

XML-API Development Guide

Yealink Redirection and Provisioning Service (RPS)



December 2011

Contents

| | |
|--------------------------------------------------------|-----------|
| 1. Protocol | 4 |
| 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: | 7 |
| 4.2 redirect.registerDevice^[2] | 10 |
| 4.2.1 XML-RPC signature: | 10 |
| 4.2.2 Parameter: | 10 |
| 4.2.3 Return values: | 10 |
| 4.2.4 Request: | 11 |
| 4.2.5 Answer: | 12 |
| 4.3 redirect.deRegisterDevice | 14 |
| 4.3.1 XML-RPCsignature: | 14 |
| 4.3.2 Parameter: | 14 |
| 4.3.3 Return values: | 14 |
| 4.3.4 Request: | 14 |
| 4.3.5 Answer: | 15 |
| 4.4 redirect.listDevices | 17 |
| 4.4.1 XML-RPCsignature: | 17 |
| 4.4.2 Return values: | 17 |
| 4.4.3 Request: | 17 |
| 4.4.4 Answer: | 18 |
| 4.5 redirect.checkDevice | 20 |
| 4.5.1 XML-RPC signature: | 20 |
| 4.5.2 Parameter: | 20 |
| 4.5.3 Return values: | 20 |
| 4.5.4 Request: | 20 |
| 4.5.5 Answer: | 21 |
| 4.6 redirect.registerDevices | 23 |
| 4.6.1 XML-RPCsignature: | 23 |
| 4.6.2 Parameter: | 23 |
| 4.6.3 Return values: | 23 |
| 4.6.4 Request: | 24 |

| | |
|---------------------------------------------------------|-----------|
| 4.6.5 Answer:..... | 25 |
| 4.7 redirect.registerDevices^[2] | 27 |
| 4.7.1 XML-RPCsignature: | 27 |
| 4.7.2 Parameter: | 27 |
| 4.7.3 Return values: | 27 |
| 4.7.4 Request:..... | 28 |
| 4.7.5 Answer:..... | 29 |
| 4.8 redirect.deRegisterDevices | 31 |
| 4.8.1 XML-RPCsignature: | 31 |
| 4.8.2 Parameter: | 31 |
| 4.8.4 Request:..... | 32 |
| 4.8.5 Answer:..... | 33 |
| 4.9 redirect.addServer | 35 |
| 4.9.1 XML-RPCsignature: | 35 |
| 4.9.2 Parameter: | 35 |
| 4.9.3 Return values: | 35 |
| 4.9.4 Request:..... | 36 |
| 4.9.5 Answer:..... | 37 |
| 4.10 redirect.editDevice | 39 |
| 4.10.1 XML-RPCsignature: | 39 |
| 4.10.2 Parameter: | 39 |
| 4.10.3 Return values: | 39 |
| 4.10.4 Request:..... | 40 |
| 4.10.5 Answer:..... | 41 |
| 4.11 redirect.editDevice^[2] | 43 |
| 4.11.1 XML-RPCsignature: | 43 |
| 4.11.2 Parameter: | 43 |
| 4.11.3 Return values: | 43 |
| 4.11.4 Request:..... | 44 |
| 4.11.5 Answer:..... | 45 |
| About Yealink | 47 |

Document Revision History

| Release | Version | Reason for Change | Date | Author |
|---------|---------|-------------------|--------------|----------|
| 1.0.2 | 1.0 | | Dec 1, 2011 | Mike Hou |
| 1.0.5 | 1.1 | | Feb 1, 2012 | Mike Hou |
| 1.0.10 | 1.2 | | Apr 1, 2012 | Mike Hou |
| 1.0.53 | 1.3 | | Apr 23, 2012 | Mike Hou |

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 specified further down 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://rps.yealink.com/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 redirect a specific device to a different configuration server by its MAC ID.

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 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**: If the MAC address has been registered successfully to a specific server.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address
 - 2) *Existing MAC(s):mac*→The MAC address has been registered by yourself.
 - 3) *Error:Invalid server*→Invalid server name
 - 4) *Are you sure you want to override MAC addresses:* →The MAC address has been registered by another RPS user, see more details on session 4.2 below.

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>
```

Notes:

1. "<![CDATA[]>" is not necessary, but with it you can enter special characters, such as "&,\$,#,@,<,>" and so on.
2. Four types of valid MAC address format are 001565000001, 00-15-65-00-00-01, 00 15 65 00 00 01, 00:15:65:00:00:00:01.

4.1.5 Answer:

4.1.5.1 Positive Answer:

```
<?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 Negative Answer:

```
<?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^[2]

If the MAC address you are registering has been registered by another RPS user, 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 already added to RPS which the device should be redirected to.
3. **isOverride**: Confirm whether to override the previous registration or not.
 - (1) *1*: Override confirmation
 - (2) *0*: Cancel registration

4.2.3 Return values:

1. Result code: *True*, *False*
2. Result text:
 - (1) **OK**: If the MAC address has been registered successfully to a specific server.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address
 - 2) *Existing MAC(s):mac* →The MAC address has been registered by yourself.
 - 3) *Error:Invalid server*→Invalid server name
 - 4) *Only 0 or 1 can be used:1 to override, 0 to cancel*→Invalid *isOverride* value
 - 5) *MAC address(es) overriding failed*→Failed to override the previous MAC address

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 Positive Answer:

```
<?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 Negative Answer:

```
<?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 deregistering its MAC address.

4.3.1 XML-RPC signature:

redirect.deRegisterDevice(mac)

4.3.2 Parameter:

mac :MAC address you would like to remove from RPS

4.3.3 Return values:

1. Result code: True, False
2. Result text:
 - (1) **OK**: If the MAC address has been deregistered 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 Positive Answer:

```
<?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 Negative Answer:

```
<?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 devices found*→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 Filled Answer:

```
<?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 Empty Answer:

```
<?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(MACs) is registered toRPS.

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* → Not assigned to a specific server.
 - 2) *Registered* →MAC has been assigned to a specific server.
 - 3) *Registered Elsewhere*→ MAC has been registered by another RPS user.
 - 4) *Unknown*→ MAC has not been added to RPS.

4.5.4 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 Positive Answer:

```
<?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 Negative Answer:

```
<?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 a different configuration server by MAC IDs.

4.6.1 XML-RPC signature:

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

4.6.2 Parameter:

1. **mac** : MAC address 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**: If the MAC address has been registered successfully to a specific server.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address
 - 2) *Existing MAC(s):mac* →The MAC address has been registered by yourself.
 - 3) *Error:Invalid server*→Invalid server name
 - 4) *Are you sure you want to override MAC addresses:* →The MAC address has been registered by another RPS user,see more details on session 4.2 above..

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 Positive Answer:

```
<?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 Negative Answer:

```
<?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>
```

4.7 redirect.registerDevices^[2]

If the MAC addresses you are registering have been registered by other RPS users, 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 already added to RPS which the device should be redirected to.
3. **isOverride**: Confirm whether to override the previous registration or not.
 - (1) *1*: Override confirmation
 - (2) *0*: Cancel registration

4.7.3 Return values:

1. Result code: *True*, *False*
2. Result text:
 - (1) **OK**: If the MAC address has been registered successfully to a specific server.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address
 - 2) *Existing MAC(s):mac* →The MAC address has been registered by yourself.
 - 3) *Error:Invalid server*→Invalid server name
 - 4) Only 0 or 1 can be used:1 to override, 0 to cancel→Invalid *isOverride* value
 - 5) MAC address(es) overriding failed→Failed to override the previous MAC address

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 Positive Answer:

```
<?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 Negative Answer:

```
<?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 deregistering these MAC addresses.

4.8.1 XML-RPC signature:

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

4.8.2 Parameter:

macList :MAC addresses you would like to remove from RPS

4.8.3 Return values:

1. Result code: True, False
2. Result text:
 - (1) **OK**: If the MAC address has been deregistered successfully.
 - (2) **Error**:
 - 1) *Invalid MAC(s):mac*→Inalid 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 Positive Answer:

```
<?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 Negative Answer:

```
<?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 different configuration server which is used to redirected devices to.

4.9.1 XML-RPC signature:

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

4.9.2 Parameter:

1. **serverName**: The server name you already added to RPS which the device will be redirected to.
2. **serverUrl**: The specific 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 been used.
 - 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 Positive Answer:

```
<?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 Negative Answer:

```
<?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 the configuration server that the device(s) has registered to to another server.

4.10.1 XML-RPC signature:

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

4.10.2 Parameter:

1. **mac** :The MAC address you would like to edit.
2. **serverName**: The server name you would like to re-register the MAC to.

4.10.3 Return values:

1. Result code: *True, False*
2. Result text:
 - (1) **OK**: If the MAC address has been registered successfully to a specific server.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address
 - 2) *Error:Invalid server*→Invalid server name
 - 3) *The MAC is unknown* → *MAC has not been registered*
 - 4) *Are you sure you want to override MAC addresses:* →The MAC address has been registered by another RPS user,see more details on session 4.2 below.

4.10.4 Request:

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


4.10.5 Answer:

4.10.5.1 Positive Answer:

```
<?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 Negative Answer:

```
<?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^[2]

If the MAC address you are registering has been registered by another RPS user, you can confirm whether to override the previous registration by *isOverride* parameter.

4.11.1 XML-RPC signature:

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

4.11.2 Parameter:

1. **mac** :The MAC address you would like to edit.
2. **serverName**: The server name you would like to re-register the MAC to.
3. **isOverride**: Confirm whether to override the previous registration or not.
 - (1) 1: Override confirmation
 - (2) 0: Cancel registration

4.11.3 Return values:

1. Result code: *True*, *False*
2. Result text:
 - (1) **OK**: If the MAC address has been registered successfully to a specific server.
 - (2) **Errors**:
 - 1) *Invalid MAC(s):mac*→Invalid format of MAC address
 - 2) *The MAC is unknown* → *MAC has not been registered*
 - 3) *Error:Invalid server*→Invalid server name
 - 4) Only 0 or 1 can be used:1 to override, 0 to cancel→Invalid isOverride value
 - 5) MAC address(es) overriding failed→Failed to override the previous MAC address

4.11.4 Request:

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

4.11.5 Answer:

4.11.5.1 Positive Answer:

```
<?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 Negative Answer:

```
<?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

Founded in 2001, Yealink is the professional designer and manufacturer of IP Phone and IP Video Phone for the world-wide broadband telephony market. For more than ten years, Yealink has focused on VoIP products characterized by high cost-performance, easy deployment and affordable price. Yealink benefits customers from flexibility and broad interoperability with the major IP-PBX, softswitch and IMS in the market. Chosen by famous carriers, enterprises and clients from over 80 countries, Yealink provides SMB and diversified customers from all over the world a best choice to extend the value of network communication.



Contact us

Yealink Network Technology Co., Ltd.

Addr: 4th-5th Floor, South Building, No.63 Wanghai Road,
2nd Software Park, Xiamen, China (361008).

Tel: 86-592-570-2000

www.yealink.com