



XMPP

XEP-0249: Direct MUC Invitations

Peter Saint-Andre
<mailto:xsf@stpeter.im>
<xmpp:peter@jabber.org>
<http://stpeter.im/>

2011-09-22
Version 1.2

Status	Type	Short Name
Draft	Standards Track	x-conference

This specification defines a method for inviting a contact to a multi-user chat room directly, instead of sending the invitation through the chat room.

Legal

Copyright

This XMPP Extension Protocol is copyright © 1999 – 2020 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	How It Works	1
3	Determining Support	2
4	Implementation Notes	3
5	Security Considerations	3
6	IANA Considerations	4
7	XMPP Registrar Considerations	4
	7.1 Protocol Namespaces	4
8	XML Schema	4
9	Acknowledgements	5

1 Introduction

[Multi-User Chat \(XEP-0045\)](#)¹ defines a protocol for groupchat over XMPP. That specification includes a method for inviting a contact to a room, where the invitation is mediated by the room itself: the user sends the invitation to the room, which in turn sends it to the contact. Unfortunately, a mediated invitation might not be delivered to the contact, for example if the contact blocks communication with entities not in its roster as specified in [Privacy Lists \(XEP-0016\)](#)². As privacy lists have become more common, MUC invitations have been increasingly blocked at the server side, resulting in an undesirable user experience. Therefore, this specification defines a method for sending an invitation directly from the user to the contact, which re-uses the original 'jabber:x:conference' namespace in use before XEP-0045 was written (with the addition of 'reason', 'continue', and 'thread' attributes for feature parity with mediated invitations).

2 How It Works

A user invites a contact to a room by sending a <message/> stanza containing only an <x/> element qualified by the 'jabber:x:conference' namespace.

Listing 1: A direct invitation

```
<message
  from='crone1@shakespeare.lit/desktop'
  to='hecate@shakespeare.lit'>
  <x xmlns='jabber:x:conference'
    jid='darkcave@macbeth.shakespeare.lit'
    password='cauldronburn'
    reason='Hey_Hecate,_this_is_the_place_for_all_good_witches!' />
</message>
```

The 'jid' attribute, which is REQUIRED, specifies the address of the groupchat room to be joined.

The 'password' attribute, which is OPTIONAL, specifies a password needed for entry into a password-protected room and maps to the <password/> element in a mediated invitation.

The 'reason' attribute, which is OPTIONAL, specifies a human-readable purpose for the invitation and maps to the <reason/> element in a mediated invitation.

The 'continue' and 'thread' attributes, which are OPTIONAL, specify that the groupchat room continues a one-to-one chat having the designated thread; these map to the <continue/> element (with 'thread' attribute) in a mediated invitation. Note that the 'continue' attribute has a datatype of boolean.³

¹XEP-0045: Multi-User Chat <<https://xmpp.org/extensions/xep-0045.html>>.

²XEP-0016: Privacy Lists <<https://xmpp.org/extensions/xep-0016.html>>.

³In accordance with Section 3.2.2.1 of XML Schema Part 2: Datatypes, the allowable lexical representations for the xs:boolean datatype are the strings "0" and "false" for the concept 'false' and the strings "1" and "true" for

Listing 2: A direct invitation that continues a one-to-one chat

```
<message
  from='crone1@shakespeare.lit/desktop'
  to='hecate@shakespeare.lit'>
  <x xmlns='jabber:x:conference'
    continue='true'
    jid='darkcave@macbeth.shakespeare.lit'
    password='cauldronburn'
    reason='Hey_Hecate,_this_is_the_place_for_all_good_witches!'
    thread='e0ffe42b28561960c6b12b944a092794b9683a38' />
</message>
```

Upon receiving a direct invitation, the contact's client SHOULD present the invitation to the contact so that the contact can accept it or decline it. If the contact accepts the invitation, the contact's client shall join the room as described in XEP-0045. If the contact declines the invitation, it shall silently discard the invitation.

3 Determining Support

If an entity supports the protocol specified herein, it MUST advertise that fact by returning a feature of "jabber:x:conference" in response to [Service Discovery \(XEP-0030\)](#)⁴ information requests (see [Protocol Namespaces](#) regarding issuance of one or more permanent namespaces).

Listing 3: Service discovery information request

```
<iq from='crone1@shakespeare.lit/desktop'
  id='disco1'
  to='hecate@shakespeare.lit/broom'
  type='get'>
  <query xmlns='http://jabber.org/protocol/disco#info' />
</iq>
```

Listing 4: Service discovery information response

```
<iq from='hecate@shakespeare.lit/broom'
  id='disco1'
  to='crone1@shakespeare.lit/desktop'
  type='result'>
  <query xmlns='http://jabber.org/protocol/disco#info'>
    <feature var='jabber:x:conference' />
  </query>
</iq>
```

the concept 'true'; implementations MUST support both styles of lexical representation.

⁴XEP-0030: Service Discovery <<https://xmpp.org/extensions/xep-0030.html>>.

In order for an application to determine whether an entity supports this protocol, where possible it SHOULD use the dynamic, presence-based profile of service discovery defined in [Entity Capabilities \(XEP-0115\)](#)⁵. However, if an application has not received entity capabilities information from an entity, it SHOULD use explicit service discovery instead.

4 Implementation Notes

Before inviting a contact to a members-only room, a user SHOULD check to see if the contact is already a member and, if not, add the contact to the member list (or ask a room administrator to do so).

If a client receives multiple invitations to the same room (e.g., a mediated invitation as defined in XEP-0045 and a direct invitation as defined here), the client SHOULD present only one of the invitations to a human user. If a client receives an invitation to a room in which the user is already an occupant, the client SHOULD silently discard the invitation.

5 Security Considerations

The following attacks are possible, in roughly the order of probability. See also [Best Practices to Discourage Denial of Service Attacks \(XEP-0205\)](#)⁶ and [RFC 3552](#)⁷.

1. The sender of an invitation could overload the 'reason' attribute with malicious or offensive text. The recipient can mitigate this attack by blocking the sender using technologies such as [Privacy Lists \(XEP-0016\)](#)⁸ and [Blocking Command \(XEP-0191\)](#)⁹.
2. A passive attacker could flood the recipient with a large number of chatroom invitations. This attack, too, can be mitigated with Privacy Lists or Simple Communications Blocking.
3. A passive attacker could use a mimicked JID to fool the recipient into thinking that the sender is a known or trusted contact. This attack requires knowledge of the recipient's known or trusted contacts, and can be mitigated by following the recommendations in [Best Practices to Prevent JID Mimicking \(XEP-0165\)](#)¹⁰.

⁵XEP-0115: Entity Capabilities <<https://xmpp.org/extensions/xep-0115.html>>.

⁶XEP-0205: Best Practices to Discourage Denial of Service Attacks <<https://xmpp.org/extensions/xep-0205.html>>.

⁷RFC 3552: Guidelines for Writing RFC Text on Security Considerations <<http://tools.ietf.org/html/rfc3552>>.

⁸XEP-0016: Privacy Lists <<https://xmpp.org/extensions/xep-0016.html>>.

⁹XEP-0191: Blocking Command <<https://xmpp.org/extensions/xep-0191.html>>.

¹⁰XEP-0165: Best Practices to Prevent JID Mimicking <<https://xmpp.org/extensions/xep-0165.html>>.

4. In the absence of end-to-end encryption, a passive attacker could eavesdrop on the chatroom invitations that a user sends or receives. This is especially threatening if the invitation includes a 'password' attribute for a password-protected room.
5. In the absence of end-to-end encryption or signing, an active attacker could modify the invitation in transit so that the recipient is directed to a different room than intended by the sender.

6 IANA Considerations

This document requires no interaction with the [Internet Assigned Numbers Authority \(IANA\)](#) ¹¹.

7 XMPP Registrar Considerations

7.1 Protocol Namespaces

The [XMPP Registrar](#) ¹² includes 'jabber:x:conference' in its registry of protocol namespaces at <https://xmpp.org/registrar/namespaces.html>, as described in Section 4 of [XMPP Registrar Function \(XEP-0053\)](#) ¹³.

8 XML Schema

```
<?xml version='1.0' encoding='UTF-8'?>

<xs:schema
  xmlns:xs='http://www.w3.org/2001/XMLSchema'
  targetNamespace='jabber:x:conference'
  xmlns='jabber:x:conference'
  elementFormDefault='qualified'>

  <xs:annotation>
    <xs:documentation>
      The protocol documented by this schema is defined in
```

¹¹The 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/>.

¹²The 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/>.

¹³XEP-0053: XMPP Registrar Function <https://xmpp.org/extensions/xep-0053.html>.

```
XEP-0249: http://www.xmpp.org/extensions/xep-0249.html
</xs:documentation>
</xs:annotation>

<xs:element name='x'>
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base='empty'>
        <xs:attribute
          name='continue'
          type='xs:boolean'
          use='optional' />
        <xs:attribute
          name='jid'
          type='xs:string'
          use='required' />
        <xs:attribute
          name='password'
          type='xs:string'
          use='optional' />
        <xs:attribute
          name='reason'
          type='xs:string'
          use='optional' />
        <xs:attribute
          name='thread'
          type='xs:string'
          use='optional' />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>

<xs:simpleType name='empty'>
  <xs:restriction base='xs:string'>
    <xs:enumeration value='' />
  </xs:restriction>
</xs:simpleType>

</xs:schema>
```

9 Acknowledgements

Thanks to Joe Hildebrand for his feedback.