XEP-0500: MUC Slow Mode

John Livingston
mailto:contact@john-livingston.fr
xmpp:john@im.yiny.org

2024-11-28
Version 0.1.0

Status	Type	Short Name
Experimental	Standards Track	muc-slow-mode

This specification describes a way to rate limit messages a single user can send to a MUC room. It includes room configuration option, and how servers and clients can handle such a feature.

Legal

Copyright

Permissions

Warranty

Liability

Conformance

Contents

1 Introduction

There are some contexts in which you want to be able to rate limit Multi-User Chat (XEP-0045) [1] messages. This could have multiple motivations: avoid flooding, garantee a better readability of the room when there are hundreds of active users, ...
This specification propose a new option to MUC rooms, allowing room owners to fix a duration that users MUST wait between two messages. We will also specify how the server MUST reject messages send too quickly, and how clients SHOULD handle this feature (by preventing users to send messages without waiting the delay to be over, and displaying some information to them).

2 Requirements

This document addresses the following requirements:

- How to allow room owners to enable and configure the feature by editing the MUC room discovery information.
- How to enable and configure the feature without allowing room owners to change the configuration.
- How the server MUST reject messages that does not respect the parameters.
- How clients SHOULD handle rooms with such feature enabled.

3 Glossary

Clients The client software used by end-users to join MUC rooms.

Moderator A room role that is usually associated with room admins but that can be granted to non-admins.

MUC The multi-user chat protocol for text-based conferencing (e.g. Multi-User Chat (XEP-0045) XEP-0045: Multi-User Chat <https://xmpp.org/extensions/xep-0045.html>.).

Participant An occupant who does not have admin status; in a moderated room, a participant is further defined as having voice (in contrast to a visitor). A participant has a role of ”participant”.

Role A temporary position or privilege level within a room, distinct from a user’s long-lived affiliation with the room; the possible roles are ”moderator”, ”participant”, and ”visitor” (it is also possible to have no defined role). A role lasts only for the duration of an

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

occupant's visit to a room. See Multi-User Chat (XEP-0045) XEP-0045: Multi-User Chat <https://xmpp.org/extensions/xep-0045.html>..

Room administrator A user empowered by the room owner to perform administrative functions such as banning users; however, a room administrator is not allowed to change the room configuration or to destroy the room. An admin has an affiliation of "admin". See Multi-User Chat (XEP-0045) XEP-0045: Multi-User Chat <https://xmpp.org/extensions/xep-0045.html>..

Room owner Users that have special access to a room, and that can edit room configuration. See Multi-User Chat (XEP-0045) XEP-0045: Multi-User Chat <https://xmpp.org/extensions/xep-0045.html>. - Owner Use Cases.

Service Discovery Extensions See Service Discovery Extensions (XEP-0128) XEP-0128: Service Discovery Extensions <https://xmpp.org/extensions/xep-0128.html>.

Slow Mode Feature allowing to rate limit user messages in a MUC room.

Slow Mode duration When the Slow Mode feature is active, specifies the duration, in seconds, users must wait between two text messages.

Visitor In a moderated room, an occupant who does not have voice (in contrast to a participant). A visitor has a role of "visitor".

4 Use Cases

The MUC Slow Mode can for example be used when a MUC room is associated to a live video stream. In such case, there are often hundreds of people writing at the same time. To avoid flooding, and encourage participants to post relevant messages, slow mode can be usefull.

5 MUC configuration

5.1 Activating Slow Mode in the MUC Room configuration

Your implementation MAY allow the Slow Mode feature to be set room by room, by their owners.
If room owners can configure the Slow Mode feature, the server MUST add a 'muc#roomconfig_slow_mode_duration' field in the room configuration form.
This field MUST have its type equal to 'text-single'.
This field SHOULD use Data Forms Validation , having its datatype equal to 'xs:integer'.
The 'value' of the field MUST be a positive integer, so you MUST add a 'range' validation, as described in Data Forms Validation (XEP-0122) [2].
Value '0' means that the slow mode is disabled for this room.

[2] XEP-0122: Data Forms Validation <https://xmpp.org/extensions/xep-0122.html>.

Any positive value is the duration, in seconds, users must wait between two messages.
Here is an example of response the server could send when a client is querying room configuration form:

Listing 1: Room configuration example

```
<iq from='coven@chat.shakespeare.lit'
  id='config1'
  to='crone1@shakespeare.lit/desktop'
  type='result'>
  <query xmlns='http://jabber.org/protocol/muc#owner'>
    <x xmlns='jabber:x:data' type='form'>
      <title>Configuration for "coven" Room</title>
      <instructions>
        Complete this form to modify the
        configuration of your room.
      </instructions>
      <field
          type='hidden'
          var='FORM_TYPE'>
        <value>http://jabber.org/protocol/muc#roomconfig</value>
      </field>
      <field
        var='muc#roomconfig_slow_mode_duration'
        type='text-single'
        label='Slow Mode (0=disabled, any positive integer= users can
           send a message every X seconds.)'
      >
        <validate xmlns='http://jabber.org/protocol/xdata-validate'
           datatype='xs:integer'>
          <range min='0'/>
        </validate>
        <value>20</value>
      </field>
      <!-{}- and any other field... -{}->
    </x>
  </query>
</iq>
```

If the configuration is changed, the server SHOULD send a status code '104', as specified in XEP-0045 - Notification of configuration changes .

5.2 Client discovering

The feature can be enabled on a room:

- by the room owner, if your implementation allow them to set this option,

- by a server-wide parameter,
- by any other criteria, specific to the room or not.

In other words: you can enable this feature, without adding the field in the room configuration form. This allows for example server admins to apply a rate limit server-wide, or to set the slow mode programmatically on any wanted criteria (number of users in the room, current server load, room context, ...).
In any case, to allow clients to discover that the feature is active, the server MUST respond on room information queries by adding a 'muc#roominfo_slow_mode_duration' field. This field type MUST be 'text-single', and its value MUST be a positive integer.
Value '0' means that the slow mode is disabled for this room.
Any positive value is the duration, in seconds, users must wait between two messages.
Any invalid (non-positive integer) value sent by the server MUST be considered as equal to '0' (in case of a bad implementation).
Here is an example of response the server could send when a client is querying room information:

Listing 2: Room information example

```
<iq from='coven@chat.shakespeare.lit'
    id='ik3vs715'
    to='hag66@shakespeare.lit/pda'
    type='result'>
  <query xmlns='http://jabber.org/protocol/disco#info'>
    <identity
        category='conference'
        name='The␣place␣to␣be'
        type='text'/>
    <feature var='http://jabber.org/protocol/muc'/>
    <x xmlns='jabber:x:data' type='result'>
      <field var='FORM_TYPE' type='hidden'>
        <value>http://jabber.org/protocol/muc#roominfo</value>
      </field>
      <field var='muc#roominfo_slow_mode_duration' type='text-single'>
        <value>20</value>
      </field>

      <!-{}- and any other field... -{}->
    </x>
  </query>
</iq>
```

If the slow mode duration has changed (either because the room configuration was modified, or because a server parameter has changed, or any other reason), the server SHOULD send a status code '104', as specified in XEP-0045 - Notification of configuration changes.

6 Server-side rate limiting

When the Slow Mode is enabled, server MUST NOT accept two consecutive messages from the same user, to the same room, until the slow mode duration has elapsed.
Only messages containing at least one '<body/>' element must be taking into account (to avoid counting 'chatstate' messages for example).
Room administrators and owners MUST NOT be rate limited.
If a user bypass the limit, the server MUST reply an error stanza, that respects RFC 6120 [3] stanza erors (see RFC6120 - stanza errors), to reject the message. Especially:

- 'error_type' MUST be 'wait', as described in RFC6120 - Stanzas error - Syntax .
- 'error_condition' MUST be 'policy-violation', as described in RFC 6120 - Stanzas error - Defined Stream Error Conditions,
- the stanza SHOULD contain a <text>' element explaining why the message was rejected, and this message SHOULD mention the slow mode duration so that users can understand why they can't post their message.

Here is an example or error stanza:

Listing 3: Example error stanza

```
<message
  xmlns="jabber:client"
  type="error"
  to="crone1@shakespeare.lit/desktop"
  id="528df978-aa6b-422a-b987-056a810c4733"
  from="coven@chat.shakespeare.lit"
>
  <error type="wait">
    <policy-violation xmlns="urn:ietf:params:xml:ns:xmpp-stanzas" />
    <text xmlns="urn:ietf:params:xml:ns:xmpp-stanzas">
      You have exceeded the limit imposed by the slow mode in this
         room.
      You have to wait 20 seconds between messages. Please try again
         later.
    </text>
  </error>
</message>
```

The rejected message MUST NOT be forwarded to other room occupants.

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

7 Client handling

When a participant joins a room, the client SHOULD request room information as described in section Client discovering", and look for the 'muc#roominfo_slow_mode_duration' field.
If this field is present, and contains a valid strictly positive integer value, the client SHOULD display an information somewhere, to tell users that there is a slow mode limitation that applies to this room. This information MAY also be displayed to users for which the rate limit does not apply (administrators, owners, ...).
Moreover, each time a participant sends a text message, the client SHOULD prevent the user to send another message before the timeout is passed. This MAY be done either by disabling the input field, or the submit button. If the user has at least the administrator acces level, the client SHOULD NOT disable the input field or the submit button.
To avoid some frustrating behaviour, in case there is some lag on the server for example, the client MAY start counting time after receiving the message echo. Indeed, if the first message is processed with some delay by the server, it could consider that the duration is not passed yet when receiving the next one.

8 Security Considerations

As a same user can join a room with multiple sessions and/or nicknames, the server MUST use the appropriate criteria to identify the account, and apply the same limits to all user's sessions.
The slow mode duration parameter is a positive integer. When parsing its value, clients and servers MUST ignore invalid value. They MUST also check that this value is not to big to store in memory (this depends on the data type used by the implementation).

9 IANA Considerations

This document requires no interaction with the Internet Assigned Numbers Authority (IANA).

10 XMPP Registrar Considerations

10.1 Field Standardization

Field Standardization for Data Forms (XEP-0068) [4] defines a process for standardizing the fields used within Data Forms scoped by a particular namespace, and the XMPP Registrar maintains a registry of such FORM_TYPES (see <https://xmpp.org/registrar/formtypes.html>). In this XEP there are two uses of such form fields:

[4] XEP-0068: Field Data Standardization for Data Forms <https://xmpp.org/extensions/xep-0068.html>.

1. Configuration of the slow mode duration using the 'muc#roomconfig_slow_mode_duration' field.
2. Discovery of the slow mode duration using the 'muc#roominfo_slow_mode_duration' field.

The registry submissions associated with these namespaces are defined below.

10.1.1 muc#roomconfig_slow_mode_duration

The registrar shall add the following field to the 'muc#roomconfig' data form:

Listing 4: Registry Submission

```
<form_type>
  <name>http://jabber.org/protocol/muc#roomconfig</name>
  <doc>XEP-XXXX</doc>
  <desc>
    Forms extension for slow mode support in a MUC room.
  </desc>
  <field
    var='muc#roomconfig_slow_mode_duration'
    type='text-single'
    label='Slow Mode (0=disabled, any positive integer= users can send
       a message every X seconds.)'
  />
</form_type>
```

10.1.2 muc#roominfo_slow_mode_duration

The registrar shall add the following field to the 'muc#roominfo' data form:

Listing 5: Registry Submission

```
<form_type>
  <name>http://jabber.org/protocol/muc#roominfo</name>
  <doc>XEP-XXXX</doc>
  <desc>
    Forms extension for slow mode support in a MUC room.
  </desc>
  <field
    var='muc#roominfo_slow_mode_duration'
    label='Slow Mode (0=disabled, any positive integer= users can send
       a message every X seconds.)'
  />
</form_type>
```