



XMPP

XEP-0440: SASL Channel-Binding Type Capability

Florian Schmaus

<mailto:flo@geekplace.eu>

<xmpp:flo@geekplace.eu>

2024-01-30

Version 0.4.1

Status	Type	Short Name
Experimental	Standards Track	sasl-cb-types

This specification allows servers to announce their supported SASL channel-binding types to clients.

Legal

Copyright

This XMPP Extension Protocol is copyright © 1999 – 2024 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	Announcing the SASL Channel-Binding Type Capability	1
3	Interaction with SASL mechanisms	2
4	Security Considerations	2
5	IANA Considerations	3
6	XMPP Registrar Considerations	3
7	XML Schema	3
8	Acknowledgements	3

1 Introduction

SASL channel-binding is a technique to increase the security of connections (RFC 5056 ¹). Unfortunately, the SASL profile specified in RFC 6120 ² lacks a method for the server to announce its supported channel-binding types. This hinders the adoption of channel-binding, especially since the error protocol to execute after a client requested a channel-binding type unsupported by the server is basically unspecified.

The extension defined herein fills the gap left by RFC 6120 ³, by allowing the server to announce its supported channel-binding types.

2 Announcing the SASL Channel-Binding Type Capability

This protocol consists of a stream feature named 'sasl-channel-binding' qualified by the 'urn:xmpp:sasl-cb:0' namespace. The 'sasl-channel-binding' element MUST contain one or more 'channel-binding' elements, of which each MUST have an attribute with the name 'type'. The value of the 'type' attribute SHOULD be the "Channel-binding unique prefix" of a channel-binding type which was registered with the IANA Channel-Binding Types Registry ⁴. A server declares that it supports particular channel-binding types by listing the supported types via the 'sasl-channel-binding' stream feature defined herein. The 'sasl-channel-binding' element could appear next to the SASL <mechanisms/> stream-feature element, qualified by the 'urn:ietf:params:xml:ns:xmpp-sasl' namespace, as specified in RFC 6120 ⁵. Another potential appearance of <sasl-channel-binding> is next to the <authentication/> stream-feature element as specified in the Extensible SASL Profile (XEP-0388) ⁶.

Listing 1: Example <mechanisms/> stream feature with SASL Channel-Binding Type Capability.

```
<stream:features>
  <sasl-channel-binding xmlns='urn:xmpp:sasl-cb:0'>
    <channel-binding type='tls-server-end-point' />
    <channel-binding type='tls-exporter' />
  </sasl-channel-binding>
  <mechanisms xmlns='urn:ietf:params:xml:ns:xmpp-sasl'>
    <mechanism>EXTERNAL</mechanism>
    <mechanism>SCRAM-SHA-1-PLUS</mechanism>
    <mechanism>PLAIN</mechanism>
  </mechanisms>
</stream:features>
```

¹RFC 5056: On the Use of Channel Bindings to Secure Channels <<http://tools.ietf.org/html/rfc5056>>.

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

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

⁴IANA Channel-Binding Types Registry <<https://www.iana.org/assignments/channel-binding-types/channel-binding-types.xhtml>>.

⁵RFC 6120: Extensible Messaging and Presence Protocol (XMPP): Core <<http://tools.ietf.org/html/rfc6120>>.

⁶XEP-0388: Extensible SASL Profile <<https://xmpp.org/extensions/xep-0388.html>>.

3 Interaction with SASL mechanisms

Some channel-binding enabled SASL mechanisms reflect the server's presumed channel-binding abilities back to the server. This prevents SASL-mechanism stripping attacks, where a Man in the Middle (MITM) removes certain SASL mechanisms in an attempt to downgrade the mechanism chosen for authentication to a non-channel-binding enabled one. An example of a SASL mechanism family with this feature is [RFC 5802](#)⁷. This standard specifies the `gs2-cbind-flag`. The flag has a tristate value of "I don't support channel-binding" (n), "I think you do not support channel-binding, but I do" (y), or, "Let us use channel-binding type X" (p). Clients using the information provided via `<sasl-channel-binding/>` MAY want to indicate to the server that they do not support channel-binding (even if they do) if no mutual supported channel-binding type was found. The only alternative is, that the client signals the server that he believes that the server does not support channel binding. But this may cause the server to terminate the connection, because it indicates a potential ongoing SASL-mechanism stripping attack.

4 Security Considerations

If a client signals to the server that he does not support channel binding, because it found no mutual supported channel-binding types, another MITM attack vector is introduced. An active attacker could replace the `<sasl-channel-binding;>` list with channel bindings unlikely (or impossible) to be supported by the client. If the client is configured to use non-channel-binding SASL mechanisms as a fallback, this could be used to downgrade the connection security. Note that this attack is a different one than the SASL-mechanism stripping one: Here the attacker tempers with the announced channel-binding types, i.e., the values within `<sasl-channel-binding;>`

Depending on the application's security policy, clients may refrain from falling back to non-channel-binding SASL mechanisms if no mutual supported channel-binding type is available. Alternatively, they may try channel-binding with a supported type nevertheless. To mitigate the attack describe above, clients could "pin" the announced channel bindings types by a service. In that case, implementations may want to allow the set of pinned channel-binding types to be extended to stronger ones.

As further mitigation, servers MUST and clients are RECOMMENDED to at least implement the channel-binding type `tls-server-end-point` ([RFC 5929](#)⁸) to increase the probability of a mutual supported channel-binding type. However, due its improved security properties, the `tls-exporter` ([RFC 9266](#)⁹) channel-binding type should be preferred over `tls-server-end-point`.

⁷RFC 5802: Salted Challenge Response Authentication Mechanism (SCRAM) SASL and GSS-API Mechanisms <http://tools.ietf.org/html/rfc5802>.

⁸RFC 5929: Channel Bindings for TLS <http://tools.ietf.org/html/rfc5929>.

⁹RFC 9266: Channel Bindings for TLS 1.3 <http://tools.ietf.org/html/rfc9266>.

5 IANA Considerations

This document requires no interaction with the [Internet Assigned Numbers Authority \(IANA\)](#)¹⁰.

6 XMPP Registrar Considerations

This document requires no interaction with the XMPP registrar.

7 XML Schema

TODO: Add if the XEP is scheduled for the state after 'experimental'.

8 Acknowledgements

Thanks to Sam Whited for the discussion about the underlying issue and incentivizing me to come up with this extension. Further thanks goes to Ruslan N. Marchenko for pointing out the possible MITM attack vector. Last but not least, Dave Cridland, Thilo Molitor, and Simon Josefsson provided valuable feedback.

¹⁰The 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/>.