



XMPP

XEP-0018: Invisible Presence

Ben Schumacher
<mailto:ben@blahr.com>
<xmpp:rynok@jabber.com>

Jeremie Miller
<mailto:jer@jabber.org>
<xmpp:jer@jabber.org>

2003-09-26
Version 0.2

| Status | Type | Short Name |
|----------|---------------|------------|
| Rejected | Informational | N/A |

Documentation of invisible presence.

Legal

Copyright

This XMPP Extension Protocol is copyright © 1999 - 2014 by the [XMPP Standards Foundation](#) (XSF).

Permissions

Permission is hereby granted, free of charge, to any person obtaining a copy of this specification (the "Specification"), to make use of the Specification without restriction, including without limitation the rights to implement the Specification in a software program, deploy the Specification in a network service, and copy, modify, merge, publish, translate, distribute, sublicense, or sell copies of the Specification, and to permit persons to whom the Specification is furnished to do so, subject to the condition that the foregoing copyright notice and this permission notice shall be included in all copies or substantial portions of the Specification. Unless separate permission is granted, modified works that are redistributed shall not contain misleading information regarding the authors, title, number, or publisher of the Specification, and shall not claim endorsement of the modified works by the authors, any organization or project to which the authors belong, or the XMPP Standards Foundation.

Warranty

NOTE WELL: This Specification is provided on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE.

Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall the XMPP Standards Foundation or any author of this Specification be liable for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising from, out of, or in connection with the Specification or the implementation, deployment, or other use of the Specification (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if the XMPP Standards Foundation or such author has been advised of the possibility of such damages.

Conformance

This XMPP Extension Protocol has been contributed in full conformance with the XSF's Intellectual Property Rights Policy (a copy of which can be found at <http://xmpp.org/about-xmpp/xsf/xsf-ipr-policy/> or obtained by writing to XMPP Standards Foundation, 1899 Wynkoop Street, Suite 600, Denver, CO 80202 USA).

Contents

| | | |
|----------|--|----------|
| 1 | Introduction | 1 |
| 2 | Requirements | 1 |
| 3 | Use Cases | 1 |
| 3.1 | User Becomes Invisible to All Contacts | 1 |
| 3.2 | User Becomes Visible to All Contacts | 2 |
| 3.3 | User Becomes Invisible to Selected Contact | 2 |
| 3.4 | User Becomes Visible to Selected Contact | 2 |
| 4 | Server Implementation Details | 2 |
| 4.1 | Undirected Presence While Invisible to Selected Contacts | 3 |
| 5 | Security Considerations | 5 |
| 6 | IANA Considerations | 5 |
| 7 | XMPP Registrar Considerations | 5 |

1 Introduction

Jabber users have long wanted the ability to come online using Jabber, but appear "unavailable" to a subset of their roster, an ability which is commonly referred to as invisible mode. While several current implementations have various ways of providing this functionality, the systems currently in place often lack clear behavior and documentation. This document provides requirements for invisible mode and clarifies the existing protocol-in-use to provide the desired functionality.

Note well: The functionality described herein can and should be implemented by means of the Privacy Rules namespace defined in [XMPP IM](#)¹. This document is provided for historical purposes only.

2 Requirements

The requirements for invisible mode are basic:

1. A user should be able to become visible or invisible at any time.
2. When in either mode, users should be able to selectively allow or block presence updates to be distributed to their contacts.

3 Use Cases

For the purposes of this discussion, we will assume this terminology.

- Undirected Presence - A presence stanza without a "to" attribute.
- Directed Presence - A presence stanza with a "to" attribute.

3.1 User Becomes Invisible to All Contacts

Assuming the user's client has authenticated with the system normally, it would send the following presence stanza to become invisible (note: this can be the initial presence stanza sent by the user). To do so, the client will send undirected presence with a type="invisible" attribute.

Listing 1: Invisible presence stanza

```
<presence type="invisible"/>
```

¹RFC 6121: Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence <<http://tools.ietf.org/html/rfc6121>>.

3.2 User Becomes Visible to All Contacts

If a user chooses to become visible after being invisible, the client will send undirected presence with a type="visible" attribute.

Listing 2: Visible presence stanza

```
<presence type="visible"/>
```

3.3 User Becomes Invisible to Selected Contact

If a user is visible to all of their contacts, and wishes to become invisible to selected contacts on their roster, then their client will send the following directed presence stanza.

Listing 3: User becomes invisible to joe@foo.com

```
<presence to="joe@foo.com" type="invisible"/>
```

In this case, the server begins to track that the user is invisible to the contact joe@foo.com, and does not forward further presence updates to joe@foo.com. If the user wishes joe@foo.com to receive further presence updates, then their client must send either an undirected presence stanza with a type="visible" attribute (see Example 2), or a directed presence stanza to joe@foo.com with a type="visible" attribute as below.

Listing 4: User becomes visible to joe@foo.com

```
<presence to="joe@foo.com" type="visible"/>
```

3.4 User Becomes Visible to Selected Contact

The opposite behavior of the previous section should, also, be present in the server. For example, if a user is invisible to all of their contacts, they should be able to become visible to a selected contact by sending a stanza similar to the one in Example 4.

As with the previous section, a user can later choose to hide his presence updates for joe@foo.com by either sending an undirected presence stanza (see Example 1), or by sending a directed presence stanza to joe@foo.com (see Example 3).

4 Server Implementation Details

The above section documents the major points of the invisible mode protocol. It is important to understand how the server should interact with the protocol documented above.

Normally, when a server receives a presence stanza, it stamps the appropriate 'from' attribute on the stanza and sends it on its way. If the stanza does not possess a 'to' attribute, the process

of sending the stanza on its way involves broadcasting that stanza to all contacts listed in the user's roster. If the stanza does have a 'to' attribute, the stanza is simply routed to the correct contact. With the addition of invisible mode presence, the server **MUST** now verify that a stanza should be directed to contacts on a user's roster before sending it on. In addition, stanzas that are directly related to invisible presence (stanzas with a 'type' attribute of either "visible" or "invisible") **MUST NOT** be sent to any contacts, but **MAY** cause other presence stanzas to be sent instead.

4.1 Undirected Presence While Invisible to Selected Contacts

One of the biggest failings of older invisible mode implementations were client features like auto-away. In most cases, these auto-away stanza would inadvertently cause a user to become visible (although away) to all contacts on its roster. This invisible presence protocol has been designed to deal with that problem elegantly.

Consider the following situation (the protocol for which is shown in the examples below):

1. joe@foo.com has contacts stpeter@foo.com, pgmillard@foo.com and rynok@foo.com on his roster.
2. joe@foo.com has logged into the foo.com sever, but has set his presence to invisible.
3. Assume that stpeter@foo.com, pgmillard@foo.com and rynok@foo.com are all online and visible.
4. joe@foo.com wants rynok@foo.com to see his presence updates, but not stpeter@foo.com or pgmillard@foo.com
5. joe@foo.com sets his presence to extended away.
6. joe@foo.com decides to become visible to all contacts on his roster.
7. joe@foo.com decides to become invisible to pgmillard@foo.com.
8. joe@foo.com set his presence to free to chat.

Listing 5: joe@foo.com sends initial presence stanza

```
<presence type="invisible"/>
```

Since joe@foo.com has just come online, the server **SHOULD NOT** forward any presence stanzas to contacts on his roster.

Listing 6: joe@foo.com sends visible presence to rynok@foo.com

```
<presence to="rynok@foo.com" type="visible"/>
```

Listing 7: Server forwards presence update to rynok@foo.com

```
<presence from="joe@foo.com/resource" to="rynok@foo.com"/>
```

Listing 8: joe@foo.com sets his presence to extended away

```
<presence><show>xa</show><status>Away (due to idle)</status></presence>
```

Listing 9: Server forwards presence update to rynok@foo.com

```
<presence from="joe@foo.com/resource" to="rynok@foo.com">
  <show>away</show><status>Meeting</status>
</presence>
```

Note that in this example, presence updates from joe@foo.com are routed only to rynok@foo.com, despite the fact that stpeter@foo.com and pgmillard@foo.com are both online and available. This is because joe@foo.com has chosen to be visible only to rynok@foo.com. If joe@foo.com chooses to become visible to his entire roster, he simply sends an undirected presence stanza with the type="visible" attribute. In this case the server SHOULD forward the last undirected presence stanza received to stpeter@foo.com and pgmillard@foo.com.

Listing 10: joe@foo.com sends undirected visible stanza

```
<presence type="visible"/>
```

Listing 11: Server forwards last presence update to stpeter@foo.com and pgmillard@foo.com

```
<presence from="joe@foo.com/resource" to="stpeter@foo.com"/>
<presence from="joe@foo.com/resource" to="pgmillard@foo.com"/>
```

Now Joe has decided that he doesn't want pgmillard@foo.com to receive further presence updates. Since he is currently visible to all contacts, he simply needs to send a directed presence stanza to pgmillard@foo.com with the type="invisible" attribute set.

Listing 12: joe@foo.com sends directed presence stanza to pgmillard@foo.com

```
<presence to="pgmillard@foo.com" type="invisible"/>
```

In this case, the server will send a directed presence stanza to pgmillard@foo.com with the type="unavailable" attribute set, which indicates to pgmillard@foo.com's client that joe@foo.com has gone "offline."

Listing 13: Server sends unavailable presence stanza to pgmillard@foo.com

```
<presence from="joe@foo.com/resource" to="pgmillard@foo.com" type="unavailable"/>
```

Further presence updates sent by joe@foo.com will now be forwarded to stpeter@foo.com and rynok@foo.com, and MUST NOT be forwarded pgmillard@foo.com.

Listing 14: joe@foo.com set his presence to free to chat

```
<presence><show>chat</show></presence>
```

Listing 15: Server forwards presence update to stpeter@foo.com and rynok@foo.com

```
<presence from="joe@foo.com/resource" to="stpeter@foo.com">
  <show>chat</show>
</presence>

<presence from="joe@foo.com/resource" to="rynok@foo.com/resource">
  <show>chat</show>
</presence>
```

The server MUST NOT forward presence stanzas containing the type="visible" or type="invisible" attribute to a client.

5 Security Considerations

There are no security features or concerns related to this proposal.

6 IANA Considerations

No interaction with the [Internet Assigned Numbers Authority \(IANA\)](http://www.iana.org/)² is required as a result of this document.

7 XMPP Registrar Considerations

No namespaces or parameters need to be registered with the [XMPP Registrar](http://xmpp.org/registrar/)³ as a result of this document.

²The Internet Assigned Numbers Authority (IANA) is the central coordinator for the assignment of unique parameter values for Internet protocols, such as port numbers and URI schemes. For further information, see <http://www.iana.org/>.

³The XMPP Registrar maintains a list of reserved protocol namespaces as well as registries of parameters used in the context of XMPP extension protocols approved by the XMPP Standards Foundation. For further information, see <http://xmpp.org/registrar/>.