ï»¿

Internet Engineering Task Force (IETF)
Request for Comments: 8797
Updates: 8166
Category: Standards Track
ISSN: 2070-1721

C. Lever
Oracle

June 2020

Remote Direct Memory Access - Connection Manager (RDMA-CM) Private Data for RPC-over-RDMA Version 1

Abstract

This document specifies the format of Remote Direct Memory Access - Connection Manager (RDMA-CM) Private Data exchanged between RPC-over-RDMA version 1 peers as part of establishing a connection. The addition of the Private Data payload specified in this document is an optional extension that does not alter the RPC-over-RDMA version 1 protocol. This document updates RFC 8166.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8797.

Copyright Notice

Copyright (c) 2020 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction

The RPC-over-RDMA version 1 transport protocol [RFC8166] enables payload data transfer using Remote Direct Memory Access (RDMA) for upper-layer protocols based on Remote Procedure Calls (RPCs) [RFC5531]. The terms "Remote Direct Memory Access" (RDMA) and "Direct Data Placement" (DDP) are introduced in [RFC5040].

The two most immediate shortcomings of RPC-over-RDMA version 1 are as follows:

1. Setting up an RDMA data transfer (via RDMA Read or Write) can be costly. The small default size of messages transmitted using RDMA Send forces the use of RDMA Read or Write operations even for relatively small messages and data payloads.

   The original specification of RPC-over-RDMA version 1 provided an out-of-band protocol for passing inline threshold values between connected peers [RFC5666]. However, [RFC8166] eliminated support for this protocol, making it unavailable for this purpose.

2. Unlike most other contemporary RDMA-enabled storage protocols, there is no facility in RPC-over-RDMA version 1 that enables the use of remote invalidation [RFC5042].

Each RPC-over-RDMA version 1 Transport Header follows the External Data Representation (XDR) definition [RFC4506] specified in [RFC8166]. However, RPC-over-RDMA version 1 has no means of extending this definition in such a way that interoperability with existing implementations is preserved. As a result, an out-of-band mechanism is needed to help relieve these constraints for existing RPC-over-RDMA version 1 implementations.

This document specifies a simple, non-XDR-based message format designed to be passed between RPC-over-RDMA version 1 peers at the time each RDMA transport connection is first established. The mechanism assumes that the underlying RDMA transport has a Private Data field that is passed between peers at connection time, such as is present in the Marker PDU Aligned Framing (MPA) protocol (described in Section 7.1 of [RFC5044] and extended in [RFC6581]) or the InfiniBand Connection Manager [IBA].

To enable current RPC-over-RDMA version 1 implementations to interoperate with implementations that support the message format described in this document, implementation of the Private Data exchange is OPTIONAL. When Private Data has been successfully exchanged, peers may choose to perform extended RDMA semantics. However, this exchange does not alter the XDR definition specified in [RFC8166].

The message format is intended to be further extensible within the normal scope of such IETF work (see Section 6 for further details). Section 8 of this document defines an IANA registry for this purpose. In addition, interoperation between implementations of RPC-over-RDMA version 1 that present this message format to peers and those that do not recognize this message format is guaranteed.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Advertised Transport Properties

3.1. Inline Threshold Size

Section 3.3.2 of [RFC8166] defines the term "inline threshold". An inline threshold is the maximum number of bytes that can be transmitted using one RDMA Send and one RDMA Receive. There are a pair of inline thresholds for a connection: a client-to-server threshold and a server-to-client threshold.

If an incoming RDMA message exceeds the size of a receiver's inline threshold, the Receive operation fails and the RDMA provider typically terminates the connection. To convey an RPC message larger than the receiver's inline threshold without risking receive failure, a sender must use explicit RDMA data transfer operations, which are more expensive than an RDMA Send. See Sections 3.3 and 3.5 of [RFC8166] for a complete discussion.

The default value of inline thresholds for RPC-over-RDMA version 1 connections is 1024 bytes (as defined in Section 3.3.3 of [RFC8166]). This value is adequate for nearly all NFS version 3 procedures.

NFS version 4 COMPOUND operations [RFC7530] are larger on average than NFS version 3 procedures [RFC1813], forcing clients to use explicit RDMA operations for frequently issued requests such as LOOKUP and GETATTR. The use of RPCSEC_GSS security also increases the average size of RPC messages, due to the larger size of RPCSEC_GSS credential material included in RPC headers [RFC7861].

If a sender and receiver could somehow agree on larger inline thresholds, frequently used RPC transactions avoid the cost of explicit RDMA operations.

3.2. Remote Invalidation

After an RDMA data transfer operation completes, an RDMA consumer can request that its peer's RDMA Network Interface Card (RNIC) invalidate the Steering Tag (STag) associated with the data transfer [RFC5042].

An RDMA consumer requests remote invalidation by posting an RDMA Send with Invalidate operation in place of an RDMA Send operation. Each RDMA Send with Invalidate carries one STag to invalidate. The receiver of an RDMA Send with Invalidate performs the requested invalidation and then reports that invalidation as part of the completion of a waiting Receive operation.

If both peers support remote invalidation, an RPC-over-RDMA responder might use remote invalidation when replying to an RPC request that provided chunks. Because one of the chunks has already been invalidated, finalizing the results of the RPC is made simpler and faster.

However, there are some important caveats that contraindicate the blanket use of remote invalidation:

* Remote invalidation is not supported by all RNICs.

* Not all RPC-over-RDMA responder implementations can generate RDMA Send with Invalidate operations.

* Not all RPC-over-RDMA requester implementations can recognize when remote invalidation has occurred.

* On one connection in different RPC-over-RDMA transactions, or in a single RPC-over-RDMA transaction, an RPC-over-RDMA requester can expose a mixture of STags that may be invalidated remotely and some that must not be. No indication is provided at the RDMA layer as to which is which.

A responder therefore must not employ remote invalidation unless it is aware of support for it in its own RDMA stack, and on the requester. And, without altering the XDR structure of RPC-over-RDMA version 1 messages, it is not possible to support remote invalidation with requesters that include an STag that must not be invalidated remotely in an RPC with STags that may be invalidated. Likewise, it is not possible to support remote invalidation with requesters that mix RPCs with STags that may be invalidated with RPCs with STags that must not be invalidated on the same connection.

There are some NFS/RDMA client implementations whose STags are always safe to invalidate remotely. For such clients, indicating to the responder that remote invalidation is always safe can enable such invalidation without the need for additional protocol elements to be defined.

4. Private Data Message Format

With an InfiniBand lower layer, for example, RDMA connection setup uses a Connection Manager (CM) when establishing a Reliable Connection [IBA]. When an RPC-over-RDMA version 1 transport connection is established, the client (which actively establishes connections) and the server (which passively accepts connections) populate the CM Private Data field exchanged as part of CM connection establishment.

The transport properties exchanged via this mechanism are fixed for the life of the connection. Each new connection presents an opportunity for a fresh exchange. An implementation of the extension described in this document MUST be prepared for the settings to change upon a reconnection.

For RPC-over-RDMA version 1, the CM Private Data field is formatted as described below. RPC clients and servers use the same format. If the capacity of the Private Data field is too small to contain this message format or the underlying RDMA transport is not managed by a CM, the CM Private Data field cannot be used on behalf of RPC-over-RDMA version 1.

The first eight octets of the CM Private Data field are to be formatted as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Format Identifier                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Version    |   Reserved  |R|   Send Size   | Receive Size  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Format Identifier: This field contains a fixed 32-bit value that identifies the content of the Private Data field as an RPC-over-RDMA version 1 CM Private Data message. In RPC-over-RDMA version 1 Private Data, the value of this field is always 0xf6ab0e18, in network byte order. The use of this field is further expanded upon in Section 5.2.

Version: This 8-bit field contains a message format version number. The value "1" in this field indicates that exactly eight octets are present, that they appear in the order described in this section, and that each has the meaning defined in this section. Further considerations about the use of this field are discussed in Section 6.

Reserved: This 7-bit field is unused. Senders MUST set these bits to zero, and receivers MUST ignore their value.

R: This 1-bit field indicates that the sender supports remote invalidation. The field is set and interpreted as described in Section 4.1.

Send Size: This 8-bit field contains an encoded value corresponding to the maximum number of bytes this peer is prepared to transmit in a single RDMA Send on this connection. The value is encoded as described in Section 4.2.

Receive Size: This 8-bit field contains an encoded value corresponding to the maximum number of bytes this peer is prepared to receive with a single RDMA Receive on this connection. The value is encoded as described in Section 4.2.

4.1. Using the R Field

The R field indicates limited support for remote invalidation as described in Section 3.2. When both connection peers have set this bit flag in their CM Private Data, the responder MAY use RDMA Send with Invalidate operations when transmitting RPC Replies. Each RDMA Send with Invalidate MUST invalidate an STag associated only with the Transaction ID (XID) in the rdma_xid field of the RPC-over-RDMA Transport Header it carries.

When either peer on a connection clears this flag, the responder MUST use only RDMA Send when transmitting RPC Replies.

4.2. Send and Receive Size Values

Inline threshold sizes from 1024 to 262144 octets can be represented in the Send Size and Receive Size fields. The inline threshold values provide a pair of 1024-octet-aligned maximum message lengths that guarantee that Send and Receive operations do not fail due to length errors.

The minimum inline threshold for RPC-over-RDMA version 1 is 1024 octets (see Section 3.3.3 of [RFC8166]). The values in the Send Size and Receive Size fields represent the unsigned number of additional kilo-octets of length beyond the first 1024 octets. Thus, a sender computes the encoded value by dividing its actual buffer size, in octets, by 1024 and subtracting one from the result. A receiver decodes an incoming Size value by performing the inverse set of operations: it adds one to the encoded value and then multiplies that result by 1024.

The client uses the smaller of its own send size and the server's reported receive size as the client-to-server inline threshold. The server uses the smaller of its own send size and the client's reported receive size as the server-to-client inline threshold.

5. Interoperability Considerations

The extension described in this document is designed to allow RPC-over-RDMA version implementations that use CM Private Data to interoperate fully with RPC-over-RDMA version 1 implementations that do not exchange this information. Implementations that use this extension must also interoperate fully with RDMA implementations that use CM Private Data for other purposes. Realizing these goals requires that implementations of this extension follow the practices described in the rest of this section.

5.1. Interoperability with RPC-over-RDMA Version 1 Implementations

When a peer does not receive a CM Private Data message that conforms to Section 4, it needs to act as if the remote peer supports only the default RPC-over-RDMA version 1 settings, as defined in [RFC8166]. In other words, the peer MUST behave as if a Private Data message was received in which (1) bit 15 of the Flags field is zero and (2) both Size fields contain the value zero.

5.2. Interoperability amongst RDMA Transports

The Format Identifier field defined in Section 4 is provided to enable implementations to distinguish the Private Data defined in this document from Private Data inserted at other layers, such as the additional Private Data defined by the MPAv2 protocol described in [RFC6581], and others.

As part of connection establishment, the buffer containing the received Private Data is searched for the Format Identifier word. The offset of the Format Identifier is not restricted to any alignment. If the RPC-over-RDMA version 1 CM Private Data Format Identifier is not present, an RPC-over-RDMA version 1 receiver MUST behave as if no RPC-over-RDMA version 1 CM Private Data has been provided.

Once the RPC-over-RDMA version 1 CM Private Data Format Identifier is found, the receiver parses the subsequent octets as RPC-over-RDMA version 1 CM Private Data. As additional assurance that the content is valid RPC-over-RDMA version 1 CM Private Data, the receiver should check that the format version number field contains a valid and recognized version number and the size of the content does not overrun the length of the buffer.

6. Updating the Message Format

Although the message format described in this document provides the ability for the client and server to exchange particular information about the local RPC-over-RDMA implementation, it is possible that there will be a future need to exchange additional properties. This would make it necessary to extend or otherwise modify the format described in this document.

Any modification faces the problem of interoperating properly with implementations of RPC-over-RDMA version 1 that are unaware of the existence of the new format. These include implementations that do not recognize the exchange of CM Private Data as well as those that recognize only the format described in this document.

Given the message format described in this document, these interoperability constraints could be met by the following sorts of new message formats:

* A format that uses a different value for the first four bytes of the format, as provided for in the registry described in Section 8.

* A format that uses the same value for the Format Identifier field and a value other than one (1) in the Version field.

Although it is possible to reorganize the last three of the eight bytes in the existing format, extended formats are unlikely to do so. New formats would take the form of extensions of the format described in this document with added fields starting at byte eight of the format or changes to the definition of bits in the Reserved field.

7. Security Considerations

The reader is directed to the Security Considerations section of [RFC8166] for background and further discussion.

The RPC-over-RDMA version 1 protocol framework depends on the semantics of the Reliable Connected (RC) queue pair (QP) type, as defined in Section 9.7.7 of [IBA]. The integrity of CM Private Data and the authenticity of its source are ensured by the exclusive use of RC QPs. Any attempt to interfere with or hijack data in transit on an RC connection results in the RDMA provider terminating the connection.

The Security Considerations section of [RFC5042] refers the reader to further relevant discussion of generic RDMA transport security. That document recommends IPsec as the default transport-layer security solution. When deployed with the Remote Direct Memory Access Protocol (RDMAP) [RFC5040], DDP [RFC5041], and MPA [RFC5044], IPsec establishes a protected channel before any operations are exchanged; thus, it protects the exchange of Private Data. However, IPsec is not available for InfiniBand or RDMA over Converged Ethernet (RoCE) deployments. Those fabrics rely on physical security and cyclic redundancy checks to protect network traffic.

Exchanging the information contained in the message format defined in this document does not expose upper-layer payloads to an attacker. Furthermore, the behavior changes that occur as a result of exchanging the Private Data described in the current document do not introduce any new risk of exposure of upper-layer payload data.

Improperly setting one of the fields in version 1 Private Data can result in an increased risk of disconnection (i.e., self-imposed Denial of Service). A similar risk can arise if non-RPC-over-RDMA CM Private Data inadvertently contains the Format Identifier that identifies this protocol's data structure. Additional checking of incoming Private Data, as described in Section 5.2, can help reduce this risk.

In addition to describing the structure of a new format version, any document that extends the Private Data format described in the current document must discuss security considerations of new data items exchanged between connection peers. Such documents should also explore the risks of erroneously identifying non-RPC-over-RDMA CM Private Data as the new format.

8. IANA Considerations

IANA has created the "RDMA-CM Private Data Identifiers" subregistry within the "Remote Direct Data Placement" protocol category group. This is a subregistry of 32-bit numbers that identify the upper-layer protocol associated with data that appears in the application-specific RDMA-CM Private Data area. The fields in this subregistry include the following: Format Identifier, Length (format length, in octets), Description, and Reference.

The initial contents of this registry are a single entry:

Format Identifier	Length	Description	Reference
0xf6ab0e18	8	RPC-over-RDMA version 1 CM Private Data	RFC 8797

Table 1: New "RDMA-CM Private Data Identifiers" Registry

IANA is to assign subsequent new entries in this registry using the Specification Required policy as defined in Section 4.6 of [RFC8126].

8.1. Guidance for Designated Experts

The Designated Expert (DE), appointed by the IESG, should ascertain the existence of suitable documentation that defines the semantics and format of the Private Data, and verify that the document is permanently and publicly available. Documentation produced outside the IETF must not conflict with work that is active or already published within the IETF. The new Reference field should contain a reference to that documentation.

The Description field should contain the name of the upper-layer protocol that generates and uses the Private Data.

The DE should assign a new Format Identifier so that it does not conflict with existing entries in this registry and so that it is not likely to be mistaken as part of the payload of other registered formats.

The DE shall post the request to the NFSV4 Working Group mailing list (or a successor to that list, if such a list exists) for comment and review. The DE shall approve or deny the request and publish notice of the decision within 30 days.

9. References

9.1. Normative References

[IBA] InfiniBand Trade Association, "InfiniBand Architecture Specification Volume 1", Release 1.3, March 2015, <https://www.infinibandta.org/>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4506] Eisler, M., Ed., "XDR: External Data Representation Standard", STD 67, RFC 4506, DOI 10.17487/RFC4506, May 2006, <https://www.rfc-editor.org/info/rfc4506>.

[RFC5040] Recio, R., Metzler, B., Culley, P., Hilland, J., and D. Garcia, "A Remote Direct Memory Access Protocol Specification", RFC 5040, DOI 10.17487/RFC5040, October 2007, <https://www.rfc-editor.org/info/rfc5040>.

[RFC5042] Pinkerton, J. and E. Deleganes, "Direct Data Placement Protocol (DDP) / Remote Direct Memory Access Protocol (RDMAP) Security", RFC 5042, DOI 10.17487/RFC5042, October 2007, <https://www.rfc-editor.org/info/rfc5042>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC8166] Lever, C., Ed., Simpson, W., and T. Talpey, "Remote Direct Memory Access Transport for Remote Procedure Call Version 1", RFC 8166, DOI 10.17487/RFC8166, June 2017, <https://www.rfc-editor.org/info/rfc8166>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

9.2. Informative References

[RFC1813] Callaghan, B., Pawlowski, B., and P. Staubach, "NFS Version 3 Protocol Specification", RFC 1813, DOI 10.17487/RFC1813, June 1995, <https://www.rfc-editor.org/info/rfc1813>.

[RFC5041] Shah, H., Pinkerton, J., Recio, R., and P. Culley, "Direct Data Placement over Reliable Transports", RFC 5041, DOI 10.17487/RFC5041, October 2007, <https://www.rfc-editor.org/info/rfc5041>.

[RFC5044] Culley, P., Elzur, U., Recio, R., Bailey, S., and J. Carrier, "Marker PDU Aligned Framing for TCP Specification", RFC 5044, DOI 10.17487/RFC5044, October 2007, <https://www.rfc-editor.org/info/rfc5044>.

[RFC5531] Thurlow, R., "RPC: Remote Procedure Call Protocol Specification Version 2", RFC 5531, DOI 10.17487/RFC5531, May 2009, <https://www.rfc-editor.org/info/rfc5531>.

[RFC5666] Talpey, T. and B. Callaghan, "Remote Direct Memory Access Transport for Remote Procedure Call", RFC 5666, DOI 10.17487/RFC5666, January 2010, <https://www.rfc-editor.org/info/rfc5666>.

[RFC6581] Kanevsky, A., Ed., Bestler, C., Ed., Sharp, R., and S. Wise, "Enhanced Remote Direct Memory Access (RDMA) Connection Establishment", RFC 6581, DOI 10.17487/RFC6581, April 2012, <https://www.rfc-editor.org/info/rfc6581>.

[RFC7530] Haynes, T., Ed. and D. Noveck, Ed., "Network File System (NFS) Version 4 Protocol", RFC 7530, DOI 10.17487/RFC7530, March 2015, <https://www.rfc-editor.org/info/rfc7530>.

[RFC7861] Adamson, A. and N. Williams, "Remote Procedure Call (RPC) Security Version 3", RFC 7861, DOI 10.17487/RFC7861, November 2016, <https://www.rfc-editor.org/info/rfc7861>.

Acknowledgments

Thanks to Christoph Hellwig and Devesh Sharma for suggesting this approach, and to Tom Talpey and David Noveck for their expert comments and review. The author also wishes to thank Bill Baker and Greg Marsden for their support of this work. Also, thanks to expert reviewers Sean Hefty and Dave Minturn.

Special thanks go to document shepherd Brian Pawlowski, Transport Area Director Magnus Westerlund, NFSV4 Working Group Chairs David Noveck and Spencer Shepler, and NFSV4 Working Group Secretary Thomas Haynes.

Author's Address

Charles Lever
Oracle Corporation
United States of America

Email: chuck.lever@oracle.com