

XML-API Development Guide

Yealink Redirection and Provisioning Service (RPS)



September 2018

Contents

Contents	2
1. Protocol	5
2. Service Port	5
3. User Accounts	5
4. Available XML-RPC Commands	6
4.1 redirect.registerDevice	6
4.1.1 XML-RPC signature:	6
4.1.2 Parameter:	6
4.1.3 Return values:	6
4.1.4 Request:	7
4.1.5 Answer:	8
4.2 redirect.registerDevice^[1]	10
4.2.1 XML-RPC signature:	10
4.2.2 Parameter:	10
4.2.3 Return values:	10
4.2.4 Request:	12
4.2.5 Answer:	13
4.3 redirect.deRegisterDevice	15
4.3.1 XML-RPC signature:	15
4.3.2 Parameter:	15
4.3.3 Return values:	15
4.3.4 Request:	16
4.3.5 Answer:	17
4.4 redirect.listDevices	19
4.4.1 XML-RPC signature:	19
4.4.2 Return values:	19
4.4.3 Request:	20
4.4.4 Answer:	21
4.5 redirect.checkDevice	23
4.5.1 XML-RPC signature:	23
4.5.2 Parameter:	23
4.5.3 Return values:	23
4.5.4 Request:	24
4.5.5 Answer:	25
4.6 redirect.registerDevices	27
4.6.1 XML-RPC signature:	27
4.6.2 Parameter:	27

4.6.3 Return values:	27
4.6.4 Request:	28
4.6.5 Answer:	29
4.7 redirect.registerDevices ^[1]	31
4.7.1 XML-RPC signature:	31
4.7.2 Parameter:	31
4.7.3 Return values:	31
4.7.4 Request:	33
4.7.5 Answer:	34
4.8 redirect.deRegisterDevices	36
4.8.1 XML-RPC signature:	36
4.8.2 Parameter:	36
4.8.3 Return values :	36
4.8.4 Request:	37
4.8.5 Answer:	38
4.9 redirect.addServer	40
4.9.1 XML-RPC signature:	40
4.9.2 Parameter:	40
4.9.3 Return values:	40
4.9.4 Request:	41
4.9.5 Answer:	42
4.10 redirect.editDevice	44
4.10.1 XML-RPC signature:	44
4.10.2 Parameter:	44
4.10.3 Return values:	44
4.10.4 Request:	45
4.10.5 Answer:	46
4.11 redirect.editDevice ^[1]	48
4.11.1 XML-RPC signature:	48
4.11.2 Parameter:	48
4.11.3 Return values:	48
4.11.4 Request:	49
4.11.5 Answer:	50
4.12 redirect.registerDeviceWithUniqueId	52
4.12.1 XML-RPC signature:	52
4.12.2 Parameter:	52
4.12.3 Return values:	52
4.12.4 Request:	53
4.12.5 Answer:	54
4.13 redirect.editDeviceWithUniqueId	56
4.13.1 XML-RPC signature:	56
4.13.2 Parameter:	56
4.13.3 Return values:	56

- 4.13.4 Request:..... 57
- 4.13.5 Answer:..... 58
- 4.14 **redirect.checkDeviceExt**..... 60
 - 4.14.1 XML-RPC signature: 60
 - 4.14.2 Parameter: 60
 - 4.14.3 Return values: 60
 - 4.14.4 Request:..... 61
 - 4.14.5 Answer:..... 62
- 4.15 **redirect.registerDeviceWithUniqueUrl[1]**..... 64
 - 4.15.1 XML-RPC signature: 64
 - 4.15.2 Parameter: 64
 - 4.15.3 Return values: 64
 - 4.15.4 Request:..... 66
 - 4.15.5 Answer:..... 67
- 4.16 **redirect.editDeviceWithUniqueUrl[1]**..... 69
 - 4.16.1 XML-RPC signature: 69
 - 4.16.2 Parameter: 69
 - 4.16.3 Return values: 69
 - 4.16.4 Request:..... 70
 - 4.16.5 Answer:..... 71
- About Yealink**.....**73**

1. Protocol

- The API of RPS is built as standard XML-RPC services. Two ways are possible to access the interface:
 1. You can use XML-RPC libraries available in the Internet.
 2. You can send XML commands via HTTP POST using Content-Type:text/xml.
- TLS/SSL is used to encrypt the communications between client and server.
- User authentication is done via basic authentication using username and password.

2. Service Port

The XML-RPC RPS service can be accessed at:

<https://api-dm.yealink.com:8443/xmlrpc>

3. User Accounts

In order to access the RPS, you need to have a valid account that you can apply for from Yealink or our local distributors.

4. Available XML-RPC Commands

4.1 redirect.registerDevice

You can register a device to RPS by binding its MAC address and server name.

4.1.1 XML-RPC signature:

```
redirect.registerDevice(mac,serverName)
```

4.1.2 Parameter:

1. **mac**:MAC address you want to register to RPS.
2. **serverName**: The server name you have already added to RPS which the device should be redirected to.

4.1.3 Return values:

1. Result code: *True, False*
2. Result text:
 - (1) **OK**: The MAC address is registered successfully.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address
 - 2) *Error: Invalid server*→Invalid server name
 - 3) *Error: The server name can only contain 'A-z 0-9 - _ . '*(include space)→ The serverName contains illegal characters.
 - 4) *Are you sure you want to override MAC addresses.* →The MAC address has been registered by yourself, see more details on section 4.2 below.
 - 5) Error: 001565000001(MAC address)
Please check the MAC address you entered
 - If you need to continue adding this MAC, please create a ticket on <https://ticket.yealink.com>.
We will review your complaint. You need to provide:
 - (1)Device MAC and SN
 - (2)Enterprise Name
 - (3)Firmware Version
 - (4)Login Account

4.1.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
<methodName>redirect.registerDevice</methodName>
<params>
  <param>
    <value><string><![CDATA[001565000001]]></string></value>
  </param>
  <param>
    <value><string><![CDATA[serverName]]></string></value>
  </param>
</params>
</methodCall>
```

Note:

1. Four types of valid MAC address format are 001565000001, 00-15-65-00-00-01, 00 15 65 00 00 01, 00:15:65:00:00:01.

4.1.5 Answer:

4.1.5.1 Register Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            OK
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```


4.1.5.2 Register Failed:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:Invalid MAC(s):00s1122334400
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.2 redirect.registerDevice^[1]

If the MAC address you are registering has been registered by you before, you can confirm whether to override the previous registration by "isOverride" parameter.

4.2.1 XML-RPC signature:

redirect.registerDevice(mac,serverName,isOverride)

4.2.2 Parameter:

1. **mac**: MAC address you want to register to RPS
2. **serverName**: The server name you have already added to RPS which the device should be redirected to.
3. **isOverride**: Confirm whether to override the previous registration or not.
 - 1: Override
 - 0: Cancel, other values are illegal.

4.2.3 Return values:

1. Result code: *True, False*
2. Result text :
 - (1) **OK**: The MAC address is registered successfully.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address
 - 2) *Error:Invalid server*→Invalid server name
 - 3) *Error: The server name can only contain 'A-z 0-9 - _ . '(include space)*→ The serverName contains illegal characters.
 - 4) *Only 0 or 1 can be used:1 to override, 0 to cancel*→Invalid isOverride value
 - 5) *MAC address(es) overriding failed*→Fail to override the previous MAC address and add the MAC.
 - 6) Error: 001565000001(MAC address)
Please check the MAC address you entered
 - If you need to continue adding this MAC, please create a ticket on <https://ticket.yealink.com>. We will review your complaint. You need to provide:
 - (1)Device MAC and SN
 - (2)Enterprise Name
 - (3)Firmware Version
 - (4)Login Account
 →MAC(001565000001) has been added to other enterprise.

[1] Compared with the last one, this API has an extra parameter - *isOverride*.

4.2.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
<methodName>redirect.registerDevice</methodName>
<params>
  <param>
    <value><string><![CDATA[001565000001]]></string></value>
  </param>
  <param>
    <value><string><![CDATA[serverName]]></string></value>
  </param>
  <param>
    <value><string><![CDATA[0]]></string></value>
  </param>
</params>
</methodCall>
```

4.2.5 Answer:

4.2.5.1 Register Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            OK
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.2.5.2 Register Failed:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:Invalid MAC(s):00s1122334400
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.3 redirect.deRegisterDevice

You can remove a specific device from RPS by specifying its MAC address.

4.3.1 XML-RPC signature:

```
redirect.deRegisterDevice(mac)
```

4.3.2 Parameter:

mac :MAC address you want to remove from RPS.

4.3.3 Return values:

1. Result code: *True, False*
2. Result text :
 - (1) **OK**: The MAC address is registered successfully.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address or the MAC address has not been registered by you.

4.3.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
<methodName>redirect.deRegisterDevice</methodName>
<params>
  <param>
    <value> <string> <![CDATA[001122334401]]> </string> </value>
  </param>
</params>
</methodCall>
```


4.3.5 Answer:

4.3.5.1 De-register Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            OK
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.3.5.2 De-register Failed:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:Invalid MAC(s):001122334401
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.4 redirect.listDevices

You can list all the devices(MACs) that you have already registered.

4.4.1 XML-RPC signature:

redirect.listDevices()

4.4.2 Return values:

1. Result code: *True, False*
2. Result text:

(1) **OK:**

```
<array>
  <data>
    <value>MAC1</value>
    <value>MAC2</value>
    ...
  </data>
</array>
```

(2) **Error:** *No device(s) exists*→No registered devices found

4.4.3 Request:

```
<?xml version='1.0' encoding='UTF-8'?>  
<methodCall>  
  <methodName>redirect.listDevices</methodName>  
</methodCall>
```

4.4.4 Answer:

4.4.4.1 There are registered devices:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            <array>
              <data>
                <value>
                  00-11-22-33-44-02
                </value>
                ...
              </data>
            </array>
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.4.4.2 There are no registered devices:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
  <params>
    <param>
      <value>
        <array>
          <data>
            <value>
              <boolean>0</boolean>
            </value>
            <value>
              No device(s) exists
            </value>
          </data>
        </array>
      </value>
    </param>
  </params>
</methodResponse>
```

4.5 redirect.checkDevice

You can check if a device(MAC) is registered to RPS.

4.5.1 XML-RPC signature:

```
redirect.checkDevice(mac)
```

4.5.2 Parameter:

mac :MAC address you want to check.

4.5.3 Return values:

1. Result code: true, false
2. Result text:
 - (1) **Error**: Invalid MAC→Invalid format of MAC address.
 - (2) **OK**:
 - 1) *Unregistered*→MAC has not been bound to a specific server.
 - 2) *Registered*→MAC has been bound to a specific server by you.
 - 3) *Registered Elsewhere*→ MAC has been registered by another RPS user.
 - 4) *Unknown*→MAC has not been added to RPS.

4.5.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
<methodName>redirect.checkDevice</methodName>
<params>
  <param>
    <value> <string> <![CDATA[001565000001]]> </string> </value>
  </param>
</params>
</methodCall>
```


4.5.5 Answer:

4.5.5.1 Check Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            Registered
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.5.5.2 Fail to Check:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:Invalid MAC(s)
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.6 redirect.registerDevices

You can register a list of devices to RPS by binding the MAC addresses and server name.

4.6.1 XML-RPC signature:

```
redirect.registerDevices(macList,serverName)
```

4.6.2 Parameter:

1. **macList**: MAC address(es) you want to register to RPS.
2. **serverName**: The server name you already added to RPS which the device should be redirected to.

4.6.3 Return values:

1. Result code: *True, False*
2. Result text:
 - (1) **OK**: The MAC address is registered successfully.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address
 - 2) *Error:Invalid server*→Invalid server name
 - 3) *Error*: The server name can only contain 'A-z 0-9 - _ .' (include space)→ The serverName contains illegal characters.
 - 4) *Error:Repeated MAC(s):00000000000d* →Repeated MAC
 - 5) *Are you sure you want to override MAC addresses.*→The MAC address has been registered by another RPS user, see more details on section 4.7 below.
 - 6) *Error: 001565000001(MAC address)*
 - Please check the MAC address you entered
 - If you need to continue adding this MAC, please create a ticket on <https://ticket.yealink.com>. We will review your complaint. You need to provide:
 - (1)Device MAC and SN
 - (2)Enterprise Name
 - (3)Firmware Version
 - (4)Login Account

→ MAC(001565000001) has been added to other enterprise.

4.6.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
<methodName>redirect.registerDevices</methodName>
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <string><![CDATA[001565000001]]></string>
          </value>
          <value>
            <string><![CDATA[001565000002]]></string>
          </value>
          ...
        </data>
      </array>
    </value>
  </param>
  <param>
    <value><string><![CDATA[serverName]]></string></value>
  </param>
</params>
</methodCall>
```

4.6.5 Answer:

4.6.5.1 Register Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            OK
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.6.5.2 Register Failed:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:Invalid MAC(s):00s1122334400, ...
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
```

4.7 redirect.registerDevices^[1]

If the MAC addresses you are registering have been registered by you before, you can confirm whether to override the previous registrations by `isOverride` parameter.

4.7.1 XML-RPC signature:

```
redirect.registerDevices(macList,serverName,isOverride)
```

4.7.2 Parameter:

1. **macList** :MAC addresses you want to register to RPS.
2. **serverName**: The server name you have already added to RPS which the device should be redirected to.
3. **isOverride**: Confirm whether to override the previous registration or not.
 - 1: Override
 - 0: Cancel, other values are illegal.

4.7.3 Return values:

1. Result code: *True, False*
2. Result text:
 - (1) **OK**: The MAC address is registered successfully and is bound to the specified server.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address(es)
 - 2) *Error:Invalid server*→Invalid server name
 - 3) *Error*: The server name can only contain 'A-z 0-9 - _ .' (include space)→ The serverName contains illegal characters.
 - 4) *Error:Repeated MAC(s):00000000000d* →Repeated MAC
 - 5) Only 0 or 1 can be used:1 to override, 0 to cancel→Invalid isOverride value
 - 6) MAC address(es) overriding failed→Fail to override the previous MAC address(es) and add the MAC.
 - 7) *Error: 001565000001(MAC address)*
 - Please check the MAC address you entered
 - If you need to continue adding this MAC, please create a ticket on <https://ticket.yealink.com>. We will review your complaint. You need to provide:
 - (1)Device MAC and SN
 - (2)Enterprise Name
 - (3)Firmware Version

(4)Login Account
→ MAC(001565000001) has been added to other enterprise.

[1] Compared with the last one, this API has an extra parameter - *isOverride*.

4.7.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
<methodName>redirect.registerDevices</methodName>
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <string><![CDATA[001565000001]]></string>
          </value>
          <value>
            <string><![CDATA[001565000002]]></string>
          </value>
          ...
        </data>
      </array>
    </value>
  </param>
  <param>
    <value><string><![CDATA[serverName]]></string></value>
  </param>
  <param>
    <value><string><![CDATA[0/1]]></string></value>
  </param>
</params>
</methodCall>
```

4.7.5 Answer:

4.7.5.1 Register Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            OK
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.7.5.2 Register Failed:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:Invalid MAC(s):00s1122334400
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.8 redirect.deRegisterDevices

You can remove a list of s devices from RPS by specifying these MAC addresses.

4.8.1 XML-RPC signature:

```
redirect.deRegisterDevices(macList)
```

4.8.2 Parameter:

macList :MAC addresses you want to remove from RPS

4.8.3 Return values:

1. Result code: True, False
2. Result text:
 - (1) **OK**: The MAC address is deregistered successfully.
 - (2) **Error**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address or the MAC address has not been registered by you.

4.8.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
<methodName>redirect.deRegisterDevices</methodName>
<params>
<param>
<value>
<array>
<data>
<value>
<string><![CDATA[001565000001]]></string>
</value>
<value>
<string><![CDATA[001565000002]]></string>
</value>
...
</data>
</array>
</value>
</param>
</params>
</methodCall>
```

4.8.5 Answer:

4.8.5.1 De-register Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            OK
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.8.5.2 De-register Failed:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:Invalid MAC(s):001122334401
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.9 redirect.addServer

You can add a configuration server that the device is redirected to.

4.9.1 XML-RPC signature:

```
redirect.addServer(serverName,serverUrl)
```

4.9.2 Parameter:

1. **serverName**: The server name you want to add to RPS which the device will be redirected to.
2. **serverUrl**: The URL of your configuration server.

4.9.3 Return values:

1. *Result code*: True, False
2. *Result text*:
 - (1) **OK**: The server has been added successfully.
 - (2) **Errors**:
 - 1) *The url can only begin with 'http://' or 'https://' or 'ftp://' or 'tftp://'* →Invalid format of URL
 - 2) *The server name can not be empty.*→The serverName should not be empty.
 - 3) *The server url can not be empty*→The serverUrl should not be empty.
 - 4) *The server name has been used*→The serverName has existed.
 - 5) *The server name can only contain 'A-z 0-9 - _ . '(include space)* →Invalid format of serverName.

4.9.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
<methodName>redirect.addServer</methodName>
<params>
  <param>
    <value>
      <string><![CDATA[serverName]]></string>
    </value>
  </param>
  <param>
    <value>
      <string><![CDATA[serverUrl]]></string>
    </value>
  </param>
</params>
</methodCall>
```

4.9.5 Answer:

4.9.5.1 Add Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            OK
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.9.5.2 Fail to Add:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:The server name has been used
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.10 redirect.editDevice

You can change (or specify) the configuration server for the device that is added to RPS.

4.10.1 XML-RPC signature:

```
redirect.editDevice(mac,serverName)
```

4.10.2 Parameter:

1. **mac** :The MAC address you want to change.
2. **serverName**: The server name you want to redirect the device to.

4.10.3 Return values:

1. Result code: True, False
2. Result text:
 - (1) **OK**: The configuration server is changed successfully.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→The format of MAC address is invalid or the MAC device does not belong to you.
 - 2) *Error: Invalid server*→Invalid server name
 - 3) *The MAC is unknown*→MAC has not been added to RPS.

4.10.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
<methodName>redirect.editDevice</methodName>
<params>
  <param>
    <value><string><![CDATA[00156500001]]></string></value>
  </param>
  <param>
    <value><string><![CDATA[serverName]]></string></value>
  </param>
</params>
</methodCall>
```

4.10.5 Answer:

4.10.5.1 Edit Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            OK
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.10.5.2 Fail to Edit:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:Invalid MAC(s):00s1122334400
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.11 redirect.editDevice^[1]

Edit the device and you can bind the device to another server.

This interface will no longer be available, please use the interface provided in section 4.10.

4.11.1 XML-RPC signature:

```
redirect.editDevice(mac,serverName,isOverride)
```

4.11.2 Parameter:

1. **mac**: The MAC address you want to change.
2. **serverName**: The server name you want to redirect the device to.
3. **isOverride**: The parameter is invalid.

4.11.3 Return values:

1. Result code: True, False
2. Result text:
 - (1) **OK**: The configuration server is changed successfully.
 - (2) **Errors**:
 - 1) Invalid MAC(s):mac→ The format of MAC address is invalid or the MAC device does not belong to you.
 - 2) Error: Invalid server→Invalid server name
 - 3) The MAC is unknown→MAC has not been added to RPS.
 - 4) Only 0 or 1 can be used:1 to override, 0 to cancel→Invalid isOverride value

[1] Compared with the last one, this API has an extra parameter - *isOverride*.

4.11.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
<methodName>redirect.editDevice</methodName>
<params>
  <param>
    <value><string><![CDATA[00156500001]]></string></value>
  </param>
  <param>
    <value><string><![CDATA[serverName]]></string></value>
  </param>
  <param>
    <value><string><![CDATA[0]]></string></value>
  </param>
</params>
</methodCall>
```

4.11.5 Answer:

4.11.5.1 Edit Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            OK
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.11.5.2 Fail to Edit:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:Invalid MAC(s):00s1122334400
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.12 redirect.registerDeviceWithUniqueUrl

You can register a device and specify the configuration server specified in `serverName` or `uniqueServerUrl` (the `uniqueServerUrl` takes precedence over the `serverName`).

4.12.1 XML-RPC signature:

`redirect.registerDeviceWithUniqueUrl (mac,serverName, uniqueServerUrl)`

4.12.2 Parameter:

1. **mac**: The MAC address you want to register to RPS.
2. **serverName**: The server name you want to added to RPS.
3. **uniqueServerUrl**: The server URL contains validation string. For example: `http://httpconfig.vonag.net/xxxttyy`.

4.12.3 Return values:

1. Result code: *True, False*
2. Result text:
 - (1) **OK**: The MAC address has been registered successfully to a specific server (specified in `uniqueServerUrl/serverName`).
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address
 - 2) *Error:Invalid server*→Invalid server name
 - 3) *Error*: The server name can only contain 'A-z 0-9 - _ .' (include space)→ The `serverName` contains illegal characters.
 - 4) *Invalid server url*→The value of `uniqueServerUrl` does not begin with `http/https/ftp/tftp`.
 - 5) *Are you sure you want to override MAC addresses*: →The MAC address has been registered by another RPS user.
 - 6) *Error: 001565000001(MAC address)*
 - Please check the MAC address you entered
 - If you need to continue adding this MAC, please create a ticket on <https://ticket.yealink.com>. We will review your complaint. You need to provide:
 - (1)Device MAC and SN
 - (2)Enterprise Name
 - (3)Firmware Version
 - (4)Login Account
 → `MAC(001565000001)` has been added to other enterprise.

4.12.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
  <methodName>redirect.registerDeviceWithUniqueUrl</methodName>
  <params>
    <param>
      <value><string><![CDATA[1A-6F-65-4D-83-9B]]></string></value>
    </param>
    <param>
      <value><string><![CDATA[serverName]]></string></value>
    </param>
    <param>
      <value><string><![CDATA[http://httpconfig.vonag.net/xxtty]]></string></value>
    </param>
  </params>
</methodCall>
```

4.12.5 Answer:

4.12.5.1 Register Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            OK
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.12.5.2 Register Failed:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:Invalid MAC(s):00s1122334400
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.13 redirect.editDeviceWithUniqueUrl

You can change (or specify) the configuration server for a device in the parameter `serverName` or `uniqueServerUrl` (the `uniqueServerUrl` takes precedence over the `serverName`).

4.13.1 XML-RPC signature:

`redirect.editDeviceWithUniqueUrl (mac,serverName, uniqueServerUrl)`

4.13.2 Parameter:

1. **mac**: The MAC address you want to change.
2. **serverName**: The server name you want to redirect the device to.
3. **uniqueServerUrl**: The server URL contains validation string. For example: `http://httpconfig.vonag.net/xxxttyy`.

4.13.3 Return values:

1. Result code: *True, False*
2. Result text:
 - (1) **OK**: The configuration server is changed successfully.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac* → The format of MAC address is invalid or the MAC device does not belongs to you.
 - 2) *Invalid server* → Invalid server name
 - 3) *The MAC is unknown* → MAC has not been added to RPS.
 - 4) *Invalid server url* → The value of `uniqueServerUrl` does not begin with `http/https/ftp/tftp`.

4.13.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
  <methodName>redirect.editDeviceWithUniqueUrl</methodName>
  <params>
    <param>
      <value><string><![CDATA[1A-6F-65-4D-83-9B]]></string></value>
    </param>
    <param>
      <value><string><![CDATA[serverName]]></string></value>
    </param>
    <param>
      <value><string><![CDATA[http://httpconfig.vonag.net/xxtty]]></string></value>
    </param>
  </params>
</methodCall>
```

4.13.5 Answer:

4.13.5.1 Edit Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            OK
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.13.5.2 Fail to Edit:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:Invalid MAC(s):00s1122334400
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.14 redirect.checkDeviceExt

You can check if a device (MAC) is registered to RPS. If the registration is successful, return server URL that the device is redirected to.

4.14.1 XML-RPC signature:

```
redirect.checkDeviceExt(mac)
```

4.14.2 Parameter:

mac :The MAC address you want to check.

4.14.3 Return values:

1. Result code: *True, False*
2. Result text:
 - (1) **Error**: Invalid MAC(s)→Invalid format of MAC address(es).
 - (2) **OK**:
 - 1) *Unregistered*→MAC has not been bound to a specific server.
 - 2) *Registered*→MAC has been bound to a specific server.
 - 3) *Registered Elsewhere*→MAC has been registered by another RPS user.
 - 4) *Unknown*→MAC has not been added to RPS.

4.14.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
  <methodName>redirect.checkDeviceExt</methodName>
  <params>
    <param>
      <value><string><![CDATA[001122334400]]></string></value>
    </param>
  </params>
</methodCall>
```

4.14.5 Answer:

4.14.5.1 Check Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>1</boolean>
          </value>
          <value>
            Registered
          </value>
          <value>
            http://www.test.com
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.14.5.2 Fail to Check:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
  <param>
    <value>
      <array>
        <data>
          <value>
            <boolean>0</boolean>
          </value>
          <value>
            Error:Invalid MAC(s)
          </value>
        </data>
      </array>
    </value>
  </param>
</params>
</methodResponse>
```

4.15 redirect.registerDeviceWithUniqueUrl[1]

You can register a device and specify the configuration server specified in `serverName` or `uniqueServerUrl` (the `uniqueServerUrl` takes precedence over the `serverName`). If you have registered the device before, you can confirm whether to override the added device by the parameter `isOverride`.

4.15.1 XML-RPC signature:

```
redirect.registerDeviceWithUniqueUrl(mac,serverName,
uniqueServerUrl,isOverride)
```

4.15.2 Parameter:

1. **mac**: The MAC address you want to register.
2. **serverName**: The server name you want to added to RPS.
3. **uniqueServerUrl**: The server URL contains validation string. For example: `http://httpconfig.vonag.net/xxxttyy`.
4. **isOverride**: Confirm whether to override the previous registration or not.
 - 1: Override
 - 0: Cancel, other values are illegal.

4.15.3 Return values:

1. Result code: *True, False*
2. Result text:
 - (1) **OK**: The configuration server is changed successfully.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→ Invalid format of MAC address
 - 2) *Invalid server*→Invalid server name
 - 3) *Error*: The server name can only contain 'A-z 0-9 - _ .' (include space)→ The `serverName` contains illegal characters.
 - 4) *Invalid server url*→The value of `uniqueServerUrl` does not begin with `http/https/ftp/tftp`.
 - 5) Only 0 or 1 can be used:1 to override, 0 to cancel→Invalid `isOverride` value
 - 6) *MAC address(es) overriding failed*→Fail to override the previous MAC address(es) and add the MAC.
 - 7) *Error: 001565000001(MAC address)*
 - Please check the MAC address you entered
 - If you need to continue adding this MAC, please create a ticket on <https://ticket.yealink.com>.

We will review your complaint. You need to provide:

(1)Device MAC and SN

(2)Enterprise Name

(3)Firmware Version

(4)Login Account

→MAC has been added to other enterprise.

[1] Compared with the last one, this API has an extra parameter - *isOverride*.

4.15.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
<methodName>redirect.registerDeviceWithUniqueUrl</methodName>
<params>
<param>
<value><string><![CDATA[1A-6F-65-4D-83-9B]]></string></value>
</param>
<param>
<value><string><![CDATA[serverName]]></string></value>
</param>
<param>
<value><string><![CDATA[http://httpconfig.vonag.net/xxtty]]></string></value>
</param>
<param>
<value><string><![CDATA[0]]></string></value>
</param>
</params>
</methodCall>
```

4.15.5 Answer:

4.15.5.1 Register Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
  <params>
    <param>
      <value>
        <array>
          <data>
            <value>
              <boolean>1</boolean>
            </value>
            <value>
              OK
            </value>
          </data>
        </array>
      </value>
    </param>
  </params>
</methodResponse> </params>
</methodResponse>
```

4.15.5.2 Register Failed:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
  <params>
    <param>
      <value>
        <array>
          <data>
            <value>
              <boolean>0</boolean>
            </value>
            <value>
              Error:Invalid MAC(s):00s1122334400
            </value>
          </data>
        </array>
      </value>
    </param>
  </params>
</methodResponse>
```

4.16 redirect.editDeviceWithUniqueUrl[1]

You can change (or specify) the configuration server for a device in the parameter `serverName` or `uniqueServerUrl` (the `uniqueServerUrl` takes precedence over the `serverName`).

This interface will no longer be available, please use the interface provided in section 4.13.

4.16.1 XML-RPC signature:

```
redirect.editDeviceWithUniqueUrl(mac,serverName,uniqueServerUrl,  
isOverride)
```

4.16.2 Parameter:

1. **mac**: The MAC address you want to register.
2. **serverName**: The server name you want to add to RPS.
3. **uniqueServerUrl**: The server URL contains validation string. For example: `http://httpconfig.vonag.net/xxxttyy`.
4. **isOverride**: Confirm whether to override the previous registration or not.
1: Override
0: Cancel, other values are illegal.
The parameter is invalid for the current version.

4.16.3 Return values:

1. Result code: *True, False*
2. Result text:
 - (1) **OK**: The configuration server is changed successfully.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac* → The format of MAC address is invalid or the MAC device does not belong to you.
 - 2) *Invalid server* → Invalid server name
 - 3) *The MAC is unknown* → MAC has not been added to RPS.
 - 4) *Invalid server url* → The value of `uniqueServerUrl` does not begin with `http/https/ftp/tftp`.

[1] Compared with the last one, this API has an extra parameter - *isOverride*.

4.16.4 Request:

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
<methodName>redirect.editDeviceWithUniqueUrl</methodName>
<params>
<param>
<value><string><![CDATA[1A-6F-65-4D-83-9B]]></string></value>
</param>
<param>
<value><string><![CDATA[serverName]]></string></value>
</param>
<param>
<value><string><![CDATA[http://httpconfig.vonag.net/xxxtty]]></string></value>
</param>
</params>
</methodCall>
```

4.16.5 Answer:

4.16.5.1 Edit Successfully:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
  <params>
    <param>
      <value>
        <array>
          <data>
            <value>
              <boolean>1</boolean>
            </value>
            <value>
              OK
            </value>
          </data>
        </array>
      </value>
    </param>
  </params>
</methodResponse>
```

4.16.5.2 Fail to Edit:

```
<?xml version="1.0" encoding="UTF-8"?>
<methodResponse
xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
  <params>
    <param>
      <value>
        <array>
          <data>
            <value>
              <boolean>0</boolean>
            </value>
            <value>
              Error:Invalid MAC(s):00s1122334400
            </value>
          </data>
        </array>
      </value>
    </param>
  </params>
</methodResponse>
```


About Yealink

Yealink (Stock Code: 300628) is a global leading unified communication (UC) terminal solution provider that primarily offers video conferencing systems and voice communication solutions. Founded in 2001, Yealink leverages its independent research and development and innovation to pursue its core mission: "Easy collaboration, high productivity." The company's high-quality UC terminal solutions enhance the work efficiency and competitive advantages of its customers in over 100 countries. Yealink is the world's second-largest SIP phone provider and is number one in the China market.



Contact us

YEALINK(XIAMEN) NETWORK TECHNOLOGY CO.,LTD.

Addr: 309, 3th Floor, No.16, Yun Ding North Road, Huli
District, Xiamen City, Fujian, P.R. China

Tel: 86-592-570-2000

www.yealink.com