P2PSIP
Internet-Draft
Intended status: Standards Track
Expires: June 2, 2012

N. Zong, Ed.
X. Jiang
R. Even
Huawei Technologies
Y. Zhang
China Mobile

November 30, 2011

An extension to RELOAD to support Relay Peer Routing
draft-ietf-p2psip-rpr-01

Abstract

This document proposes an optional extension to RELOAD to support relay peer routing mode. RELOAD recommends symmetric recursive routing for routing messages. The new optional extension provides a shorter route for responses reducing the overhead on intermediary peers and describes the potential cases where this extension can be used.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on June 2, 2012.

Copyright Notice

Table of Contents

1. Introduction

1.1. Backgrounds

RELOAD [I-D.ietf-p2psip-base] recommends symmetric recursive routing (SRR) for routing messages and describes the extensions that would be required to support additional routing algorithms. Other than SRR, two other routing options: direct response routing (DRR) and relay peer routing (RPR) are also discussed in Appendix D in [I-D.ietf-p2psip-base]. DRR is specified in [I-D.ietf-p2psip-drr]. As we show in section 3, RPR is advantageous over SRR in some scenarios reducing load (CPU and link BW) on intermediary peers . RPR works better in a network where relay peers are provisioned in advance so that relay peers are publicly reachable in the P2P system. In other scenarios, using a combination of RPR and SRR together is more likely to bring benefits than if SRR is used alone. Some discussion on connectivity is in Non-Transitive Connectivity and DHTs [http://srhea.net/papers/ntr-worlds05.pdf].

Note that in this draft, we focus on RPR routing mode and its extensions to RELOAD. Some text such as modification to RELOAD message structure, optional methods to investigate node connectivity described in DRR draft [I-D.ietf-p2psip-drr] are also relevent to RPR.

We first discuss the problem statement in Section 3, then how to combine RPR and SRR is presented in Section 4. In Section 5, we give comparison on the cost of SRR and RPR in both managed and open networks. An extension to RELOAD to support RPR is proposed in

Section 6. Discovery of relay peers is introduced in Section 7. Some optional methods to check node connectivity is introduced in Section 8.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

We use the terminology and definitions from the Concepts and Terminology for Peer to Peer SIP [I-D.ietf-p2psip-concepts] draft extensively in this document. We also use terms defined in NAT behavior discovery [I-D.ietf-behave-nat-behavior-discovery]. Other terms used in this document are defined inline when used and are also defined below for reference.

There are two types of roles in the RELOAD architecture: peer and

client. Node is used when describing both peer and client. In discussions specific to behavior of a peer or client, the term peer or client is used instead.

Publicly Reachable: A node is publicly reachable if it can receive unsolicited messages from any other node in the same overlay. Note: "publicly" does not mean that the nodes must be on the public Internet, because the RELOAD protocol may be used in a closed system.

Relay Peer: A type of publicly reachable peer that can receive unsolicited messages from all other nodes in the overlay and forward the responses from destination peers towards the request sender.

Relay Peer Routing (RPR): refers to a routing mode in which responses to P2PSIP requests are sent by the destination peer to a relay peer transport address who will forward the responses towards the sending peer. For simplicity, the abbreviation RPR is used instead in the following text.

Symmetric Recursive Routing (SRR): refers to a routing mode in which responses follow the request path in the reverse order to get back to the sending peer. For simplicity, the abbreviation SRR is used

instead in the following text.

3. Problem Statement

RELOAD is expected to work under a great number of application scenarios. The situations where RELOAD is to be deployed differ greatly. For instance, some deployments are global, such as a Skype-like system intended to provide public service. Some run in closed networks of small scale. SRR works in any situation, but RPR may work better in some specific scenarios.

3.1. Overview

RELOAD is a simple request-response protocol. After sending a request, a node waits for a response from a destination node. There are several ways for the destination node to send a response back to the source node. In this section, we will provide detailed information on RPR.

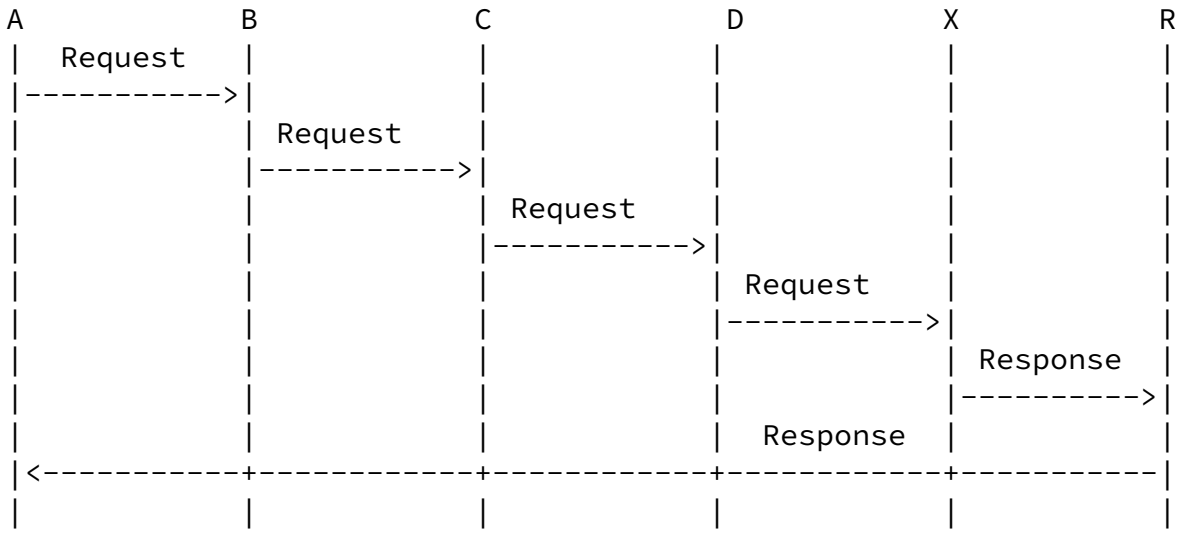
Note that the same illustrative settings can be found in DRR draft [I-D.ietf-p2psip-drr].

3.1.1. Relay Peer Routing (RPR)

If peer A knows it is behind a NAT or NATs, and knows one or more relay peers with whom they have a prior connections, peer A can try RPR. Assume A is associated with relay peer R. When sending the request, peer A includes information describing peer R transport address in the request. When peer X receives the request, peer X sends the response to peer R, which forwards it directly to Peer A on the existing connection. Note that RPR also allows a shorter route for responses compared to SRR, which means less overhead on intermediary peers. Establishing a connection to the relay with TLS requires multiple round trips. Please refer to Section 5 for cost comparison between SRR and RPR.

This technique relies on the relative population of nodes such as A

that require relay peers and peers such as R that are capable of serving as a relay peers. It also requires mechanism to enable peers to know which nodes can be used as their relays. This mechanism may be based on configuration, for example as part of the overlay configuration an initial list of relay peers can be supplied. Another option is in a response to ATTACH request the peer can signal that it can be used as a relay peer.



3.2. Scenarios Where RPR Benefits

In this section, we will list several scenarios where using RPR would provide improved performance.

3.2.1. Managed or Closed P2P System

As described in Section 3.2.1, many P2P systems run in a closed or managed environment so that network administrators can better manage their system. For example, the network administrator can deploy

several relay peers which are publicly reachable in the system and indicate their presence in the configuration file. After learning where these relay peers are, peers behind NATs can use RPR with the help from these relay peers. Peers must also support SRR in case RPR fails.

Another usage is to install relay peers on the managed network boundary allowing external peers to send responses to peers inside

the managed network.

3.2.2. Using Bootstrap Peers as Relay Peers

Bootstrap peers must be publicly reachable in a RELOAD architecture. As a result, one possible architecture would be to use the bootstrap peers as relay peers for use with RPR. The requirements for being a relay peer are publicly accessible and maintaining a direct connection with its client. As such, bootstrap peers are well suited to play the role of relay peers.

3.2.3. Wireless Scenarios

While some mobile deployments may use clients, in mobile networks using peers, RPR may reduce radio battery usage and bandwidth usage by the intermediary peers. The service provider may recommend in the configuration using RPR based on his knowledge of the topology. Such relay peers may also help connectivity to external networks.

4. Relationship Between SRR and RPR

4.1. How RPR Works

Peers using RPR must maintain a connection with their relay peer(s). This can be done in the same way as establishing a neighbor connection between peers by using the Attach method.

A requirement for RPR is for the source peer to convey their relay peer (or peers) transport address in the request, so the destination peer knows where the relay peer are and send the response to a relay peer first. The request should include also the requesting peer information enabling the relay peer to route the response back to the right peer.

Note that being a relay peer does not require that the relay peer have more functionality than an ordinary peer. As discussed later, relay peers comply with the same procedure as an ordinary peer to forward messages. The only difference is that there may be a larger traffic burden on relay peers. Relay peers can decide whether to

accept a new connection based on their current burden.

4.2. How SRR and RPR Work Together

RPR is not intended to replace SRR. As seen from Section 3, RPR has better performance in some scenarios, but have limitations as well, see for example section 4.3 in Non-Transitive Connectivity and DHTs [http://srhea.net/papers/ntr-worlds05.pdf]. As a result, it is better to use these two modes together to adapt to each peer's specific situation. Note that the informative suggestions on how to transition between SRR and RPR (e.g. compute success rate of RPR, fall back to SRR, etc) are same with that on DRR and RPR. Please refer to DRR draft [I-D.ietf-p2psip-drr] for more details. Similarly, the node can decide whether to try RPR based on other information such as configuration file information. If a relay peer is provided by the service provider, nodes may prefer RPR over SRR.

5. Comparison on cost of SRR and RPR

The major advantages in using RPR are in going through less intermediary peers on the response. By doing that it reduces the load on those peers' resources like processing and communication bandwidth.

5.1. Closed or managed networks

As described in Section 3, many P2P systems run in a closed or managed environment (e.g. carrier networks) so that network administrators would know that they could safely use RPR.

The number of hops for a response in SRR and RPR are listed in the following table. Note that the same illustrative settings can be found in DRR draft [I-D.ietf-p2psip-drr].

Mode	Success	No. of Hops	No. of Msgs
SRR	Yes	logN	logN
RPR	Yes	2	2
RPR(DTLS)	Yes	2	7+2

From the above comparison, it is clear that:

1) In most cases of N > 4 (2^2), RPR has fewer hops than SRR. Shorter route means less overhead and resource usage on intermediary peers, which is an important consideration for adopting RPR in the cases where the resource such as CPU and BW is limited, e.g. the case of mobile, wireless network.

2) In the cases of N > 512 (2^9), RPR also has fewer messages than SRR.

3) In the cases where N < 512, RPR has more messages than SRR (but still has fewer hops than SRR). So the consideration to use RPR or SRR depends on other factors like using less resources (bandwidth and processing) from the intermediaries peers. Section 4 provides use cases where RPR has better chance to work or where the intermediary resources considerations are important.

5.2. Open networks

In open network where RPR is not guaranteed, RPR can fall back to SRR If it fails after trial, as described in Section 4. Based on the same settings in Section 5.1, the number of hops, number of messages for a response in SRR and RPR are listed in the following table.

Mode	Success	No. of Hops	No. of Msgs
SRR	Yes	logN	logN
RPR	Yes	2	2
	Fail&Fall back to SRR	2+logN	2+logN
RPR(DTLS)	Yes	2	7+2
	Fail&Fall back to SRR	2+logN	9+logN

From the above comparison, it can be observed that:

1) Trying RPR would still have a good chance of fewer hops than SRR. The detailed analysis is same as DRR case and can be found in DRR draft [I-D.ietf-p2psip-drr].

2) In the cases of large network and the success rate of RPR is good, it is still possible that RPR has fewer messages than SRR. Otherwise, the consideration to use RPR or SRR depends on other factors like using less resources from the intermediaries peers.

6. Extensions to RELOAD

Adding support for RPR requires extensions to the current RELOAD protocol. In this section and in DRR[I-D.ietf-p2psip-drr], we define the changes required to the protocol, including changes to message structure and to message processing.

6.1. Basic Requirements

The basic requirements to peers for supporting RPR are same as DRR case. Please refer to DRR draft [I-D.ietf-p2psip-drr].

6.2. Modification To RELOAD Message Structure

RELOAD provides an extensible framework to accommodate future extensions. In this section and in DRR[I-D.ietf-p2psip-drr], we define a ForwardingOption structure to support RPR mode.

6.2.1. State-keeping Flag

The state-keeping flag to support RPR is same as DRR case. Please refer to DRR draft [I-D.ietf-p2psip-drr].

6.2.2. Extensive Routing Mode

The ForwardingOption structure to support RPR is same as DRR case. Please refer to DRR draft [I-D.ietf-p2psip-drr]. The definition of the fields is as follow:

RouteMode: refers to which type of routing mode is indicated to the destination peer. Currently, only DRR (specified in DRR draft [I-D.ietf-p2psip-drr]) and RPR are defined.

OverlayLinkType: refers to the transport type which is used to deliver responses from the destination peer to the relay peer.

IpAddressPort: refers to the transport address that the destination peer should use to send the response to. This will be a relay peer address for RPR.

Destination: refers to the relay peer itself. If the routing mode is RPR, then the destination contains two destinations, which are the relay peer's node-id and the sending node's node-id.

6.3. Creating a Request

6.3.1. Creating a request for RPR

When using RPR for a transaction, the sending peer MUST set the IGNORE-STATE-KEEPING flag in the ForwardingHeader. Additionally, the peer MUST construct and include a ForwardingOptions structure in the

ForwardingHeader. When constructing the ForwardingOption structure, the fields MUST be set as follows:

1) The type MUST be set to extensive_routing_mode.

2) The ExtensiveRoutingModeOption structure MUST be used for the option field within the ForwardingOptions structure. The fields MUST be defined as follows:

2.1) routemode set to 0x02 (RPR).

2.2) transport set as appropriate for the relay peer.

2.3) ipaddressport set to the transport address of the relay peer that the sender wishes the message to be relayed through.

2.4) destination structure MUST contain two values. The first MUST be defined as type peer and set with the values for the relay peer. The second MUST be defined as type peer and set with the sending peer's own values.

6.4. Request And Response Processing

This section gives normative text for message processing after RPR is introduced. Here, we only describe the additional procedures for supporting RPR. Please refer to [I-D.ietf-p2psip-base] for RELOAD base procedures.

6.4.1. Destination Peer: Receiving a Request And Sending a Response

When the destination peer receives a request, it will check the options in the forwarding header. If the destination peer can not understand extensive_routing_mode option in the request, it MUST attempt to use SRR to return an "Error_Unknown_Extension" response (defined in Section 5.3.3.1 and Section 13.9 in [I-D.ietf-p2psip-base]) to the sending peer.

If the routing mode is RPR, the destination peer MUST construct a Destination list for the response with two entries. The first MUST be set to the relay peer node-id from the option in the request and the second MUST be the sending node node-id from the option of the

request.

In the event that the routing mode is set to RPR and there are not exactly two destinations the destination peer MUST try to send an "Error_Unknown_Extension" response (defined in Section 5.3.3.1 and Section 13.9 in [I-D.ietf-p2psip-base]) to the sending peer using SRR.

After the peer constructs the destination list for the response, it sends the response to the transport address which is indicated in the ipaddressport field in the option using the specific transport mode in the Forwardingoption. If the destination peer receives a retransmit with SRR preference on the message it is trying to response to now, the responding peer should abort the RPR response and use SRR.

6.4.2. Sending Peer: Receiving a Response

Upon receiving a response, the peer follows the rules in [I-D.ietf-p2psip-base]. If the sender used RPR and does not get a response until the timeout, it MAY either resend the message using RPR but with a different relay peer (if available), or resend the message using SRR.

6.4.3. Relay Peer Processing

Relay peers are designed to forward responses to nodes who are not publicly reachable. For the routing of the response, this draft still uses the destination list. The only difference from SRR is that the destination list is not the reverse of the via-list, instead it is constructed from the forwarding option as described below.

When a relay peer receives a response, it MUST follow the rules in [I-D.ietf-p2psip-base]. It receives the response, validates the message, re-adjust the destination-list and forward the response to the next hop in the destination list based on the connection table. There is no added requirement for relay peer.

7. Discovery Of Relay Peer

There are several ways to distribute the information about relay peers throughout the overlay. P2P network providers can deploy some relay peers and advertise them in the configuration file. With the configuration file at hand, peers can get relay peers to try RPR. Another way is to consider relay peer as a service and then some service advertisement and discovery mechanism can also be used for discovering relay peers, for example, using the same mechanism as used in TURN server discovery in base RELOAD [I-D.ietf-p2psip-base]. Another option is to let a peer advertise his capability to be a relay in the response to ATTACH or JOIN.

8. Optional Methods to Investigate Node Connectivity

This section is for informational purposes only for providing some mechanisms that can be used when the configuration information does not specify if RPR can be used. It summarizes some methods which can be used for a node to determine its own network location compared with NAT. These methods may help a node to decide which routing mode it may wish to try. Note that there is no foolproof way to determine if a node is publically reachable, other than via out- of-band mechanisms. As such, peers using these mechanisms may be able to optimize traffic, but must be able to fall back to SRR routing if the

other routing mechanisms fail.

For RPR to function correctly, a node may attempt to determine whether it is publicly reachable. If it is not, RPR may be chosen to route the response with the help from relay peers, or the peers should fall back to SRR. NATs and firewalls are two major contributors preventing RPR from functioning properly. There are a number of techniques by which a node can get its reflexive address on the public side of the NAT. After obtaining the reflexive address, a peer can perform further tests to learn whether the reflexive address is publicly reachable. If the address appears to be publicly reachable, the nodes to which the address belongs can be a candidate to serve as a relay peer. Nodes which are not publicly reachable may still use RPR to shorten the response path with the help from relay peers.

Some conditions are unique in P2PSIP architecture which could be leveraged to facilitate the tests. In P2P overlay network, each node

only has partial a view of the whole network, and knows of a few nodes in the overlay. P2P routing algorithms can easily deliver a request from a sending node to a peer with whom the sending node has no direct connection. This makes it easy for a node to ask other nodes to send unsolicited messages back to the requester.

The approaches for a node to get the addresses needed for the further tests, as well as the test for learning whether a peer may be publicly reacheable is same as the DRR case. Please refer to DRR draft [I-D.ietf-p2psip-drr] for more details.

9. Security Considerations

As a routing alternative, the security part of RPR conforms to section 12.6 in based draft[I-D.ietf-p2psip-base] which describes routing security.

10. IANA Considerations

No IANA action is needed.

11. Acknowledgements

David Bryan has helped extensively with this document, and helped provide some of the text, analysis, and ideas contained here. The authors would like to thank Ted Hardie, Narayanan Vidya, Dondeti Lakshminath, Bruce Lowekamp, Stephane Bryant and Marc Petit-Huguenin

for their constructive comments.

12. References

12.1. Normative References

[I-D.ietf-p2psip-base] Jennings, C., Lowekamp, B., Rescorla, E., Baset, S., and H. Schulzrinne, "REsource LOcation And Discovery (RELOAD) Base Protocol", draft-ietf-p2psip-base-19 (work in progress), October 2011.

[I-D.ietf-p2psip-concepts] Bryan, D., Matthews, P., Shim, E., Willis, D., and S. Dawkins, "Concepts and Terminology for Peer to Peer SIP", draft-ietf-p2psip-concepts-03 (work in progress), October 2010.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[I-D.ietf-p2psip-drr] Zong, N., Jiang, X., Even, R. and Zhang, Y., "An extension to RELOAD to support Direct Response Routing", draft-ietf-p2psip-drr-01, November 2011.

12.2. Informative References

[ChurnDHT] Rhea, S., "Handling Churn in a DHT", Proceedings of the USENIX Annual Technical Conference. Handling Churn in a DHT, June 2004.

[DTLS] Modadugu, N., Rescorla, E., "The Design and Implementation of Datagram TLS", 11th Network and Distributed System Security Symposium (NDSS), 2004.

[I-D.ietf-behave-nat-behavior-discovery] MacDonald, D. and B. Lowekamp, "NAT Behavior Discovery Using STUN", draft-ietf-behave-nat-behavior-discovery-04 (work in progress), July 2008.

[I-D.ietf-behave-tcp] Guha, S., Biswas, K., Ford, B., Sivakumar, S., and P. Srisuresh, "NAT Behavioral Requirements for TCP", draft-ietf-behave-tcp-08 (work in progress), September 2008.

[I-D.lowekamp-mmusic-ice-tcp-framework] Lowekamp, B. and A. Roach, "A Proposal to Define Interactive Connectivity Establishment for the Transport Control Protocol (ICE-TCP) as an Extensible Framework", draft-lowekamp-mmusic-ice-tcp-framework-00 (work in progress), October 2008.

[RFC4787] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.

Authors' Addresses

Ning Zong (editor)
Huawei Technologies

Email: zongning@huawei.com

Xingfeng Jiang
Huawei Technologies

Email: jiang.x.f@huawei.com

Roni Even
Huawei Technologies

Email: even.roni@huawei.com

Yunfei Zhang
China Mobile

Email: zhangyunfei@chinamobile.com