

API User Guide

Change records

- JiangYan Yu 20-11-2017 The first draft is completed;
- JiangYan Yu 30-11-2017 The content is changed as follows:
 1. Optimize the visit process.
 2. Reclassify the interface object.
 3. Add parameter description.
- JiangYan Yu 21-12-2017 The content is changed as follows:
 1. Modify the response message of each method
 2. Modify the description of each method parameter.
 3. Modify the user.logOut request contents.
 4. Modify the group.get response contents.
 5. Modify the wlan object.
- JiangYan Yu 08-01-2018 The content is changed as follows:
 1. Delete wlan.enable.
- JiangYan Yu 29-01-2018 The content is changed as follows:
 1. Delete group object.
 2. Modify wlan object.
- JiangYan Yu 09-02-2018 The content is changed as follows:
 1. user.logIn added a limit.
 2. wlan info added two parameters : e0211kStatus, e0211vStatus.
 3. wlan.create added a limit.
- JiangYan Yu 02-03-2018 The content is changed as follows:
 1. Modify the 'Request period Parameters' of the wlan.create method.
- JiangYan Yu 12-03-2018 The content is changed as follows:
 1. Add system.getApList method.
 2. Modify the location fields in the return value of the ap.getSystemInfo method.
 3. Modify the wlanTimer of the wlan object.
 4. Delete all the fields that show the plaintext password in the return value of the wlan.get method.
(passphrase, authSecret, accountSecret)
- JiangYan Yu 27-03-2018 The content is changed as follows:
 1. Modify System.getPvc method so that its username can only be Administrator.
- JiangYan Yu 20-06-2018 The content is changed as follows:
 1. Add ap.setStaticIp method.
 2. Add system.setPassword method.
 3. Added description of API support version.
 4. Modify the note of the wlan.create method.

5. Modify the description of the system.saveConfig method.
6. Modify the description of the password parameter in user.logIn and system.getPvc.

- JiangYan Yu 15-07-2018 The content is changed as follows:
 1. Add system.setGroupInfo method.
 2. Add wlan.list method.
 3. Modify the request parameters of the ap.setStaticIp.
- JiangYan Yu 17-07-2018 The content is changed as follows:
 1. Modify the Note description of the Request in the system.setGroupInfo method.
 2. Add the Note description of the Request in the user.logIn method.
- JiangYan Yu 26-09-2018 The content is changed as follows:
 1. Add description of the ap_bridge parameter.
- Daniel Duan 19-03-2019 The content is changed as follows:
 1. Add how to modify WLAN attributes by wlan.create method.
- JiangYan Yu 09-07-2019 The content is changed as follows:
 1. Add rf.setRadio method.

How to access API interface in Express mode

Northbound API supports R3.0.3 (include 3.0.3) or later.

For the Northbound requirements, we provide five objects: user, ap, system, wlan, rf.

The visit process is as follows:

Step 1: Get PVC IP address

The web developer first needs to call the system.getPvc method via the following url to get the pvcip.

Url:

```
http://anyip address:8080/apiaccess
```

Step 2: Login to PVC

After that through the following url with pvcip call user.logIn method to get the session.

Url:

```
http://pvcip address:8080/apiaccess
```

If the PVC switch, you need to re-run Step1 to get a new pvcip.

Step 3: API access

After that through the following url with pvcip call other methods, each call must carry the session.

Url:

```
http://pvcip address:8080/apiaccess
```

If no method is called within 10 minutes, the session will be invalid.

Only by re-executing Step2 to get a valid session, other methods can be called normally.

Step 4: Logout

Finally, when you no longer call other methods, you need to call the `user.log_out` method to delete the session by using the following url.

Url:

```
http://pvcip address:8080/apiaccess
```

1.user object

For the user object, provide two methods: `user.logIn`, `user.logOut`.

user.logIn

Description:

```
This is the login method.
```

Request:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "user.logIn",
  "params": {
    "username": "Administrator",
    "password": "21232f297a57a5a743894a0e4a801fc3"
  }
}
```

Note:

1. Only use the Administrator account to log in.
2. The value of id ranges from -2147483648 to 2147483647. If the value you set for the id is greater than the maximum, it will return you 2147483647. If the value you set for the id is less than the minimum value, it will return you -2147483648.
3. The id in all of the following methods are like this. I would like to explain here, and I won't repeat it later.

Request Parameters:

params	type	description
username	string	The username is Administrator.
password	string	The default password is 21232f297a57a5a743894a0e4a801fc3. This is the md5 hash of "admin".

Response:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "success": "true/false",
    "message": "No id/No method/Login success/Login log account lock/Login log Password error xxx/Login log account banned/Login log saveSessionid error/login log saveSessionValid error/Please access pvc/No params/No username/No password/Can not get password/The username is not Administrator",
    "sessionId": "2eb79fc698f7894349fe2605f37fdf3e"
  }
}
```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.
sessionId	string	Valid session, call other methods must be carry it. If success is false, then there will be no sessionId parameter.

user.logOut

Description:

This is the logout method.

Request:

```
{
  "id": 4,
  "jsonrpc": "2.0",
  "method": "user.logOut",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {}
}
```

Response:

```

{
  "id": 4,
  "jsonrpc": "2.0",
  "result": {
    "success": "true/false",
    "message": "No id/No method/Invalid session/Log out success/Log out fail/Please access
pvc"
  }
}

```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.

2.ap object

For the ap object, provide three methods: ap.getSystemInfo, ap.getClient, ap.setStaticIp.

ap.getSystemInfo

Description:

Show AP details (AP Model, AP MAC, AP IP, UP time, AP SW version running).

Request:

```

{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "ap.getSystemInfo",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {
    "apmac": "aa:bb:cc:dd:ee:ff"
  }
}

```

Request Parameters:

params	type	description
apmac	string	An ap mac.

Response:

```

{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "apinfo":{
      "model": "OAW-AP1221",
      "mac": "aa:bb:cc:dd:ee:ff",
      "ip": "192.168.20.1",
      "uptime": 95458,
      "version": "3.0.0.57",
      "location": "home"
    },
    "success": "true/false",
    "message": "No id/No method/Invalid session/Please access pvc/No params/Ubush lookup id failed/Ubush call nbm failed, status = xxx/Failed to get nbm return value/No topic/Mqtt pub fail/Mqtt timeout/Get ap info success"
  }
}

```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.
apinfo	object	The details of ap. If success is false, then there will be no apinfo object.
model	string	Return ap's model, it belongs to apinfo object.
mac	string	Return ap's mac, it belongs to apinfo object.
ip	string	Return the ip address of ap, it belongs to apinfo object.
uptime	int	Return the total seconds of ap run time.
version	string	Return ap's version, it belongs to apinfo object.
location	string	Return ap's location, it belongs to apinfo object.

ap.getClient

Description:

Get the total number of clients connected on an ap, and also get the list of client connected on an ap.

Request:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "ap.getClient",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {
    "apmac": "aa:bb:cc:dd:ee:ff"
  }
}
```

Request Parameters:

params	type	description
apmac	string	An ap mac.

Response:

```

{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "total": 1,
    "users": [
      {
        "on_line": true,
        "mac": "b0:e2:35:24:e6:54",
        "ap": "46:07:5a:fa:a2:6a",
        "vap": "46:07:5a:fa:a2:77",
        "wlanid": 3,
        "radioid": 1,
        "AssociationSSID": "123",
        "AttachedBand": 5,
        "Channel": 161,
        "AuthType": "OPEN",
        "Portal_username": "",
        "ip": "192.168.20.189",
        "uptime": 15,
        "livetime": 0,
        "signal": 35,
        "mode": 19,
        "rx_rate": 390000,
        "rx_wifirate": 1733300,
        "rx_bytes": 72129,
        "tx_rate": 27000,
        "tx_wifirate": 1733300,
        "tx_bytes": 53132,
        "portal_enable": false,
        "portal_type": "wifidog",
        "portal_state": "know",
        "devtype": "Mobile",
        "hostname": "MI5-xiaomishouji",
        "ostype": "Android",
        "devmodel": "",
        "cputype": "",
        "delaytime": 9999999,
        "packetloss": 100,
        "rx_retr": 0,
        "tx_retr": 0,
        "up_package": 165,
        "down_package": 735,
        "up_rate": 6,
        "down_rate": 4
      }
    ]
  },
  "success": "true/false",
  "message": "No id/No method/Invalid session/Please access pvc/No params/Ubuss lookup id failed/Ubuss call nbm failed, status = xxx/Failed to get nbm return value/No topic/Mqtt pub fail/Mqtt timeout/Get clients success/Um return error"
}

```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.
total	int	Return the total number of clients connected on an ap. If success parameter is false, then there will be no total parameter.
users	array	Return the list of clients info connected on an ap. If success parameter is false, then there will be no users parameter.

ap.setStaticIp

Description:

Set the static IP of an AP.

After the ap.setStaticIp method is executed, if the result shows that the setting is successful, the set Ap takes about 2 minutes to resume communication.

Request:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "ap.setStaticIp",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {
    "apmac": "aa:bb:cc:dd:ee:ff",
    "staticip": "192.168.20.188",
    "netmask": "255.255.254.0",
    "gateway": "192.168.21.254",
    "dns": "219.141.136.10"
  }
}
```

Request Parameters:

params	type	description
apmac	string	An mac of the ap in the group
ip	string	A valid IP address
netmask	string	A valid netmask address
gateway	string	A valid gateway value
The netmask	string	A valid dns value

Response:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "success": "true/false",
    "message": "No id/No method/Invalid session/No params/Ubuss lookup id failed/Ubuss call nbm failed, status = xxx/Failed to get nbm return value/Please access pvc/The parameter is wrong: xxx/Mqtt pub fail/Mqtt timeout/Apmanage return error/Set ip fail/Set ip success"
  }
}
```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.

3.system object

For the system object, provide five methods: system.saveConfig, system.getPvc,system.getApList, system.setPassword, system.setGroupInfo.

system.saveConfig

Description:

Save the configuration for the cluster.
This method is to prevent configuration loss when the AP suddenly loses power.

Request:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "system.saveConfig",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {}
}
```

Response:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "success": "true/false",
    "message": "No id/No method/Invalid session/Please access pvc/Ubush lookup id failed/Ubush call nbm failed, status = xxx/Failed to get nbm return value/Mqtt pub fail/Mqtt timeout/Uc return error/Save config success"
  }
}
```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.

system.getPvc

Description:

Show which ap is the pvc. Accessed by username & password

Request:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "system.getPvc",
  "params": {
    "username": "Administrator",
    "password": "21232f297a57a5a743894a0e4a801fc3"
  }
}
```

Note: Only use the Administrator account to log in.

Request Parameters:

params	type	description
username	string	The username is Administrator.
password	string	The default password is 21232f297a57a5a743894a0e4a801fc3. This is the md5 hash of “admin”.

Response:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "pvcip": "192.168.20.1",
    "success": "true/false",
    "message": "No id/No method/Get pvc success/Can not get password/Password wrong/No
params/No username/No password/The username is not Administrator"
  }
}
```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.
pvcip	string	Return the ip address of pvc. The web developer first needs to call the method to get the pvcip. If success is false, then there will be no pvcip parameter.

system.getApList

Description:

Get the Ap List for the cluster.

Request:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "system.getApList",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {}
}
```

Response:

```

{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "aps": [
      {
        "ap_name": "AP-2B:E0",
        "mac": "b0:e2:35:24:2b:e0",
        "ip": "192.168.20.189",
        "status": 3,
        "role": 1,
        "version": "3.0.3.13"
      },
      {
        "ap_name": "AP-E6:54",
        "mac": "b0:e2:35:24:e6:54",
        "ip": "192.168.20.182",
        "status": 3,
        "role": 2,
        "version": "3.0.3.13"
      }
    ],
    "success": "true/false",
    "message": "No id/No method/Invalid session/Please access pvc/Ubush lookup id failed/Ubush call nbm failed, status = xxx/Failed to get nbm return value/Can't execute command/Get aplist success/The aplist is null"
  }
}

```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.
aps	array	Please refer to [Return aps Values].

Return aps Values:

aps	type	description
ap_name	string	Return ap's name.
mac	string	Return ap's mac.
ip	string	Return the ip address of ap.
status	integer	Return ap's status. 1:Joining; 2:Initializing;3:Working;4:Down.
role	integer	Return ap's role. 1:PVC;2:SVC;3:VC.
version	string	Return ap's version.

system.setPassword

Description:

Set the admin password.

Request:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "system.setPassword",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {
    "password": "e10adc3949ba59abbe56e057f20f883e"
  }
}
```

Request Parameters:

params	type	description
password	string	The password is the md5 hash of a string.

Response:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "success": "true/false",
    "message": "No id/No method/Invalid session/No params/Ubuss lookup id failed/Ubuss call nbm failed, status = xxx/Failed to get nbm return value/Please access pvc/No password/Mqtt pub fail/Mqtt timeout/Apmanage return error/Set password success/Set password fail"
  }
}
```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.

system.setGroupInfo

Description:

Set the group info: name, location, management IP, management netmask, group ID.

Request:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "system.setGroupInfo",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {
    "group_name": "AP-Group",
    "group_location": "office",
    "group_mgtip": "192.168.20.46",
    "group_mgtnetmask": "255.255.254.0",
    "group_id": 8888
  }
}
```

Note:

- 1.Group will be reorganized **if** you modify the group_id, at the same time pvc may change.
After setting the group_id successfully, please wait 20s and then log **in** again.
- 2.The lighttpd certificate will be recreated and loaded **if** you modify the group_mgtip.
So you may not receive a reply.

Request Parameters:

params	type	validation	description
group_name	string	1-25 letters, digits, underlines or middle bars	The group name must be filled.
group_location	string	1-32 digits, letters or underlines	If you don't want to set the group location, this field can not exist.
group_mgtip	string	valid IP address	The group_mgtip and group_mgtnetmask must be set or not set together.
group_mgtnetmask	string	valid netmask address	The group_mgtip and group_mgtnetmask must be set or not set together.
group_id	integer	1-9999	If you don't want to set the group id, this field can not exist. Its default value is 100.

Response:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "success": "true/false",
    "message": "No id/No method/Invalid session/No params/Ubush lookup id failed/Ubush call nbm failed, status = xxx/Failed to get nbm return value/Please access pvc/The parameter is wrong:xxx/The group ip and netmask must be set together/This group_mgtip can't be the same as the ip of the aps in the group/The obtained pvcip is incorrect, and the groupip cannot be verified/This group_mgtip and pvcip are not in a network segment/Mqtt pub fail/Mqtt timeout/Apmanage return error/Set group info success/Set group info fail/Set group id success, Please wait for 20s and then log in again"
  }
}
```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true. If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.

4.wlan object

For the wlan object, provide four methods: wlan.create, wlan.delete, wlan.get, wlan.list.

wlan.create

Description:

Create or modify the various type of essid. It can be controllable time based.
Allow creation of a essid as standard open.

Request:

```

{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "wlan.create",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {
    "ssid": "mywifi",
    "actionrange": "group",    //"group" or "all"
    "groupList": [
      "aa:bb:cc:dd:ee:f0",
      "aa:bb:cc:dd:ee:f1",
      "aa:bb:cc:dd:ee:f2"
    ],
    "allowBand": "all",
    "ssidEnable": "enable",
    "hideSSID": "disable",
    "securityLevel": "Personal",    //"Personal" or "Open" or "Enterprise"
    "encryptionType": "wpa-psk-aes",
    "passphrase": "12345678",
    "ap_bridge": "enable",
    "broadcastFilterARP": "disable",
    "multicastOptimization": "disable",
    "maxClientsPerBand": 13,
    "e0211rStatus": "disable",
    "e0211kStatus": "disable",
    "e0211vStatus": "disable",
    "okcStatus": "enable",
    "wlanTimer": {
      "enable": false,
      "periodType": "daily",        //"daily" or "weekend" or "weekday"
      "period": {
        "enable": false,           //false:all day, true:period start.
        "startTime": "10:59",      // hour and mins
        "endTime": "22:59"
      }
    },
    "upstreamBandwidth": 3,
    "downstreamBandwidth": 4,
    "redirectStatus": "disable",
    "vlanNumber": 123,
    "AuthServer": {
      "authenticationPort": 1812,
      "authIp": "192.168.21.123",
      "authSecret": "2345678"
    },
    "radiusAccounting": true,
    "AccServer": {
      "accountingPort": 1813,
      "accountIp": "192.168.21.111",
      "accountSecret": "123456"
    }
  }
}

```

Note:

- 1.No matter which ap is applied to each WLAN, the number of wlan configured for each radio band can not more than seven.
- 2.If you want to modify an existing wlan, just execute the wlan.create method again.

Request Parameters:

params	type	validation	description
ssid	string	1-32 characters except space. It also can't begin with *. and end with □	Name that uniquely identifies a wireless network. If the name repeats, it will overwrite the wlan with the same name.
actionrange	enum	group/all	If the actionrange is group, the ssid will be enabled/disabled in the group. If the actionrange is all, the ssid will be enabled/disabled in all aps.
groupList	array	Valid apmac list	If the actionrange is group, the groupList need to be filled. If not, it does not need to be filled.
allowBand	enum	2.4/5/all	Configures the radio frequency band on which this ssid will be broadcast. You can select either 2.4 GHz, 5 GHz, or all to specify both bands.
ssidEnable	enum	enable/disable	Enables or disables the ssid.
hideSSID	enum	enable/disable	Enables or disables hiding of the ssid name in beacon frames. Note that hiding the ssid does very little to increase security.
securityLevel	enum	Open/Personal/Enterprise	
encryptionType	enum	wpa-psk-tkip, wpa-psk-aes, wpa-psk-aes-tkip, wpa2-psk-tkip, wpa2-psk-aes, dynamic-wep, wpa-tkip, wpa-aes, wpa2-tkip, wpa2-aes	If the securityLevel is Open, the encryptionType does not need to be filled. If the securityLevel is Personal, the encryptionType will be one of the following values: wpa-psk-tkip, wpa-psk-aes, wpa-psk-aes-tkip, wpa2-psk-tkip, wpa2-psk-aes. If the securityLevel is Enterprise, the encryptionType will be one of the following values: dynamic-wep, wpa-tkip, wpa-aes, wpa2-tkip, wpa2-aes.
passphrase	string	8-63 characters except space. It also can't include `=`	WPA passphrase. If the securityLevel is Personal, this parameter needs to be filled. If not, it does not need to be filled.
ap_bridge	enum	enable/disable	If the ap_bridge is enable, then Client-Isolation is off. If the ap_bridge is disable, then Client-Isolation is on.
broadcastFilterARP	enum	enable/disable	

multicastOptimization	enum	enable/disable	
maxClientsPerBand	integer	1-120	
e0211rStatus	enum	enable/disable	If the securityLevel is Personal and Enterprise, this parameter need to be filled. If not, it does not need to be filled.
e0211kStatus	enum	enable/disable	
e0211vStatus	enum	enable/disable	
okcStatus	enum	enable/disable	If the securityLevel is Enterprise, this parameter need to be filled. If not, it does not need to be filled.
wlanTimer	object	Please refer to [Request wlanTimer Parameters]	
upstreamBandwidth	integer	0-65536	
downstreamBandwidth	integer	0-65536	
redirectStatus	enum	enable/disable	If the securityLevel is Open, this parameter need to be filled. If not, it does not need to be filled.
vlanNumber	integer	0-4094	
AuthServer	object	authenticationPort: 1-65535, authIp: valid IP, authSecret: 1-64 characters except space. It also can't include `=`	If the securityLevel is Enterprise, these parameters need to be filled. If not, it does not need to be filled.
radiusAccounting	bool	true/false	If the securityLevel is Enterprise, these parameters need to be filled.If not, it does not need to be filled.
AccServer	object	accountingPort: 1-65535, accountIp: valid IP, accountSecret: 1-64 characters except space. It also can't include `=`	If the securityLevel is Enterprise and the radiusAccounting is true, these parameters need to be filled.If not, it does not need to be filled.

Request wlanTimer Parameters:

wlanTimer	type	validation	description
enable	bool	true/false	If this parameter is true, time control is valid. If not, time control is invalid.
periodType	enum	daily/weekend/weekday	If the enable is true, this parameter need to be filled. If not, it does not need to be filled.
period	object	Please refer to [Request period Parameters]	If the enable is true, the period object need to be filled. If not, it does not need to be filled.

Request period Parameters:

Period	type	validation	description
enable	bool	true/false	If the enable is false, this wlan effective all day. If not, this wlan effective from startTime to endTime.
startTime	string	hour:mins	If the enable is true, this parameter need to be filled. If not, it does not need to be filled.
endTime	string	hour:mins	If the enable is true, this parameter need to be filled. If not, it does not need to be filled.

Response:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "success": "true/false",
    "message": "No id/No method/Invalid session/No params/Ubuss lookup id failed/Ubuss call nbm failed, status = xxx/Failed to get nbm return value/Please access pvc/The parameter is wrong: xxx/Mqtt pub fail/Mqtt timeout/Wlan return error/Create wlan fail/Create wlan success/The number of wlan configured for each radio band can not more than seven"
  }
}
```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.

wlan.create(For modify)

Description:

Use wlan.create method modify the wlan attributes.

For example:

If you want to disable the WLAN "mywifi", you just need to modify the value of the "ssidEnable" to disable, other attributes should be kept consistent with before.

```

{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "wlan.create",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {
    "ssid": "mywifi",
    "actionrange": "group",    //"group" or "all"
    "groupList": [
      "aa:bb:cc:dd:ee:f0",
      "aa:bb:cc:dd:ee:f1",
      "aa:bb:cc:dd:ee:f2"
    ],
    "allowBand": "all",
    "ssidEnable": "disable",
    "hideSSID": "disable",
    "securityLevel": "Personal",    //"Personal" or "Open" or "Enterprise"
    "encryptionType": "wpa-psk-aes",
    "passphrase": "12345678",
    "ap_bridge": "enable",
    "broadcastFilterARP": "disable",
    "multicastOptimization": "disable",
    "maxClientsPerBand": 13,
    "e0211rStatus": "disable",
    "e0211kStatus": "disable",
    "e0211vStatus": "disable",
    "okcStatus": "enable",
    "wlanTimer": {
      "enable": false,
      "periodType": "daily",        //"daily" or "weekend" or "weekday"
      "period": {
        "enable": false,    //false:all day, true:period start.
        "startTime": "59:10",    // mins and hour
        "endTime": "59:22"
      }
    },
    "upstreamBandwidth": 3,
    "downstreamBandwidth": 4,
    "redirectStatus": "disable",
    "vlanNumber": 123,
    "AuthServer": {
      "authenticationPort": 1812,
      "authIp": "192.168.21.123",
      "authSecret": "2345678"
    },
    "radiusAccounting": true,
    "AccServer": {
      "accountingPort": 1813,
      "accountIp": "192.168.21.111",
      "accountSecret": "123456"
    }
  }
}

```

Note: No matter which ap is applied to each WLAN, the number of wlan configured for each radio band can not more than seven.

wlan.delete

Description:

Delete the essid in a group.

Request:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "wlan.delete",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {
    "ssid": "mywifi"
  }
}
```

Request Parameters:

params	type	description
ssid	string	Name that uniquely identifies a wireless network.

Response:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "success": "true/false",
    "message": "No id/No method/Invalid session/No params/Ubuss lookup id failed/Ubuss call nbm failed, status = xxx/Failed to get nbm return value/Please access pvc/No ssid/Mqtt pub fail/Mqtt timeout/Wlan return error/The ssid is not exist/Delete wlan fail/Delete wlan success"
  }
}
```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.

wlan.get

Description:

Check if a given essid exists.
If it exists, return wlan information.

Request:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "wlan.get",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {
    "ssid": "mywifi"
  }
}
```

Request Parameters:

params	type	description
ssid	string	Name that uniquely identifies a wireless network.

Response:

```

{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "success": "true/false",
    "message": "No id/No method/Invalid session/No params/Ubus lookup id failed/Ubus call
nbm failed, status = xxx/Failed to get nbm return value/Please access pvc/The essid is not
exist/No essid/Get wlan info error/Get wlan info success",
    "wlaninfo": {
      "ssid": "mywifi",
      "actionrange": "group",    //"group" or "all"
      "groupList": [
        "aa:bb:cc:dd:ee:f0",
        "aa:bb:cc:dd:ee:f1",
        "aa:bb:cc:dd:ee:f2"
      ],
      "allowBand": "all",
      "ssidEnable": "enable",
      "hideSSID": "disable",
      "securityLevel": "Enterprise",
      "encryptionType": "wpa-tkip",
      "ap_bridge": "enable",
      "broadcastFilterARP": "disable",
      "multicastOptimization": "disable",
      "maxClientsPerBand": 13,
      "e0211rStatus": "disable",
      "e0211kStatus": "disable",
      "e0211vStatus": "disable",
      "okcStatus": "enable",
      "wlanTimer": {
        "enable": true,
        "periodType": "daily",    //"daily" or "weekend" or "weekday"
        "period": {
          "enable": true,
          "startTime": "10:59",    // hour and mins
          "endTime": "22:59"
        }
      },
      "upstreamBandwidth": 3,
      "downstreamBandwidth": 4,
      "redirectStatus": "disable",
      "vlanNumber": 123,
      "AuthServer": {
        "authenticationPort": 1812,
        "authIp": "192.168.21.123"
      },
      "radiusAccounting": true,
      "AccServer": {
        "accountingPort": 1813,
        "accountIp": "192.168.21.111"
      }
    }
  }
}

```

```
}
```

Return Values:

result	type	description
success	string	If the interface is called successfully and the essid exists, this parameter is true.If the interface fails to call or the essid doesn't exist, this parameter is false.
message	string	Return the result information of the call.
wlaninfo	object	If success is true, wlaninfo will display the details of the wlan. Please refer to [Return wlanInfo Values] for details.

Return wlanInfo Values:

params	type	description
ssid	string	Name that uniquely identifies a wireless network.
actionrange	enum	It means the application scope of the ssid.
groupList	string	If the actionrange is group, the groupList will be displayed. If not, it will not to be displayed.
allowBand	enum	Configures the radio frequency band on which this ssid will be broadcast.
ssidEnable	enum	Enables or disables the ssid.
hideSSID	enum	Enables or disables hiding of the ssid name in beacon frames. Note that hiding the ssid does very little to increase security.
securityLevel	enum	
encryptionType	enum	If the securityLevel is Open, the encryptionType does not to be displayed. If not, it will to be displayed.
ap_bridge	enum	If the ap_bridge is enable, then Client-Isolation is off. If the ap_bridge is disable, then Client-Isolation is on.
broadcastFilterARP	enum	
multicastOptimization	enum	
maxClientsPerBand	integer	
e0211rStatus	enum	If the securityLevel is Personal and Enterprise, it will be displayed. If not, it will not to be displayed.
e0211kStatus	enum	
e0211vStatus	enum	
okcStatus	enum	If the securityLevel is Enterprise, it will be displayed. If not, it will not to be displayed.
wlanTimer	object	
upstreamBandwidth	integer	
downstreamBandwidth	integer	
redirectStatus	enum	If the securityLevel is Open, it will be displayed. If not, it will not to be displayed.
vlanNumber	integer	
AuthServer	object	If the securityLevel is Enterprise, these parameters will be displayed. If not, these parameters will not to be displayed.
		If the securityLevel is Enterprise, it will be displayed. If not, it will not to be displayed.

radiusAccounting	bool	to be displayed.
AccServer	object	If the securityLevel is Enterprise and the radiusAccounting is true, these parameters will be displayed. If not, these parameters will not to be displayed.

wlan.list

Description:

List all existing SSID.

Request:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "wlan.list",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {}
}
```

Response:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "success": "true/false",
    "message": "No id/No method/Invalid session/Ubuss lookup id failed/Ubuss call nbm failed, status = xxx/Failed to get nbm return value/Please access pvc/Get essid list fail/Get essid list success",
    "ssidList": [
      "mywifi1",
      "mywifi2",
      "mywifi3"
    ]
  }
}
```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.
ssidList	array	Display ssid list.

5. rf object

For the rf object, provide one method: rf.setRadio.

rf.setRadio

Description:

Can turn on/off the radio of the AP.

Request:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "method": "rf.setRadio",
  "session": "038e1d7b1735c6a5436ee9eae095879e",
  "params": {
    "actionrange": "group",    //"group" or "all"
    "groupList": [
      "aa:bb:cc:dd:ee:f0",
      "aa:bb:cc:dd:ee:f1",
      "aa:bb:cc:dd:ee:f2"
    ],
    "2.4G": "enable",
    "5G_all": "enable",
    "5G_high": "enable",
    "5G_low": "enable"
  }
}
```

Note: For the four parameters of 2.4G/5G_all/5G_high/5G_low, you only need to send the parameters which you want to set, and don't need to send them all.

Request Parameters:

params	type	validation	description
actionrange	enum	group/all	If the actionrange is group, the ap's radio will be enabled/disabled in the group. If the actionrange is all, the ap's radio will be enabled/disabled in all aps.
groupList	array	Valid apmac list	If the actionrange is group, the groupList need to be filled. If not, it does not need to be filled.
2.4G	enum	enable/disable	Enables or disables the 2.4g radio of the AP. This parameter is valid for all AP models.
5G_all	enum	enable/disable	Enables or disables the 5g radio of the AP. This parameter is valid for other AP models except AP1231/AP1232.
5G_high	enum	enable/disable	Enables or disables the 5g high frequency radio of the AP. This parameter is only valid for AP1231/AP1232 models.
5G_low	enum	enable/disable	Enables or disables the 5g low frequency radio of the AP. This parameter is only valid for AP1231/AP1232 models.

Response:

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "result": {
    "success": "true/false",
    "message": "No id/No method/Invalid session/No params/Ubuss lookup id failed/Ubuss call nbm failed, status = xxx/Failed to get nbm return value/Please access pvc/The parameter is wrong: xxx/Mqtt pub fail/Mqtt timeout/RF return error/Set radio fail/Set radio success"
  }
}
```

Return Values:

result	type	description
success	string	If the interface is called successfully, this parameter is true.If the interface fails to call, this parameter is false.
message	string	Return the result information of the call.