



XMPP

XEP-0441: Message Archive Management Preferences

Matthew Wild

<mailto:mwild1@gmail.com>
<xmpp:me@matthewwild.co.uk>

Kevin Smith

<mailto:kevin@kismith.co.uk>
<xmpp:kevin@doomsong.co.uk>

2020-08-25
Version 0.2.0

Status	Type	Short Name
Experimental	Standards Track	mamprefs

This document defines a protocol to control a user's archiving preferences.

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	Archiving Preferences	1
2.1	Simple configuration	1
2.1.1	Default behaviour	3
2.1.2	Always archive	3
2.1.3	Never archive	3
2.2	Advanced configuration	3
2.3	JID matching	4
2.3.1	General rules	4
2.3.2	Outgoing messages	4
2.3.3	Incoming messages	4

1 Introduction

This specification describes a protocol for a server to allow a client to configure a user's message archiving preferences. It is intended to be used in conjunction with [Message Archive Management \(XEP-0313\)](#)¹.

After observing XEP-0313 usage in the wild, it became apparent that preferences were not often used, and can interfere with clients that use the archive for synchronization of messages received by the user while disconnected. Therefore it is not actively encouraged for an implementation/deployment to offer this functionality.

2 Archiving Preferences

Depending on implementation and deployment policies, a server MAY allow the user to have control over the server's archiving behaviour. This specification defines a basic protocol for this, and also allows a server to offer more advanced configuration to a user.

2.1 Simple configuration

If the server supports and allows configuration of the preferences described below then it SHOULD implement the protocol defined in this section. This allows the user to retrieve and configure the following preferences:

- A list of JIDs that should always have messages to/from archived in the user's store.
- A list of JIDs that should never have messages to/from archived in the user's store.
- The default archiving behaviour (for JIDs in neither of the above lists).

Listing 1: Retrieving archiving preferences

```
<iq type='get' id='juliet2'>
  <prefs xmlns='urn:xmpp:mam:2' />
</iq>
```

The server replies with the user's current archiving preferences. The `<prefs>` element MUST be present and contain the current default archiving policy. The `<always>` and `<never>` MUST also be present (even if empty), and contain a list of JIDs enclosed in `<jid>` elements.

Listing 2: Server responds with current preferences

```
<iq type='result' id='juliet2'>
  <prefs xmlns='urn:xmpp:mam:2' default='roster'>
```

¹XEP-0313: Message Archive Management <<https://xmpp.org/extensions/xep-0313.html>>.

```

    <always/>
    <never/>
  </prefs>
</iq>

```

It is also possible that the server may respond with a stanza error, for example the standard 'feature-not-implemented' (server does not support MAM configuration) defined in [RFC 6120](https://tools.ietf.org/html/rfc6120)².

Listing 3: Server does not support archive configuration

```

<iq type='error' id='juliet2'>
  <error type='cancel'>
    <feature-not-implemented xmlns='urn:ietf:params:xml:ns:xmpp-
      stanzas' />
  </error>
</iq>

```

To update the preferences, the client can simply send an iq stanza with a type of 'set':

Listing 4: Updating archiving preferences

```

<iq type='set' id='juliet3'>
  <prefs xmlns='urn:xmpp:mam:2' default='roster'>
    <always>
      <jid>romeo@montague.lit</jid>
    </always>
    <never>
      <jid>montague@montague.lit</jid>
    </never>
  </prefs>
</iq>

```

The server then replies with the applied preferences (note that due to server policies these MAY be different to the preferences sent by the client):

Listing 5: Server responds with updated preferences

```

<iq type='result' id='juliet3'>
  <prefs xmlns='urn:xmpp:mam:2' default='roster'>
    <always>
      <jid>romeo@montague.lit</jid>
    </always>
    <never>
      <jid>montague@montague.lit</jid>
    </never>
  </prefs>
</iq>

```

²RFC 6120: Extensible Messaging and Presence Protocol (XMPP): Core <<http://tools.ietf.org/html/rfc6120>>.

It is also possible for the server to respond with an error, for example (but not limited to) the standard 'feature-not-implemented' (the server does not support configuration of preferences), 'forbidden' (the user is not authorized to change their preferences) or 'not-allowed' (the server generally does not allow changing of configuration preferences).

2.1.1 Default behaviour

If a JID is in neither the 'always archive' nor the 'never archive' list then whether it is archived depends on this setting, the default.

The 'default' attribute of the 'prefs' element MUST be one of the following values:

- 'always' - all messages are archived by default.
- 'never' - messages are never archived by default.
- 'roster' - messages are archived only if the contact's bare JID is in the user's roster.

2.1.2 Always archive

The <prefs/> element MAY contain an <always/> child element. If present, it contains a list of <jid/> elements, each containing a single JID. The server SHOULD archive any messages to/from this JID (see 'JID matching').

If missing from the preferences, <always/> SHOULD be assumed by the server to be an empty list.

2.1.3 Never archive

The <prefs/> element MAY contain an <never/> child element. If present, it contains a list of <jid/> elements, each containing a single JID. The server SHOULD NOT archive any messages to/from this JID (see 'JID matching').

If missing from the preferences, <never/> SHOULD be assumed by the server to be an empty list.

2.2 Advanced configuration

In addition to this protocol, a server MAY offer more advanced configuration to the user through [Ad-Hoc Commands \(XEP-0050\)](#)³. Such an interface might, for example, allow the user to configure what types of messages to store, or set a limit on how long messages should remain in the archive.

If supported, such a configuration command SHOULD be presented on the well-defined

³XEP-0050: Ad-Hoc Commands <<https://xmpp.org/extensions/xep-0050.html>>.

command node of "urn:xmpp:mam#configure".

2.3 JID matching

2.3.1 General rules

When comparing the message target JID against the user's roster (ie. when the user has set default='roster') the comparison MUST use the bare target JID (that is, stripped of any resource).

For matching against entries in either the 'allow' or 'never' lists, for each listed JID:

- If the listed JID contains a resource, compare against the target JID as-is.
- If the listed JID has no resource (it is a bare JID) then first strip any resource from the target JID prior to comparison.

2.3.2 Outgoing messages

For outgoing messages, the server MUST use the value of the 'to' attribute as the target JID.

2.3.3 Incoming messages

For incoming messages, the server MUST use the value of the 'from' attribute as the target JID.