XEP-0415: XMPP Over RELOAD (XOR)

Evgeny Khramtsov
mailto:ekhramtsov@process-one.net
xmpp:xram@zinid.ru

2019-03-06
Version 0.1.0

Status	Type	Short Name
Deferred	Standards Track	NOT_YET_ASSIGNED

This specification defines an XMPP Usage of REsource LOcation And Discovery (RELOAD). The XMPP usage provides an ability for XMPP clients to discover other peers' location through the peer-to-peer overlay. Once a peer location is determined, the RELOAD AppAttach method is used to establish a direct connection between peers through which XMPP streams are exchanged.

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).

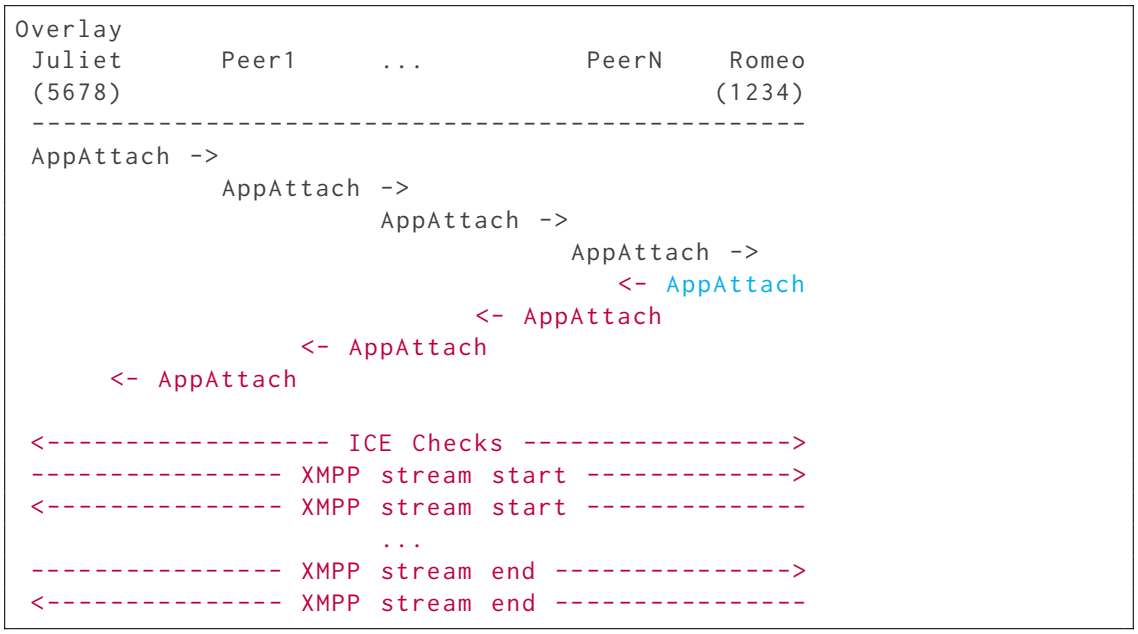
Contents

1 Introduction

REsource LOcation And Discovery (RELOAD) (RFC 6940 [1]) specifies a peer-to-peer (P2P) signaling protocol for general use on the Internet. This document defines an XMPP Usage of RELOAD that allows XMPP clients to establish peer-to-peer XMPP streams without routing them through XMPP servers.
The XMPP Usage involves two basic functions:

1. **Address Location**: XMPP clients can use the RELOAD data storage functionality to store a mapping from their XMPP address to their Node-ID in the overlay and to retrieve the Node-ID of other clients.
2. **Rendezvous**: Once an XMPP client has identified the Node-ID for an XMPP address it wishes to contact, it can use the RELOAD message routing system to set up a direct connection for exchanging XMPP streams.

Mappings are stored in the XmppLocation Resource Record defined in this document. All operations required to perform an XMPP address location or rendezvous are standard RELOAD protocol methods.
Note: XMPP stanzas are not routed through the overlay and are not stored therein.
For example, Romeo registers location of his XMPP address, ”romeo@montague.lit”, for his Node-ID ”1234”. When Juliet wants to contact Romeo, she queries the overlay for ”romeo@montague.lit” and receives Node-ID ”1234” in return. She then uses the overlay routing to establish a direct connection with Romeo and can directly start a standard XMPP stream. In detail, this works along the following steps:



[1] RFC 6940: REsource LOcation And Discovery (RELOAD) Base Protocol <http://tools.ietf.org/html/rfc6940>.

Direct XMPP streams exchange will be documented in follow-up extensions. So far, a possible way is described in Link-Local Messaging (XEP-0174) [2], although this method interacts badly with the ordinary XMPP client-to-server connection and message replication accross user devices.
It is important to note that the XMPP Usage of RELOAD is not intended to replace the existing XMPP servers infrastructure as it looks unrealistic, at least currently. Instead, the overlay connection is designed to be working along with the ordinary XMPP client-to-server connection in order to provide backward compatibility, reliable offline message delivery and multicasting. However, some clients MAY decide to maintain the overlay connection only. As an example, such scenario is possible in the video game industry where all clients are stationary (e.g. desktop) clients with persistent broadband Internet connection, without battery restrictions and no need to receive offline messages.

2 Requirements

TBD

3 Glossary

RELOAD REsource LOcation And Discovery (RFC 6940 RFC 6940: REsource LOcation And Discovery (RELOAD) Base Protocol <http://tools.ietf.org/html/rfc6940>.) - a P2P signaling protocol for general use on the Internet. The terminology and definitions from this protocol are used extensively in this document.

Address Location One or many RELOAD Node-IDs to which a peer-to-peer connection can be established in order to contact an owner of the XMPP address.

4 Storing an Address Location

4.1 Overview

In XMPP Core RFC 6120 [3], a client fully relies on servers for its XMPP address location. In XMPP Usage of RELOAD, this location function is provided by the overlay as a whole. To register its location, a RELOAD peer stores an XmppLocation Resource Record for its own XMPP address using the XMPP-LOCATION Kind, which is formally defined below. Note that if a client wishes to set the location lifetime it MUST use lifetime of the basic RELOAD StoredData structure (see Section 7 of RFC 6940 [4]).
As a simple example, consider Juliet with an XMPP address "juliet@capulet.lit" at Node-ID

[2] XEP-0174: Link-Local Messaging <https://xmpp.org/extensions/xep-0174.html>.
[3] RFC 6120: Extensible Messaging and Presence Protocol (XMPP): Core <http://tools.ietf.org/html/rfc6120>.
[4] RFC 6940: REsource LOcation And Discovery (RELOAD) Base Protocol <http://tools.ietf.org/html/rfc6940>.

”1234”. She might store the mapping ”juliet@capulet.lit -> 1234” telling anyone who wants to contact her to establish a direct XMPP stream with node ”1234”.
RELOAD peers can store two kinds of XMPP mappings,

- from an XMPP address to a destination list (a single Node-ID is just a trivial destination list), or
- from one XMPP address to another.

The meaning of the first kind of mapping is ”in order to contact me, form a connection with this Peer”. The meaning of the second kind of mapping is ”in order to contact me, dereference this XMPP address”. The latter allows for forwarding. For instance, if Juliet wants her messages to be forwarded to Romeo, she might insert the following mapping: ”juliet@capulet.lit -> romeo@montague.lit”.

4.2 Data Structure

The XmppLocation Resource Record is defined as follows:

```
enum {
     xmpp_location_address(1),
     xmpp_location_route(2),
     (255)
 } XmppLocationType;

 select (XmppLocation.type) {
   case xmpp_location_address:
     opaque            address<0..2^16-1>;

   case xmpp_location_route:
     uint8             priority;
     Destination       destination_list<0..2^16-1>;

   /* This type can be extended */

 } XmppLocationData;

 struct {
    XmppLocationType   type;
    uint16             length;
    XmppLocationData   data;
 } XmppLocation;
```

The contents of the XmppLocation Resource Record are:

type the type of the location

length the length of the rest of the PDU

data the location data

- If the location is of type ”xmpp_location_address”, then the contents are an opaque string containing the XMPP address. The address MUST be bare (i.e. without a resource part) and MUST be prepared for comparison using PRECIS rules from RFC 7622 [5].
- If the location is of type ”xmpp_location_route”, then the contents are an integer representing a route priority and an opaque string containing a destination list for the Peer. The meaning of a priority is described below in this document.

4.3 Multiple Locations

The XMPP Usage explicitly supports multiple locations for a single XMPP address. The locations are stored in a dictionary with Node-IDs as the dictionary keys. Consider, for instance, the case where Juliet has two Peers:

- her desktop client (1234)
- her cell phone (5678)

Juliet might store the following in the overlay at resource ”juliet@capulet.lit”:

- an XmppLocation of type ”xmpp_location_route” with dictionary key ”1234” and value ”1234”, both referring to Node-IDs
- an XmppLocation of type ”xmpp_location_route” with dictionary key ”5678” and value ”5678”

4.4 Access Control

In order to prevent hijacking or other misuse, locations are subject to access control rules. Two kinds of restrictions apply:

- A Store is permitted for the owner of this XMPP address, e.g. its certificate is signed by the trusted CA.
- Storing requests are performed according to the USER-NODE-MATCH access control policy of RELOAD.

[5] RFC 7622: Extensible Messaging and Presence Protocol (XMPP): Address Format <http://tools.ietf.org/html/rfc7622>.

Before a Store is permitted, the Storing Peer MUST check that:

- The XMPP address of the request is a valid Resource Name, e.g. the corresponding certificate is signed by the trusted CA.
- The certificate contains a username that is an XMPP address that hashes to the Resource-ID it is being stored at.
- The certificate contains a Node-ID that is the same as the dictionary key it is being stored at.

If any of these checks fail, the request MUST be rejected with an Error_Forbidden error.
The Storing Peer MUST NOT apply the PRECIS profile to any XMPP addresses. It is the responsibility of the Peer issuing the Store request. This allows to join XMPP agnostic RELOAD nodes to the overlay and protects intermediate peers from excessive computations, as well as possible bugs related to XMPP addresses comparison.

5 Looking Up an Address Location

In order to locate a peer in the current overlay, a RELOAD Peer MUST execute the following steps:

1. MUST remove the resource part of the XMPP address and prepare it for comparison using PRECIS rules defined in RFC 7622 [6].
2. MUST perform a Fetch for Kind XMPP-LOCATION at the Resource-ID corresponding to this prepared bare XMPP address. This Fetch SHOULD NOT indicate any dictionary keys, so that it will fetch all the stored values.
3. MUST remove duplicate destination lists and MUST initiate direct connections to all Peers as described in the following sections.

6 Forming a Direct Connection

6.1 Setting Up a Connection

Once the Peer has translated the XMPP address into a set of destination lists, it then uses the overlay to route AppAttach messages to each of those Peers. It is RECOMMENDED to route AppAttach messages to the Peers in parallel. If the Peer chooses sequential routing, it is RECOMMENDED to sort the destination lists by priority in ascending order and perform the

[6] RFC 7622: Extensible Messaging and Presence Protocol (XMPP): Address Format <http://tools.ietf.org/html/rfc7622>.

routing and connection attempts in this order (i.e. from the destination list with the smallest priority to the biggest, assuming standard integer comparison).
The ”application” field of AppAttach message MUST be 5222. The responding Peer MUST present a certificate with a Node-ID matching the terminal entry in the destination list. Otherwise, the connection MUST NOT be used and MUST be closed.
Once the AppAttach succeeds, the Peer MUST start TLS-encrypted XMPP connection. A STARTTLS procedure MUST NOT be used. For better censorship resistance, the Peer MUST NOT use ALPN extension (RFC 7301 [7]): since the endpoints are negotiated during the ICE phase, protocol multiplexing is not needed at all.
A peer (device) of an XMPP user at any time MAY close connections to some peers (devices) of another user while keeping the rest of connections to this user's peers opened. However, only connections corresponding to the destination lists with higher priorities (biggest integer values) MUST be considered for closing as redundant.
At startup, the peer MUST try to establish connections to all its user's devices. The Peer MUST strive to maintain connections to all its user's devices. It MUST NOT voluntary close some of them.

6.2 Stanza Routing

A stanza to an XMPP user MUST be sent to all connected peers (devices) of this user. Upon reception of a stanza, the peer MUST forward it to all its user's devices. An XMPP peer MUST be prepared to deal with duplicates and forwards. The follow-up extensions are supposed to clarify this.

7 Interaction with XMPP Core

The XMPP Usage of RELOAD is designed to work along with standard XMPP client-to-server (c2s) connection defined in RFC 6120 [8]. Depending on the user preferences or application usage, a peer MAY treat either c2s or RELOAD connection as primary.

- If the c2s connection is primary, the Peer MAY use the overlay in the case when its XMPP server is unavailable. This allows the XMPP service to ”degrade gracefully”: it is better to keep basic functionality working rather than completely halt the whole service. This is assumed to be the main use case of the current specification.
- If the RELOAD connection is considered as primary, a client MAY use the c2s connection to send stanzas when it has failed to locate the destination XMPP address in the overlay or when all connection attempts to the destination peer have failed.

[7] RFC 7301: Transport Layer Security (TLS) Application-Layer Protocol Negotiation Extension <https://tools.ietf.org/html/rfc7301>.
[8] RFC 6120: Extensible Messaging and Presence Protocol (XMPP): Core <http://tools.ietf.org/html/rfc6120>.

8 Enrollment and Authentication

Sybil attacks are the major threat of any peer-to-peer system. A successful Sybil attack may degrade or completely paralyze the overlay, e.g. by mounting a consequent Eclipse attack. It is asserted that under realistic assumptions, without a logically centralized authority, Sybil attacks are always possible in peer-to-peer systems [9]. To address this, the RELOAD specification relies on certificate-based authentication with a central authority. The authority's ability to ensure attackers cannot get a large number of certificates for the overlay is one of the cornerstones of RELOAD's security.
In the case of a public XMPP overlay based on existing network of federated XMPP servers, RELOAD peers MUST rely on e2e authentication defined in XEP-EAX. The document also specifies a location of the enrollment server.
In order to build an isolated XMPP overlay the reader is suggested to follow directly the approach described in the RELOAD document itself.

9 XMPP-LOCATION Kind Definition

This section defines the XMPP-LOCATION Kind.

Name XMPP-LOCATION

Kind IDs The Resource Name for the XMPP-LOCATION Kind-ID is the bare XMPP address of the user prepared for comparison using PRECIS. The data stored is an XmppLocation, which can contain either another XMPP address or a destination list to the Peer that is acting for the user.

Data Model The data model for the XMPP-LOCATION Kind-ID is a dictionary. The dictionary key is the Node-ID of the Storing Peer. This allows each Peer (presumably corresponding to a single device) to store a single route mapping.

Access Control USER-NODE-MATCH. Note that this matches the XMPP address against the "id-on-xmppAddr" Object Identifier (as defined in RFC 6120 RFC 6120: Extensible Messaging and Presence Protocol (XMPP): Core <http://tools.ietf.org/html/rfc6120>.) in the X.509 v3 certificate.

Data stored under the XMPP-LOCATION Kind is of type XmppLocation, containing one of two data types:

xmpp_location_address An XMPP address that the user can be reached at.

xmpp_location_route A destination list that can be used to reach the user's Peer.

[9] Douceur, John R. "The sybil attack." International workshop on peer-to-peer systems. Springer, Berlin, Heidelberg, 2002.

10 Security Considerations

10.1 RELOAD Security

This Usage for RELOAD does not define new protocol elements or operations. Hence, no new threats arrive from message exchanges in RELOAD.

10.2 SPAM

Successful SPAM dissemination is possible as long as the malicious entity is able to create a lot of accounts in the overlay. In other words, SPAM is a derivative of a Sybil attack. Since the overlay is designed to be Sybil resistant, SPAM is expected to be negligible.

10.3 Accounts Harvesting

TBD

10.4 Network Address Disclosure

TBD

11 IANA Considerations

11.1 Data Kind-ID

The specification introduces the following code point in the ”RELOAD Data Kind-ID” Registry (cf., RFC6940) to represent the XMPP-LOCATION Kind:

Kind	Kind-ID	Reference
XMPP-LOCATION	0x5	XEP-0415

12 XMPP Registrar Considerations

This document defines no new XMPP namespaces.