XEP-0399: Client Key Support

Dave Cridland
mailto:dave@hellopando.com
xmpp:dwd@dave.cridland.net

2018-01-25
Version 0.1.0

Status	Type	Short Name
Deferred	Standards Track	client-key

This specification defines an XMPP binding of the supporting functions for the CLIENT-KEY SASL mechanism.

Legal

Copyright

Permissions

Warranty

Liability

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

The CLIENT-KEY SASL mechanism defined in draft-cridland-kitten-clientkey-00.txt suggests supporting protocol messages to be present in the application protocol. This specification defines these for XMPP.

2 Typical Flow

A typical client might use this protocol alongside that of TOTP, Extensible SASL Profile (XEP-0388) [1], and draft-cridland-kitten-clientkey-00.txt as follows.
On first use, a client will use a traditional SASL mechanism using SASL2, such as SCRAM. The server will then prompt using <next-authenticate/> to initiate, or perform, TOTP.
The client will then request a Client Key to reauthenticate later. This may be one or both of a short-term Client Key intended for in-memory storage, perhaps for use with ISR, and a longer-term Client Key used for a "remember this client" to suppress 2FA for a period.
Later authentications will use CLIENT-KEY or CLIENT-KEY-PLUS, and the server SHOULD suppress TOTP in such cases.

3 Client Key Support Operations

3.1 Client Registration

Client registration requests a Client Key from the server. It is typically used to speed reauthentication during a session, and to elide a full reauthentication at the start of a subsequent session.
In order to register and obtain a Client Key, a client sends an <iq/> of type "set" containing an XML representation of the data required, within a <register/> element qualified by the 'urn:xmpp:client-key:0' namespace, containing four elements in any order. Descriptions of values are here informative; the canonical definition is in draft-cridland-kitten-clientkey-00.txt.
<id/> has a text value of the ClientID, a suitable identifier for the client instance, unique within the scope of the authenticated (<localpart@domain.tld> or <domain.tld>).
<name/> has a text value of the Client Name, a human-readable name for the client instance.
<key/> has a text value of the ValidationKey, encoded using Base 64. Implementors are strongly advised to take careful note of the requirements of the ValidationKey as discussed in draft-cridland-kitten-clientkey-00.txt.
<ttl/> has a text value containing a integer string representation of the number of seconds the Client Key is requested to last for.
In the following example, the ValidationKey is H("Random"), and the TTL is for 30 days - a

[1] XEP-0388: Extensible SASL Profile <https://xmpp.org/extensions/xep-0388.html>.

reasonable "Remember this client" option.

```
<iq type='set' id='123456'>
  <register xmlns='urn:xmpp:client-key:0'>
    <id>213456-987123-123987</iq>
    <name>SuperChatBiscuit on Honest Pete's␣Mobile␣OS</name>
␣␣␣␣<key>WNiIwIqlYfNw44zul2EhUyqIPXE=</key>
␣␣␣␣<ttl>2592000</ttl>
␣␣</register>
</iq>
```

The server responds with two items of information in a <registered/> element qualified by the 'urn:xmpp:client-key:0' namespace. The EncryptedSecret is contained within a <encrypted-secret/> element as a base64-encoded value, and the <expiry/> element contains a timestamp for expiry.

```
<iq type='result' id='123456'>
  <registered xmlns='urn:xmpp:client-key:0'>
    <encrypted-secret>WNiIwIqlYfNw44zul2EhUyqIPXE=</encrypted-secret>
    <expiry>2017-10-15T12:00:00Z</expiry>
  </registered>
</iq>
```

Note that the expiry time might not be 30 days simply because the client has requested it - the server is free to shorten expiry times.

3.2 Key Revocation

Any authenticated client may revoke a key belonging to the same user by sending an <iq/> of type "set" containing a <revoke/> element qualified by the 'urn:xmpp:client-key:0' namespace, containing a <key/> element whose text value is the ClientID corresponding to the key to be revoked.

```
<iq type='set' id='123456'>
  <revoke xmlns='urn:xmpp:client-key:0'>
    <id>213456-987123-123987</iq>
  </revoke>
</iq>
```

3.3 Key Enumeration

Any authenticated client may enumerate keys belonging to the same user by sending an <iq/> of type "get" containing a <list/> element qualified by the 'urn:xmpp:client-key:0' namespace.

```
<iq type='get' id='123456'>
  <list xmlns='urn:xmpp:client-key:0'/>
</iq>
```

The server responds with an <iq/> of type 'result', containing the <list/> element qualified by the 'urn:xmpp:client-key:0' namespace. This element contains a sequence of <key/> elements each containing (in any order) the <id/>, <name/> and <expiry/> elements as in registration.

```
<iq type='result' id='123456'>
  <list xmlns='urn:xmpp:client-key:0'>
    <key>
      <id>213456-987123-123987</iq>
      <name>SuperChatBiscuit on Honest Pete's Mobile OS</name>
      <expiry>2017-10-15T12:00:00Z</expiry>
    </key>
    <key>
      <id>313456-987123-123987</iq>
      <name>SuperChatChocolate on Honest Bob's Mobile OS</name>
      <expiry>2018-01-08T12:00:00Z</expiry>
    </key>
  </list>
</iq>
```

4 Determining Support

Support for this protocol is advertised as the Disco feature 'urn:xmpp:client-key:0'; however clients MAY infer support if the CLIENT-KEY or CLIENT-KEY-PLUS SASL mechanism is supported.

5 Security Considerations

Security considerations for this specification are covered within the Internet-Draft draft-cridland-kitten-clientkey-00.txt - this specification introduces no further considerations by design, but relies heavily on the guidance given there.