XEP-0361: Zero Handshake Server to Server Protocol

Steve Kille
mailto:steve.kille@isode.com
xmpp:steve.kille@isode.com

2017-09-11
Version 0.3

Status	Type	Short Name
Deferred	Informational	X2X

This specification defines an approach for a pair of servers to eliminate initial handshakes and associated data transfer when using the XMPP S2S Protocol. This approach may only be used with a priori agreement and configuration of the two servers involved. This is of significant benefit in high latency environments.

Legal

Copyright

Permissions

Warranty

Liability

Conformance

Contents

1 Introduction

This specification arose from work on deploying XMPP in high latency environments, with round trips of several second. Even with data transfer rates as low as 2400 bit per second, XMPP works well once connections are established as compressed messages are small and the protocols are fully asynchronous. However the combination of low data rate and high latency led to connection establishment times of several minutes. This was unworkable, particularly when connections were prone to failure.
The solution set out here is to eliminate all the intial handshaking and to start the S2S communication as if the handshaking had been correctly completed. This cannot be used for communication between an arbitrary pair of servers, as in general the negotiation associated with the handshaking is vital for correctly determining a variety of parameters for use in the connection. However, a pair of servers may operate by locally configuring information that would have been negotiated. This enables the pair of servers to eliminate initial handshaking and data exchange.

2 Requirements

This specification can be considered as a profile for server to server XMPP communication, to enable XMPP deployment over high latency links. This profile MUST only be used where its use has been pre-agreed and configured for both participating servers.

3 Use Cases

An example scenario where this protocol is important is to support use of XMPP communication on an aircraft which only has slow high latency air ground communication, with round trips of several seconds (e.g., UHF or Satcom). Use of stanard XMPP protocols (server to server or client to server) leads to long setup times. This protocol can be deployed between an XMPP server on the aircraft and one on the ground. This will be operated over a closed private network, where security considerations can be addressed primarily at the network level.

4 Business Rules

4.1 General

In simple terms, this can be considered as operation of RFC 6121 communication between a pair of XMPP servers without the preliminary negotiation done in RFC 6120. It might be considered that the start point is the DONE box in Figure 3 of RFC 6121. The TCP connection is opened and messages start to flow, as if the preceding RFC 6120 exchanges had taken place. All configuration informaition, including choice of port is handled by the a priori configuration.

Note that stream open is not sent. The interaction takes place as if this had completed. All XML elements received over the stream are treated as if they were inside a normal S2S stream that has been opened with the following stream opening tag, defining the default namespace of jabber:server and the stream namespace of <stream:stream xmlns:stream='...' xmlns='...'>. When a stream is closed, a stream close SHOULD be exchanged.

4.2 Identity Determination

A connection between a pair of servers will be by TCP, with or without TLS. The pair of servers need to identify each other at the connection level. Three mechanisms are noted:

1. Implicit. The responder assumes identity of sender because it knew where to connect to. This is NOT RECOMMENDED.
2. Validation of source IP and port.
3. Validation of digital signature using a certificate. This requires TLS to be used.

The server will then associate one or more XMPP domains with this connection level identity.

4.3 Connection Direction

Connections may be opened by one server only or by either server. The choice is part of the a priori configured agreement. It is generally recommended to allow connections to be opened by either server. However policy or network constraints may require that the connection is initiated by one server only. When a server initiates a connection it will generally use this connection to send messages to the other server. The server opening a connection is responsible for closing it at the end of its use.
Consider a scenario with two servers: server A and server B. When a connection is opened by server A to server B, the server B MAY use this connection to send messages to server A or MAY open a connection to server A. It is recommended that only a single connection is used in this scenario and so in many cases this protocol will proceed with a single bidirectional TCP connection and messages flowing in both directions. In the event of both servers opening connections at the same time, both TCP connections SHOULD be used unidirectional with messages sent on the connection opened by the message sender only.

4.4 Multiple Domains

Typically a pair of XMPP servers connecting using this protocl will communcicate with multiple domains (e.g., a base domain and a MUC domain). It is generally desirable to configure things so that all communications will share the same link, rather than establishing separate links for each domain, essentially piggy-backing multiple logical connections onto a single

TCP connection. Two or more connections MAY be initiated from one server to the other but this is NOT RECOMMENDED.

4.5 Message Validation

An XMPP server receiving data over such a link should appropriately validate to and from elements of stream child elements. The rules for this SHOULD be controlled by an priori agreement. An inbound connection will generally be associated with several peer domains. A RECOMMENDED approach is to consider each of these peers in turn and validate in the manner of a peer XMPP server connected using RFC 6020 for that domain. In the event that an inbound message is not considered to be valid, it should be handled in a manner that this invalid message would be handled if it arrived over standard S2S.

4.6 Use of TLS

This protocol MAY be deployed directly over TCP. This will often be appropriate for environments where network security is handled at IP or lower layers or where the system is operated in closed network environment.
This protocol may be deployed over TLS operating over TCP. If this is done, TLS client and or server X.509 based authentication may be used, with certificate validation achieved by PKI or simply pinning (configuring) a trusted certificate. This configuration and authentication is a part of the a priori configuration.

5 Security Considerations

This protocol operates without the standard XMPP security negotiation. It is imperative that consideration is given to link security whenever this protocol is set up. The identity verification facilities of Server Dialback and SASL EXTERNAL are not available in this protocol so other mechanisms are needed. Use of TLS with mutual authentication is the approach that provides best security with this protocol.
When TLS is not used, the only option available is for the responder to identify the initiator based on source IP address and port. This mechanism is prone to attacks, and so should be used with care. Where source IP address is checked, this may be done directly by match of IP address or by use of reverse DNS lookup to identify the connecting server. If reverse DNS Lookup is used, it is RECOMMENDED to use DNS SEC to mitigate against DNS attacks.

6 IANA Considerations

None.

7 XMPP Registrar Considerations

None.

8 Acknowledgements

Dave Cridland, Curtis King, Kevin Smith and Kurt Zeilenga worked out and validated the approach documented in this XEP.
Kevin Smith and Kurt Zeilenga provided review and input to this document.