

# OpenFire

- [Unified presence](#)
- [Get presence information](#)
- [Get XMPPU user name](#)
- [Get presence state](#)
- [Set presence state](#)
- [Set presence status](#)
- [Get presence status](#)
- [Ping](#)
- [Register presence monitor](#)

## Unified presence

XML-RPC presence java clients provided by sipxecs (sipXopenfire project) can be used using the SHARED\_SECRET

NOTE: Calls are secured using BASIC Authentication

Example using curl CLI tool to get presence information

## Get presence information

**Method name:** getUnifiedPresenceInfo

**Description:** This method returns XMPP information being given the sip extension, as described below.

Step 1. Create a file called *get-presence.xml* with content, replace 200 with the SIP user ID you wish to get presence information on:

```
<methodCall>
<methodName>presenceServer.getUnifiedPresenceInfo</methodName>
<params>
<param>
<value>
<string>2011</string>
</value>
</param>
</params>
</methodCall>
```

Step 2. Then run this command, replacing *mypbx.example.com* with the *FQDN* of your sipXecs system:

```
curl {color}
-H "Content-Type: application/xml" {color}
-X POST {color}
--data-binary @get-presence.xml {color}
http://{user}:<password>@mypbx.example.com:9094/plugins/sipx-openfire-presence/status
```

Response example:

```
<methodResponse xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions">
<params>
<param>
<value>
<struct>
<member>
<name>custom-presence-message</name>
<value></value>
</member>
<member>
<name>sip-id</name>
<value>2011</value>
</member>
<member>
<name>status-code</name>
<value>ok</value>
</member>
<member>
<name>unified-presence</name>
<value>available-for-both</value>
</member>
<member>
<name>jabber-id</name>
<value>mirceac@openuc.ezuze.com</value>
</member>
<member>
<name>xmpp-presence</name>
<value>AVAILABLE</value>
</member>
<member>
<name>sip-presence</name>
<value>IDLE</value>
</member>
</struct>
</value>
</param>
</params>
</methodResponse>
```

## Get XMPPU user name

**Method name:** getXmppUsername

**Description:** This method returns the xmpp username being given the email address saved in user-portal: contact information page.

<b>XML Request</b>	<pre>&lt;methodCall&gt; &lt;methodName&gt;presenceServer.getXmppUsername&lt;/methodName&gt; &lt;params&gt; &lt;param&gt; &lt;value&gt; &lt;string&gt;mirceac@ezuze.com&lt;/string&gt; &lt;/value&gt; &lt;/param&gt; &lt;/params&gt; &lt;/methodCall&gt;</pre>
--------------------	---

<b>Response</b>	<pre> &lt;methodResponse xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions"&gt;   &lt;params&gt;     &lt;param&gt;       &lt;value&gt;         &lt;struct&gt;           &lt;member&gt;             &lt;name&gt;sip-id&lt;/name&gt;             &lt;value&gt;mirceac&lt;/value&gt;           &lt;/member&gt;           &lt;member&gt;             &lt;name&gt;status-code&lt;/name&gt;             &lt;value&gt;ok&lt;/value&gt;           &lt;/member&gt;         &lt;/struct&gt;       &lt;/value&gt;     &lt;/param&gt;   &lt;/params&gt; </pre>
-----------------	--

## Get presence state

**Method name:** getPresenceState

**Description:** This method returns the presence state being given the XMPP ID. The string representation of the presence might be: AVAILABLE, AWAY, BUSY, OFFLINE, CHAT, EXTENDED\_AWAY.

<b>XML Request</b>	<pre> &lt;methodCall&gt;   &lt;methodName&gt;presenceServer.getPresenceState&lt;/methodName&gt;   &lt;params&gt;     &lt;param&gt;       &lt;value&gt;         &lt;string&gt;mirceac&lt;/string&gt;       &lt;/value&gt;     &lt;/param&gt;   &lt;/params&gt; </pre>
<b>Response</b>	<pre> &lt;methodResponse xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions"&gt;   &lt;params&gt;     &lt;param&gt;       &lt;value&gt;         &lt;struct&gt;           &lt;member&gt;             &lt;name&gt;status-code&lt;/name&gt;             &lt;value&gt;ok&lt;/value&gt;           &lt;/member&gt;           &lt;member&gt;             &lt;name&gt;xmpp-presence&lt;/name&gt;             &lt;value&gt;AVAILABLE&lt;/value&gt;           &lt;/member&gt;         &lt;/struct&gt;       &lt;/value&gt;     &lt;/param&gt;   &lt;/params&gt; &lt;/methodResponse&gt; &lt;/methodCall&gt; </pre>

## Set presence state

**Method name:** setPresenceState

**Description:** This method changes the presence state of the given XMPP user id. The xmppPresenceAsString parameter can have the following String representations: AVAILABLE, AWAY, BUSY, OFFLINE, CHAT, EXTENDED\_AWAY.

<b>XML Request</b>	? (String xmppUsername, String xmppPresenceAsString)? XML to send: <pre>&lt;methodCall&gt; &lt;methodName&gt;presenceServer.setPresenceState&lt;/methodName&gt; &lt;params&gt; &lt;param&gt; &lt;value&gt; &lt;string&gt;mirceac&lt;/string&gt; &lt;/value&gt; &lt;/param&gt; &lt;param&gt; &lt;value&gt; &lt;string&gt;AWAY&lt;/string&gt; &lt;/value&gt; &lt;/param&gt; &lt;/params&gt; &lt;/methodCall&gt;</pre>
<b>Response</b>	? (String xmppUsername, String xmppPresenceAsString)? XML to send: <pre>&lt;methodCall&gt; &lt;methodName&gt;presenceServer.setPresenceState&lt;/methodName&gt; &lt;params&gt; &lt;param&gt; &lt;value&gt; &lt;string&gt;mirceac&lt;/string&gt; &lt;/value&gt; &lt;/param&gt; &lt;param&gt; &lt;value&gt; &lt;string&gt;AWAY&lt;/string&gt; &lt;/value&gt; &lt;/param&gt; &lt;/params&gt; &lt;/methodCall&gt;</pre>

If you now call the `getPresenceState` method it will show you AWAY as xmpp-presence value

## Set presence status

**Method name:** `setPresenceStatus`

**Description:** This method sets the given status message of the given XMPP user ID.  
(String id, String status)

<b>XML Request</b>	<pre>&lt;methodCall&gt; &lt;methodName&gt;presenceServer.setPresenceStatus&lt;/methodName&gt; &lt;params&gt; &lt;param&gt; &lt;value&gt; &lt;string&gt;mirceac&lt;/string&gt; &lt;/value&gt; &lt;/param&gt; &lt;param&gt; &lt;value&gt; &lt;string&gt;Here&lt;/string&gt; &lt;/value&gt; &lt;/param&gt; &lt;/params&gt; &lt;/methodCall&gt;</pre>
<b>Response</b>	<pre>&lt;methodResponse xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions"&gt; &lt;params&gt; &lt;param&gt; &lt;value&gt; &lt;struct&gt; &lt;member&gt; &lt;name&gt;status-code&lt;/name&gt; &lt;value&gt;ok&lt;/value&gt; &lt;/member&gt; &lt;/struct&gt; &lt;/value&gt; &lt;/param&gt; &lt;/params&gt; &lt;/methodResponse&gt;</pre>

# Get presence status

**Method name:** getPresenceStatus

**Description:** This method returns the presence status message of the given XMPP user ID.  
public Map getPresenceStatus(String id)

<b>XML Request</b>	<pre>&lt;methodCall&gt; &lt;methodName&gt;presenceServer.getPresenceStatus&lt;/methodName&gt; &lt;params&gt; &lt;param&gt; &lt;value&gt; &lt;string&gt;mirceac&lt;/string&gt; &lt;/value&gt; &lt;/param&gt; &lt;/params&gt; &lt;/methodCall&gt;</pre>
<b>Response</b>	<pre>&lt;methodResponse xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions"&gt; &lt;params&gt; &lt;param&gt; &lt;value&gt; &lt;struct&gt; &lt;member&gt; &lt;name&gt;custom-presence-message&lt;/name&gt; &lt;value&gt;Here&lt;/value&gt; &lt;/member&gt; &lt;member&gt; &lt;name&gt;status-code&lt;/name&gt; &lt;value&gt;ok&lt;/value&gt; &lt;/member&gt; &lt;/struct&gt; &lt;/value&gt; &lt;/param&gt; &lt;/params&gt; &lt;/methodResponse&gt;</pre>

# Ping

**Method name:** setPresenceState

**Description:** This method verifies if the XMPP server is running.  
(String originatorName)

<b>XML Request</b>	<pre>&lt;methodCall&gt; &lt;methodName&gt;presenceServer.ping&lt;/methodName&gt; &lt;params&gt; &lt;param&gt; &lt;value&gt; &lt;string&gt;mirceac&lt;/string&gt; &lt;/value&gt; &lt;/param&gt; &lt;/params&gt; &lt;/methodCall&gt;</pre>
--------------------	--

<b>Response</b>	<pre> &lt;methodResponse xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions"&gt;   &lt;params&gt;     &lt;param&gt;       &lt;value&gt;         &lt;struct&gt;           &lt;member&gt;             &lt;name&gt;status-code&lt;/name&gt;             &lt;value&gt;ok&lt;/value&gt;           &lt;/member&gt;           &lt;member&gt;             &lt;name&gt;instance-handle&lt;/name&gt;             &lt;value&gt;of:821711375117195347&lt;/value&gt;           &lt;/member&gt;         &lt;/struct&gt;       &lt;/value&gt;     &lt;/param&gt;   &lt;/params&gt; &lt;/methodResponse&gt; </pre>
-----------------	---

## Register presence monitor

**Method name:** registerPresenceMonitor

**Description:** This method registers a listener to be notified when the presence changes.  
(String protocol, String serverUrl )

<b>XML Request</b>	<pre> ? (String xmppUsername, String xmppPresenceAsString)? XML to send: &lt;methodCall&gt;   &lt;methodName&gt;presenceServer.setPresenceState&lt;/methodName&gt;   &lt;params&gt;     &lt;param&gt;       &lt;value&gt;         &lt;string&gt;mirceac&lt;/string&gt;       &lt;/value&gt;     &lt;/param&gt;     &lt;param&gt;       &lt;value&gt;         &lt;string&gt;AWAY&lt;/string&gt;       &lt;/value&gt;     &lt;/param&gt;   &lt;/params&gt; &lt;/methodCall&gt; </pre>
<b>Response</b>	<pre> ? (String xmppUsername, String xmppPresenceAsString)? XML to send: &lt;methodCall&gt;   &lt;methodName&gt;presenceServer.setPresenceState&lt;/methodName&gt;   &lt;params&gt;     &lt;param&gt;       &lt;value&gt;         &lt;string&gt;mirceac&lt;/string&gt;       &lt;/value&gt;     &lt;/param&gt;     &lt;param&gt;       &lt;value&gt;         &lt;string&gt;AWAY&lt;/string&gt;       &lt;/value&gt;     &lt;/param&gt;   &lt;/params&gt; &lt;/methodCall&gt; </pre>