



XEP-0318: Best Practices for Client Initiated Presence Probes

Tobias Markmann

<mailto:tobias.markmann@isode.com>

<xmpp:tm@ayena.de>

2013-08-06

Version 0.2

Status	Type	Short Name
Deferred	Informational	last-presence

This specification defines a way to determine the time when a XMPP entity has last changed its presence. Using client initiated presence probes the current presence of subscribed XMPP users can be requested. In addition a protocol to request the uptime of servers and components is defined herein.

Legal

Copyright

This XMPP Extension Protocol is copyright © 1999 – 2017 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 <https://xmpp.org/about/xsf/ipr-policy>) or obtained by writing to XMPP Standards Foundation, P.O. Box 787, Parker, CO 80134 USA).

Contents

1	Introduction	1
2	Rationale	1
3	Use Cases	1
3.1	Requesting up-to-date Presence of a XMPP Entity	1
3.2	Requesting Uptime of a XMPP Server	2
4	Acknowledgements	3
5	Security Considerations	3
6	IANA Considerations	3
7	XMPP Registrar Considerations	3

1 Introduction

[RFC 6121](#)¹, section 4.3, defines presence probes as a way for servers to actively ping for presence status of XMPP contacts. In some scenarios however, clients want to request an update on the status of a XMPP contact.

2 Rationale

While [RFC 6121](#)² specifically advises against (SHOULD NOT) clients sending presence probes to XMPP contacts, there are valid scenarios where XMPP clients want to send presence probes. According to [RFC 6121](#)³, contacts a client doesn't have presence information on, are expected to be offline and server aren't mandated to explicitly send offline presence to the client for offline users.

In addition, clients in constrained environments (i.e. mobile clients), could explicitly tell the server to filter out presence stanzas of certain kind, to keep communication to a minimum. One such protocol is [Stanza Interception and Filtering Technology \(XEP-0273\)](#)⁴.

This causes presence information to be outdated and any information, that might have been attached to the offline presence, i.e. [Delayed Delivery \(XEP-0203\)](#)⁵, to be missing at client side.

Sending presence probes from the client should be based on human request, i.e. opening a chat dialog to an offline contact when missing full presence information for that contact. Clients MUST NOT send presence probes to all contacts that they think are offline after login.

3 Use Cases

This section describes two major use cases of the described protocol, client initiated presence probes.

3.1 Requesting up-to-date Presence of a XMPP Entity

In some situations, after login, the client has incomplete presence information for offline contacts. The user might be interested in status text of the offline presence of a contact or when a contact went offline. This information can be requested, i.e. when the user opens a

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

²[RFC 6121: Extensible Messaging and Presence Protocol \(XMPP\): Instant Messaging and Presence <http://tools.ietf.org/html/rfc6121>](#).

³[RFC 6121: Extensible Messaging and Presence Protocol \(XMPP\): Instant Messaging and Presence <http://tools.ietf.org/html/rfc6121>](#).

⁴[XEP-0273: Stanza Interception and Filtering Technology <https://xmpp.org/extensions/xep-0273.html>](#).

⁵[XEP-0203: Delayed Delivery <https://xmpp.org/extensions/xep-0203.html>](#).

chat dialog to an offline user, using a client initiated presence probe and is described in the following two examples.

Initially a client requests the current presence information of a contact by sending out a presence probe.

Listing 1: Request for up-to-date Presence using Presence Probe

```
<presence from='juliet@capulet.com/balcony' to='romeo@montague.com'
  type='probe' />
```

The other side's server, in this example montague.com, then responds with the last known presence of the user, including [Delayed Delivery \(XEP-0203\)](#)⁶ and other information provided by the user.

Listing 2: Presence Reply in Response to Presence Probe

```
<presence from='romeo@montague.com' to='juliet@capulet.com/balcony'
  type='unavailable'>
  <status>Going offline. Out of battery.</status>
  <delay xmlns='urn:xmpp:delay'
    from='romeo@montague.com/balcony'
    stamp='2012-09-10T23:41:07Z' />
</presence>
```

3.2 Requesting Uptime of a XMPP Server

XMPP servers typically don't have all the properties known from XMPP clients, like presence or rosters. However, [Server Buddies \(XEP-0267\)](#)⁷ for example added rosters to XMPP servers. In a similar manner, this extension describes the use of presence for XMPP servers and XMPP components. Basically, when a XMPP server or component starts up it's expected to set its presence to online.

With this concept, any party could easily request the time a XMPP server or component went online, by sending a presence probe to it.

Listing 3: Request for Presence of a XMPP Server

```
<presence from='juliet@capulet.com/balcony' to='montague.com' type='
  probe' />
```

In response, the requester receives a presence stanza, which contains [Delayed Delivery \(XEP-0203\)](#)⁸ information, indicating the time the server went online.

⁶XEP-0203: Delayed Delivery <<https://xmpp.org/extensions/xep-0203.html>>.

⁷XEP-0267: Server Buddies <<https://xmpp.org/extensions/xep-0267.html>>.

⁸XEP-0203: Delayed Delivery <<https://xmpp.org/extensions/xep-0203.html>>.

Listing 4: Response from XMPP Server indicating uptime

```
<presence from='montague.com' to='juliet@capulet.com/balcony'>
  <delay xmlns='urn:xmpp:delay'
    from='montague.com'
    stamp='2012-09-10T23:41:07Z' />
</presence>
```

4 Acknowledgements

Thanks to Kim Alvefur and Lance Stout for their helpful comments.

5 Security Considerations

Adding delayed delivery notation to the presence probe responses exposes information a user might not expect to have exposed. However, the information about last presence change is known to all entities subscribed to one's presence which are online. This extension just provides this information to XMPP entities which have been offline during the course of presence change.

The security considerations of [XMPP Date and Time Profiles \(XEP-0082\)](#)⁹ apply here.

6 IANA Considerations

This document requires no interaction with the [Internet Assigned Numbers Authority \(IANA\)](#)¹⁰.

7 XMPP Registrar Considerations

This document requires no interaction with the [XMPP Registrar](#)¹¹.

⁹XEP-0082: XMPP Date and Time Profiles <<https://xmpp.org/extensions/xep-0082.html>>.

¹⁰The 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/>>.

¹¹The 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 <<https://xmpp.org/registrar/>>.