XEP-0161: Abuse Reporting

Peter Saint-Andre
mailto:peter@andyet.net
xmpp:stpeter@stpeter.im
https://stpeter.im/

2007-05-06
Version 0.4

Status	Type	Short Name
Deferred	Standards Track	NOT_YET_ASSIGNED

This document specifies an XMPP protocol extension for reporting abusive XMPP stanzas.

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

Unfortunately, not all XMPP entities are well-behaved -- they may send spam of various kinds, harrass chat rooms, generate large amounts of traffic, etc. Currently, if an XMPP entity (the "attacker") sends abusive stanzas to another XMPP entity (the "victim"), there is no way for the victim or the victim's server to inform the attacker's server that the attacker is generating abusive traffic. In current practice, the victim's server may have no choice but to terminate the server-to-server connection rather than continue to accept the abusive traffic.
This situation is far from desirable. Therefore, this specification defines several protocol functions that can help to discourage abuse on the XMPP network:

1. A method by which a victim or the victim's server can send an abuse report to the attacker's server.
2. A method by which the attacker's server can inform other servers (or dedicated reporting services) that the IP address from which the attacker connected may be compromised.
3. A method by which the victim's server can inform other servers (or dedicated reporting services) that the attacker's server may be a rogue server.
4. An application-specific stanza error condition that can be combined with the standard <not-acceptable/> stanza error condition to inform the attacker's server that a particular XMPP stanza is considered abusive.
5. An application-specific stream error condition that can be combined with the standard <policy-violation/> stream error condition to inform the attacker's server about the reason for termination of an XML stream (if necessary).

2 Reporting Abuse

An abuse report shall be sent in an IQ stanza of type "set" containing an <abuse/> element qualified by the 'urn:xmpp:tmp:abuse' namespace (see Protocol Namespaces regarding issuance of one or more permanent namespaces).

Listing 1: Abuse report

```
<iq from='example.org'
    id='rep1'
    to='example.com'
    type='set'>
  <abuse xmlns='urn:xmpp:tmp:abuse'>
    <condition>
      <muc/>
    </condition>
```

```
    <description xml:lang='en'>This is a test.</description>
    <jid>abuser@example.com/foo</jid>
    <pointer>http://pastebin.ca/1006003</pointer>
    <stanzas>
    </stanzas>
  </abuse>
</iq>
```

The children of the <abuse/> element are as follows:

- A <condition/> element containing a child element that specifies the abuse condition (see list of conditions).
- Optionally a <description/> that provides a natural-language description of the abusive incident(s).
- A <jid/> element that specifies the JID of the abusive sender.
- Optionally a <pointer/> element that specifies a URI at which further information can be found (e.g., a web page that contains a server log).
- Optionally a <stanzas/> element that contains one or more XMPP <presence/>, <iq/>, or <message/> stanzas.

This specification intentionally does not define exactly what constitutes abuse, since ”abuse is in the eye of the beholder”. However, the following machine-readable conditions are defined as children of the <reason/> element.

Condition	Definition
<gateway/>	Attempting to inappropriately use a gateway on the receiving server (see Gateway Interaction (XEP-0100) XEP-0100: Gateway Interaction <https://xmpp.org/extensions/xep-0100.html>.)
<muc/>	Attempting to take over or otherwise abuse Multi-User Chat (XEP-0045) XEP-0045: Multi-User Chat <https://xmpp.org/extensions/xep-0045.html>. rooms on the receiving server
<proxy/>	Attempting to inappropriately use a SOCKS5 Bytestreams (XEP-0065) XEP-0065: SOCKS5 Bytestreams <https://xmpp.org/extensions/xep-0065.html>. proxy, TURN server, or other proxy on the receiving server
<pubsub/>	Attempting to inappropriately use a Publish-Subscribe (XEP-0060) XEP-0060: Publish-Subscribe <https://xmpp.org/extensions/xep-0060.html>. service on the receiving server
<service/>	Attempting to inappropriately use any other kind of service on the receiving server

Condition	Definition
<spam/>	Sending spam (unsolicited bulk messages)
<stanza-too-big/>	Sending extremely large stanzas
<too-many-recipients/>	Sending messages that contain too many recipients (see Extended Stanza Addressing (XEP-0033) XEP-0033: Extended Stanza Addressing <https://xmpp.org/extensions/xep-0033.html>.)
<too-many-stanzas/>	Sending an extremely large number of stanzas
<unacceptable-payload/>	Sending messages that contain unacceptable payloads such as malicious executables
<unacceptable-text/>	Sending messages that contain unacceptable human-readable text
<undefined-abuse/>	The abuse condition is undefined (should be used with an application-specific condition)

Note: The foregoing list of conditions is not exhaustive. The list may be augmented or otherwise modified in a future version of this specification as a result of implementation and deployment experience.
An abuse report can be generated by the victim or the victim's server. If the report is generated by the victim's server, the report MUST be sent to the attacker's server. If the report is generated by the victim, the report MUST be sent to the attacker's server and SHOULD also be sent to the victim's server as well (since the victim may not know if the attacker's server is a rogue server).
The following rules and guidelines apply to the act of reporting abuse.

1. The recipient SHOULD NOT report the abuse stanza to a server or service until it determines that the server or service supports the Abuse Reporting protocol (see the Discovering Support section of this document).

2. The recipient SHOULD report the abuse to the suspected abuser's server.

3. If the recipient's home server (i.e., the server with which it has a registered account or other trust relationship) supports the abuse Reporting protocol, the recipient SHOULD also report the offending stanza to its own server.

4. If the recipient's home server does not support the Abuse Reporting protocol, the recipient SHOULD report the abuse stanza to one or more dedicated abuse reporting services if available.

5. The recipient SHOULD NOT report the abuse stanza to the suspected abuser.

Listing 2: Unsuspecting User Receives Abuse from Evil Bot

```
<presence from='abuser@example.com'
          to='victim@example.org'
          type='subscribe'>
  <status>
    You too can be rich! Find out how at
    http://clickhere.info/makemoney
    Let&apos;s chat to make your dreams
    come true!
  </status>
</presence>
```

Listing 3: Unsuspecting User Reports Abuse

```
<iq from='unsuspectinguser@example.org/foo'
    to='example.net'
    type='set'
    id='report1'>
  <spim xmlns='urn:xmpp:tmp:abuse'>
    <presence from='abuser@example.com'
              to='victim@example.net'
              type='subscribe'
              xmlns='jabber:client'>
      <status>
        You too can be rich! Find out how at
        http://clickhere.info/makemoney
        Let&apos;s chat to make your dreams
        come true!
      </status>
    </presence>
  </spim>
</iq>
```

Upon receiving the abuse report, the attacker's server MUST proceed as follows.
If the sending server does not understand the abuse reporting protocol, it MUST return a <service-unavailable/> error.

Listing 4: Abuse reporting not supported

```
<iq from='example.com'
    id='rep1'
    to='example.org'
    type='error'>
  <error type='cancel'>
    <service-unavailable xmlns='urn:ietf:params:xml:ns:xmpp-stanzas'/>
  </error>
</iq>
```

If the JID contained in the abuse report does not exist at the attacker's server, it MUST return an <item-not-found/> error.

Listing 5: Attacker not found

```
<iq from='example.com'
    id='rep1'
    to='example.org'
    type='error'>
  <error type='cancel'>
    <item-not-found xmlns='urn:ietf:params:xml:ns:xmpp-stanzas'/>
  </error>
</iq>
```

Otherwise, if the sending server accepts the abuse report, it MUST return an IQ stanza of type "result".

Listing 6: Abuse report accepted

```
<iq from='example.com'
    id='rep1'
    to='example.org'
    type='result'/>
```

This specification does not define how a sending server shall behave when it receives an abuse report. In general it is expected that the sending server (1) will notify the human administrators of the server in some implementation-specific or deployment-specific fashion, and (2) may employ the abuse report in an automated fashion (e.g., as input to a rate-limiting algorithm, reputation system, or decision about temporarily suspending the privileges of the sending entity based on JabberID or IP address).
The following rules and guidelines apply to the processing of an abuse report:

1. Before processing the report, the processor MAY respond to the report by sending a challenge to the sender (e.g., using CAPTCHA Forms (XEP-0158) [1]) in order to make sure that the sender is not sending spurious reports or otherwise abusing the abuse reporting system.
2. The processor MUST add the report to a list or database of pending abuse reports.
3. If the suspected abuser is not already on the processor's list of known abusers, the processor SHOULD use the report in determining whether the suspected abuser is an actual abuser (see next section).
4. The processor MAY report the abuse stanza to one or more dedicated abuse reporting services if available.

[1] XEP-0158: CAPTCHA Forms <https://xmpp.org/extensions/xep-0158.html>.

5. The processor MAY report the abuse stanza to other servers it trusts.

6. The processor MAY report the abuse stanza to the suspected abuser's server (if the server is otherwise trusted, i.e., is not considered a rogue server).

7. The processor SHOULD NOT report the abuse stanza to the suspected abuser.

Not all abuse reports are valid, and not all JIDs that send stanzas reported as abuse are abusers. Care must be taken in correctly determining if a suspected abuser is an actual abuser. The following rules apply:

1. The processor SHOULD NOT add a suspected abuser to its list of known abusers until it has received at least three (3) valid abuse reports with regard to that suspected abuser, or until it has independently verified the veracity of the report.

2. If the processor determines that the suspected abuser is an actual abuser, the processor:

   a) MUST add that JID to its list of known abusers.
   b) SHOULD add the abuser's IP address to its list of known bad IPs.
   c) MAY send an abuser report to abuse reporting services or reputation services.
   d) MAY send an abuser report to other servers it trusts.
   e) SHOULD NOT send an abuser report to the abuser.

3 Reporting an Abuser

If the attacker's server determines that the suspected attacker is an actual attacker, it might decide that both the JabberID and the IP address associated with the attacker's JabberID are compromised. If it does so, it SHOULD report its conclusions to appropriate other entities (e.g., trusted peer servers or dedicated abuse reporting services). The protocol is quite simple: include the JabberID and IP address as children of an <abuser/> element qualified by the 'urn:xmpp:tmp:abuse' namespace and send an IQ stanza of type "set" to the entity that shall receive the report. This protocol SHOULD NOT be used directly by victims of abusive stanzas and if an entity receives such a report from an end user it SHOULD ignore it. The following is an example:

Listing 7: Entity Reports Abuser

```
<iq from='example.net'
    to='abuse.xmpp.net'
    type='set'
    id='abuser1'>
  <abuser xmlns='urn:xmpp:tmp:abuse'>
```

```
    <jid>abuser@example.net</jid>
    <ip>204.8.219.178</ip>
  </abuser>
</iq>
```

4 Reporting a Rogue Server

If the victim's server determines that the attacker's server is a bad actor on the network (i.e., a rogue server), it SHOULD report its conclusions to appropriate other entities (e.g., trusted peer servers or dedicated abuse reporting services). This is done by including the domain name (and optionally IP address) of the rogue server in a <rogue/> element qualified by the 'urn:xmpp:tmp:abuse' namespace and send an IQ stanza of type "set" to the entity that shall receive the report. This protocol SHOULD NOT be used directly by victims of abusive stanzas and if an entity receives such a report from an end user it SHOULD ignore it. The following is an example:

Listing 8: Entity Reports Rogue Server

```
<iq from='example.net'
    to='abuse.xmpp.net'
    type='set'
    id='rogue'>
  <rogue xmlns='urn:xmpp:tmp:abuse'>
    <jid>rogueserver.example.org</jid>
    <ip>204.8.219.178</ip>
  </rogue>
</iq>
```

5 Stanza Error

The receiving server MAY report that a particular stanza is considered abusive. The stanza error condition MUST be <not-acceptable/> and the error stanza MUST include an application-specific error condition of <abuse/> qualified by the 'urn:xmpp:tmp:abuse' (see Protocol Namespaces regarding issuance of one or more permanent namespaces). The <abuse/> element MUST include one or more <jid/> elements whose XML character data specifies the JID(s) of the abusive sender(s).

Listing 9: Abusive stanza

```
<message from='abuser@example.org/foo'
         to='victim@example.org'>
  [ ... some abusive payload here ... ]
</message>
```

Listing 10: Stanza error

```
<message from='example.com'
         to='example.org'>
  <error type='cancel'>
    <not-acceptable xmlns='urn:ietf:params:xml:ns:xmpp-stanzas'/>
  </error>
  <abuse xmlns='urn:xmpp:tmp:abuse'>
    <condition>
      <unacceptable-payload/>
    </condition>
    <jid>abuser@example.com/foo</jid>
  </abuse>
</message>
```

6 Stream Error

If the sending entity continues to generate abusive stanzas via the sending server, the receiving server MAY close the stream between the receiving server and the sending server. The stream error condition MUST be <policy-violation/> and the stream error MUST include an application-specific error condition of <abuse/> qualified by the 'urn:xmpp:tmp:abuse'. The <abuse/> element MUST include one or more <jid/> elements whose XML character data specifies the JID(s) of the abusive sender(s).

Listing 11: Stream Error

```
<stream:error>
  <policy-violation xmlns='urn:ietf:params:xml:ns:xmpp-streams'/>
  <abuse xmlns='urn:xmpp:tmp:abuse'>
    <condition>
      <too-many-stanzas/>
    </condition>
    <jid>abuser@example.com/foo</jid>
  </abuse>
</stream:error>
</stream:stream>
```

The receiving entity then SHOULD terminate the TCP connection between the receiving server and the sending server.

7 Discovering Support

If a server supports the abuse reporting protocol, it MUST report that fact by including a service discovery feature of "urn:xmpp:tmp:abuse" (see Protocol Namespaces regarding issuance of one or more permanent namespaces) in response to a Service Discovery (XEP-0030)

[2] information request:

Listing 12: Service Discovery information request

```
<iq from='example.org'
    id='disco1'
    to='example.com'
    type='get'>
  <query xmlns='http://jabber.org/protocol/disco#info'/>
</iq>
```

Listing 13: Service Discovery information response

```
<iq from='example.com'
    id='disco1'
    to='example.org'
    type='result'>
  <query xmlns='http://jabber.org/protocol/disco#info'>
    ...
    <feature var='urn:xmpp:tmp:abuse'/>
    ...
  </query>
</iq>
```

8 Security Considerations

8.1 False Reports

Not all reported abuse is actual abuse, and not all reported abusers are actual abusers. Processors must take care to ensure that processing of one or a few reports does not result in branding of a legitimate sender as an abuser, since otherwise the sender could effectively be the subject of a denial of service attack.

8.2 Rogue Servers

It is NOT RECOMMENDED for victims to send abuse reports to the server of a suspected abuser, since that server could be a rogue domain capable of sending abuse from any JID at that domain.

8.3 Denial of Service Attacks

It is possible for an abusive sender to launch a denial of service attack against legitimate users of the sending server by generating abusive traffic over the server-to-server connection (in

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

fact such attacks have already been observed on XMPP networks). Although use of the abuse reporting protocol does not completely prevent such attacks, it may at least enable sending servers to react to abusive traffic in close to real time, thus helping to ”heal” the network when denial of service attacks are launched.

8.4 Man in the Middle Attacks

If a malicious entity can inject information into the server-to-server connection, it can falsely send abuse reports to the sending server. Therefore the connection SHOULD be encrypted using Transport Layer Security as specified in XMPP Core [3].

9 IANA Considerations

This document requires no interaction with the Internet Assigned Numbers Authority (IANA) [4].

10 XMPP Registrar Considerations

10.1 Protocol Namespaces

Until this specification advances to a status of Draft, its associated namespace shall be ”urn:xmpp:tmp:abuse”; upon advancement of this specification, the XMPP Registrar [5] shall issue a permanent namespace in accordance with the process defined in Section 4 of XMPP Registrar Function (XEP-0053) [6].

10.2 Application-Specific Errors

The XMPP Registrar shall add <abuse/> to its registry of application-specific error conditions (see <https://xmpp.org/registrar/errors.html>), where the element is qualified by the ’urn:xmpp:tmp:abuse’ namespace (see Protocol Namespaces regarding issuance of one or more permanent namespaces).
The registry submission is as follows:

[3] RFC 6120: Extensible Messaging and Presence Protocol (XMPP): Core <http://tools.ietf.org/html/rfc6120>.
[4] The 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/>.
[5] The 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/>.
[6] XEP-0053: XMPP Registrar Function <https://xmpp.org/extensions/xep-0053.html>.

```
<condition>
  <ns>urn:xmpp:tmp:abuse</ns>
  <element>abuse</element>
  <desc>the sending entity has generated traffic that the receiving
     server considers abusive</desc>
  <doc>XEP-xxxx</doc>
</condition>
```

11 XML Schema

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

<xs:schema
    xmlns:xs='http://www.w3.org/2001/XMLSchema'
    targetNamespace='urn:xmpp:tmp:abuse'
    xmlns='urn:xmpp:tmp:abuse'
    elementFormDefault='qualified'>

  <xs:import
      namespace='jabber:client'
      schemaLocation='http://www.xmpp.org/schemas/jabber-client.xsd'/>

  <xs:element name='abuse'>
    <xs:complexType>
      <xs:sequence>
        <xs:element ref='condition'/>
        <xs:element ref='description'/>
        <xs:element ref='jid'/>
        <xs:element ref='pointer'/>
        <xs:element ref='stanzas'/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:element name='condition'>
    <xs:complexType>
      <xs:choice>
        <xs:element name='gateway' type='empty'/>
        <xs:element name='muc' type='empty'/>
        <xs:element name='proxy' type='empty'/>
        <xs:element name='pubsub' type='empty'/>
        <xs:element name='service' type='empty'/>
        <xs:element name='spam' type='empty'/>
        <xs:element name='stanza-too-big' type='empty'/>
        <xs:element name='too-many-recipients' type='empty'/>
        <xs:element name='too-many-stanzas' type='empty'/>
        <xs:element name='unacceptable-payload' type='empty'/>
```

```
        <xs:element name='unacceptable-text' type='empty'/>
        <xs:element name='undefined-abuse' type='empty'/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

  <xs:element name='stanzas'>
    <xs:complexType>
      <xs:choice xmlns:xmpp='jabber:client'>
        <xs:element ref='xmpp:iq'/>
        <xs:element ref='xmpp:message'/>
        <xs:element ref='xmpp:presence'/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

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

</xs:schema>
```