTTP/1.1

Response:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "devaddr": "FE4E939A",
  "mask": "FFFFFFFF"
}
```

Request : Delet single ignored node

DELETE /api/ignored\_nodes/{devaddr} HTTP/1.1

Example DELETE /api/ignored\_nodes/FE4E939A HTTP/1.1

Request : update single ignored node

PUT /api/ignored\_nodes/{devaddr} HTTP/1.1

Example PUT /api/ignored\_nodes/FE4E939A HTTP/1.1

```
{
  "devaddr": "FE4E939A",
  "mask": "FFFFFFFF"
}
```

## /api/txframes

Request : Get txframes list

GET /api/txframes HTTP/1.1

Response:

HTTP/1.1 200 OK

Content-Type: application/json

```
[
  {
    "datetime": "2019-03-06T13:38:02Z",
    "devaddr": "27002F71",
    "frid": "1589622AB2B480CF",
    "txdata": {
      "data": "11",
      "port": 1
    }
  }
]
```

```
}  
]
```

Request : Add txframes

POST /api/txframes HTTP/1.1

```
{  
  "devaddr": "27002F71",  
  "txdata": {  
    "data": "22",  
    "port": 22,  
    "confirmed": false  
  }  
  , "immediately": false  
}
```

Response:

HTTP/1.1 204 No Content

## /api/rxframes

Request : Get rxframes list

GET /api/rxframes HTTP/1.1

Response:

HTTP/1.1 200 OK

Content-Type: application/json

```
[
  {
    "app": "WISE6610_Handler",
    "average_qs": {
      "rssi": -34.181814909744446,
      "snr": 7.632117928628288
    },
    "confirm": true,
    "data": "01000000",
    "datetime": "2019-02-21T11:01:40Z",
    "devaddr": "27002F71",
    "fcnt": 0,
    "frid": "15855C16FBD62544",
    "gateways": [
      {
        "mac": "000A14FFFEDEF0A1",
        "rxq": {
          "codr": "4/5",
          "datr": "SF10BW125",
          "freq": 902.3,
          "lsnr": 11.5,
          "rssi": -18,
          "time": "2019-02-21T11:01:40.188282Z",
          "tmst": 343687364
        }
      }
    ],
    "network": "US902",
    "port": 64,
    "powe": 0
  },
  ...
  ...
  ..
]
```

## Filtering

To list only some items the REST API accepts the `_filters` query parameter, which shall contain URL encoded JSON. For instance:

```
http://server:81/api/rxframes?_filters={"devaddr":"22222222"}
```

## Sorting

The REST API accepts `_sortField` and `_sortDir` query parameters to sort the list. The `_sortDir` can be either ASC or DESC. For instance:

```
http://server:81/api/rxframes?_sortField=datetime&_sortDir=ASC
```

## Pagination

The REST API accepts `_page` and `_perPage` query parameters to paginate lists, for instance:

```
http://server:8080/api/rxframes?_page=2&_perPage=20
```

The server also inserts the HTTP header `x-Total-Count` indicating the total item count.

## /api/handlers

```
Request : Get handlers list
```

```
GET /api/handlers HTTP/1.1
```

```
Response:
```

```
HTTP/1.1 200 OK
```

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

```
[
  {
    "app": "WISE6610_Handler",
    "downlink_expires": "never",
    "fields": [
      "devaddr",
      "fcnt",
      "port",
      "data",
      "datetime",
      "rssi",
```

```
        "appargs",
        "lsnr"
    ]
}
]
```

Request : Get single handlers list

GET /api/handlers/WISE6610\_Handler HTTP/1.1

Response:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "app": "WISE6610_Handler",
  "downlink_expires": "never",
  "fields": [
    "devaddr",
    "fcnt",
    "port",
    "data",
    "datetime",
    "rssi",
    "appargs",
    "lsnr"
  ]
}
```

Request : Delete single handlers list

DELETE /api/handlers/WISE6610\_Handler HTTP/1.1

Response:

HTTP/1.1 204 No Content

## /api/connectors

Request : Get connectors list

GET /api/connectors HTTP/1.1

Response:

HTTP/1.1 200 OK

Content-Type: application/json

```
[
  {
    "app": "WISE6610_Handler",
```

```
    "connid": "WISE6610_Broker",
    "enabled": true,
    "format": "json",
    "publish_uplinks": "uplink/{devaddr}",
    "received": "downlink/{devaddr}",
    "subscribe": "downlink/#",
    "uri": "mqtt://127.0.0.1:1883"
  },
  {
    "app": "WISE6610_Handler",
    "connid": "WISE6610_Websocket",
    "enabled": true,
    "format": "json",
    "publish_events": "/ws/events/{devaddr}",
    "publish_uplinks": "/ws/uplink/{devaddr}",
    "uri": "ws:"
  }
]
```

Request : Get single connectors list

GET /api/connectors/WISE6610\_Broker HTTP/1.1

Response:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "app": "WISE6610_Handler",
  "connid": "WISE6610_Broker",
  "enabled": true,
  "format": "json",
  "publish_uplinks": "uplink/{devaddr}",
  "received": "downlink/{devaddr}",
  "subscribe": "downlink/#",
  "uri": "mqtt://127.0.0.1:1883"
}
```

Request : Delete single connectors list

DELETE /api/connectors/WISE6610\_Broker HTTP/1.1

Response:

HTTP/1.1 204 No Content



## /api/events

Request : Get events list

GET /api/events HTTP/1.1

Response:

HTTP/1.1 200 OK

Content-Type: application/json

```
[
  {
    "args": "{405, \"/api/txframes\"}",
    "count": 1,
    "entity": "server",
    "evid": "8AA816EC9B4CA78AAB07340C04BF28CE",
    "first_rx": "2019-03-13T15:19:10Z",
    "last_rx": "2019-03-13T15:19:10Z",
    "severity": "warning",
    "text": "http_error"
  },
  {
    "args": "{404, \"/api/queued\"}",
    "count": 1,
    "entity": "server",
    "evid": "62648ED119D938C3CEC4464FDC19F0C1",
    "first_rx": "2019-03-13T11:59:02Z",
    "last_rx": "2019-03-13T11:59:02Z",
    "severity": "warning",
    "text": "http_error"
  },
  {
    "args": "{405, \"/api/networks\"}",
    "count": 1,
    "entity": "server",
    "evid": "08E786A6B62AEA23AB7895AA5010D00D",
    "first_rx": "2019-03-13T11:43:31Z",
    "last_rx": "2019-03-13T11:43:31Z",
    "severity": "warning",
    "text": "http_error"
  },
  {
    "count": 56876,
    "eid": "FE7E32E1",
```

```
"entity": "node",
"eid": "983C219CA211D9B24B62D091EB5283B3",
"first_rx": "2019-03-06T02:10:10Z",
"last_rx": "2019-03-13T15:31:54Z",
"severity": "error",
"text": "unknown_devaddr"
},
{
"args": "{404, \"/api/groups\"}",
"count": 1,
"entity": "server",
"eid": "682965FC6645B9EA30F526AF79870407",
"first_rx": "2019-03-13T11:52:19Z",
"last_rx": "2019-03-13T11:52:19Z",
"severity": "warning",
"text": "http_error"
},
{
"count": 184,
"eid": "FE17FA5F",
"entity": "node",
"eid": "CC2CBF4E1425A2AA9E5DA514B20E5505",
"first_rx": "2019-03-13T09:01:27Z",
"last_rx": "2019-03-13T13:32:07Z",
"severity": "error",
"text": "unknown_devaddr"
},
{
"args": "{405, \"/api/ignored_nodes/FE4E939A\"}",
"count": 1,
"entity": "server",
"eid": "6E86477FCDA593DFB34BC7BC6EACDF8",
"first_rx": "2019-03-13T15:14:47Z",
"last_rx": "2019-03-13T15:14:47Z",
"severity": "warning",
"text": "http_error"
},
{
"args": "{404, \"/api/groups\"}",
"count": 1,
"entity": "server",
"eid": "51BD884E13985D9B0B6C69A3860820CF",
```

```
    "first_rx": "2019-03-13T11:52:14Z",
    "last_rx": "2019-03-13T11:52:14Z",
    "severity": "warning",
    "text": "http_error"
  },
  {
    "args": "{404, \"/api/gateway\"}",
    "count": 1,
    "entity": "server",
    "evid": "861736E3D205465683C6FB28D4DA70A6",
    "first_rx": "2019-03-13T11:34:42Z",
    "last_rx": "2019-03-13T11:34:42Z",
    "severity": "warning",
    "text": "http_error"
  },
  {
    "args": "{404, \"/api/areas\"}",
    "count": 1,
    "entity": "server",
    "evid": "FE6B62E633B5BCA18388CCB12B7B16B5",
    "first_rx": "2019-03-13T11:12:59Z",
    "last_rx": "2019-03-13T11:12:59Z",
    "severity": "warning",
    "text": "http_error"
  },
  {
    "args": "{404, \"/api/applications\"}",
    "count": 1,
    "entity": "server",
    "evid": "028A86F723AD3212A211EC8353EF46B9",
    "first_rx": "2019-03-13T11:29:36Z",
    "last_rx": "2019-03-13T11:29:36Z",
    "severity": "warning",
    "text": "http_error"
  }
]
```

## **/api/wise6610**

Request : Get WISE6610 information

GET /api/wise6610 HTTP/1.1

Response:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "Router IP": "192.168.1.1",
  "Model Name": "WISE-6610-E100-A",
  "Temperature": 32,
  "CPU": " Generic AM33XX (Flattened Device Tree)"
}
```

## /api/lorawan\_radio

Request : Get LoRaWAN Radio setting

GET /api/lorawan\_radio HTTP/1.1

Response:

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "Radio Eanble": 1,
  "Main_Freq_0": 902700,
  "Main_Freq_1": 903400,
  "Channel0 Enable": 1,
  "Channel0 Select": 0,
  "Channel0 Offset": -400,
  "Channel1 Enable": 1,
  "Channel1 Select": 0,
  "Channel1 Offset": -200,
  "Channel2 Enable": 1,
  "Channel2 Select": 0,
  "Channel2 Offset": 0,
  "Channel3 Enable": 1,
  "Channel3 Select": 0,
  "Channel3 Offset": 200,
  "Channel4 Enable": 1,
  "Channel4 Select": 1,
  "Channel4 Offset": -300,
  "Channel5 Enable": 1,
  "Channel5 Select": 1,
  "Channel5 Offset": -100,
  "Channel6 Enable": 1,
  "Channel6 Select": 1,
}
```

```
"Channel6 Offset": 100,  
"Channel7 Enable": 1,  
"Channel7 Select": 1,  
"Channel7 Offset": 300,  
"ChannelSTD Enable": 1,  
"ChannelSTD Select": 0,  
"ChannelSTD Offset": 300,  
"ChannelSTD SF": 8,  
"ChannelSTD BW": "500KHz"  
}
```

Request : Set LoRaWAN Radio setting

PUT /api/lorawan\_radio HTTP/1.1

```
{  
  "Radio Eanble": 1,  
  "Main_Freq_0": 902700,  
  "Main_Freq_1": 903400,  
  "Channel0 Enable": 0,  
  "action": "restart"  
}
```

Response:

HTTP/1.1 204 No Content

## /api/packet\_forward

Request : Get packet forward setting

GET /api/packet\_forward HTTP/1.1

Response:

HTTP/1.1 200 OK

Content-Type: application/json

```
{  
  "Network Server": "127.0.0.1",  
  "Upstream Port": 1680,  
  "Downstream Port": 1680  
}
```

Request : Set packet forward setting

PUT /api/packet\_forward HTTP/1.1

```
{  
  "Network Server": "127.0.0.1",  
  "Upstream Port": 1680,  
}
```

```
"Downstream Port": 1680,
```

```
"action": "restart"
```

```
}
```

Response:

```
HTTP/1.1 204 No Content
```

## /api/MQTT\_broker

Request : Get MQTT broker setting

```
GET /api/MQTT_broker HTTP/1.1
```

Response:

```
HTTP/1.1 200 OK
```

```
{
```

```
  "Enable": 1,
```

```
  "Port": 1883
```

```
}
```

Request : Set MQTT broker setting

```
PUT /api/MQTT_broker HTTP/1.1
```

```
{
```

```
  "Enable": 1,
```

```
  "Port": 1883,
```

```
  "action": "restart"
```

```
}
```

Response:

```
HTTP/1.1 204 No Content
```

## /api/MQTT\_Bridge

Request : Get MQTT Bridge setting

```
GET /api/MQTT_Bridge HTTP/1.1
```

Response:

```
HTTP/1.1 200 OK
```

```
{
```

```
  "enable": 0,
```

```
  "port": 1883,
```

```
  "bridge address": "",
```

```
  "username": "",
```

```
  "password": "",
```

```
  "identifier": ""
```

```
}
```

Request : Set MQTT Bridge setting

PUT /api/MQTT\_Bridge HTTP/1.1

```
{  
  "enable": 0,  
  "port": 1883,  
  "bridge address": "",  
  "username": "",  
  "password": "",  
  "identifier": "",  
  "action": "restart"  
}
```

```
}
```

Response:

HTTP/1.1 204 No Content

## /api/network\_server

Request : Get network server setting

GET /api/network\_server HTTP/1.1

Response:

HTTP/1.1 200 OK

```
{  
  "Enable": 1,  
  "Listen Port": 1680,  
  "HTTP Port": 8080,  
  "HTTPS Port": 8443,  
  "HTTPS Enable": 0  
}
```

```
}
```

Request : Set network server setting

PUT /api/network\_server HTTP/1.1

```
{  
  "Enable": 1,  
  "Listen Port": 1680,  
  "HTTP Port": 8080,  
  "HTTPS Port": 8443,  
  "HTTPS Enable": 0,  
  "action": "restart"  
}
```

```
}
```

Response:

HTTP/1.1 204 No Content

## /api/adv\_application

Request : Get Advantech application server setting

GET /api/adv\_application HTTP/1.1

Response:

HTTP/1.1 200 OK

```
{
  "Enable": 1,
  "MQTT address": "127.0.0.1",
  "MQTT port": 1883,
  "MQTT username": "",
  "MQTT password": "",
  "Uplink topic": "uplink/#",
  "Downlink topic": "downlink/",
  "MQTT Retain": 0,
  "MQTT QoS": 0,
  "Modbus TCP Enable": 0,
  "Modbus TCP port": 502,
  "Modbus TCP Timeout": 2
}
```

Request : Set Advantech application server setting

PUT /api/network\_server HTTP/1.1

```
{
  "Enable": 1,
  "MQTT address": "127.0.0.1",
  "MQTT port": 1883,
  "MQTT username": "",
  "MQTT password": "",
  "Uplink topic": "uplink/#",
  "Downlink topic": "downlink/",
  "MQTT Retain": 0,
  "MQTT QoS": 0,
  "Modbus TCP Enable": 0,
  "Modbus TCP port": 502,
  "Modbus TCP Timeout": 2,
  "action": "restart"
}
```

Response:



HTTP/1.1 204 No Content

## /api/advantech/node

Request : Get Advantech's node list

GET /api/advantech/node HTTP/1.1

Response:

HTTP/1.1 200 OK

```
[
  {
    "devaddr": "27002F71",
    "model": "BB-WSW2C42000",
    "datetime": "2000-01-01T04:29:13Z",
    "fcnt": 1,
    "rssi": -30,
    "packet_loss": 0,
    "bat": 0
  },
  {
    "devaddr": "FF19D133",
    "model": "WISE4610-S614",
    "datetime": "2000-01-01T04:29:46Z",
    "fcnt": 1,
    "rssi": -30,
    "packet_loss": 0,
    "bat": 100
  }
]
```

## /api/advantech/node/ABC

GET /api/advantech/node/27002F71 HTTP/1.1

Response:

HTTP/1.1 200 OK

```
{
  "info": {
    "devaddr": "27002F71",
    "model": "BB-WSW2C42000",
    "datetime": "2000-01-01T04:29:13Z",
    "fcnt": 1,
```

```
"rssi": -30,
"packet_loss": 0,
"bat": 0
},
"data": {
  "AI1": {
    "RAW": 65281,
    "Mode": "0~10V",
    "Value": 9.96,
    "Unit": "V"
  },
  "AI2": {
    "RAW": 1,
    "Mode": "Disable",
    "Value": 0,
    "Unit": " "
  },
  "AI3": {
    "RAW": 43981,
    "Mode": "0~20mA",
    "Value": 13.42,
    "Unit": "mA"
  },
  "AI4": {
    "RAW": 513,
    "Mode": "+-20mA",
    "Value": -19.69,
    "Unit": "mA"
  },
  "DI1": {
    "Value": 1,
    "EN": 1
  },
  "DI2": {
    "Value": 1,
    "EN": 1
  },
  "DO": {
    "Value": 1,
    "EN": 1
  }
}
```

```
}
```

```
GET /api/advantech/node/FF19D133 HTTP/1.1
```

```
Response:
```

```
HTTP/1.1 200 OK
```

```
{
```

```
  "info": {
```

```
    "devaddr": "FF19D133",
```

```
    "model": "WISE4610-S614",
```

```
    "datetime": "2000-01-01T04:29:46Z",
```

```
    "fcnt": 1,
```

```
    "rssi": -30,
```

```
    "packet_loss": 0,
```

```
    "bat": 100
```

```
  },
```

```
  "data": {
```

```
    "DIO": {
```

```
      "value": 0,
```

```
      "status": {
```

```
        "Signal Logic": 0,
```

```
        "Start Counter": 1,
```

```
        "Get/Clean Counter Overflow": 0,
```

```
        "Get/Clean L2H Latch": 0,
```

```
        "Get/Clean H2L Latch": 0,
```

```
        "DI Change": 0
```

```
      }
```

```
    },
```

```
    "DI1": {
```

```
      "value": 0,
```

```
      "status": {
```

```
        "Signal Logic": 0,
```

```
        "Start Counter": 1,
```

```
        "Get/Clean Counter Overflow": 0,
```

```
        "Get/Clean L2H Latch": 0,
```

```
        "Get/Clean H2L Latch": 0,
```

```
        "DI Change": 0
```

```
      }
```

```
    },
```

```
    "DI2": {
```

```
      "value": 0,
```

```
      "status": {
```

```
        "Signal Logic": 0,
```

```
        "Start Counter": 1,
        "Get/Clean Counter Overflow": 0,
        "Get/Clean L2H Latch": 0,
        "Get/Clean H2L Latch": 0,
        "DI Change": 0
    }
},
"DI3": {
    "value": 0,
    "status": {
        "Signal Logic": 0,
        "Start Counter": 1,
        "Get/Clean Counter Overflow": 0,
        "Get/Clean L2H Latch": 0,
        "Get/Clean H2L Latch": 0,
        "DI Change": 0
    }
},
"AIO": {
    "Range": 4,
    "Raw Data": 32774,
    "Event": 0,
    "MaxVal": 32775,
    "MinVal": 32762,
    "status": {
        "Low Alarm": 0,
        "High Alarm": 0
    }
},
"AI1": {
    "Range": 9,
    "Raw Data": 0,
    "Event": 0,
    "MaxVal": 2,
    "MinVal": 0,
    "status": {
        "Low Alarm": 1,
        "High Alarm": 0
    }
},
"AI2": {
    "Range": 11,
```

```
    "Raw Data": 32767,  
    "Event": 0,  
    "MaxVal": 32768,  
    "MinVal": 32767,  
    "status": {  
        "Low Alarm": 1,  
        "High Alarm": 0  
    }  
},  
"AI3": {  
    "Range": 5,  
    "Raw Data": 2,  
    "Event": 0,  
    "MaxVal": 18,  
    "MinVal": 0,  
    "status": {  
        "Low Alarm": 0,  
        "High Alarm": 0  
    }  
},  
"AI4": {  
    "Range": 0,  
    "Raw Data": 0,  
    "Event": 0,  
    "MaxVal": 0,  
    "MinVal": 0,  
    "status": {  
        "Low Alarm": 0,  
        "High Alarm": 0  
    }  
},  
"GNSS": {  
    "Latitude": "0.000000 N",  
    "Longitude": "0.000000 E"  
}  
}  
}
```

**/api/ping\_req**

PUT /api/ping\_req HTTP/1.1

```
{
  "IPAddress": "192.168.1.100"
}
```

Response:

HTTP/1.1 204 No Content

## /api/ping\_res

GET /api/ping\_res HTTP/1.1

Response:

HTTP/1.1 200 OK

Response:

```
{
  "IPAddress": "192.168.1.100",
  "status": "Success"
}
```

## /api/port\_list

GET /api/port\_list HTTP/1.1

Response:

HTTP/1.1 200 OK

```
[
  {
    "protocol": "tcp",
    "port": "127.0.0.1:5354",
    "name": "dnsmasq"
  },
  {
    "protocol": "tcp",
    "port": "0.0.0.0:8080",
    "name": "beam"
  },
  {
    "protocol": "tcp",
    "port": "0.0.0.0:81",
    "name": "RESTful"
  },
  {
    "protocol": "tcp",
```

```
    "port": "0.0.0.0:4369",
    "name": "epmd"
  },
  {
    "protocol": "tcp",
    "port": "0.0.0.0:22",
    "name": "sshd"
  },
  {
    "protocol": "tcp",
    "port": "0.0.0.0:1880",
    "name": "node-red"
  },
  {
    "protocol": "tcp",
    "port": "0.0.0.0:1883",
    "name": "broker_mqtts"
  },
  {
    "protocol": "tcp",
    "port": "0.0.0.0:8445",
    "name": "RESTful"
  },
  {
    "protocol": "tcp",
    "port": "0.0.0.0:47101",
    "name": "beam"
  },
  {
    "protocol": "tcp",
    "port": "::1:5354",
    "name": "dnsmasq"
  },
  {
    "protocol": "tcp",
    "port": ":::80",
    "name": "mhttpd"
  },
  {
    "protocol": "tcp",
    "port": ":::4369",
    "name": "epmd"
  }
```

```
},
{
  "protocol": "tcp",
  "port": ":::53",
  "name": "totd"
},
{
  "protocol": "tcp",
  "port": ":::22",
  "name": "sshd"
},
{
  "protocol": "tcp",
  "port": ":::443",
  "name": "mhttpd"
},
{
  "protocol": "udp",
  "port": "0.0.0.0:67",
  "name": "dhcpd"
},
{
  "protocol": "udp",
  "port": "127.0.0.1:1884",
  "name": "broker_mqtts"
},
{
  "protocol": "udp",
  "port": "0.0.0.0:6004",
  "name": "AppServer"
},
{
  "protocol": "udp",
  "port": "0.0.0.0:1680",
  "name": "beam"
},
{
  "protocol": "udp",
  "port": "0.0.0.0:161",
  "name": "snmpd"
},
{
```



```
    "protocol": "udp",
    "port": "127.0.0.1:5354",
    "name": "dnsmasq"
  },
  {
    "protocol": "udp",
    "port": ":::53",
    "name": "totd"
  },
  {
    "protocol": "udp",
    "port": ":::161",
    "name": "snmpd"
  },
  {
    "protocol": "udp",
    "port": "::1:5354",
    "name": "dnsmasq"
  }
]
```

## /api/ethernet

GET /api/ethernet HTTP/1.1

Response:

HTTP/1.1 200 OK

```
{
  "DHCP Client": 0,
  "IP Address": "192.168.1.1",
  "netmask": "255.255.255.0",
  "gateway": "",
  "DNS": "",
  "DHCP leases": 1,
  "IP Pool Start": "192.168.1.2",
  "IP Pool End": "192.168.1.254"
}
```

Setting DHCP client

PUT /api/ethernet HTTP/1.1

```
{
  "DHCP Client": 1,
```

```
"DHCP leases": 0  
}
```

Response:

```
HTTP/1.1 204 No Content
```

**Note:Effective after reboot**

Setting static IP

```
PUT /api/ethernet HTTP/1.1
```

```
{  
  "DHCP Client": 0,  
  "IP Address": "192.168.1.1",  
  "netmask": "255.255.255.0",  
  "gateway": "",  
  "DNS": "",  
  "DHCP leases": 1,  
  "IP Pool Start": "192.168.1.2",  
  "IP Pool End": "192.168.1.254"  
}
```

Response:

```
HTTP/1.1 204 No Content
```

**Note:Effective after reboot**

## /api/reboot

```
PUT /api/reboot HTTP/1.1
```

Response:

```
HTTP/1.1 204 No Content
```