XEP-0076: Malicious Stanzas

Peter Saint-Andre

mailto:peter@andyet.net

xmpp:stpeter@stpeter.im

https://stpeter.im/

Joe Hildebrand

mailto:jhildebr@cisco.com

xmpp:hildjj@jabber.org

2003-04-01

Version 1.0

Status	Type	Short Name
Active	Humorous	evil

This document defines an XMPP protocol extension for flagging malicious stanzas.

Legal

Copyright

This XMPP Extension Protocol is copyright © 1999 - 2014 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 <http://xmpp.org/about-xmpp/xsf/xsf-ipr-policy/> or obtained by writing to XMPP Standards Foundation, 1899 Wynkoop Street, Suite 600, Denver, CO 80202 USA).

Contents

1 Introduction

RFC 3514 [1], published just today (2003-04-01), defines a mechanism for specifying the "evil bit" in IPv4 in order to determine if a packet was sent with malicious intent. In Section 5 ("Related Work") of that RFC, reference is made to complementary mechanisms for other forms of evil such as IPv6 support and the application/evil MIME type. Because the XMPP Standards Foundation (XSF) [2] desires to maintain compliance with protocols developed by core Internet standards bodies, the current document defines a complementary mechanism for XMPP support of evil.

2 Requirements and Approach

There are three basic XMPP stanza types that may be sent within XML streams:

- <message/> -- a "push" medium for sending information to other entities.

- <presence/> -- a "broadcast" medium for publishing information to entities that have subscribed to an entity's availability status.

- <iq/> -- a "request-response" medium for executing basic but structured transactions with other entities.

Any one of the foregoing data elements can be used with malicious intent. Therefore a generalized mechanism is needed. Because XML namespaces are used within XMPP to properly scope data, this document proposes a new namespace ('http://jabber.org/protocol/evil') to implement the desired functionality.

3 Use Cases

3.1 Evil Messages

If an evil entity sends an evil message, it MUST include an appropriately namespaced extension in the message stanza:

Listing 1: Evil Entity Sends Evil Message

```
<message
    from='iago@shakespeare.lit/pda'
```

[1]RFC 3514: The Security Flag in the IPv4 Header <http://tools.ietf.org/html/rfc3514>.

[2]The XMPP Standards Foundation (XSF) is an independent, non-profit membership organization that develops open extensions to the IETF's Extensible Messaging and Presence Protocol (XMPP). For further information, see <http://xmpp.org/xsf/>.

```
    to='emilia@shakespeare.lit/mobile'>
  <body>
      I told him what I thought, and told no more
      Than what he found himself was apt and true.
  </body>
  <evil xmlns='http://jabber.org/protocol/evil'/>
</message>
```

3.2 Evil Presence

If an evil entity sends evil presence information, it MUST include an appropriately namespaced extension in the presence stanza:

Listing 2: Evil Entity Sends Evil Presence

```
<presence from='iago@shakespeare.lit/pda'>
  <show>dnd</show>
  <status>Fomenting dissension</status>
  <evil xmlns='http://jabber.org/protocol/evil'/>
</presence>
```

3.3 Evil IQs

If an evil entity provides evil information in an IQ exchange, it MUST include an appropriately namespaced extension in the IQ stanza:

Listing 3: Evil Entity Sends Evil Message

```
<iq from='iago@shakespeare.lit/pda'
    id='evil1'
    to='emilia@shakespeare.lit/mobile'>
  <query xmlns='jabber:iq:version'>
    <name>Stabber</name>
    <version>666</version>
    <os>FiendOS</os>
  </query>
  <evil xmlns='http://jabber.org/protocol/evil'/>
</iq>
```

4 Determining Support

Evil entities MUST advertise their support for this protocol in their responses to Service Discovery (XEP-0030) [3] information ("disco#info") requests by returning a feature of

---

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

"http://jabber.org/protocol/evil":

Listing 4: A disco#info query

```
<iq from='emilia@shakespeare.lit/mobile'
    id='disco1'
    to='iago@shakespeare.lit/pda'
    type='get'>
  <query xmlns='http://jabber.org/protocol/disco#info'/>
</iq>
```

Listing 5: A disco#info response

```
<iq from='iago@shakespeare.lit/pda'
    id='disco1'
    to='emilia@shakespeare.lit/mobile'
    type='result'>
  <query xmlns='http://jabber.org/protocol/disco#info'>
    <feature var='http://jabber.org/protocol/evil'/>
  </query>
</iq>
```

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) [4]. However, if an application has not received entity capabilities information from an entity, it SHOULD use explicit service discovery instead.

5   Security Considerations

Because the 'http://jabber.org/protocol/evil' namespace flags an XML stanza as malicious, it is critically important that an entity appropriately process an XML stanza that contains the evil extension. Mission-critical applications SHOULD ignore any stanzas tagged with the evil extension. Evil servers MAY pass through evil stanzas unmodified. Really evil servers MAY silently delete the evil extension. Entities that are evil to the core SHOULD support channel-level evil as defined in RFC 3514, since this document defines per-stanza evil only.

6   IANA Considerations

This document requires no interaction with the Internet Assigned Numbers Authority (IANA) [5].

---

[4]XEP-0115: Entity Capabilities <http://xmpp.org/extensions/xep-0115.html>.

[5]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/>.

7 XMPP Registrar Considerations

The XMPP Registrar [6] shall register the 'http://jabber.org/protocol/evil' namespace as a result of this document.

---

[6]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 <http://xmpp.org/registrar/>.