Peplink Router API Documentation

Overview

The API is a set of HTTP endpoints. Each endpoint is an HTTP GET requests or POST requests with JSON arguments and JSON responses.

The access port is same as that configured for Web Admin access. For security reason, however, the API should always be used under Secure HTTP (HTTPS) access.

Getting Started

API Resource URL

https://<device_ip_address>/api/<function endpoint>
e.g. https://192.168.1.1/api/status.wan.connection

Authentication - with Admin User account

As in Web Admin Access, Admin User account can access the API with the user name and password. After successfully login, the session will be authorized for subsequent access to the allowed APIs.

The session id is returned from cookie named "bauth" under Secure HTTP access.

Authentication - with Client ID

API can be accessed with Client ID / secret, generated in advanced from the authenticated user, without disclosing the user name and password information

Successfully authorization with Client ID / secret with be granted with access token, which can be used along subsequent access to the allowed APIs

Permission

- Read-Only Permission - It can only read the status and the config.
- Read-Write Permission - It can read the status and the config. It can also change the config.
- Admin Permission - It can manage the client and the token. It also have the "Read-Write Permission"

Admin Permission can only be granted by admin user account login

Create Client

Admin Permission is needed to create the client

POST the name and scope by using the API call /api/auth.client endpoint

Example:

```
POST /api/auth.client HTTP/1.1
Host: 192.168.1.1
Content-Type: application/json

{
        "name": "Client 1",
        "scope": "api.read-only"
}
```

Successful request will return client ID and client secret.

Generate token

POST the client ID, client secret and scope(optional) by using the API call /api/auth.token.grant

Example:

```
POST /api/auth.token.grant HTTP/1.1
Host: 192.168.1.1
Content-Type: application/json

{
```

```
        "clientId": "9270c250111cabab02058007bb72217e",
        "clientSecret": "cf5fe1c51252a058ebd6bd7d5f493cf5"
}
```

Matched client ID and secret will return access token.

How to use the access token

Add the access token as a GET parameter

Example:

```
GET /api/status.wan.connection?accessToken=43c65216eb16d779092fc40b184a1794 HTTP/1.1
Host: 192.168.1.1
```

Valid access token will get resource.

HTTP Method

- **GET** to retrieves simple data
- **POST** to manipulate configuration or execute various actions, along with supplied arguments in JSON format

GET Request Parameter

Parameters are passed in the query string (after the ? in the URL)

Example:

```
GET /api/status.wan.connection?id=1&lite=yes HTTP/1.1
Host: 192.168.1.1
```

POST Request Parameter

Parameters in POST requests must be in JSON-encoded format

Example:

```
POST /api/login HTTP/1.1
Host: 192.168.1.1
Content-Type: application/json

{
        "username": "admin",
        "password": "admin"
}
```

Response

API response are in JSON-encoded format. The JSON response is an JSON object, with "stat" to indicate if the request is done successfully (ok) or not (fail)

Typically, a successfully response will have an "response" describe the retrieved information or result of the request

In failed responses, "code" is provided for the error code, and message about the failure, if any, will be described in "message"

	Type	*Notation*	*Description*
stat	String	{ok fail}	ok - API call success fail - API call not success
response	Any	-	Any additional information of the success call will be here
code	Number	<int>	Error code of the API call, only appear if the API call not success
message	String	<String>	Error message of the API call, only appear if the API call not success
notice	Object	<Object>	Extra information about this API request (but not part of the normal response). For example, the notice to inform when the API is undocumented (for experimental / beta), or when it is in deprecate state or already replace with another API endpoint.

For success API call

```
{
        "stat": "ok"
}
```

Or

```
{
        "stat": "ok",
        "response": <Any JSON support type>
}
```

For success API call (beta)

```
{
        "stat": "ok",
        "notice": {
                "status": "beta"
        },
        "response": <Any JSON support type>
}
```

For fail API call

```
{
        "stat": "fail",
        "code": <int>,
        "message": <string>
}
```

API Reference List

- POST login
- POST logout
- GET auth.client
- POST auth.client
- GET auth.client.token
- POST auth.token.grant
- POST auth.token.revoke
- POST cmd.billing.newCycle
- GET cmd.carrier.scan
- POST cmd.carrier.scan
- POST cmd.carrier.select
- POST cmd.channelPci.lock
- POST cmd.channelPci.scan
- POST cmd.config.apply
- POST cmd.config.discard
- POST cmd.config.restore
- POST cmd.mesh.discover
- GET cmd.mesh.discover.result
- POST cmd.mesh.request
- POST cmd.port.poe.disable
- POST cmd.port.poe.enable
- POST cmd.sendUssd
- GET cmd.sms.get
- POST cmd.sms.sendMessage
- POST cmd.starlink
- GET cmd.ap
- POST cmd.ap
- POST cmd.cellularModule.rescanNetwork
- POST cmd.cellularModule.reset
- POST cmd.system.reboot
- POST cmd.wan.cellular
- POST cmd.wifi.connect
- POST cmd.wifi.disconnect
- POST cmd.wifi.forget
- GET cmd.wifi.result
- GET cmd.wifi.scan
- POST config.gpio
- GET config.mesh
- POST config.mesh
- GET config.speedfusionConnectProtect
- POST config.speedfusionConnectProtect
- GET config.ssid.profile
- POST config.ssid.profile
- GET config.wan.connection
- POST config.wan.connection
- POST config.wan.connection.priority

GET info.firmware
GET info.location
GET info.time
POST status.cellularModule.temperature
GET status.client
GET status.extap.mesh
GET status.extap.mesh.link
GET status.gpio.input
GET status.gpio.output
GET status.lan.profile
GET status.pepvpn
GET status.wan.connection
GET status.wan.connection.allowance

API Reference

POST /api/login

API

Acquire proper authorization for other API requests.
After a successful authentication, the obtained cookie session can be used for other API requests.
Permission GET is granted for Read-only user access, while Permission GET and POST are granted for Read-write user access.
The session is similar to that being used in Web Admin Access, and governed by the same session idle timeout.
For a more persistent API access, consider authorization with Client ID / Secret

Avaliable in 7.0.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
username	String	<string>	require	Username
password	String	<string>	require	Password

Return Parameters

Return JSON

	Type	*Notation*	*Description*
permission	Object	<Permission_Obj>	Permission granted. Most APIs require a proper permission to access.

<Permission_Obj>

	Type	*Notation*	*Description*
GET	Number	{ 0, 1 }	1 - Allow retrieving data from the device 0 - Not allow retrieving data from the device
POST	Number	{ 0, 1 }	1 - Allow changing device settings 0 - Not allow changing device settings

cURL Example

```
> curl -c cookies.txt -H "Content-Type: application/json" -X POST -d
'{"username":"user","password":"pass"}' http://192.168.1.1/api/login
```

```
{
    "stat": "ok",
    "response": {
        "permission": {
            "GET": 1,
            "POST": 1
        }
    }
}
```

POST /api/logout

API

Properly logout the current session.
It is advised to logout immediately after use.

Avaliable in 7.0.0 or later

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST http://192.168.1.1/api/logout
```

```
{
    "stat": "ok"
}
```

GET /api/auth.client

Auth

Get the authentication client list. Only Admin Permission can access this information.

Avaliable in 7.1.1 or later

Return Parameters

Return JSON

	Type	*Notation*	*Description*
-	Array	list of <Client_Obj>	List of the auth client.

<Client_Obj>

	Type	*Notation*	*Description*
name	String	<string>	Name of the client
clientId	String	<hash>	Client ID for granting the access token
clientSecret	String	<hash>	Client Secret for granting the access token
confidential	Boolean	<boolean>	Confidential or public client type
createTimestamp	Number	<integer>	Create timestamp of the client
scope	String	{ api, api.read-only }	The scope of the client

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/auth.client
```

```
{
    "stat": "ok",
    "response": [
        {
            "name": "Client 1",
            "clientId": "9270c250111cabab02058007bb72217e",
            "clientSecret": "cf5fe1c51252a058ebd6bd7d5f493cf5",
            "confidential": false,
            "createTimestamp": 32172904,
            "scope": "api.read-only"
        }
    ]
}
```

POST /api/auth.client

Auth

Create a new client

Create a new client by giving the name and scope. Only Admin Permission can access this information.

Avaliable in 7.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
action	String	{ add }	require	

	Type	Notation	Mandatory	Description
name	String	<string>	require	Client name
scope	String	{ api, api.read-only }	require	Scope of the client api - Read-Write permission api.read-only - Read-Only permission

Return Parameters

Return JSON

	Type	Notation	Description
name	String	<string>	Name of the client
clientId	String	<hash>	Client ID for granting the access token
clientSecret	String	<hash>	Client Secret for granting the access token
confidential	Boolean	<boolean>	Confidential or public client type
createTimestamp	Number	<integer>	Create timestamp of the client
scope	String	{ api, api.read-only }	The scope of the client

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"action":"add","name":"Client
2","scope":"api"}' http://192.168.1.1/api/auth.client
```

```
{
    "stat": "ok",
    "response": {
        "name": "Client 2",
        "clientId": "0396c250111dcaef02058007bb72217e",
        "clientSecret": "de5cd1c51252a13854d6bd7ddeabbcf5",
        "confidential": false,
        "createTimestamp": 32175831,
        "scope": "api"
    }
}
```

Remove a client

Remove the client by giving the client ID. Only Admin Permission can access this information.

Avaliable in 7.1.1 or later

Input Parameters

	Type	Notation	Mandatory	Description
action	String	{ remove }	require	
clientId	String	<hash>	require	Client ID

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d
'{"action":"remove","clientId":"0396c250111dcaef02058007bb72217e"}'
http://192.168.1.1/api/auth.client
```

```
{
    "stat": "ok"
}
```

GET /api/auth.client.token

Auth

Obtain the access token list by providing the client ID
Only Admin Permission can access this information.

Avaliable in 7.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
clientId	String	<hash>	optional	Client ID. If this field is absent, all the access tokens will be obtained.

Return Parameters

Return JSON

	Type	*Notation*	*Description*
-	Array	list of <Access_Token_Obj>	List of access token information

<Access_Token_Obj>

	Type	*Notation*	*Description*
accessToken	String	<hash>	Access token
clientId	String	<hash>	Client ID
clientName	String	<string>	Client Name
authorizationType	Number	{ 3 }	Authorization type. Always get 3 for client credentials grant
scope	String	{ api, api.read-only }	The scope of the access token
createTimestamp	Number	<integer>	Issued date in timestamp

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/auth.client.token?
clientId=0396c250111dcaef02058007bb72217e
```

```
{
    "stat": "ok",
    "response": [
        {
            "accessToken": "43c65216eb16d779092fc40b184a1794",
            "clientId": "0396c250111dcaef02058007bb72217e",
            "clientName": "Client 1",
            "authorizationType": 3,
            "scope": "api.read-only",
            "createTimestamp": 32177831
        }
    ]
}
```

POST /api/auth.token.grant

Auth

Generate a new access token by giving the clientId and clientSecret.

Avaliable in 7.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
clientId	String	<hash>	require	Client ID
clientSecret	String	<hash>	require	Client Secret
scope	String	{ api, api.read-only }	optional	Scope of the access token generated api - Read-write permission of API api.read-only - Read-only permission of API

Return Parameters

Return JSON

	Type	*Notation*	*Description*
accessToken	String	<hash>	Access token
authorizationType	Number	{ 3 }	Authorization type. Always out 3 for client credentials grant
scope	String	{ api, api.read-only }	The scope of the access token
expiresIn	Number	<integer>	Expires in seconds

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d
'{"clientId":"0396c250111dcaef02058007bb72217e","clientSecret":"de5cd1c51252a13854d6bd7ddeabbcf5","s
ope":"api"}' http://192.168.1.1/api/auth.token.grant
```


```
{
    "stat": "ok",
    "response": {
        "accessToken": "43c65216eb16d779092fc40b184a1794",
        "authorizationType": 3,
        "scope": "api",
        "expiresIn": 172800
    }
}
```


POST /api/auth.token.revoke

Auth

Revoke the access token provided.
Only Admin Permission or self revoke can access this information.

Avaliable in 7.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
accessToken	String	<hash>	require	Access token desired to revoke

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d
'{"accessToken":"0396c250111dcaef02058007bb72217e"}' http://192.168.1.1/api/auth.token.revoke
```


```
{
    "stat": "ok"
}
```


POST /api/cmd.billing.newCycle

API

Start the new billing cycle by Connection ID and SIM ID

Avaliable in 8.1.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	WAN Connection ID to be renew billing cycle
simId	Number	[1,2]	optional	SIM ID to be renew billing cycle 1 is for SIM A, and 2 is for SIM B Always send 1 for single SIM model If the WAN Connection is not support cellular, the param will be ignored.

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"connId":4,"simId":1}'
http://192.168.1.1/api/cmd.billing.newCycle
```

```
{
    "stat": "ok"
}
```

GET /api/cmd.carrier.scan

API

Obtain the result of discovered cellular network.
The API will always return fail when the WAN connection does not support carrier scan.

Avaliable in 8.0.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	Specify which WAN connection ID is wanted to scan cellular network
reference	String	{ yes, no }	require	The cellular network which is wanted to select

Return Parameters

Return JSON

	Type	*Notation*	*Description*
scanStatus	String	{ scanning, done }	Report the scanning status
timestamp	Number	<integer>	Timestamp of the carrier list
list	Array	list of <Scan_Carrier_Obj>	List of discovered carrier
reference	Object	<Reference_Obj>	Current configuration

<Scan_Carrier_Obj>

	Type	*Notation*	*Description*
name	String	<string>	Name of the carrier
mobileType	String	{ 2G, 3G, LTE }	-
mcc	String	3 digits <string>	Mobile Country Code
mnc	String	2-3 digits <string>	Mobile Network Code
pcs	Number	[0, 1]	-

<Reference_Obj>

	Type	*Notation*	*Description*
activeSim	Object NULL	<In_Use_SIM_Obj> NULL	Active SIM information. If there is no active SIM, this value is JSON NULL

<In_Use_SIM_Obj>

	Type	*Notation*	*Description*
simId	Number	{ 1, 2 }	SIM ID of the active SIM
selectedCarrier	Object NULL	<Carrier_Obj> NULL	The selected network If is it auto, this value is JSON NULL.

<Carrier_Obj>

	Type	*Notation*	*Description*
name	String	<string>	Name of the carrier

<Carrier_Obj>

	Type	*Notation*	*Description*
mcc	String	3 digits <string>	-
mnc	String	2-3 digits <string>	-
pcs	Number	[0, 1]	-

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/cmd.carrier.scan?connId=4&reference=yes
```


```
{
    "stat": "ok",
    "response": {
        "scanStatus": "scanning",
        "list": [
            {
                "name": ".csl",
                "mobileType": "LTE",
                "mcc": "454",
                "mnc": "0",
                "pcs": 0
            },
            {
                "name": "SMT HK",
                "mobileType": "LTE",
                "mcc": "454",
                "mnc": "6",
                "pcs": 0
            }
        ],
        "reference": {
            "activeSim": {
                "simId": 1,
                "cellularNetwork": null
            }
        }
    }
}
```


POST /api/cmd.carrier.scan

API

Obtain the result of discovered cellular network.
The API will always return fail when the WAN connection does not support carrier scan.

Avaliable in 8.1.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
action	String	{start}	optional	Trigger the scan start action
connId	Number	<conn_id>	require	Specify which WAN connection ID is wanted to scan cellular network
reference	String	{ yes, no }	optional	The cellular network which is wanted to select

Return Parameters

Return JSON

	Type	*Notation*	*Description*
scanStatus	String	{ scanning, done }	Report the scanning status
timestamp	Number	<integer>	Timestamp of the carrier list

Return JSON

	Type	*Notation*	*Description*
list	Array	list of <Scan_Carrier_Obj>	List of discovered carrier
reference	Object	<Reference_Obj>	Current configuration

<Scan_Carrier_Obj>

	Type	*Notation*	*Description*
name	String	<string>	Name of the carrier
mobileType	String	{ 2G, 3G, LTE }	-
mcc	String	3 digits <string>	Mobile Country Code
mnc	String	2-3 digits <string>	Mobile Network Code
pcs	Number	[0, 1]	-

<Reference_Obj>

	Type	*Notation*	*Description*
activeSim	Object NULL	<In_Use_SIM_Obj> NULL	Active SIM information. If there is no active SIM, this value is JSON NULL

<In_Use_SIM_Obj>

	Type	*Notation*	*Description*
simId	Number	{ 1, 2 }	SIM ID of the active SIM
selectedCarrier	Object NULL	<Carrier_Obj> NULL	The selected network If is it auto, this value is JSON NULL.

<Carrier_Obj>

	Type	*Notation*	*Description*
name	String	<string>	Name of the carrier
mcc	String	3 digits <string>	-
mnc	String	2-3 digits <string>	-
pcs	Number	[0, 1]	-

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d
'{"action":"start","connId":"4","reference":"yes"}' http://192.168.1.1/api/cmd.carrier.scan
```


```
{
    "stat": "ok",
    "response": {
        "scanStatus": "scanning",
        "list": [
            {
                "name": ".csl",
                "mobileType": "LTE",
                "mcc": "454",
                "mnc": "0",
                "pcs": 0
            },
            {
                "name": "SMT HK",
                "mobileType": "LTE",
                "mcc": "454",
                "mnc": "6",
                "pcs": 0
            }
        ],
        "reference": {
```

```
            "activeSim": {
                "simId": 1,
                "cellularNetwork": null
            }
        }
    }
}
```


POST /api/cmd.carrier.select

API

Update the cellular network selection

Avaliable in 8.0.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	Specify which WAN connection ID is wanted to change the carrier selection
simId	Number	{ 1, 2 }	optional	Specify which SIM is wanted to change the carrier selection
selectedCarrier	Object	<Carrier_Obj>	require	The carrier which is wanted to select

<Carrier_Obj>

	Type	*Notation*	*Mandatory*	*Description*
mcc	String	3 digits <string>	require	-
mnc	String	2-3 digits <string>	require	-
pcs	Number	[0, 1]	require	-
name	String	<String>	optional	-

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"connId":4,"selectedCarrier"
{"mcc":"345","mnc":"23","pcs":0}}' http://192.168.1.1/api/cmd.carrier.select
```


```
{
    "stat": "ok"
}
```


POST /api/cmd.channelPci.lock

API Cellular

Lock the connected LTE network on specific channel number (and Physical Layer Cell Identity(PCI)).
The API will always return fail when the WAN connection does not support channel PCI lock

Avaliable in 8.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	Specify which WAN connection ID is wanted to lock
sim	Array	list of <SIM_Obj>	require	Specify channel and PCI for the SIM card

<SIM_Obj>

	Type	*Notation*	*Mandatory*	*Description*
id	Number	<sim_id>	require	SIM ID 1 for SIM A

<SIM_Obj>

	Type	*Notation*	*Mandatory*	*Description*
				2 for SIM B
value	Object Null	<CH_PCI_Obj>	require	Specify channel PCI to lock Provide a JSON Null here to clear the lock for the SIM

<CH_PCI_Obj>

	Type	*Notation*	*Mandatory*	*Description*
channel	Number	[0, 65535]	require	Specify channel to lock
pci	Number	[0, 503]	require	Specify PCI to lock

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"connId":4,"sim":
[{"id":1,"value":{"channel":1350,"pci":77}}]}' http://192.168.1.1/api/cmd.channelPci.lock
```

```
{
    "stat": "ok"
}
```

POST /api/cmd.channelPci.scan

API Cellular

Obtain the result of descoverd LTE cellular network.
Provide action=start as parameter to rescan the channel PCI
The API will always return fail when the WAN connection does not support channel PCI scan

Avaliable in 8.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
action	String	{start}	optional	Trigger the scan start action
connId	Number	<conn_id>	require	Specify which WAN connection ID is wanted to scan channel PCI

Return Parameters

Return JSON

	Type	*Notation*	*Description*
scanStatus	String	{ scanning, done }	Report the scanning status
timestamp	Number	<integer>	TImestamp of the carrier list
list	Array	list of <CH_PCI_Obj>	List of discovered channel PCI

<CH_PCI_Obj>

	Type	*Notation*	*Description*
pci	Number	<integer>	Physical-layer Cell Identity
earfcn	Number	<integer>	E-UTRA Absolute radio-frequency channel number
cellUtranId	Number	<integer>	Cell UTRAN ID
plmn	Array	list of <PLMN_Obj>	Public land mobile network information

<PLMN_Obj>

	Type	*Notation*	*Description*
mcc	String	3 digits <string>	Three decimal digits as Mobile Country Code(MCC)
mnc	String	2/3 digits <string>	Two or Three decimal digits as Mobile Network Code(MNC)

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"action":"start","connId":4}
http://192.168.1.1/api/cmd.channelPci.scan
```


```
{
    "stat": "ok",
    "response": {
        "scanStatus": "scanning",
        "timestamp": 1577836800,
        "list": [
            {
                "pci": 371,
                "earfcn": 3000,
                "cellUtranId": 23574039,
                "plmn": [
                    {
                        "mcc": "454",
                        "mnc": "00"
                    }
                ]
            }
        ]
    }
}
```


POST /api/cmd.config.apply

API internal testing

Apply changes

Apply the changes on pending config

Avaliable in 7.1.1 or later

Return Parameters

Return JSON

	Type	*Notation*	*Description*
warning	String	<string>	Changes are applied with a warning message. If there is no warning message, this field will not appear

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST
http://192.168.1.1/api/cmd.config.apply
```


```
{
    "stat": "ok"
}
```


POST /api/cmd.config.discard

API internal testing

Discard changes

Discard changes of pending config

Avaliable in 7.1.1 or later

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST
http://192.168.1.1/api/cmd.config.discard
```

```
{
    "stat": "ok"
}
```

POST /api/cmd.config.restore

API internal testing

Restore the config to the factory default

Avaliable in 8.5.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
timestamp	Number	<integer>	require	A timestamp value which is get from /api/info.time

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"timestamp":1723077803}'
http://192.168.1.1/api/cmd.config.restore
```

```
{
    "stat": "ok"
}
```

POST /api/cmd.mesh.discover

API alpha

Trigger Integrated AP/External AP to scan for default APs.

Avaliable in 8.4.2 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
apList	Array	list of <AP Serial Numbers>	require	Specify the APs to scan.

Return Parameters

Return JSON

	Type	*Notation*	*Description*
apList	Array	list of <Capable_AP_Obj>	List of available APs for discovery

<Capable_AP_Obj>

	Type	*Notation*	*Description*
serialNumber	String	<sn>	Serial Number of the capable AP
status	String	{online, scanning}	Current status of the capable AP

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"apList":["19FF-ABCD-
1111","19FF-ABCE-2222"]}' http://192.168.1.1/api/cmd.mesh.discover
```

```
{
    "stat": "ok",
    "response": {
        "apList": [
            {
```

```
                "serialNumber": "19FF-ABCD-1111",
                "status": "scanning"
            },
            {
                "serialNumber": "19FF-ABCE-2222",
                "status": "scanning"
            }
        ]
    }
}
```

GET /api/cmd.mesh.discover.result

API alpha

Retrieve discovery results for default APs.

Avaliable in 8.4.2 or later

Return Parameters

Return JSON

	Type	*Notation*	*Description*
apList	Array	list of <Capable_AP_Obj>	List of available APs for discovery
candidateList	Array	list of <Candidate_Obj>	List of candidates that can accept mesh requests.

<Capable_AP_Obj>

	Type	*Notation*	*Description*
serialNumber	String	<sn>	Serial Number of the capable AP
status	String	{online, scanning}	Current status of the capable AP

<Candidate_Obj>

	Type	*Notation*	*Description*
id	String	<mac>	Candidate ID.
bssList	Array	list of <Candidate_BSS_Obj>	Array of discovered BSS Objects from candidate.
status	String	{new, adopting}	Status of the candidate.

<Discovered_BSS_Obj>

	Type	*Notation*	*Description*
bssid	String	<mac>	BSSID of the candidate.
serialNumber	String	<sn>	Serial Number of scanning AP.
rssi	Number	<integer>	Signal strength of the BSSID observed by the scanning AP.
channel	Number	<Number>	Channel number.
lastUpdate	Number	<Number>	Time since the last update in seconds.

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/cmd.mesh.discover.result
```

```
{
    "stat": "ok",
    "response": {
        "apList": [
            {
                "serialNumber": "19FF-ABCD-1111",
                "status": "scanning"
```

```
            },
            {
                "serialNumber": "19FF-ABCE-2222",
                "status": "scanning"
            }
        ],
        "candidates": {
            "order": [
                "A8:C0:EA:34:C7:A0"
            ],
            "A8:C0:EA:34:C7:A0": {
                "bssList": [
                    {
                        "bssid": "A8:C1:EA:34:C7:A4",
                        "serialNumber": "19FF-ABCD-1111",
                        "rssi": -25,
                        "channel": 11,
                        "lastUpdate": 1
                    },
                    {
                        "bssid": "A8:C1:EA:34:C7:A4",
                        "serialNumber": "19FF-ABCE-2222",
                        "rssi": -33,
                        "channel": 11,
                        "lastUpdate": 1
                    },
                    {
                        "bssid": "A8:C1:EA:34:C7:A8",
                        "serialNumber": "19FF-ABCD-1111",
                        "rssi": -32,
                        "channel": 36,
                        "lastUpdate": 1
                    },
                    {
                        "bssid": "A8:C1:EA:34:C7:A8",
                        "serialNumber": "19FF-ABCE-2222",
                        "rssi": -32,
                        "channel": 36,
                        "lastUpdate": 1
                    }
                ],
                "status": "new"
            }
        }
    }
}
```


POST /api/cmd.mesh.request

API alpha

Trigger Integrated AP/External AP to send mesh request to the target candidate.

Avaliable in 8.4.2 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
candidateId	String	<mac>	require	Target candidate ID

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"candidateId":
["00:1A:DD:FF:FF:FF"]}' http://192.168.1.1/api/cmd.mesh.request
```

```
{
    "stat": "ok"
}
```

POST /api/cmd.port.poe.disable

API

Disable the PoE of the port.

For Switch and Balance/MAX device:
Only Port ID is needed. To success turn off the PoE, the port must be enabled.
In Balance or MAX device, the port must be LAN port.

For modular devices like the EPX
If the device only has a single module or a fixed module, only the Port ID is needed.
If the device has more than one module, the portId, moduleType, and moduleId are all required.
To successfully turn off the PoE, the port must be enabled as a WAN or a LAN.

When the device does not support PoE or the port does not support PoE, then the API will return as fail.

Avaliable in 8.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
port	Number Object Array	<integer> <Port_Obj> list of {<integer>, <Port_Obj>}	require	This field support a single port or multiple port. User can port provide a single port ID, or array of port ID. Provide a <Port_Obj>, or array of <Port_Obj>

<Port_Obj>

	Type	*Notation*	*Mandatory*	*Description*
id	Number	<integer>	require	Port ID
moduleType	String	<string>	optional	Module Type of the slot NOTE: This parameter is mandatory for modulized device.
moduleId	Number	<integer>	optional	Module ID of the slot NOTE: This parameter is mandatory for modulized device.

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"port":[2,
{"id":1,"moduleType":"E8","moduleId":2}]}' http://192.168.1.1/api/cmd.port.poe.disable
```

```
{
    "stat": "ok"
}
```

POST /api/cmd.port.poe.enable

API

Enable the PoE of the port.

For Switch and Balance/MAX device:
Only Port ID is needed. To success turn on the PoE, the port must be enabled.
In Balance or MAX device, the port must be LAN port.

For modular devices like the EPX
If the device only has a single module or a fixed module, only the Port ID is needed.
If the device has more than one module, the portId, moduleType, and moduleId are all required.
To successfully turn on the PoE, the port must be enabled as a WAN or a LAN.

When the device does not support PoE or the port does not support PoE, then the API will return as fail.

Avaliable in 8.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
port	Number Object Array	<integer> <Port_Obj> list of {<integer>, <Port_Obj>}	require	This field support a single port or multiple port. User can port provide a single port ID, or array of port ID. Provide a <Port_Obj>, or array of <Port_Obj>

<Port_Obj>

	Type	*Notation*	*Mandatory*	*Description*
id	Number	<integer>	require	Port ID
moduleType	String	<string>	optional	Module Type of the slot NOTE: This parameter is mandatory for modulized device.
moduleId	Number	<integer>	optional	Module ID of the slot NOTE: This parameter is mandatory for modulized device.

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"port":[2,
{"id":1,"moduleType":"E8","moduleId":2}]}' http://192.168.1.1/api/cmd.port.poe.enable
```

```
{
    "stat": "ok"
}
```

POST /api/cmd.sendUssd

API

Send USSD to the target address, if there is any SIM card supported.

Avaliable in 8.1.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	Specify which WAN connection ID sends USSD
simId	Number	<sim_id>	optional	Specify which SIM ID sends USSD. If the information is absent, the call will choose the active SIM
ussd	String	{1234567890*#}	require	USSD code

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"connId":2,"ussd":"*109#"}'
http://192.168.1.1/api/cmd.sendUssd
```

```
{
    "stat": "ok",
    "response": {
        "message": "Request is sent successfully"
    }
}
```

GET /api/cmd.sms.get

API

Fetch the active SIM SMS according to connId.

Avaliable in 8.1.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	Get the SMS according to WAN connection ID

Return Parameters

Return JSON

	Type	*Notation*	*Description*
connId	Number	<conn_id>	Connection ID
simId	Number	{ 1, 2 }	SIM ID of the SMS message
imsi	String	<string>	International Mobile Subscriber Identity (IMSI) For fw8.2.0 or late
iccid	String	<string>	Integrate circuit card identity (ICCID). For fw8.2.0 or late
mtn	String	<string>	Mobile Telecommunications Network (MTN) For fw8.2.0 or late
sms	Array	list of <SMS_Obj>	List of SMS message

<SMS_Obj>

	Type	*Notation*	*Description*
sender	String	<string>	Sender of the SMS
message	Array	list of <Message_Obj>	The list of the message

<Message_Obj>

	Type	*Notation*	*Description*
id	Number	<integer>	The ID of the SMS
date	String	<string>	Date of the SMS
timestamp	Number	<timestamp>	Timestamp of the SMS
length	Number	<integer>	The lenght of the SMS message content
content	String	<string>	SMS content

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/cmd.sms.get?connId=6
```


```
{
    "stat": "ok",
    "response": {
        "connId": 6,
        "simId": 1,
        "sms": [
            {
                "sender": "988",
                "message": [
                    {
                        "id": 1,
                        "date": "Feb 17 13:55",
                        "timestamp": 1581774925,
                        "length": "50",
                        "message": "The is the 1st line SMS,\nand this is the 2nd line."
                    }
                ]
            },
            {
                "sender": "+81325359875",
                "message": [
                    {
                        "id": 2,
                        "date": "Feb 05 01:55",
                        "timestamp": 1580867113,
                        "length": "24",
```

```
                    "message": "Multipart message part 1"
                },
                {
                    "id": 6,
                    "date": "Feb 05 01:55",
                    "timestamp": 1580867113,
                    "length": "24",
                    "message": "Multipart message part 2"
                }
            ]
        }
    ]
  }
}
```


POST /api/cmd.sms.sendMessage

API

Send SMS message to the target address, if there is any SIM card supported.

Avaliable in 8.0.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	optional	Specify which WAN connection ID sends the SMS message
address	String	<string>	require	Target address of the SMS message, the address must begin with '+' and follow with 2 to 15 digits. and the first digit cannot be '0'
content	String	<string>	optional	Content of the SMS message

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d
'{"address":"+85235984335","content":"SMS Content"}' http://192.168.1.1/api/cmd.sms.sendMessage
```


```
{
    "stat": "ok"
}
```


POST /api/cmd.starlink

API alpha

Controls Starlink WAN to Reboot/Stow/Unstow Dish(Care Plan required).

Avaliable in 8.5.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	WAN connection ID
action	String	<string>	require	{reboot, stow, unstow}

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"connId":1,"action":"unstow"}'
http://192.168.1.1/api/cmd.starlink
```


```
{
    "stat": "ok"
}
```

GET /api/cmd.ap

API alpha

Returns the status of the device Access Point

Avaliable in 7.0.2 or later

Return Parameters

Return JSON

	Type	*Notation*	*Description*
support	Boolean	<boolean>	Indicates the support of Access Point. Products without Access Point will return false, and provides no further information.
enable	Boolean	<boolean>	Indicates if Access point is currently turned on
wanDependent	Boolean	<boolean>	[Experimental] Returns true when the engineering setting "Turn off AP when there is n Internet connectivity" is currently enabled. (This value is not officially supported and is subject to change in future

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/cmd.ap
```

```
{
    "stat": "ok",
    "response": {
        "support": true,
        "enable": true,
        "wanDependent": true
    }
}
```

POST /api/cmd.ap

API alpha

Switch on or shut down the device Access Point.

Avaliable in 7.0.2 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
enable	Boolean	<boolean>	require	true to Switch on the device Access Point; otherwise to turn off the Access Point.

Return Parameters

Return JSON

	Type	*Notation*	*Description*
support	Boolean	<boolean>	Indicates the support of Access Point. Products without Access Point will return false, and provides no further information.
enable	Boolean	<boolean>	Indicates if Access point is currently turned on
wanDependent	Boolean	<boolean>	[Experimental] Returns true when the engineering setting "Turn off AP when there is n Internet connectivity" is currently enabled. (This value is not officially supported and is subject to change in future

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"enable":true}'
http://192.168.1.1/api/cmd.ap
```

```
{
    "stat": "ok",
    "response": {
        "support": true,
        "enable": true,
        "wanDependent": true
    }
}
```

POST /api/cmd.cellularModule.rescanNetwork

API

Rescan the network of the corresponding WAN connection

Avaliable in 8.0.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	WAN connection ID of the cellular module to rescan

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"connId":"4"}'
http://192.168.1.1/api/cmd.cellularModule.rescanNetwork
```

```
{
    "stat": "ok"
}
```

POST /api/cmd.cellularModule.reset

API

Reset the cellular module of the corresponding WAN connection

Avaliable in 8.0.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	WAN connection ID of the cellular module to reset

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"connId":"4"}'
http://192.168.1.1/api/cmd.cellularModule.reset
```

```
{
    "stat": "ok"
}
```

POST /api/cmd.system.reboot

API

Reboot Device

Reboot the device.
API allows specifying which firmware will be loaded after reboot when providing firmware ID.

To lookup the available firmware ID, please call /api/info.firmware.

Avaliable in 8.2.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
firmwareId	Number	[1, 2]	optional	

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST
http://192.168.1.1/api/cmd.system.reboot
```

```
{
    "stat": "ok"
}
```

POST /api/cmd.wan.cellular

API

Change the SIM priority
The items in the simPriority will be enabled and act like the order in the array.
The cellular scheme type will change to custom automatically.

Avaliable in 8.3.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	WAN Connection of the cellular module
simPriority	Array	list of {1, 2, remoteSim speedfusionConnect5gLte}	require	The priority of the SIM

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"connId":4,"simPriority":
["1","remoteSim"]}' http://192.168.1.1/api/cmd.wan.cellular
```

```
{
    "stat": "ok"
}
```

POST /api/cmd.wifi.connect

API

Connect the Wi-Fi with provide SSID if profile is defined.
If the SSID profile is not defined, connection will require additional information.
WEP or WPA-PSK connection require 'key'
WPA-EAP and 802.1x connection require the Extensible Authentication Protocol(EAP) related information.
When credential cannot be obtained from existing SSID profile, nor supplied parameters, connection cannot be done.

Avaliable in 7.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	Wi-Fi with the WAN connection ID to be used.
ssid	String	<string>	require	SSID to be connected
security	Object	<Security_Obj>	require	Security information to connect the SSID Start support in fw8.2.0
securityPolicy	String	{ open, wep, wpa-eap, wpa-	require	Security Policy to connect the SSID Deprecated in fw8.2.0, please use <Security_Obj>.

	Type	Notation	Mandatory	Description
		psk, 8021x }		
key	String	<string>	optional	Key for WEP and WAP-PSK security policy Deprecated in fw8.2.0, please use <Security_Obj>.
preferredBssid	String	<mac>	optional	Preferred BSSID of the Wi-Fi connection

<Security_Obj>

	Type	Notation	Mandatory	Description
policy	Object	<Security_Policy_Obj>	require	Security policy to connect the SSID

<Security_Policy_Obj>

	Type	Notation	Mandatory	Description
type	String	{ WPA3-Personal, WPA2/WPA3-Personal, 802.1x with dynamic WEP key, WPA/WPA2-Personal, WPA/WPA2-Enterprise, Open, WEP }	require	Security policy to connect the SSID
detail	Object	<Security_Policy_Detail_Obj>	require	Security policy detail to connect the SSID

<Security_Policy_Detail_Obj>

	Type	Notation	Mandatory	Description
key	String	<string>	optional	PSK (Pre-shared Key)The field only for policy type is WPA3-Personal, WPA2/WPA3-Personal, WPA/WPA2-Personal or WEP.

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"connId":1,"ssid":"Main SSID"}' http://192.168.1.1/api/cmd.wifi.connect
```

```
{
    "stat": "ok"
}
```

POST /api/cmd.wifi.disconnect

API

Disconnect the Wi-Fi if it is connected

Avaliable in 7.1.1 or later

Input Parameters

	Type	Notation	Mandatory	Description
connId	Number	<conn_id>	require	Wi-Fi with the WAN connection ID to be used.
ssid	String	<string>	optional	SSID to be disconnected. When omitted, the current connected SSID will be disconnected.

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"connId":1,"ssid":"Main SSID"}' http://192.168.1.1/api/cmd.wifi.disconnect
```

```
{
    "stat": "ok"
}
```

POST /api/cmd.wifi.forget

API

Remove existing SSID profile, if any, by giving the SSID and Authentication method.
Wi-Fi will also disconnect if it is using this SSID.

Avaliable in 7.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	Wi-Fi with the WAN connection ID to be used.
ssid	String	<string>	require	SSID to be forgotten
securityPolicy	String	{ open, wep, wpa-eap, wpa-psk, 8021x }	require	Security Policy of the SSID

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"connId":1,"ssid":"Main SSID","securityPolicy":"wpa-psk"}' http://192.168.1.1/api/cmd.wifi.forget
```

```
{
    "stat": "ok"
}
```

GET /api/cmd.wifi.result

API

Obtain the last known result of Wi-Fi WAN Connection

Avaliable in 7.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	Wi-Fi with the WAN connection ID to be used.

Return Parameters

Return JSON

	Type	*Notation*	*Description*
timestamp	Number	<timestamp>	Timestamp of the last know result
result	String	{ CONNECTED, TIMEOUT, PSK_AUTH_FAIL, EAP_AUTH_FAIL, AP_NOT_FOUND, UNKNOWN_FAIL }	CONNECTED - Wi-Fi is success connected TIMEOUT - Wi-Fi connect timeout AP_NOT_FOUND - Cannot found the AP PSK_AUTH_FAIL - Wi-Fi connect fail and the reason is PSK not match EAP_AUTH_FAIL - Wi-Fi connect fail and the reason is username and password of EAP not match UNKNOWN_FAIL - Wi-Fi connect fail but the error cannot be classified
bssid	String	<mac>	BSSID of the connected AP
ssid	String	<string>	SSID of the connected AP
securityPolicy	String	{ open, wep, wpa-eap, wpa-psk, 8021x }	Security Policy of the connected AP
message	String	<string>	Additional information of the status

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/cmd.wifi.result?connId=1
```

```
{
    "stat": "ok",
    "response": {
        "result": "CONNECTED",
```

```
        "timestamp": 1529899328,
        "ssid": "Main SSID",
        "bssid": "A2:E5:B8:55:89:DF",
        "securityPolicy": "wpa-psk",
        "message": "connected to Main SSID (A2:E5:B8:55:89:DF)"
    }
}
```

GET /api/cmd.wifi.scan

API

Discover nearby Wi-Fi access points

Avaliable in 7.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	Wi-Fi with the WAN connection ID to be used.
infoType	String	{ status, config }	optional	Additional information can be requested along with discovered Wi-Fi access point. config - indicates if the connect profile is present status - indicates if the SSID is connected, or with connect profile
sortBy	String	{ name, security, signal, channel }	optional	Sort by name, security method, signal or channel. When omitted, it will sort by name and the defined SSID will be on the head of the array
sortOrder	String	{ asc, desc }	optional	Sort with descending or ascending order

Return Parameters

Return JSON

	Type	*Notation*	*Description*
-	Array	list of <Wifi_Obj>	List of discovered Wi-Fi Access Points

<Wifi_Obj>

	Type	*Notation*	*Description*
ssid	String	<string>	Service Set Identifier (SSID)
bssid	String	<mac>	Basic Service Set Identifier (BSSID)
signal	Number	<Number>	Signal in dBm Deprecated in firmware 8.1.0
signalStrength	Number	<Number>	Signal in dBm Introduced in firmware 8.1.0
signalLevel	Number	[0, 5]	Signal level Introduced in firmware 8.1.0
channel	Number	<Number>	Channel
securityPolicy	String	{ open, wep, wpa-eap, wpa-psk, 8021x }	Security Policy
status	Object	<Status_Obj>	Status information
config	Object	<Config_Obj>	Config information

<Status_Obj>

	Type	*Notation*	*Description*
inUse	Boolean	<boolean>	SSID profile is targeted as connection.
connected	Boolean	<boolean>	Wi-Fi is currently connected to this SSID.

<Config_Obj>

	Type	*Notation*	*Description*
profileId	Number	<integer>	ID of the connect profile for this SSID.
automatic	Boolean	<boolean>	Indicates if Wi-Fi is configured to connect this SSID automatically.

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/cmd.wifi.scan?connId=1&infoType=status
```


```
{
    "stat": "ok",
    "response": [
        {
            "ssid": "Main SSID",
            "bssid": "A2:E5:B8:55:89:DF",
            "signal": -68,
            "channel": 10,
            "securityPolicy": "wpa-psk",
            "status": {
                "inUse": true,
                "connected": true
            }
        }
    ]
}
```


POST /api/config.gpio

API beta

Obtain and updated the GPIO
The API will return the updated config as return.

If the passing a empty 'list', it will return the current config,
no update will be made

Avaliable in 8.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
list	Array	list of <GPIO_Obj>	optional	List of GPIO config for updating
reference	Boolean	<boolean>	optional	GPIO reference or not

<GPIO_Obj>

	Type	*Notation*	*Mandatory*	*Description*
id	Number	<integer>	require	
enable	Boolean	<boolean>	optional	GPIO enable
name	String	<String>	optional	GPIO custom name
type	String	{ digital_input, digital_output, analog_input }	optional	GPIO type
mode	String	{ input_sensing, ignition_sensing } { wan_status } { input_sensing, voltage_measurement, analog_testing }	optional	For type=digital_input, { input_sensing, ignition_sensing } For type=digital_output, { wan_status } For type=analog_input, { input_sensing, voltage_measurement analog_testing }
customStates	Object	<Custom_Input_State_Obj>	optional	Custom input state
delay	Number	[1, 3600]	optional	GPIO delay ONLY for input type

<Custom_Input_State_Obj>

	Type	*Notation*	*Mandatory*	*Description*
high	String	<String>	optional	Custom input state (High)
low	String	<String>	optional	Custom input state (Low)

Return Parameters

Return JSON

	Type	*Notation*	*Description*
<gpio_id>	Object	<GPIO_Obj>	GPIO information for the <gpio_id>
order	Array	list of <gpio_id>	The order of the ids
reference	Object	<GPIO_Ref_Map_Obj>	Provide the support type and mode for each <gpio_id>

<GPIO_Obj>

	Type	*Notation*	*Description*
enable	Boolean	<boolean>	GPIO enable
name	String	<String>	GPIO custom name
type	String	{ digital_input, digital_output, analog_input }	GPIO type
mode	String	{ input_sensing, ignition_sensing } { wan_status } { input_sensing, voltage_measurement, analog_testing }	For type=digital_input, { input_sensing, ignition_sensing } For type=digital_output, { wan_status } For type=analog_input, { input_sensing, voltage_measurement, analog_testing }
customStates	Object	<Custom_Input_State_Obj>	Custom input state
delay	Number	[1, 3600]	GPIO delay ONLY for input type

<Custom_Input_State_Obj>

	Type	*Notation*	*Description*
high	String	<String>	Custom input state (High)
low	String	<String>	Custom input state (Low)

<GPIO_Ref_Map_Obj>

	Type	*Notation*	*Description*
<gpio_id>	Object	<GPIO_Ref_Obj>	GPIO reference for the <gpio_id>
order	Array	list of <gpio_id>	The order of the ids

<GPIO_Ref_Obj>

	Type	*Notation*	*Description*
name	String	<string>	GPIO name
type	Array	list of { digital_input, digital_output, analog_input }	Which GPIO type support for the <gpio_id>
mode	Object	<GPIO_Ref_Mode_Obj>	Which GPIO mode is support for specific GPIO type

<GPIO_Ref_Mode_Obj>

	Type	*Notation*	*Description*
digital_input	Array	list of { input_sensing, ignition_sensing }	Support mode for digital_input type
digital_output	Array	list of { wan_status }	Support mode for digital_output type
analog_input	Array	list of { input_sensing, voltage_measurement,	Support mode for analog_input type

<GPIO_Ref_Mode_Obj>

	Type	*Notation*	*Description*
		analog_testing }	

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"list":
[{"id":1,"enable":true,"type":"digital_output","mode":"toggle_high"},
{"id":2,"enable":true,"name":"PowerInput","type":"digital_input","mode":"input_sensing","customState
":{"high":"Battery Backup","low":"Grid"},"delay":3}]}' http://192.168.1.1/api/config.gpio
```


```
{
    "stat": "ok",
    "response": {
        "1": {
            "enable": true,
            "type": "digital_output",
            "mode": "toggle_high"
        },
        "2": {
            "enable": true,
            "name": "PowerInput",
            "type": "digital_input",
            "mode": "input_sensing",
            "customStates": {
                "high": "Battery Backup",
                "low": "Grid"
            },
            "delay": 3
        },
        "order": [
            1,
            2
        ]
    }
}
```


GET /api/config.mesh

API

Obtain the wireless mesh config

Avaliable in 8.3.2 or later

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <profile_id>	The order of the SSID Profile ID
<profile_id>	Object	<Mesh_Profile_Obj>	SSID Profile information

<Mesh_Profile_Obj>

	Type	*Notation*	*Description*
enable	Boolean	<boolean>	Enable the profile
profileId	Array	list of <integer>	Which AP profile is using this mesh network
meshId	String	<string>	Mesh ID
passphrase	String	<string>	Passphrase of the mesh profile, the value must be at least 8 characters long
frequencyBand	String	{2.4 GHz, 5 GHz}	2.4 GHz or 5 GHz frequency band

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/config.mesh
```

```
{
    "stat": "ok",
    "response": {
        "1": {
            "enable": true,
            "meshId": "522",
            "passphrase": "222246645646",
            "frequencyBand": "2.4 GHz"
        },
        "order": [
            1
        ]
    }
}
```

POST /api/config.mesh

API

Add the Mesh profile

Add the Mesh profile according to the given information.

Avaliable in 8.3.2 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
action	String	{ add }	require	State the add action
list	Array	List of <Mesh_Profile_Obj>	require	List of profiles

<Mesh_Profile_Obj>

	Type	*Notation*	*Mandatory*	*Description*
id	Number	<integer>	require	Profile ID which is wanted to update
meshId	String	<string>	require	Mesh ID of the profile
profileId	Boolean NULL	list of <integer> NULL	optional	Assign this mesh network to the AP profile by the AP profile ID If the value is NULL or absent, no profile will be assigned.
passphrase	String	<string>	require	Passphrase of the mesh profile, the value must be at least 8 characters long
frequencyBand	String	{2.4 GHz, 5 GHz}	require	Choose 2.4 GHz or 5 GHz for frequency band

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <profile_id>	The order of the SSID Profile ID
<profile_id>	Object	<Mesh_Profile_Obj>	SSID Profile information

<Mesh_Profile_Obj>

	Type	*Notation*	*Description*
enable	Boolean	<boolean>	Enable the profile
profileId	Array	list of <integer>	Which AP profile is using this mesh network
meshId	String	<string>	Mesh ID
passphrase	String	<string>	Passphrase of the mesh profile, the value must be at least 8 characters

<Mesh_Profile_Obj>

	Type	*Notation*	*Description*
			long
frequencyBand	String	{2.4 GHz, 5 GHz}	2.4 GHz or 5 GHz frequency band

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"action":"update","list":
[{"id":1,"meshId":"mesh_id","enable":true,"passphrase":"passphraseString","frequencyBand":"5 GHz"}]
http://192.168.1.1/api/config.mesh
```

```
{
    "stat": "ok",
    "response": {
        "1": {
            "enable": true,
            "meshId": "mesh_id",
            "passphrase": "passphraseString",
            "frequencyBand": "5 GHz"
        },
        "order": [
            1
        ]
    }
}
```

Update the Mesh profile

Update the Mesh profile according to the given information.
Only given information will be affected.

Avaliable in 8.3.2 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
action	String	{ update }	require	State the update action
list	Array	List of <Mesh_Profile_Obj>	require	List of profiles

<Mesh_Profile_Obj>

	Type	*Notation*	*Mandatory*	*Description*
id	Number	<integer>	require	Profile ID which is wanted to update
meshId	String	<string>	optional	Mesh ID of the profile
profileId	Boolean NULL	list of <integer> NULL	optional	Assign this mesh network to the AP profile by the AP profile ID If the value is NULL, no profile will be assigned. If the value is absent, this field will not be updated.
passphrase	String	<string>	optional	Passphrase of the mesh profile, the value must be at least 8 characters long
frequencyBand	String	{2.4 GHz, 5 GHz}	optional	Choose 2.4 GHz or 5 GHz for frequency band

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <profile_id>	The order of the SSID Profile ID
<profile_id>	Object	<Mesh_Profile_Obj>	SSID Profile information

<Mesh_Profile_Obj>

	Type	*Notation*	*Description*
enable	Boolean	<boolean>	Enable the profile

<Mesh_Profile_Obj>

	Type	*Notation*	*Description*
profileId	Array	list of <integer>	Which AP profile is using this mesh network
meshId	String	<string>	Mesh ID
passphrase	String	<string>	Passphrase of the mesh profile, the value must be at least 8 characters long
frequencyBand	String	{2.4 GHz, 5 GHz}	2.4 GHz or 5 GHz frequency band

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"action":"update","list":
[{"id":1,"frequencyBand":"2.4 GHz"}]}' http://192.168.1.1/api/config.mesh
```


```
{
    "stat": "ok",
    "response": {
        "1": {
            "enable": true,
            "meshId": "mesh_id",
            "passphrase": "passphraseString",
            "frequencyBand": "2.4 GHz"
        },
        "order": [
            1
        ]
    }
}
```


Remove the Mesh profile

Remove the Mesh profile according to the given information

Avaliable in 8.3.2 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
action	String	{ remove }	require	State the remove action
list	Array	List of <ID_Obj>	require	List of profiles

<ID_Obj>

	Type	*Notation*	*Mandatory*	*Description*
id	Number	<integer>	require	Profile ID which is wanted to remove

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <profile_id>	The order of the SSID Profile ID
<profile_id>	Object	<Mesh_Profile_Obj>	SSID Profile information

<Mesh_Profile_Obj>

	Type	*Notation*	*Description*
enable	Boolean	<boolean>	Enable the profile
profileId	Array	list of <integer>	Which AP profile is using this mesh network
meshId	String	<string>	Mesh ID
passphrase	String	<string>	Passphrase of the mesh profile, the value must be at least 8 characters long
frequencyBand	String	{2.4 GHz, 5 GHz}	2.4 GHz or 5 GHz frequency band

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"action":"remove","list":
[{"id":1}]}' http://192.168.1.1/api/config.mesh
```


```
{
    "stat": "ok",
    "response": {
        "2": {
            "enable": true,
            "meshId": "mesh_id_2",
            "passphrase": "passphraseString",
            "frequencyBand": "2.4 GHz"
        },
        "order": [
            2
        ]
    }
}
```


GET /api/config.speedfusionConnectProtect

API SpeedFusion Connect Protect

Obtain the SpeedFusion Connect Protect config of Client(in term of MAC address) and Access Point(in term of SSID)

Avaliable in 8.3.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
infoType	Array	list of {client, accessPoint}	optional	Filter of the return object

Return Parameters

Return JSON

	Type	*Notation*	*Description*
client	Array	list of <SFC_Client_Obj>	The client connected to SpeedFusion Connect Protect
accessPoint	Array	list of <SFC_SSID_Profile_Obj>	SSID Profile information

<SFC_Client_Obj>

	Type	*Notation*	*Description*
mac	String	<mac>	MAC address which will use SpeedFusion Connect Protect
cityCode	String	<string>	SpeedFusion Clould location city code, it will be empty string when tha is auto

<SFC_SSID_Profile_Obj>

	Type	*Notation*	*Description*
ssid	String	<string>	The SSID which is connected to the Cloud Location
referenceSsid	String	<uuid>	Name of the reference SSID profile
cityCode	String	<string>	SpeedFusion Clould location city code

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/config.speedfusionConnectProtect
```


```
{
    "stat": "ok",
    "response": {
        "client": [
```

```
            {
                "mac": "00:11:6E:44:44:EE",
                "cityCode": "FRA"
            }
        ],
        "accessPoint": [
            {
                "referenceSsid": "Balance_SSID_1",
                "cityCode": "FRA",
                "ssid": "Balance_SSID_1_Germany_FRA"
            },
            {
                "referenceSsid": "Balance_SSID",
                "cityCode": "FRA",
                "ssid": "Balance_SSID_Germany_FRA"
            }
        ]
    }
}
```

POST /api/config.speedfusionConnectProtect

API SpeedFusion Connect Protect

Update client / SSID connection to SpeedFusion Connect Protect

Avaliable in 8.3.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
action	String	{ update, remove, replace }	require	Action of the list update - When mac/ssid is match, the profile will be updated. When it is not found, a new profile will be added remove - Remove the item by mac/ssid replace - When client/accessPoint array is provided, whole client/accessPoint array will be removed and add all new items from the provided array
client	Array	list of <SFC_Client_Obj>	optional	The list for update, remove or replace the client
accessPoint	Array	list of <SFC_SSID_Obj>	optional	The list for update, remove or replace the access point

<SFC_Client_Obj>

	Type	*Notation*	*Mandatory*	*Description*
mac	String	<mac>	optional	MAC address
cityCode	String	<string>	optional	Cloud location Need not provide this field when action=remove

<SFC_SSID_Obj>

	Type	*Notation*	*Mandatory*	*Description*
ssid	String	<string>	optional	The SSID which is connected to the Cloud Location
referenceSsid	String	<uuid>	optional	Name of base SSID profile Need not provide this field when action=remove
cityCode	String	<string>	optional	Cloud location Need not provide this field when action=remove

Return Parameters

Return JSON

	Type	*Notation*	*Description*
client	Array	list of <SFC_Client_Obj>	The client connected to SpeedFusion Connect Protect
accessPoint	Array	list of <SFC_SSID_Profile_Obj>	SSID Profile information

<SFC_Client_Obj>

	Type	*Notation*	*Description*
mac	String	<mac>	MAC address
cityCode	String	<string>	SpeedFusion Clould location city code

<SFC_SSID_Profile_Obj>

	Type	*Notation*	*Description*
ssid	String	<string>	The SSID which is connected to the Cloud Location
referenceSsid	String	<uuid>	Name of the base SSID profile
cityCode	String	<string>	SpeedFusion Clould location city code

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"action":"replace","client":
[{"mac":"DD:11:6E:44:44:EE","cityCode":"FRA"},
{"mac":"CC:11:6E:44:44:EE","cityCode":"FRA"}],"accessPoint":
[{"referenceSsid":"Balance_SSID","cityCode":"FRA","ssid":"Balance_SSID_Germany_FRA"}]}'
http://192.168.1.1/api/config.speedfusionConnectProtect
```


```
{
    "stat": "ok",
    "response": {
        "client": [
            {
                "mac": "DD:11:6E:44:44:EE",
                "cloudLocation": "FRA"
            },
            {
                "mac": "CC:11:6E:44:44:EE",
                "cloudLocation": "FRA"
            }
        ],
        "accessPoint": [
            {
                "referenceSsid": "Balance_SSID",
                "cityCode": "FRA",
                "ssid": "Balance_SSID_Germany_FRA"
            }
        ]
    }
}
```


GET /api/config.ssid.profile

API

Obtain the SSID profile information

Avaliable in 7.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
id	Array	list of <ssid_profile_id>	optional	list the SSID Profile base on id, multiple values are accepted, When omitted, all profile will be return.

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <profile_id>	The order of the SSID Profile ID
<profile_id>	Object	<SSID_Profile_Obj>	SSID Profile information

<SSID_Profile_Obj>

	Type	*Notation*	*Description*
name	String	<string>	SSID of the profile
enable	Boolean	<boolean>	Enable the profile as the Local AP. Always false when the device is not support Wi-Fi.
frequency	Array	list of {2.4GHz, 5GHz}	The frequency of the SSID. This field will not appear when disable. Only for Wi-Fi supported device
vlanId	Number	<integer>	VLAN ID of the profile, the field will not appear if use the LAN
captivePortal	Boolean	<boolean>	Profile will use captive portal or not
incontrolManaged	Boolean	<boolean>	InControl is managed this profile or not
broadcast	Boolean	<boolean>	Broadcast the SSID or not
security	Object	<SSID_Security_Obj>	The security policy and related information

<SSID_Security_Obj>

	Type	*Notation*	*Description*
policy	String	{ WPA2 Personal, WPA/WPA2 Personal }	Security policy of the SSID proifle
wpa2Personal	Object	<WPA2_Personal_Obj>	WPA2 Personal related information
wpaWpa2Personal	Object	<WPA2_Personal_Obj>	WPA/WPA2 Personal related information
ppsk	Array	list of <PPSK_Obj>	Private Pre-Shared Key security config information

<WPA2_Personal_Obj>

	Type	*Notation*	*Description*
fastTransition	Boolean	<boolean>	Fast Transition for WPA2, this field will not appear in WPA/WPA2 Personal This config does not take effect in 7.1.1 with WPA2 Enterprise
key	String	<string>	Key for WPA2 Personal and WPA/WPA2 Personal

<PPSK_Obj>

	Type	*Notation*	*Description*
name	String	<string>	A reference name for this <PPSK_Obj> object
key	String	<string>	WPA/WPA2/WPA3 Personal Key in plaintext format
mac	String	<mac>	The MAC address this <PPSK_Obj> object will only be applied to
vlanId	Number	<integer>	The VLAN ID of the VLAN network this <PPSK_Obj> will be connectec to, or the Untagged LAN will be connected if not specified

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/config.ssid.profile?id=1 2
```


```
{
    "stat": "ok",
    "response": {
        "1": {
            "name": "Main SSID",
            "enable": true,
            "captivePortal": true,
            "incontrolManaged": false,
            "broadcast": true,
```

```
            "security": {
                "policy": "WPA2 Personal",
                "wpa2Personal": {
                    "fastTransition": true,
                    "key": "pas53or2"
                }
            }
        },
        "2": {
            "name": "Guest SSID",
            "enable": true,
            "captivePortal": true,
            "incontrolManaged": false,
            "broadcast": true,
            "vlanId": 1,
            "security": {
                "policy": "WPA2 Personal",
                "wpa2Personal": {
                    "fastTransition": false,
                    "key": "pass3ord"
                }
            }
        },
        "order": [
            1,
            2
        ]
    }
}
```

POST /api/config.ssid.profile

API

Update the SSID profile

Update the SSID profile according to the given information.
Only given information will be affected.

Avaliable in 7.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
action	String	{ update }	require	State the update action
id	Number	<integer>	require	Profile ID which is wanted to update
name	String	<string>	optional	SSID of the profile
enable	Boolean	<boolean>	optional	Enable the profile as the Local AP, only support when the device is supporting Wi-Fi.
frequency	Array	list of {2.4GHz, 5GHz}	optional	Choose the frequency when the SSID is enable as Local AP. If the SSID is enabled and this field is absent, both 2.4GHz and 5GHz will be used.Only for Wi-Fi supported device
vlanId	Number	<integer>	optional	VLAN ID of the profile, the field will not appear if use the LAN
broadcast	Boolean	<boolean>	optional	Broadcast the profile or not
security	Object	<SSID_Security_Obj>	optional	Security information

<SSID_Security_Obj>

	Type	*Notation*	*Mandatory*	*Description*
policy	String	{ "WPA2 Personal", "WPA/WPA2 Personal" }	optional	Security Policy of the SSID profile
wpa2Personal	Object	<WPA2_Personal_Obj>	optional	WPA2 Personal related information

<SSID_Security_Obj>

	Type	*Notation*	*Mandatory*	*Description*
wpaWpa2Personal	Object	<WPA2_Personal_Obj>	optional	WPA/WPA2 Personal related information
ppsk	Array	list of <PPSK_Obj>	optional	Private Pre-Shared Key security config information

<WPA2_Personal_Obj>

	Type	*Notation*	*Mandatory*	*Description*
fastTransition	Boolean	<boolean>	optional	Fast Transition for WPA2, this field cannot be set in WPA/WPA2 Personal This config does not take effect in 7.1.1 with WPA2 Enterprise
key	String	<string>	optional	Key for WPA2 Personal or WPA/WPA2 Personal The length must between 8 and 63 or HEX in 64

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <profile_id>	The order of the SSID Profile ID
<profile_id>	Object	<SSID_Profile_Obj>	SSID Profile information

<SSID_Profile_Obj>

	Type	*Notation*	*Description*
name	String	<string>	SSID of the profile
enable	Boolean	<boolean>	Enable the profile as the Local AP. Always false when the device is not support Wi-Fi.
frequency	Array	list of {2.4GHz, 5GHz}	The frequency of the SSID. This field will not appear when disable. Only for Wi-Fi supported device
vlanId	Number	<integer>	VLAN ID of the profile, the field will not appear if use the LAN
captivePortal	Boolean	<boolean>	Profile will use captive portal or not
incontrolManaged	Boolean	<boolean>	InControl is managed this profile or not
broadcast	Boolean	<boolean>	Broadcast the SSID or not
security	Object	<SSID_Security_Obj>	The security policy and related information

<SSID_Security_Obj>

	Type	*Notation*	*Description*
policy	String	{ "WPA2 Personal", "WPA/WPA2 Personal" }	Security policy of the SSID proifle
wpa2Personal	Object	<WPA2_Personal_Obj>	WPA2 Personal related information
wpaWpa2Personal	Object	<WPA2_Personal_Obj>	WPA/WPA2 Personal related information

<WPA2_Personal_Obj>

	Type	*Notation*	*Description*
fastTransition	Boolean	<boolean>	Fast Transition for WPA2, this field will not appear in WPA/WPA2 Personal This config does not take effect in 7.1.1 with WPA2 Enterprise
key	String	<string>	Key for WPA2 Personal and WPA/WPA2 Personal

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d
'{"action":"update","id":"1","enable":true,"frequency":["2.4GHz","5GHz"],"security":{"wpa2Personal"
{"key":"thisIsNewPassword"}}}' http://192.168.1.1/api/config.ssid.profile
```

```
{
    "stat": "ok",
```

```
    "response": {
        "1": {
            "name": "Main SSID",
            "enable": true,
            "captivePortal": true,
            "incontrolManaged": false,
            "broadcast": true,
            "security": {
                "policy": "WPA2 Personal",
                "wpa2Personal": {
                    "fastTransition": true,
                    "key": "thisIsNewPassword"
                }
            }
        },
        "order": [
            1
        ]
    }
}
```


GET /api/config.wan.connection

API

Get the config of the WAN settings

Avaliable in 8.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
id	Array	list of <lan_id>	optional	List the WAN config settings base on id, multiple values are accepted List all WAN if absent in id
infoType	String	{ multipleIp, connection, physical, healthcheck, bandwidthAllowanceMonitor, cellular }	optional	Filter of the return object, multiple values are accepted. All types will return if this field is empty

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <conn_id>	The order of WAN ID
<conn_id>	Object	<WAN_Config_Obj>	WAN config information

<WAN_Config_Obj>

	Type	*Notation*	*Description*
name	String	<string>	Connection Name of the WAN
asLan	Boolean	<boolean>	WAN performing WAN as LAN
enable	Boolean	<boolean>	WAN Enable
active	Boolean	<boolean>	WAN Active
multipleIp	Array	list of <ipv4>	Additional IP Address, will not appear if asLan is true
connection	Object	<Connection_Obj>	Connection Settings Object, will not appear if asLan is true
physical	Object	<Physical_Obj>	Physical Interface Settings Object, will not appear if asLan is true
healthcheck	Object	<Healthcheck_Obj>	Healthcheck Settings Object, will not appear if asLan is true
bandwidthAllowanceMonitor	Object	<Bandwidth_Monitor_Obj>	Bandwidth Allowance Monitor Object, will not appear if asLan is true

<Connection_Obj>

	Type	*Notation*	*Description*
method	String	{ DHCP, Static IP, PPPoE, L2TP, GRE, Drop In }	Connection Method of the WAN
mode	String	{ NAT, IP Forwarding }	NAT or IP Forwarding mode
icmpPing	Boolean	<boolean>	Reply ICMP Ping
priority	Number	<integer>	The priority of the WAN in MAX device, or The connection type of the WAN in Balance device
dns	Object	<DNS_Obj>	DNS Object
ddns	Object	<DDNS_Obj>	Dynamic DNS Settings Object
bandwidth	Object	<Max_Speed_Obj>	Bandwidth limit information
schedule	Number	<integer>	Schedule ID, only appear in WAN is scheduled
pepVpnNat	Boolean	<boolean>	Apply NAT on Remote SpeedFusion VPN peers's outgoing Internet traffic This field deprecated in 8.0.0
pepvpnNat	Boolean	<boolean>	Apply NAT on Remote SpeedFusion VPN peers's outgoing Internet traffic
gobi	Object	<WAN_Gobi_Obj>	Gobi information, only appear if WAN is Gobi
modem	Object	<WAN_Modem_Obj>	Modem information, only appear if WAN is Modem
hotStandBy	Boolean	<boolean>	To indicate this WAN connection be tring to get connect, when acting as a standby. Deprecated in fw 9.0.0, please use hotStandby.
hotStandby	Object	<Hot_Standby_Obj>	To indicate this WAN connection be tring to get connect, when acting as a standby.
idleTimeout	Number	<integer>	Modm idle timout in second
dhcp	Object	<DHCP_Obj>	DHCP Object, for method=DHCP
staticIp	Object	<Static_IP_Obj>	Static IP Object, for method=Static IP
pppoe	Object	<PPPoE_Obj>	PPPoE Object, for method=PPPoE
l2tp	Object	<L2TP_Obj>	L2TP Object, for method=L2TP
gre	Object	<GRE_Obj>	GRE Object, for method=GRE
dropIn	Object	<Static_IP_Obj>	DropIn Object, for method=Drop In

<Physical_Obj>

	Type	*Notation*	*Description*
type	String	{ ethernet, wireless, modem, gobi }	Port type of the WAN connection
speed	String	{ Auto, 1000baseTx-FD, 100baseTx-FD, 100baseTx-HD, 10baseT-FD, 10baseT-HD }	Port speed of WAN
advertise	Boolean	<boolean>	Advertise Speed enable Only report the value when the speed is not "Auto"
supportGigaEthernet	Boolean	<boolean>	Support Giga Ethernet
mtu	Number	[576, 1500]	MTU Value, this field will be absent if MTU is auto
mss	Number	[536, 1460]	MSS Value, this field will be absent if MSS is auto
mac	String	<mac>	MAC address, this field will be absent if MAC address is auto
vlan	Number	<integer>	Only appear if VLAN is endabled

<Healthcheck_Obj>

	Type	*Notation*	*Description*
enable	Boolean	<boolean>	Enable config of the Healthcheck function All other field in this object will be absent if this field is false
method	String	{ http, nslookup, ping, smartcheck }	Healthcheck method

<Healthcheck_Obj>

	Type	*Notation*	*Description*
timeout	Number	{ 1,2,3,4,5,10 }	Timeout in second
interval	Number	{ 5,10,20,30,60,120,1800,3600 }	Interval in second
retry	Number	{ 1,3,5,10,15,20 }	Retries times
recovery	Number	{ 1,3,5,10,15,20 }	Recovery reties times
http	Object	<Healthcheck_HTTP_Obj>	Extra information if method=http
dns	Object	<Healthcheck_DNS_Obj>	Extra information if method=dns
ping	Object	<Healthcheck_Ping_Obj>	Extra information if method=ping

<Bandwidth_Monitor_Obj>

	Type	*Notation*	*Description*
enable	Boolean	<boolean>	Enable config of Bandwidth Allowance Monitor All other field in this object will be absent if this field is false
action	Array	list of { email, disconnnect }	Action will be taken when the allowance is reach email: when the allowance reach 75% / 95% disconnect: when the allowance reach 100%
start	Number	<integer>	Start day of bandwidth allowance monitor start=0 means the alst day of the month
monthlyAllowance	Object	<Monthly_Obj>	Monthly Allowance

<DNS_Obj>

	Type	*Notation*	*Description*
auto	Boolean	<boolean>	Obtain DNS server address automatically
server	Array	list of <ipv4>	Custom DNS server addresses

<DDNS_Obj>

	Type	*Notation*	*Description*
enable	Boolean	<boolean>	Support DDNS IP update service
provider	String	{ changeip, dyndns, noip, dnsomatic, customUrl }	Provider of the dynamic DNS provider
customUrl	String	<domain>	Custom provdier of DDNS service only appear if provider=customUrl
username	String	<string>	Login information of the dynamic DNS provider service
password	String	<string>	Password of the dynamic DNS provider service
host	Array	list of <string>	Host of dynamic DNS Service

<Max_Speed_Obj>

	Type	*Notation*	*Description*
upload	Object	<Bandwidth_Obj>	Upload Limit
download	Object	<Bandwidth_Obj>	Download Limit

<WAN_Gobi_Obj>

	Type	*Notation*	*Description*
mode	String	<string>	Gobi Mode
forceSubnet	Number	<integer>	Force Subnet
operator	Object	<Operator_Obj>	Operator Object

<WAN_Modem_Obj>

	Type	*Notation*	*Description*
mobileType	String	{ 4G, 3G, 2G, 2G_3G, 3G_2G }	Mobile Type
wimaxLogin	Object	<Login_Pair_Obj>	WIMAX information, only appear if the WAN is WIMAX modem
huaweiBand	Array	list of { GSM1900, GSM900/GSM1800/WCDMA210 }	Huawei information, only appear if the WAN is Huawei mmodem
operator	Object	<Operator_Obj>	Operator Object
simPin	String	<string>	SIM Pin

<Hot_Standby_Obj>

	Type	*Notation*	*Description*
enable	Boolean	<boolean>	Enable hot standby
schedule	Number	<integer>	Schedule ID for hot standby This field will be absent if not schedule is set.

<DHCP_Obj>

	Type	*Notation*	*Description*
hostname	String	<string>	Hostname, if hostname does not set or the type of connection method L2TP this field will be absent

<Static_IP_Obj>

	Type	*Notation*	*Description*
ip	String	<ipv4>	IP address
mask	Number	<maskn>	Subnet Mask
gateway	String	<ipv4>	Default gateway
managementNetwork	Object	<Network_Obj>	Management IP address

<PPPoE_Obj>

	Type	*Notation*	*Description*
username	String	<string>	PPPoE Username
password	String	<string>	PPPoE Password
service	String	<string>	Service Name, this field will be absent if empty
ip	String	<ipv4>	IP adderss, this field will be absent if empty
managementNetwork	Object	<Network_Obj>	PPPoE management IP address for access PPPoE modem

<L2TP_Obj>

	Type	*Notation*	*Description*
username	String	<string>	L2TP Username
password	String	<string>	L2TP Password
host	String	<string>	Server IP address / host
staticIp	String	<string>	Static IP, only appear if the IP is set

<GRE_Obj>

	Type	*Notation*	*Description*
host	String	<string>	Remote GRE host
local	String	<string>	Tunnel Local IP address
remote	String	<string>	Tunnel Remote IP address
nat	String	<string>	Outgoing NAT IP address
staticIp	Object	<Static_IP_Obj>	Static IP Object

<GRE_Obj>

	Type	*Notation*	*Description*
			This field will only appear if the WAN type is ethernet only

<Healthcheck_HTTP_Obj>

	Type	*Notation*	*Description*
url	Array	list of <URL_Pattern_Obj>	Healthcheck URL list

<Healthcheck_Ping_Obj>

	Type	*Notation*	*Description*
host	Array	list of <string>	Host for pinging, this field will not appear if the list is empty

<Healthcheck_DNS_Obj>

	Type	*Notation*	*Description*
host	Array	list of <string>	DNS Server for healthcheck, this field will not appear if the list is empty
includePublic	Boolean	<boolean>	Include public DNS server

<Monthly_Obj>

	Type	*Notation*	*Description*
value	Number	<integer>	Monthly allowance value
unit	String	{ MB }	Monthly allowance unit, a constant value "MB"

<Bandwidth_Obj>

	Type	*Notation*	*Description*
value	Number	<integer>	Limiited bandwidth
unit	String	{ kbps }	Units of bandwidth limitation, a constant value "kbps"

<Operator_Obj>

	Type	*Notation*	*Description*
auto	Boolean	<boolean>	Enable auto operator
apn	String	<string>	APN
username	String	<string>	Username
password	String	<string>	Password
dialNumber	String	{0123456789*#}	Dial number

<Login_Pair_Obj>

	Type	*Notation*	*Description*
username	String	<string>	Username
password	String	<string>	Password

<Network_Obj>

	Type	*Notation*	*Description*
ip	String	<ipv4>	IP address
mask	Number	<maskn>	Subnet mask

<URL_Pattern_Obj>

	Type	*Notation*	*Description*
url	String	<string>	Healthcheck URL
pattern	String	<string>	Match string of the url

<URL_Pattern_Obj>

	Type	Notation	Description
			This field will be absent if the match string is empty or disable

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/config.wan.connection?id=1 2&infoType=connection
healthcheck
```


```
{
    "stat": "ok",
    "response": {
        "1": {
            "name": "WAN 1",
            "asLan": false,
            "enable": true,
            "active": true,
            "multipleIp": [],
            "connection": {
                "method": "dhcp",
                "mode": "NAT",
                "icmpPing": true,
                "priority": 1,
                "dns": {
                    "auto": true
                },
                "ddns": {
                    "username": "username",
                    "password": "@~HiDdEn~@",
                    "host": [
                        "kjkjkjkj.com"
                    ],
                    "provider": "noip",
                    "enable": true
                },
                "bandwidth": {
                    "upload": {
                        "value": 100000,
                        "unit": "kbps"
                    },
                    "download": {
                        "value": 100000,
                        "unit": "kbps"
                    }
                },
                "schedule": 4,
                "dhcp": {
                    "hostname": ""
                }
            },
            "healthcheck": {
                "method": "nslookup",
                "timeout": 5,
                "interval": 5,
                "retry": 3,
                "recovery": 3,
                "enable": true,
                "nslookup": {
                    "includePublic": false,
                    "host": [
                        "208.67.222.222",
                        "208.67.220.220"
                    ]
                }
            }
        },
        "2": {
```

```
            "name": "WAN2",
            "asLan": false,
            "enable": true,
            "active": true,
            "multipleIp": [],
            "connection": {
                "method": "dropIn",
                "mode": "IP Forwarding",
                "icmpPing": true,
                "priority": 0,
                "dns": {
                    "auto": false,
                    "host": [
                        "3.3.3.3"
                    ]
                },
                "ddns": {
                    "enable": false
                },
                "bandwidth": {
                    "upload": {
                        "bandwidth": 100000,
                        "unit": "kbps"
                    },
                    "download": {
                        "bandwidth": 100000,
                        "unit": "kbps"
                    }
                },
                "pepVpnNat": true,
                "dropIn": {
                    "ip": "169.254.0.1",
                    "mask": 24,
                    "gateway": "22.2.2.2"
                }
            },
            "healthcheck": {
                "method": "nslookup",
                "timeout": 5,
                "interval": 5,
                "retry": 3,
                "recovery": 3,
                "enable": true,
                "nslookup": {
                    "includePublic": true
                }
            }
        },
        "order": [
            1,
            2
        ]
    }
}
```


POST /api/config.wan.connection

API beta

Update the WAN connection settings, most of the options will update only when the information is provided.

Avaliable in 8.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
action	String	{update}	require	Action of the API, now only support update.
list	Array	list of <WAN_Config_Obj>	require	List of the WAN connection object which is going to update

<WAN_Config_Obj>

	Type	*Notation*	*Mandatory*	*Description*
id	Number	<conn_id>	optional	WAN connection ID
name	String	<string>	optional	WAN connection name
enable	Boolean	<boolean>	optional	Enable the WAN connection
schedule	Number Null	<integer> <null>	optional	Schedule ID for the WAN To disable schedule, give the JSON null
connection	Object	<Connection_Obj>	optional	Connection information
modem	Object	<Modem_Obj>	optional	Modem information Only support when the WAN is modem type
cellular	Object	<Cellular_Obj>	optional	Cellular information Only support when the WAN is cellular
wifi	Object	<Wifi_Obj>	optional	Wi-Fi WAN information Only support when the WAN is Wi-Fi
wifiProfile	-	-	optional	Wi-Fi Profile which is used by the Wi-Fi WAN Only support when the WAN is Wi-Fi NOTE: This field is not confirmed yet If you want to manage the Wi-Fi profile, try the AI below: POST cmd.wifi.connect POST cmd.wifi.disconnect POST cmd.wifi.forget Please make sure the SSID is nearby
physical	Object	<Physical_Obj>	optional	Physical information
healthcheck	Object	<Healthcheck_Obj>	optional	Healthcheck information
bandwidthAllowanceMonitor	Object	<BW_Allowance_Monitor_Obj>	optional	Bandwidth allowance monitor
multipleIp	Array	list of <ipv4>	optional	Additional IP
ddns	Object	<DDNS_Obj>	optional	Dynamic DNS service

<Connection_Obj>

	Type	*Notation*	*Mandatory*	*Description*
cellularModule	Object	<Cellular_Module_Obj>	optional	Cellular Module
routingMode	String	{ IP Forwarding, NAT }	optional	Routing Mode
pepvpnNat	Boolean	<boolean>	optional	SpeedFusion VPN traffic via this WAN connection be in bridge (IP forwarding), with no NAT involved
useLanIp	Boolean	<boolean>	optional	Local out to IP forwarding WAN traffic will SNAT to defaul trunk LAN IP instead of WAN IP
method	Object	<Connection_Method_Obj>	optional	Connection method information This field only for Static IP, DHCP, PPPoE, L2TP, GRE ar OpenVPN
dns	Object	<DNS_Obj>	optional	DNS information
priority	Number	<integer>	optional	WAN Priority
groupSet	Number	<integer>	optional	Group number if support multiple groups of WAN
ignoreDefaultGateway	Boolean	<boolean>	optional	Ignore default gateway
hotStandBy	Boolean	<boolean>	optional	Hot standby state Deprecated in fw 9.0.0, please use hotStandby.
hotStandby	Object	<Hot_Standby_Obj>	optional	Hot standby state
idleTimeout	Number Null	<integer> <null>	optional	Idle timeout To disable idle timeout, give the JSON null
icmpPing	Boolean	<boolean>	optional	ICMP Ping
bandwidth	Object	<Bandwidth_Map_Obj>	optional	Bandwidth information

<Cellular_Module_Obj>

	Type	*Notation*	*Mandatory*	*Description*
networkMode	String	<string>	optional	Network Mode

<Connection_Method_Obj>

	Type	*Notation*	*Mandatory*	*Description*
type	String	{ staticIp, dhcp, pppoe, l2tp, gre, openvpn }	optional	Connetion method type
detail	Object Object Object Object Object Object	<DHCP_Obj> <Static_IP_Obj> <PPPoE_Obj> <L2TP_Obj> <GRE_Obj> <OpenVPN_Obj>	optional	Detail of connection method To update the connection method, 'type' cannot be absent

<DHCP_Obj>

	Type	*Notation*	*Mandatory*	*Description*
hostname	String	<string>	optional	Hostname
ipPassthrough	Boolean	<boolean>	optional	IP passthrough Only valid when that is not in drop in mode and port type is cellular or ethernet
staticRoute	Array	list of <Network_Obj>	optional	Static Route for IP passthroughOnly valid when that is not in drop in mode and port type is cellular or ethernet and 'ipPassthrough' is true

<Static_IP_Obj>

	Type	*Notation*	*Mandatory*	*Description*
ip	String	<ipv4>	require	IP address
mask	Number	[0, 32]	require	Subnet mask
gateway	String	<ipv4>	require	Gateway
ipPassthrough	Boolean	<boolean>	optional	IP passthrough Only valid when that is not in drop in mode and port type is cellular or ethernet
staticRoute	Array	list of <Network_Obj>	optional	Static Route for IP passthroughOnly valid when that is not in drop in mode and port type is cellular or ethernet and 'ipPassthrough' is true

<PPPoE_Obj>

	Type	*Notation*	*Mandatory*	*Description*
username	String	<string>	require	Username
password	String	<string>	require	Password
service	String Null	<string> <null>	optional	Service Information which is provide by Internet Service Provider(ISP) To clear the setting, give the JSON null
ip	String	<ipv4>	optional	IP address Information which is provide by Internet Service Provider(ISP) To clear the setting, give the JSON null
managementNetwork	Object	<Network_Obj>	optional	Management IP Address Information which is provide by Internet Service Provider(ISP) To clear the setting, give the JSON null
keepaliveInterval	Number Null	<integer> <null>	optional	Keep alive interval To clear the setting, give the JSON null
keepaliveRetry	Number Null	<integer> <null>	optional	Keep alive retry To clear the setting, give the JSON null

<L2TP_Obj>

	Type	*Notation*	*Mandatory*	*Description*
username	String	<string>	require	Username

<L2TP_Obj>

	Type	*Notation*	*Mandatory*	*Description*
password	String	<string>	require	Password
host	String	<ipv4>	require	Host IP address
staticIp	Object Null	<Static_IP_Common_Obj> <null>	optional	Static IP To clear the setting, give the JSON null

<GRE_Obj>

	Type	*Notation*	*Mandatory*	*Description*
staticIp	Object	<Static_IP_Common_Obj>	optional	Static IP This field is mandatory when the Port is ethernet
host	String	<ipv4>	require	Host IP address
local	String	<ipv4>	require	Lcoal IP address
remote	String	<ipv4>	require	Remote IP address
nat	String	<ipv4>	optional	NAT IP address

<OpenVPN_Obj>

	Type	*Notation*	*Mandatory*	*Description*
username	String	<string>	optional	Username
password	String	<string>	optional	Password

<Static_IP_Common_Obj>

	Type	*Notation*	*Mandatory*	*Description*
ip	String	<ipv4>	require	IP address
mask	Number	[0, 32]	require	Subnet mask
gateway	String	<ipv4>	require	Gateway

<Network_Obj>

	Type	*Notation*	*Mandatory*	*Description*
ip	String	<ipv4>	require	IP address
mask	Number	[0, 32]	require	Subnet mask

<DNS_Obj>

	Type	*Notation*	*Mandatory*	*Description*
auto	Boolean	<boolean>	optional	Auto DNS
host	Array	list of <ipv4>	optional	Host IP addresses

<Hot_Standby_Obj>

	Type	*Notation*	*Mandatory*	*Description*
enable	Boolean	<boolean>	require	Enable hot standby
schedule	Number Null	<integer> <null>	optional	Schedule ID for hot standby To disable schedule, give the JSON null

<Bandwidth_Map_Obj>

	Type	*Notation*	*Mandatory*	*Description*
upload	Object	<Bandwidth_Obj>	optional	Bandwidth upload information
download	Object	<Bandwidth_Obj>	optional	Bandwidth download information

<Bandwidth_Obj>

	Type	*Notation*	*Mandatory*	*Description*
value	Number	<integer>	require	Upload / Download value Mininum - 1 kbps Maxinum - 10 Gbps
unit	String	{ kbps, Mbps, Gbps }	require	Unit

<Modem_Obj>

	Type	*Notation*	*Mandatory*	*Description*
operator	Object Null	<Operator_Obj> <null>	optional	Operator information To clear the setting, give the JSON null
simPin	String Null	<string> <null>	optional	SIM Pin
mobileType	String	{ 4G, 3G, 2G }	optional	Mobile type
huaweiBand	Array	list of { GSM1900, GSM900/GSM1800/WCDMA2100 }	optional	The Band for Huawei Modem

<Cellular_Obj>

	Type	*Notation*	*Mandatory*	*Description*
useExternalAntenna	Boolean	<boolean>	optional	Use external antenna
simCardScheme	String	{ <empty>, 1, 2, alternate, remote_sim }	optional	SIM card scheme <empty> - Default (Internal / Both SIMs) 1 - SIM A only 2 - SIM B only alternate - Alternate periodically between SIM A only and SIM B only remote_sim - RemoteSIM (The API error if the device not support RemoteSIM)
preferredSim	Number	{1, 2}	optional	Preferred SIM
idleTimeout	Number Null	<integer> <null>	optional	Idle timeout To disable idle timeout, give the JSON null
failbackTimeout	Number Null	<integer> <null>	optional	Failback timeout To disable failback timeout, give the JSON null
remoteSim	ArrayNull	list of <string> <null>	optional	RemoteSIM information
alternateSim	Object	<Alternate_SIM_Obj>	optional	Alternate SIM information Only suport when simCardScheme is alternate
sim	Array	list of <SIM_Obj>	optional	SIM information
signalThreshold	Object	<Signal_Threshold_Obj>	optional	Signal threshold

<Alternate_SIM_Obj>

	Type	*Notation*	*Mandatory*	*Description*
day	Number	<integer>	require	Alternate SIM day
hour	Number	<integer>	require	Altherate SIM hour

<SIM_Obj>

	Type	*Notation*	*Mandatory*	*Description*
id	Number	<integer>	require	SIM ID
carrierSelection	Array Null	list of <Carrier_Selection_Obj> <null>	require	Carrier selection
mobileType	String	{ LTE, 3G, 2G }	optional	Mobile type
optimalNetwork	Object Null	<Optimal_Network_Obj> <null>	optional	Optimal Network To clear the setting, give the JSON null
bandSelection	Array Null	list of <string> <null>	optional	Band Selection To disable band selection, give the JSON null

<SIM_Obj>

	Type	*Notation*	*Mandatory*	*Description*
				NOTE: The string format to be confirm
roaming	Object	<Roaming_Obj>	optional	Roaming
authentication	String Null	{pap, chap} <null>	optional	Authentication mode
operator	Object Null	<Operator_Obj> <null>	optional	Operator information
simPin	String Null	<string><null>	optional	SIM Pin
bandwidthAllowanceMonitor	Object	<BW_Allowance_Monitor_Obj>	optional	Bandwidth allowance monitor Only valid when the device support bandwidth allowance of cellular

<Carrier_Selection_Obj>

	Type	*Notation*	*Mandatory*	*Description*
mcc	String	<string>	optional	MCC
mnc	String	<string>	optional	MNC
pcs	Number	<integer>	optional	PCS
name	String	<string>	optional	Name
plmn	String	<string>	optional	PLMN

<Signal_Threshold_Obj>

	Type	*Notation*	*Mandatory*	*Description*
signalLevel	Array Null	list of [0, 5] <null>	optional	Signal Level
rsrp	Array Null	list of [-140, -44] <null>	optional	RSRP
sinr	Array Null	list of [-100, 100] <null>	optional	SINR
rssi	Array Array Null	list of [-125, -10] list of [-192, 63] <null>	optional	RSSI For Cellular WAN - [-125, -10] For Wi-Fi WAN - [-192, 63] To remove rssi, give the JSON null

<Optimal_Network_Obj>

	Type	*Notation*	*Mandatory*	*Description*
discovery	Number	[5, 480]	optional	Optimal network discovery
period	Array	list of <integer>	optional	Optimal network period

<Roaming_Obj>

	Type	*Notation*	*Mandatory*	*Description*
enable	Boolean	<boolean>	optional	Roaming enable
accessControlList	Array Null	list of <integer> <null>	optional	Access conotrol list No effect at this moment
mode	String Null	{ whitelist, blacklist }	optional	Roaming mode
name	String	<string>	optional	Name
plmn	String	<string>	optional	PLMN

<Operator_Obj>

	Type	*Notation*	*Mandatory*	*Description*
apn	String	<string>	optional	APN
username	String	<string>	optional	Username for the APN
password	String	<string>	optional	Password for the APN

<Operator_Obj>

	Type	*Notation*	*Mandatory*	*Description*
dialNumber	String	{1234567890*#}	optional	Dial Number Only support in modem type

<Wifi_Obj>

	Type	*Notation*	*Mandatory*	*Description*
country	Number	<integer>	optional	Country ID Only for beta, make sure you know the ID is representing the country you wanted.
channelWidth	String	{ 20 MHz, 20/40 MHz, 40MHz, 80 MHz, 20/40/80 MHz, auto }	optional	Channel width
channel	Array	list of <integer>	optional	Channel Only for beta, make sure all channels in the array are correct
power	String	{ custom, auto, manual, high, medium, low, max }	optional	Power
powerBoost	Boolean	<boolean>	optional	Power Boost
dataRate	String	MCS{[0, 9]}	optional	Data RateOnly for beta, make sure data string is correct and match the channel width
roaming	Object	<Wifi_Roaming_Obj>	optional	Roaming information
autoConnect	Boolean	<boolean>	optional	Auto Connect
beaconMissCounter	Number	[2, 100]	optional	Beacon miss counter
channelScanInterval	Number	[5, 1000]	optional	Channel scan interval
signalThreshold	Object	<Signal_Threshold_Obj>	optional	Signal Threshold

<Wifi_Roaming_Obj>

	Type	*Notation*	*Mandatory*	*Description*
enable	Boolean	<boolean>	optional	Enable
algorithm	Object	<Wifi_Roaming_Algo_Obj>	optional	Roaming Algorithm

<Wifi_Roaming_Algo_Obj>

	Type	*Notation*	*Mandatory*	*Description*
type	String	{ normal, advanced, express }	optional	Algorithm type
detail	Object	<Wifi_Roaming_Algo_Detail_Obj>	optional	Algorithm detail

<Wifi_Roaming_Algo_Detail_Obj>

	Type	*Notation*	*Mandatory*	*Description*
signalLevel	Object	<Wifi_Roaming_Algo_Signal_Level_Obj>	optional	Signal level
checkInterval	Number	[5, 3600]	optional	Check interval
intensiveScan	Object	<Wifi_Roaming_Algo_Adv_Intensive_Scan_Obj>	optional	Intensive scan This field only for advanced
diagnosticLevel	String	{ minimum, basic, detail }	optional	Diagnostic level This field only for express
signalMode	Object	<Wifi_Roaming_Algo_Exp_Signal_Mode_Obj>	optional	Signal mode This field only for express
forceRoam	Object	<Wifi_Roaming_Algo_Exp_Force_Roam_Obj>	optional	Intensive scan This field only for express
confirmPeriod	Number	<integer>	optional	Confirm period This field only for express
backupDisconnect	Object	<Wifi_Roaming_Algo_Exp_Backkup_Disconnect_Obj>	optional	Backup disconnect This field only for express
authenticationTimeout	Number	<integer>	optional	Authentication timeout This field only for express

<Wifi_Roaming_Algo_Signal_Level_Obj>

	Type	*Notation*	*Mandatory*	*Description*
threshold	Number	[-95, -40]	optional	Signal level threshold
gain	Number	[5, 55]	optional	Signal level gain

<Wifi_Roaming_Algo_Adv_Intensive_Scan_Obj>

	Type	*Notation*	*Mandatory*	*Description*
enable	Boolean	<boolean>	optional	Enable intensive scan
signalLevel	Number	[-95, -40]	optional	Signal level for intensive scan
scanInterval	Number	[1, 3600]	optional	Scan interval for intensive scan

<Wifi_Roaming_Algo_Exp_Signal_Mode_Obj>

	Type	*Notation*	*Mandatory*	*Description*
type	String	{ relative, absolute }	optional	Signal mode type
detail	Object	<Wifi_Roaming_Algo_Exp_Signal_Mode_Detail_Obj>	optional	Signal mode detail

<Wifi_Roaming_Algo_Exp_Signal_Mode_Detail_Obj>

	Type	*Notation*	*Mandatory*	*Description*
minimumSignalDifference	Number	[0, 94]	optional	Minimum signal difference Only valid for type is relative
signalThreshold	Object	<Wifi_Roaming_Algo_Exp_Signal_Threshold_Obj>	optional	Signal threshold Only valid for type is absolute
dynamicZone	Object	<Wifi_Roaming_Algo_Exp_Dynamic_Zone_Obj>	optional	Signal mode detail

<Wifi_Roaming_Algo_Exp_Signal_Threshold_Obj>

	Type	*Notation*	*Mandatory*	*Description*
upper	Number	[-95, -1]	optional	Upper limit
lower	Number	[-95, -1]	optional	Lower limit

<Wifi_Roaming_Algo_Exp_Dynamic_Zone_Obj>

	Type	*Notation*	*Mandatory*	*Description*
inner	Number	[0, 95]	optional	Inner limit
outer	Number	[0, 95]	optional	Outer limit

<Wifi_Roaming_Algo_Exp_Force_Roam_Obj>

	Type	*Notation*	*Mandatory*	*Description*
enable	Boolean	<boolean>	optional	Force roam enable
threshold	Number	[-95, -1]	optional	Force roam threshold

<Wifi_Roaming_Algo_Exp_Backkup_Disconnect_Obj>

	Type	*Notation*	*Mandatory*	*Description*
mode	String	{ no, immediate, delay }	optional	Backup disconnect mode
delay	Number	<integer>	optional	Delay value Only valid for mode is delay

<Signal_Threshold_Obj>

	Type	*Notation*	*Mandatory*	*Description*
signalLevel	Array Null	list of [0, 5] <null>	optional	Signal Level
rsrp	Array Null	list of [-140, -44] <null>	optional	RSRP

<Signal_Threshold_Obj>

	Type	*Notation*	*Mandatory*	*Description*
sinr	Array Null	list of [-100, 100] <null>	optional	SINR
rssi	Array Array Null	list of [-125, -10] list of [-192, 63] <null>	optional	RSSI For Cellular WAN - [-125, -10] For Wi-Fi WAN - [-192, 63] To remove rssi, give the JSON null

<Physical_Obj>

	Type	*Notation*	*Mandatory*	*Description*
speed	String	{Auto, 1000baseTx-FD, 100baseTx-FD, 100baseTx-HD, 10baseT-FD, 10baseT-HD}	optional	Speed The field only validate when the port is ethernet NOTE: 1000baseTx-FD only support Giga Ethernet port
advertise	Boolean	<boolean>	optional	Advertise The field only validate when the port is ethernet
mtu	Number Number Number Number Null	[576, 1492] [576, 1476] [576, 9000] [576, 1500] <null>	optional	MTU value For PPPoE, the max value is 1492 For GRE, the max value is 1476 For Jumbo frame, the max value is 9000 Otherwise, the max value is 1500 To clear the setting, give the JSON null
mss	Number Null	[536, 1452] <null>	optional	MSS value The value cannot be more than mtu value - 40 To clear the setting, give the JSON null
ttl	Number Null	[1, 255] <null>	optional	TTL value To clear the setting, give the JSON null
mac	String Null	<mac> <null>	optional	MAC address The field only available when the connectionType is ethernetTo clear the setting, give the JSON null
vlan	Number Null	[1, 4094] [1, 10] <null>	optional	VLAN ID The field only available when the port is ethernet or VDSLFor ethernet, the max value is 4094 For VDSL, the max value is 10 To clear the setting, give the JSON null
vpi	Number	[1, 255]	optional	VPI value The field only available when the port is ADSL or VDSL
vci	Number	[32, 65535]	optional	VCI value The field only available when the port is ADSL or VDSL
greUplink	Number	<conn_id>	optional	GRE uplink The field only available when the port is GRE
openvpn	Object	<Physical_OpenVPN_Obj>	optional	OpenVPN information

<Physical_OpenVPN_Obj>

	Type	*Notation*	*Mandatory*	*Description*
uplink	Array	list of <OpenVPN_Uplink_Priority_Obj>	optional	OpenVPN Uplink
failback	Boolean	<boolean>	optional	OpenVPN connection failback

<OpenVPN_Uplink_Priority_Obj>

	Type	*Notation*	*Mandatory*	*Description*
id	Number	<conn_id>	require	WAN connection ID
priority	Number	<integer>	require	Priority

<Healthcheck_Obj>

	Type	*Notation*	*Mandatory*	*Description*
enable	Boolean	<boolean>	require	Healthcheck enable
method	Object	<Healthcheck_Method_Obj>	optional	Healthcheck method
timeout	Number	[200, 10000]	optional	Healthcheck timeout

<Healthcheck_Obj>

	Type	*Notation*	*Mandatory*	*Description*
				Normally, the range is 801-10000 200-800 is for ping only
interval	Number	[5, 3600]	optional	Healthcheck interval
retry	Number	[1, 20]	optional	Healthcheck retry
recovery	Number	[1, 20]	optional	Healthcheck recovery

<Healthcheck_Method_Obj>

	Type	*Notation*	*Mandatory*	*Description*
type	String	{ ping, nslookup, http, smartcheck, openvpn}	require	Healthcheck enable
detail	Object	<Healthcheck_Method_Host_Obj>	require	Healthcheck detail The field has no effect for type is openvpn

<Healthcheck_Method_Host_Obj>

	Type	*Notation*	*Mandatory*	*Description*
includedPublic	Boolean	<boolean>	optional	Included public IP This field only for method type 'nslookup'
host	Array Array	list of <ipv4> list of <Healthcheck_Method_HTTP_Obj>	optional	Host IP address The maximum array size is 2 For method type ping, nslookup, smartcheck, list of <ipv4> For method type http, list of <Healthcheck_Method_HTTP_Obj>

<Healthcheck_Method_HTTP_Obj>

	Type	*Notation*	*Mandatory*	*Description*
host	Array	list of <URL_Pattern_Obj>	optional	Host URL pattern

<URL_Pattern_Obj>

	Type	*Notation*	*Mandatory*	*Description*
url	String	<string>	require	URL
pattern	String	<string>	require	Pattern

<BW_Allowance_Monitor_Obj>

	Type	*Notation*	*Mandatory*	*Description*
enable	Boolean	<boolean>	optional	Bandwidth allowance monitor enable
action	Array	list of { email, disconnect, restrict }	optional	The actions which the allowance is reach
start	Number	[0, 28]	optional	Bandwidth allowance monitor start day
monthlyAllowance	Object	<BW_Allowance_Monitor_Monthly_Obj>	optional	Bandwidth monthly allowance

<BW_Allowance_Monitor_Monthly_Obj>

	Type	*Notation*	*Mandatory*	*Description*
value	Number	<integer>	require	Bandwidth allowance monitor monthly allowance value
unit	String	{ MB, GB, TB }	require	Bandwidth allowance monitor monthly allowance unit for value

<DDNS_Obj>

	Type	*Notation*	*Mandatory*	*Description*
enable	Boolean	<boolean>	optional	DDNS enable
provider	String	{ changeip, dyndns, noip, tzo, dnsomatic, others }	optional	DDNS service provider

<DDNS_Obj>

	Type	*Notation*	*Mandatory*	*Description*
customUrl	String	<string>	optional	Custom URL This field only valid for provider is others
useWanIp	Boolean	<boolean>	require	Use WAN IP
username	String	<string>	require	Username for the service
password	String	<string>	require	Password for the service
host	Array	list of <domain>	require	Host Allow empty array when the provider is dnsmatic

Return Parameters

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"action":"update","list":
[{"id":1,"enable":true}]}' http://192.168.1.1/api/config.wan.connection
```


```
{
    "stat": "ok",
    "response": {
        "1": {
            "name": "WAN 1",
            "asLan": false,
            "enable": true,
            "active": true,
            "multipleIp": [],
            "connection": {
                "method": "dhcp",
                "mode": "NAT",
                "icmpPing": true,
                "priority": 1,
                "dns": {
                    "auto": true
                },
                "ddns": {
                    "username": "username",
                    "password": "@~HiDdEn~@",
                    "host": [
                        "kjkjkjkj.com"
                    ],
                    "provider": "noip",
                    "enable": true
                },
                "bandwidth": {
                    "upload": {
                        "bandwidth": 100000,
                        "unit": "kbps"
                    },
                    "download": {
                        "bandwidth": 100000,
                        "unit": "kbps"
                    }
                },
                "schedule": 4,
                "dhcp": {
                    "hostname": ""
                }
            },
            "healthcheck": {
                "method": "nslookup",
                "timeout": 5,
                "interval": 5,
                "retry": 3,
                "recovery": 3,
                "enable": true,
                "nslookup": {
```

```
                    "includePublic": false,
                    "host": [
                        "208.67.222.222",
                        "208.67.220.220"
                    ]
                }
            }
        },
        "2": {
            "name": "WAN2",
            "asLan": false,
            "enable": true,
            "active": true,
            "multipleIp": [],
            "connection": {
                "method": "dropIn",
                "mode": "IP Forwarding",
                "icmpPing": true,
                "priority": 0,
                "dns": {
                    "auto": false,
                    "host": [
                        "3.3.3.3"
                    ]
                },
                "ddns": {
                    "enable": false
                },
                "bandwidth": {
                    "upload": {
                        "bandwidth": 100000,
                        "unit": "kbps"
                    },
                    "download": {
                        "bandwidth": 100000,
                        "unit": "kbps"
                    }
                },
                "pepVpnNat": true,
                "dropIn": {
                    "ip": "169.254.0.1",
                    "mask": 24,
                    "gateway": "22.2.2.2"
                }
            },
            "healthcheck": {
                "method": "nslookup",
                "timeout": 5,
                "interval": 5,
                "retry": 3,
                "recovery": 3,
                "enable": true,
                "nslookup": {
                    "includePublic": true
                }
            }
        },
        "order": [
            1,
            2
        ]
    }
}
```


POST /api/config.wan.connection.priority

API

Change the priority of the WAN connection
The API will return WAN connection ID, priority and enable information which are just updated.

Avaliable in 7.1.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
instantActive	String	<boolean>	optional	Priority should be updated and take effect immediately When omitted, the configuration will be saved normally, and pending for the explicit apply changes action to take effec
list	Array	list of <WAN_Config_Priority_Obj>	optional	The list of object for changing the priority.

<WAN_Config_Priority_Obj>

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	WAN connection ID
priority	Number	<integer>	optional	Priority of the WAN connection
group	Number	[0, 1]	optional	0 means normal priority 1 means independent from backup WAN This field only available in fw9.0.0 or later
enable	Boolean	<boolean>	optional	Enable the WAN connection

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <conn_id>	The order of WAN ID
<conn_id>	Object	<WAN_Config_Priority_Obj>	WAN config information

<WAN_Config_Priority_Obj>

	Type	*Notation*	*Description*
name	String	<string>	Name of the WAN connection
priority	Number	<integer>	Priority of the WAN connection
group	Number	[0, 1]	0 means normal priority 1 means independent from backup WAN This field only available in fw9.0.0 or later
enable	Boolean	<boolean>	WAN connection enabled or not

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"instantActive":true,"list":
[{"connId":1,"priority":1},{"connId":2,"priority":2}]}'
http://192.168.1.1/api/config.wan.connection.priority
```


```
{
    "stat": "ok",
    "response": {
        "1": {
            "name": "WAN 1",
            "priority": 1,
            "group": 0,
            "enable": true
        },
        "2": {
            "name": "WAN 2",
            "priority": 2,
            "group": 0,
            "enable": true
        },
```

```
        "order": [
            1,
            2
        ]
    }
}
```

GET /api/info.firmware

API internal testing

Retrieves information on the device's firmware.
This API can also be done before login, but it will only return information on the firmware version that is currently in use.

Avaliable in 7.1.1 or later

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <fw_id>	The order of firmware information by ID.
<fw_id>	Object	<Firmware_Obj>	Firmware information.

<Firmware_Obj>

	Type	*Notation*	*Description*
version	String	<string>	Firmware version
bootable	Boolean	<boolean>	Firmware is bootable or not
inUse	Boolean	<boolean>	Firmware is running or not

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/info.firmware
```

```
{
    "stat": "ok",
    "response": {
        "1": {
            "version": "7.0.3 build 2765",
            "bootable": true,
            "inUse": false
        },
        "2": {
            "version": "7.1.0 build 2860",
            "bootable": true,
            "inUse": true
        },
        "order": [
            1,
            2
        ]
    }
}
```

GET /api/info.location

API

Obtain GPS data and other information related to location.

Avaliable in 8.0.1 or later

Return Parameters

Return JSON

	Type	*Notation*	*Description*
gps	Boolean	<boolean>	The GPS signal is valid or not
location	Object	<GPS_Location_Obj>	GPS Location information

<GPS_Location_Obj>

	Type	*Notation*	*Description*
latitude	Number	<double>	-
longitude	Number	<double>	-
altitude	Number	<double>	-
speed	Number	<double>	-
heading	Number	<double>	-
pdop	Number	<double>	Position Dilution Of Precision
hdop	Number	<double>	Horizontal Dilution Of Precision
vdop	Number	<double>	Vertical Dilution Of Precision
timestamp	Number	<integer>	-

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/info.location
```

```
{
    "stat": "ok",
    "response": {
        "gps": true,
        "location": {
            "latitude": 22.340134,
            "longitude": 114.152588,
            "altitude": 55.1,
            "speed": 0.026751,
            "heading": 356.887,
            "pdop": 1.3,
            "hdop": 1,
            "vdop": 0.8,
            "timestamp": 1311972720
        }
    }
}
```

GET /api/info.time

API

Obtain current time of the router

Avaliable in 8.5.0 or later

Return Parameters

Return JSON

	Type	*Notation*	*Description*
time	Number	<integer>	Current time of the router

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/info.time
```

```
{
    "stat": "ok",
    "response": {
```

```
        "time": 1723077803
    }
}
```

POST /api/status.cellularModule.temperature

API

Obtain the current temperature of the cellular module in degrees Celsius.

Avaliable in 8.4.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Number	<conn_id>	require	WAN connection ID of the cellular module.

Return Parameters

Return JSON

	Type	*Notation*	*Description*
temperature	Number	<integer>	Temperature in degrees Celsius.

cURL Example

```
> curl -b cookies.txt -H "Content-Type: application/json" -X POST -d '{"connId":4}'
http://192.168.1.1/api/status.cellularModule.temperature
```

```
{
    "stat": "ok",
    "response": {
        "temperature": 41
    }
}
```

GET /api/status.client

API

This API retrieves client details including client name, MAC address, IP address, signal information (if any), and other such details.

Avaliable in 8.0.1 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
vlanId	Number	<integer>	optional	Filter clients by VLAN ID. Leave blank to display all VLAN IDs and untagged LAN clients.
activeOnly	String	{ yes, no }	optional	Filter clients by active or inactive. Leave blank to display both active and inactive clients.
connectionType	Array	list of { ethernet, wireless, pptp, stroute, l2tp, openvpn, pepvpn, other }	optional	Filter clients by connection type.
size	Number	[1, 10000000]	optional	Limited the number of clients returned. Leaving this field blank will result in 1000 clients returned.
outputWeight	String	{ full, normal, lite }	optional	Set the content parameters to be returned. full - return all details. normal - return ip, connectionType, clientType, name, mac, bssid, vlanId, essid and active lite - return ip, connectionType, clientType, name, mac, bssid and vlanI

	Type	*Notation*	*Mandatory*	*Description*
infoType	Array	list of { ip, connectionType, lease, name, mac, bssid, port, vlanId, essid, active, signalStrength, speed }	optional	Set the content parameters to be returned. This field will override the outputWeight parameter. Leaving outputWeight and infoType blank will default to outputWeight=normal.

Return Parameters

Return JSON

	Type	*Notation*	*Description*
list	Array	list of <Client_Obj>	The list of the client object.

<Client_Obj>

	Type	*Notation*	*Description*
ip	String	<ipv4>	IP Address
connectionType	String	{ ethernet, wireless, pptp, stroute, l2tp, openvpn, pepvpn, other }	Connection Type of the client If the client is not active, this param will be absent. In fw 8.1.0 or before, it return 'ethernet' accidentally. Better check the 'active' param before this.
lease	Object	<Lease_Obj>	Lease type and expires in second The field only available when the connectionType is ethernet or wireles
name	String	<string>	The name of the drive if any.
mac	String	<mac>	MAC address of the client
bssid	String	<mac>	BSSID of the Wi-Fi. This field only present when connectionType=wireless
vlanId	Number	<integer>	Which VLAN the client connected. When it connects to untagged LAN, this field will be absent.
essid	String	<string>	SSID of the Wi-Fi. This field only present when connectionType=wireless
active	Boolean	<boolean>	The active state of the client
signalStrength	Object	<Signal_Obj>	Signal Strength information Deprecated in fw 8.1.0
signal	Object	<Signal_Detail_Obj>	Signal Strength and Level information First present in fw 8.1.0
speed	Object	<Bandwidth_Obj>	Speed information

<Lease_Obj>

	Type	*Notation*	*Description*
expiresIn	Number	<integer>	Lease expires in second
type	String	{ normal, dhcp, wins }	Lease Type

<Signal_Obj>

	Type	*Notation*	*Description*
value	Number	<integer>	Strength of the Wi-Fi signal
unit	String	{ dBm }	Unit of the signal

<Signal_Detail_Obj>

	Type	*Notation*	*Description*
strength	Number	<integer>	Strength of the Wi-Fi signal in dBm
level	Number	[1, 5]	Signal Level

<Bandwidth_Obj>

	Type	*Notation*	*Description*
download	Number	<integer>	Download rate
upload	Number	<integer>	Upload rate
unit	String	{ kbps }	Unit of the speed

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/status.client?connectionType=ethernet wireless
```


```
{
    "stat": "ok",
    "response": {
        "list": [
            {
                "ip": "192.168.50.4",
                "connectionType": "wireless",
                "name": "Android client",
                "mac": "9C:5C:F9:2B:85:99",
                "bssid": "00:1A:DD:ED:8F:69",
                "essid": "PEPWAVE_D3B1",
                "active": true
            },
            {
                "ip": "192.168.50.11",
                "connectionType": "ethernet",
                "name": "macOS client",
                "mac": "E4:25:E7:8A:D3:12",
                "active": false
            },
            {
                "ip": "192.168.50.17",
                "connectionType": "wireless",
                "name": "iOS client",
                "mac": "34:12:98:9B:11:D7",
                "active": false
            }
        ]
    }
}
```


GET /api/status.extap.mesh

API

Obtain the status of the mesh network

Avaliable in 8.3.2 or later

Return Parameters

Return JSON

	Type	*Notation*	*Description*
<mesh_profile_id>	Object	<Mesh_Profile_Obj>	Show the mesh ID and peer device info
order	Array	list of <mesh_profile_id>	List the mesh profile ID

<Mesh_Profile_Obj>

	Type	*Notation*	*Description*
meshId	String	<string>	Mesh ID
peer	Array	list of <Connected_Device_Obj>	List the connected device info in this mesh ID

<Connected_Device_Obj>

	Type	*Notation*	*Description*
deviceInfo	Object	<Device_Info_Obj>	The connected device information
duration	Number	<integer>	The connected duration in second
rssi	Number	[-192, 63]	Received Signal Strength Indicator (RSSI)
signalLevel	Number	[0, 5]	Signal Level
mac	String	<mac>	Peer MAC address
peerName	String	<string>	Peer name. The item will be absent if there is no peer name
linkType	String	{mesh}	Always show mesh
meshId	String	<String>	Mesh ID
protocol	String	{ng, na, ac, axg, axa}	Wi-Fi protocol
rx	Object	<Data_Rate_Obj>	Receive information
tx	Object	<Data_Rate_Obj>	Transmit information

<Device_Info_Obj>

	Type	*Notation*	*Description*
id	Number	<integer>	Device ID
serialNumber	String	<sn>	Serial Number of the device
name	String	<string>	Device name

<Data_Rate_Obj>

	Type	*Notation*	*Description*
rate	String	<string>	Physical rate of the Wi-Fi
packet	Number	<integer>	Packet count
byte	Number	<integer>	Byte count

reference

	Type	*Notation*	*Description*
assignedMeshId	Array	list of <string>	The mesh ID which is assigned to AP controller

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/status.extap.mesh
```


```
{
    "stat": "ok",
    "response": {
        "1": {
            "meshId": "mesh A",
            "peer": [
                {
                    "deviceInfo": {
                        "id": 6917676,
                        "serialNumber": "AAAABBBBCCCC",
                        "name": "AP-One"
                    },
                    "duration": 22010,
                    "rssi": -36,
                    "signalLevel": 5,
                    "mac": "edab784eea89",
                    "linkType": "mesh",
                    "meshId": "mesh A",
                    "protocol": "ac",
                    "rx": {
                        "rate": "2161.36M",
                        "packet": 12416,
                        "byte": 1458903
```

```
                    },
                    "tx": {
                        "rate": "1921.5M",
                        "packet": 30292,
                        "byte": 6105135
                    }
                }
            ]
        },
        "order": [
            1
        ]
    }
}
```


GET /api/status.extap.mesh.link

API alpha

Returns the link & node status of the mesh cluster

Avaliable in 8.4.2 or later

Return Parameters

Return JSON

	Type	*Notation*	*Description*
links	Array	list of <Mesh_Link_Obj>	Mesh links status & info of the mesh cluster
nodes	Object	<Mesh_Node_Obj>	Mesh nodes of the mesh cluster

<Mesh_Link_Obj>

	Type	*Notation*	*Description*
linkType	String	<string>	Mesh link type in string: '2.4GHz', '5GHz', 'ethernet', 'router'
from	Object	<Mesh_Link_Endpoint_Obj>	Mesh link status and data collected from the 'from' node
to	Object	<Mesh_Link_Endpoint_Obj>	Mesh link status and data collected from the 'to' node

<Mesh_Link_Endpoint_Obj>

	Type	*Notation*	*Description*
serialNumber	String	<string-sn>	Serial Number of a mesh node from a mesh link Serial Number with '-', i.e. 1234-5678-ABCD
txRate	Number	<integer>	Tx Rate in mbps of the mesh link, from the node
rxRate	Number	<integer>	Rx Rate in mbps of the mesh link, from the node
signalLevel	Number	<integer>	Mesh link's signal level (1-5), from the node
rssi	Number	<integer>	Mesh link's signal strength in dBm, from the node

<Mesh_Node_Obj>

	Type	*Notation*	*Description*
<string-sn>	Object	<Mesh_Node_Item_Obj>	A Mesh node current status

<Mesh_Node_Item_Obj>

	Type	*Notation*	*Description*
name	String	<string>	Device name
minimumHopCountToGateway	Number	<integer>	Hop count from the mesh node to its gateway -1: router node, 1: first-hop/gateway mesh node

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/status.extap.mesh.link
```

```
{
    "stat": "ok"
}
```

GET /api/status.gpio.input

API internal testing

Obtain the status of the GPIO input

Avaliable in 8.5.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
infoType	Array	list of {analog, digial}	optional	Filter of the return object

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <gpio_id>	The order of GPIO ID
<gpio_id>	Object	<GPIO_Status_Obj>	GPIO Status

<GPIO_Status_Obj>

	Type	*Notation*	*Description*
type	String	{digital_input, analog_input}	Type of the GPIO
name	String	<string>	Name of the GPIO
enable	Boolean	<boolean>	Enable Status of the GPIO
voltage	Number	<double>	Voltage for analog input
state	Number	<integer>	State for digital input

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/status.gpio.input?infoType=analog
```

```
{
    "stat": "ok",
    "response": {
        "1": {
            "name": "GPIO 1",
            "type": "analog_input",
            "enable": true,
            "voltage": 3.434
        },
        "order": [
            1
        ]
    }
}
```

GET /api/status.gpio.output

API internal testing

Obtain the status of the GPIO ouput

Avaliable in 8.5.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
infoType	Array	list of {digial}	optional	Filter of the return object

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <gpio_id>	The order of GPIO ID
<gpio_id>	Object	<GPIO_Status_Obj>	GPIO Status

<GPIO_Status_Obj>

	Type	*Notation*	*Description*
type	String	{digital_input, analog_input}	Type of the GPIO
name	String	<string>	Name of the GPIO
enable	Boolean	<boolean>	Enable Status of the GPIO
state	Number	<integer>	State for digital output

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/status.gpio.output?infoType=digital
```


```
{
    "stat": "ok",
    "response": {
        "2": {
            "name": "GPIO 2",
            "type": "digital_output",
            "enable": true,
            "voltage": 1
        },
        "order": [
            2
        ]
    }
}
```


GET /api/status.lan.profile

API

Obtain Balance LAN Status

Avaliable in 7.1.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
id	Array	list of <lan_id>	optional	list the LAN information base on id, multiple values are accepted, if id is absent, all LAN will return

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <profile_id>	The order of LAN ID
<profile_id>	Object	<LAN_Status_Obj>	LAN status information

<LAN_Status_Obj>

	Type	*Notation*	*Description*
name	String	<string>	LAN / VLAN Name
vlanId	Number	[1, 4094]	VLAN ID. This field will not appear if vlanId is empty
ip	String	<ipv4>	IP address
mask	Number	<maskn>	Subnet mask

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/status.lan.profile
```


```
{
    "stat": "ok",
    "response": {
        "0": {
            "ip": "10.6.1.231",
            "mask": 16
        },
        "1": {
            "name": "Name 1",
            "ip": "10.6.1.231",
            "vlanId": 164,
            "mask": 16
        },
        "order": [
            0,
            1
        ]
    }
}
```


GET /api/status.pepvpn

API beta

Obtain SpeedFusion VPN status

Avaliable in 7.1.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
infoType	Array	list of { profile, peer, tunnel }	optional	Choose the information which is wanted to obtain.
lite	String	{ yes, no }	optional	The call only returns limited data when the field is 'yes'. Otherwise, all status information will be got.
tunnelOption	Array	list of <peer_id>	optional	Retrieve the tunnel information base on peer ID
start	Number	<integer>	optional	Start number of the peer
size	Number	<integer>	optional	Output size of the peer
searchPattern	String	<string>	optional	Search peer by string if the field is not empty
serialNumber	String	<sn>	optional	Search peer by serial number

Return Parameters

Return JSON

	Type	*Notation*	*Description*
profile	Object	<Profile_Order_Obj>	SpeedFusion VPN profile information
peer	Array	list of <Peer_Obj>	Peer Information
tunnel	Object	<Tunnel_Order_Obj>	tunnel Information, if tunnelOption is empty, the field will not be appeared

<Profile_Order_Obj>

	Type	*Notation*	*Description*
order	Array	list of <profile_id>	Order of the profile ID
<profile_id>	Object	<Profile_Obj>	Profile information by ID
siteId	String	<string>	Local ID of the device

<Profile_Obj>

	Type	*Notation*	*Description*
name	String	<string>	SpeedFusion VPN profile Name
master	Boolean	<boolean>	State that is master profile
vlanId	Number	<integer>	VLAN ID of the profile. The field will not appear if lite=yes
status	String	{ START, AUTHEN, TUNNEL, ROUTE, CONFLICT, CONNECTED }	Status of the profile. The filed will not appear if lite=yes
conflictCount	Number	<integer>	Conflict count. The field will not appear if lite=yes
peerCount	Number	<integer>	Peer count. The field will not appear if lite=yes
userShared	Boolean	<boolean>	Allow user shared. The field will not appear if lite=yes
userCount	Number	<integer>	User count. The field will not appear if lite=yes
type	String	{ l3, l2, nats, natc }	Type of the profile. The field will not appear if lite=yes

<Peer_Obj>

	Type	*Notation*	*Description*
serialNumber	String	<sn>	Serial Number of the peer device
status	String	{ START, AUTHEN, TUNNEL, ROUTE, CONFLICT, CONNECTED }	Status of the peer
name	String	<string>	Peer device name
profileId	Number	<integer>	Profile ID of the peer connecting to
secure	Boolean	<boolean>	State the connection is sucured or not
type	String	{ l3, l2, nats, natc }	Type of profile peer connection
username	String	<string>	Account username
conflictRoute	Array	list of <cidr>	Conflict Route of the connection. The field will only appear in Layer3 connection
inactiveRoute	Array	list of <cidr>	Inactive Route of the connection. The field will only appear in Layer3 connection
route	Array	list of <cidr>	Route of the connection. The field will only appear in Layer3 connectio
server	String	<ipv4>	Server IP. The field will only appear in NAT connection
client	String	<cidr>	Client IP with subnet mask. The field will only appear in NAT connectio
bridge	String	<ipv4>	IP of the bridge. The field will only appear in Layer2 connection
vlanId	Number	<integer>	VLAN ID. The field will only appear in Layer2 connection
peerId	String	[<integer>-<integer>]	Unique ID of the peer

<Tunnel_Order_Obj>

	Type	*Notation*	*Description*
order	Array	list of <peer_id>	Order of the peer ID
<peer_id>	Object	{<Tunnel_Obj>, <WAN_Order_Obj>}	Tunnel information by peer ID For fw8.1.0 or above, use <Tunnel_Obj> Before fw8.1.0, use <WAN_Order_Obj>

<Tunnel_Obj>

	Type	*Notation*	*Description*
wan	Object	<WAN_Order_Obj>	Tunnel information by WAN
overall	Object	<Overall_Obj>	Overall tunnel Statistic information

<Overall_Obj>

	Type	*Notation*	*Description*
time	Object	<Time_Obj>	Time information of the tunnel
receive	Object	<Receive_Obj>	Receive information For fw8.1.0 or later
transmit	Object	<Transmit_Obj>	Transmit information For fw8.1.0 or later

<WAN_Order_Obj>

	Type	*Notation*	*Description*
order	Array	list of <conn_id>	Order of the WAN connection ID
<conn_id>	Object	<WAN_Obj>	Tunnel Statistic information by WAN connection ID

<WAN_Obj>

	Type	*Notation*	*Description*
id	Number	<integer>	WAN connection ID
state	String	{ INVALID, WAN_DOWN, WAN_DISABLED, DETECTING, FAILURE, REMOTE_FAILURE, COLD, STATNDBY, P-SUSPD, D-SUSPD, U-SUSPD, P-ACTIV, D-ACTIV, U-ACTIV, ACTIVE }	Status of the tunnel
name	String	<string>	WAN name
time	Object	<Time_Obj>	Time information of the tunnel
nanotime	Object	<Time_Obj>	Time information of the tunnel
rtt	Number	<integer>	Round trip delay time of the remote peer WAN
rx	Array Number	<numlist> <integer>	Receive bytes of the remote peer WAN For local tunnel information, this field is array. Otherwise this field is a number Depreated in fw8.1.0
tx	Array Number	<numlist> <integer>	Transmit bytes of the remote peer WAN For local tunnel information, this field is array. Otherwise this field is a number Deprecated in fw8.1.0
loss	Array Number	<numlist> <integer>	Package loss of the remote peer WAN For local tunnel information, this field is array. Otherwise this field is a number Deprecated in fw8.1.0
receive	Object	<Receive_Obj>	Receive information For fw8.1.0 or later
transmit	Object	<Transmit_Obj>	Transmit information For fw8.1.0 or later

<WAN_Obj>

	Type	*Notation*	*Description*
remote	Object	<WAN_Order_Obj>	Remote WAN tunnel status This field only appear in local tunnel information

<Receive_Obj>

	Type	*Notation*	*Description*
byte	Array Number	<numlist> <integer>	Receive bytes of the remote peer WAN For local tunnel information, this field is array. Otherwise this field is a number Absent for <Overall_Obj>
packet	Object	<Receive_Packet_Obj>	Receive packet of the remote peer WAN

<Transmit_Obj>

	Type	*Notation*	*Description*
byte	Array Number	<numlist> <integer>	Transmit bytes of the remote peer WAN For local tunnel information, this field is array. Otherwise this field is a number Absent for <Overall_Obj>
packet	Object	<Transmit_Packet_Obj>	Time in nano second

<Receive_Packet_Obj>

	Type	*Notation*	*Description*
wan	Array Number	<numlist> <integer>	Receive wan packet For local tunnel information, this field is array. Otherwise this field is a number Absent for <Overall_Obj>
forward	ArrayNumber	<numlist> <integer>	Receive forward packet For local tunnel information, this field is array. Otherwise this field is a number
fragment	Array Number	<numlist> <integer>	Receive fragment packet For local tunnel information, this field is array. Otherwise this field is a number Absent for <Overall_Obj>
loss	Array Number	<numlist> <integer>	Receive loss packet For local tunnel information, this field is array. Otherwise this field is a number
outOfOrder	Array Number	<numlist> <integer>	Receive out of order packet For local tunnel information, this field is array. Otherwise this field is a number
recover	Array Number	<numlist> <integer>	Receive recover packet For local tunnel information, this field is array. Otherwise this field is a number Absent for <Overall_Obj>
discard	Array Number	<numlist> <integer>	Receive discard packet For local tunnel information, this field is array. Otherwise this field is a number Absent for <Overall_Obj>

<Transmit_Packet_Obj>

	Type	*Notation*	*Description*
wan	Array Number	<numlist> <integer>	Transmit wan packet For local tunnel information, this field is array. Otherwise this field is a number Absent for <Overall_Obj>
forward	Array Number	<numlist> <integer>	Transmit forward packet For local tunnel information, this field is array.

<Transmit_Packet_Obj>

	Type	*Notation*	*Description*
			Otherwise this field is a number
fragment	Array Number	<numlist> <integer>	Transmit fragment packet For local tunnel information, this field is array. Otherwise this field is a number Absent for <Overall_Obj>
loss	Array Number	<numlist> <integer>	Transmit loss packet For local tunnel information, this field is array. Otherwise this field is a number
outOfOrder	Array Number	<numlist> <integer>	Transmit out of order packet For local tunnel information, this field is array. Otherwise this field is a number
fec	Array Number	<numlist> <integer>	Transmit forward error correct packet For local tunnel information, this field is array. Otherwise this field is a number Absent for <Overall_Obj>
redundant	Array Number	<numlist> <integer>	Transmit redundant packet For local tunnel information, this field is array. Otherwise this field is a number Absent for <Overall_Obj>

<Time_Obj>

	Type	*Notation*	*Description*
second	Number	<integer>	Time in second
nanoSecond	Number	<integer>	Time in nano second

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/status.pepvpn?infoType=profile
peer&lite=yes&tunnelOption=1-1
```


```
{
    "stat": "ok",
    "response": {
        "profile": {
            "1": {
                "name": "Next (1)",
                "master": true
            },
            "2": {
                "name": "Next (2 - 2)",
                "master": true
            },
            "siteId": "999",
            "order": [
                2,
                1
            ]
        },
        "tunnel": {
            "1-1": {
                "wan": {
                    "1": {
                        "time": {
                            "second": 1292258,
                            "nanoSecond": 485618662
                        },
                        "rtt": 1,
                        "rx": [
                            1423988
                        ],
                        "tx": [
                            1334004
```

```
                    ],
                    "loss": [
                        0
                    ],
                    "priority": 1,
                    "state": "ACTIVE",
                    "name": "WAN 1"
                },
                "2": {
                    "priority": 0,
                    "state": "WAN_DOWN",
                    "name": "WAN 2"
                },
                "3": {
                    "priority": 0,
                    "state": "WAN_DISABLED",
                    "name": "WAN 3"
                },
                "4": {
                    "priority": 0,
                    "state": "WAN_DISABLED",
                    "name": "WAN 4"
                },
                "5": {
                    "priority": 0,
                    "state": "WAN_DISABLED",
                    "name": "WAN 5"
                },
                "6": {
                    "priority": 0,
                    "state": "WAN_DISABLED",
                    "name": "Mobile Internet"
                },
                "order": [
                    1,
                    2,
                    3,
                    4,
                    5,
                    6
                ]
            },
            "overall": {
                "time": {
                    "second": 1292258,
                    "nanoSecond": 485618662
                },
                "receive": {
                    "packet": {
                        "forward": 32,
                        "loss": 1,
                        "outOfOrder": 0
                    }
                },
                "transmit": {
                    "packet": {
                        "forward": 12,
                        "loss": 0,
                        "outOfOrder": 0
                    }
                }
            }
        },
        "order": [
            "1-1"
        ]
    },
```

```
        "peer": [
            {
                "serialNumber": "1825-4131-B4E7",
                "status": "CONNECTED",
                "name": "Ke-B580-x64-30",
                "profileId": 1,
                "secure": true,
                "type": "l3",
                "username": "dev30",
                "route": [
                    "192.168.30.0/24"
                ],
                "peerId": "1-1"
            }
        ]
    }
}
```


GET /api/status.wan.connection

API

Obtain the WAN status.
In fw 8.0.0, band and signal are updated, the API supports multiple bands.

Avaliable in 8.0.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
id	Array	list of <conn_id>	optional	list the WAN information base on id, multiple values are accepted, if id is absent, all WAN will be return
lite	String	{yes, no}	optional	Limited data within the connection will be get when the field set to 'yes' Otherwise, all status information will be got. NOTE: This parameter will not have effect on MAX device.

Return Parameters

Return JSON

	Type	*Notation*	*Description*
order	Array	list of <conn_id>	The order of connection by ID
<conn_id>	Object	<WAN_Status_Obj	WAN Status information

<WAN_Status_Obj>

	Type	*Notation*	*Description*
name	String	<string>	Name of the WAN connection
statusLed	String	{ empty, gray, red, yellow, green, flash }	LED color for UI
asLan	Boolean	<boolean>	WAN port is performing WAN as LAN or not
enable	Boolean	<boolean>	WAN is enabled or not
locked	Boolean	<boolean>	WAN is locked or not.
scheduledOff	Boolean	<boolean>	Only appear if Connection is scheduled and currently off
message	String	<string>	WAN status message
uptime	Number	<integer>	Uptime in second
type	String	{ modem, wireless, gobi, cellular, ipsec, adsl, ethernet }	WAN connection type For cellular WAN In fw8.0.1 or later, it will return "cellular". Before fw8.0.1, it will return "gobi"

<WAN_Status_Obj>

	Type	*Notation*	*Description*
virtualType	String	{ modem, wireless, gobi, cellular, ipsec, adsl, ethernet }	WAN connection type For cellular WAN In fw8.0.1 or later, it will return "cellular". Before fw8.0.1, it will return "gobi"
priority	Number	<integer>	Priority of the WAN. The field will not appear if the WAN is disabled
groupSet	Number	<integer>	Group set of the WAN connection
ip	String	<ipv4>	IP address
mask	Number	<maskn>	Subnet mask. The field will not appear if ip is not exist or lite=yes
gateway	String	ipv4	Gateway. The field will not appear if ip is not exist or lite=ye
method	String	{ dhcp, static }	Connection method, DHCP or Static IP. The field will not appear if lite=yes
mode	String	{ NAT, IP Forwarding }	Connection mode. The field will not appear if lite=yes Please use routingMode in firmware 8.0.1 or later
routingMode	String	{ NAT, IP Forwarding }	Connection mode. The field will not appear if lite=yes
dns	Array	list of <ipv4>	DNS Server list. The field will not appear if lite=yes
additionalIp	Array	list of <ipv4>	Additional IP address list. The field will not appear if lite=ye
mtu	Number	[576, 9000]	MTU value. The field will not appear if auto or lite=yes
mss	Number	[536, 8960]	MSS value. This field will not appear if auto or lite=yes
mac	String	<mac>	MAC address. This field will not appear if lite=yes
wireless	Object	<Wifi_Obj>	WAN connection detail for wireless. The field will only appear if type is wifi
modem	Object	<Modem_Obj>	WAN connection detail for modem. The field will only appea if type is modem
cellular	Object	<Gobi_Obj>	WAN connection detail for gobi. The field will only appear if type is cellular
gobi	Object	<Gobi_Obj>	WAN connection detail for gobi. The field will only appear if type is gobi NOTE: This object is deprecated in firmware 8.0.1.
bandwidthAllowanceMonitor	Object	<BW_Allowance_Monitor_Obj>	Bandwidth allowance monitor. This field will not appear if auto or lite=yes

<Wifi_Obj>

	Type	*Notation*	*Description*
signal	Object	<Signal_Obj>	Signal information
ssid	String	<string>	SSID of the Wifi. The field will not appear if lite=yes
bssid	String	<string>	BSSID. The field will not appear if lite=yes

<Modem_Obj>

	Type	*Notation*	*Description*
name	String	<string>	Modem adaptor name
vendorId	Number	<integer>	Modem adaptor vendor ID
productId	Number	<integer>	Modem adaptor product ID
manufacturer	String	<string>	Modem adaptor manufacturer
carrier	Object	<Carrier_Obj>	Carrier Information
signalLevel	Number	[0, 5]	Signal level
network	String	<string>	Network name
imsi	String	<string>	International Mobile Subscriber Identity (IMSI). The field will not appea if lite=yes
iccid	String	<string>	Integrate circuit card identity (ICCID). The field will not appear if lite=ye
esn	String	<string>	Electronic Serial Number (ESN). The field will not appear if lite=yes

<Modem_Obj>

	Type	*Notation*	*Description*
mtn	String	<string>	Mobile Telecommunications Network (MTN). The field will not appear if lite=yes
apn	String	<string>	APN. The field will not appear if lite=yes
username	String	<string>	Username for APN. The field will not appear if lite=yes
password	String	<string>	Password for APN. The field will not appear if lite=yes
dialNumber	String	{0123456789*#}	Dial number for APN. The field will not appear if lite=yes
band	Array	list of <Band_Obj>	Cellular band information. Including Band Name and signal info

<Gobi_Obj>

	Type	*Notation*	*Description*
roamingStatus	Object	<Roaming_Obj>	Roaming status information
network	String	<string>	Network name This information will be deprecated in fw8.0.1
mobileType	String	<string>	Network name As "network" is deprecated in fw8.0.1, please change the key to use "mobileType" to get the information in fw8.0.1 or later
sim	Object	<SIM_Group_Obj>	SIM information
remoteSim	Object	<Remote_SIM_Obj>	RemoteSIM information, this field will only appear when RemoteSIM is enable
carrier	Object	<Carrier_Obj>	Carrier information
carrierAggregation	Boolean	<boolean>	Carrier Aggregation
signalLevel	Number	[0, 5]	Signal level
meid	Object	<MEID_Obj>	Hex and Dec value of Mobile Equipment Identifier (MEID). The field wil not appear if lite=yes
imei	String	<string>	International Mobile Equipment Identity (IMEI). The field will not appea if lite=yes
esn	String	<string>	Electronic Serial Number (ESN). The field will not appear if lite=yes
mode	String	<string>	Gobi network mode. The field will not appear if lite=yes
band	Array	list of <Band_Obj>	Gobi band information. Including Band Name and signal info NOTE: This field will be obsoleted in fw 8.1.2, please use 'rat' to get the band information
rat	Array	list of <RAT_Obj>	Radio Access Technology support Available after fw 8.1.2
mcc	String	<string>	Three decimal digits as Mobile Country Code(MCC)
mnc	String	<string>	Two or Three decimal digits as Mobile Network Code(MNC)
cellTower	Object	<Cell_Tower_Obj>	Cell Tower information

<RAT_Obj>

	Type	*Notation*	*Description*
name	String	<string>	RAT Name
band	Array	list of <Band_Obj>	Cellular band information

<Band_Obj>

	Type	*Notation*	*Description*
name	String	<string>	Band Name
channel	Number	<integer>	Band Channel Avaliable after 8.1.2
signal	Object	<Signal_Obj>	Signal information

<Signal_Obj>

	Type	*Notation*	*Description*
rssi	Number	<integer>	Received Signal Strength Indicator (RSSI), only appear in Gobi and Modem
sinr	Number	<number>	Signal to Interference plus Noise Ratio (SINR), only appear in Gobi an Modem
snr	Number	<number>	Signal-to-noise ratio (SNR), only appear in Gobi and has value
ecio	Number	<number>	Energy to Interference Ratio (Ec/Io), only appear in Gobi and has valu
rsrp	Number	<integer>	Reference Signal Received Power (RSRP), only appear in Gobi and Modem
rsrq	Number	<number>	Reference Signal Received Quality (RSRQ), only appear in Gobi
strength	Number	<integer>	Wi-Fi signal strength, only appear in Wifi

<SIM_Group_Obj>

	Type	*Notation*	*Description*
order	Array	list of <sim_id>	list of <sim_id>
<sim_id>	Object	<SIM_Obj>	SIM Information for SIM ID

<Remote_SIM_Obj>

	Type	*Notation*	*Description*
imsi	String	<string>	-
serialNumber	String	<string>	-
slot	Number	<integer>	Number of slot
autoApn	Boolean	<boolean>	Indicate the APN, Username and Password fields are auto detect or custom values Only available in fw8.1.1 or later
apn	String	<string>	APN. The field will not appear if lite=yes Only available in fw8.1.1 or later
username	String	<string>	Username for APN. The field will not appear if lite=yes Only available in fw8.1.1 or later
password	String	<string>	Password for APN. The field will not appear if lite=yes Only available in fw8.1.1 or later

<Carrier_Obj>

	Type	*Notation*	*Description*
name	String	<string>	Carrier name
country	String	<string>	Carrier country. The field will not appear if lite=yes

<MEID_Obj>

	Type	*Notation*	*Description*
hex	String	<string>	MEID value in HEX
dec	String	<string>	MEID value in DEC

<SIM_Obj>

	Type	*Notation*	*Description*
status	String	{ In Use, SIM Card Detected, No SIM Card Detected }	SIM card status
active	Boolean	<boolean>	SIM card active status
apn	String	<string>	APN. The field will not appear if lite=yes
username	String	<string>	Username for APN. The field will not appear if lite=yes
password	String	<string>	Password for APN. The field will not appear if lite=yes
imsi	String	<string>	International Mobile Subscriber Identity (IMSI). The field will not appea if lite=yes

<SIM_Obj>

	Type	*Notation*	*Description*
iccid	String	<string>	Integrate circuit card identity (ICCID). The field will not appear if lite=ye
mtn	String	<string>	Mobile Telecommunications Network (MTN). The field will not appear if lite=yes

<Roaming_Obj>

	Type	*Notation*	*Description*
code	Number	{ 0, 1, 2 }	Romaing Status Code
message	String	{ roaming, home, roaming partner }	Readable Roaming Status Code and message relation: 0 - roaming 1 - home 2 - roaming partner

<Cell_Tower_Obj>

	Type	*Notation*	*Description*
cellId	Number	<integer>	Cell ID of the each base transceiver status
cellPlmn	Number	<integer>	Cell Public Land Mobile Network (Cell PLMN) of the tower
cellUtranId	Number	<integer>	Cell UTRAN ID
tac	Number	<integer>	Tracking Area Code for LTE network
lac	Number	<integer>	Location Area Code for GSM/UMTS network

<BW_Allowance_Monitor_Obj>

	Type	*Notation*	*Description*
enable	Boolean	<boolean>	Bandwidth Allowance enable
hasSmtp	Boolean	<boolean>	Email notification is enabled or not
action	Array	list of {email, disconnect, restrict}	Action will take when reach the allowance limit email - send the email, disconnect - disconnect the WAN connection restrict - allow traffic to hostname peplink.com and user defined ICA host only for management purpose
start	Number	[0, 28]	Start date of the allowance monitor. When the value is '0', that means the start day is the last day of that month
monthlyAllowance	Object	<Monthly_Allowance_Obj>	Monthly Allowance Information

<Monthly_Allowance_Obj>

	Type	*Notation*	*Description*
value	Number	<integer>	Monthly Allowance Limit
unit	String	{MB}	The unit for 'value'.

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/status.wan.connection?id=1 2
```


```
{
    "stat": "ok",
    "response": {
        "1": {
            "name": "CUST WAN 1",
            "enable": true,
            "asLan": false,
            "message": "Connected",
            "uptime": 27037017,
            "type": "ethernet",
            "virtualType": "ethernet",
            "priority": 0,
            "ip": "192.168.123.144",
            "statusLed": "green",
```

```
            "mask": 24,
            "gateway": "12.23.34.0",
            "method": "dhcp",
            "mode": "NAT",
            "dns": [
                "12.22.32.12",
                "12.34.67.89"
            ],
            "mtu": 576
        },
        "2": {
            "name": "WAN2",
            "enable": true,
            "asLan": false,
            "message": "No Cable Detected",
            "uptime": 27066417,
            "type": "ethernet",
            "virtualType": "ethernet",
            "priority": 0,
            "statusLed": "red",
            "method": "static",
            "mode": "IP Forwarding",
            "mtu": 1440
        },
        "order": [
            1,
            2
        ]
    }
}
```


GET /api/status.wan.connection.allowance

API

Obtain the bandwidth allowance of the WAN connection or SIM

Avaliable in 8.0.0 or later

Input Parameters

	Type	*Notation*	*Mandatory*	*Description*
connId	Array	list of <conn_id>	optional	Connection ID In firmware 8.0.0, this field is mandatory and ONLY cellular WAN is allowed API user needs to provide the ID to obtain the information In firmware 8.0.1, this field is optional and allow any type of WAN when the field is absent, all WAN connection bandwidth allowance monitor information will be got.

Return Parameters

Return JSON

	Type	*Notation*	*Description*
<conn_id>	Object	{<SIM_Allowance_Obj>, <Allowance_Obj>}	In firmware 8.0.0, only cellular WAN is supported, It will return <SIM_Allowance_Obj> for the allowance monitor. In firmware 8.0.1 or later, all WAN type is supported, it will return <Allowance_Obj> if that is not cellular WAN. In firmware 8.0.1 or later, the output of Cellular WAN will same as firmware 8.0.0
order	Array	list of <conn_id>	WAN Connection ID order reference

<SIM_Allowance_Obj>

	Type	*Notation*	*Description*
<sim_id>	Object	<Allowance_Obj>	Allowance status
order	Array	list of <sim_id>	SIM ID order reference

<Allowance_Obj>

	Type	*Notation*	*Description*
enable	Boolean	<boolean>	-
usage	Number	<integer>	Data used in MB
limit	Number	<integer>	Monthly allowance in MB
percent	Number	[0, 100]	Percentage of the usage
start	Number	[0, 28]	Start day of the allowance, 0 mean the last day of the month
unit	String	{ MB }	-

cURL Example

```
> curl -b cookies.txt http://192.168.1.1/api/status.wan.connection.allowance?connId=1
```

```
{
    "stat": "ok",
    "response": {
        "1": {
            "1": {
                "enable": true,
                "usage": 5,
                "limit": 1024,
                "percent": 0,
                "start": 1,
                "unit": "MB"
            }
        },
        "order": [
            1
        ]
    }
}
```