Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: April 21, 2011

Alex RN, Ed.
Dhawal Bhagwat, Ed.
Dipankar Roy
Rishikesh Barooah
NetApp
October 18, 2010

NFS operation over IPv6
draft-ietf-nfsv4-ipv6-00.txt

Abstract

This Internet-Draft provides the description of problems faced by NFS and its various side band protocols, when implemented over IPv6 in various deployment scenarios. Solutions to the various problems are also given in the draft and are sought for approval.

Foreword

This "forward" section is an unnumbered section that is not included in the table of contents. It is primarily used for the IESG to make comments about the document. It can also be used for comments about the status of the document and sometimes is used for the RFC2119 requirements language statement.

Requirements Language

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].

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 April 21, 2011.

Copyright Notice

Copyright (c) 2010 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 (http://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. Terminology

Host: Used to refer to the client or the server where the specific(s) of client or the server does not matter.

IPv4: Internet Protocol Version 4.

IPv6: Internet Protocol Version 6.

NFS: Used to refer to Network File System irrespective of the version.

NFSv2: Network File System Protocol Version 2.

NFSv3: Network File System Protocol version 3.

NFSv4: Network File System Protocol version 4.

NFSv4.1: Network File System Protocol version 4.1.

NLM: Network Lock Manager Protocol.

NSM: Network Status Monitor Protocol.

Operation: Refers to the NFS operation when its mode of request or response is inconsequential.

2. Introduction

NFS being a application layer protocol can operate over several network layer protocols. This draft addresses problems associated with NFS operation over an IPv6 only network.

3. RPCBIND

NFS servers supporting IPv6 MUST support RPCBINDv3 as defined in [RFC1833], over IPv6. Additionally, RPCBINDv4 SHOULD be supported, as noted later in this section.

RPCBINDv3/4 protocols 'use a transport-independent format for the transport address'. Using RPCBINDv3/4, a client can clearly communicate to the server which transport (IPv4/v6, TCP/UDP) it is interested in for contacting a service. The server can communicate clearly to the client, the various transports on which a service is available. RPCBINDv2 (aka PORTMAP) provides limited support in this area.

RPCBINDv4 SHOULD be supported because it introduces useful procedures --

o RPCBPROC_GETVERSADDR - to query the server for the address of a specific version of an RPC service.

o RPCBPROC_GETADDRLIST - to query the server for a list of all addresses / transports on which an RPC service is available.

Clients SHOULD use those procedures wherever those procedures enable them to get the information of interest in one go, instead of making multiple RPCBPROC_GETADDR calls.

The netid and address formats in the RPCBINDv3/4 procedures, MUST be as per those defined for netid and universal addresses, in netid_ID draft [netid_ID]. The implementation MUST NOT use IPv4 embedded IPv6 addresses defined in Section 2.5.5 [RFC4291], for the RPCBINDv3/4 procedures.

An NFS client SHOULD specify a proper universal address in a RPCBPROC_GETADDR call; specifically, it SHOULD match the server's IP address on which the client made the call.

While processing the RPCBPROC_GETADDR call, the NFS server needs to know which local address the client is querying on; the server SHOULD pull that address from the network layer instead (the local address on which the RPCBPROC_GETADDR call was received; similar to what [RFC1833] recommends for the "r_netid" parameter -

The "r_netid" field of the argument is ignored and the "r_netid" is inferred from the network identifier of the transport on which the request came in.)

4. NFSv4 Callback Information

In the case of NFSv4.0 procedure SETCLIENTID, the netid and address formats in the callback information MUST be as per those defined for netid and universal addresses, in netid_ID draft [netid_ID]. The implementation MUST NOT use IPv4 embedded IPv6 addresses defined in Section 2.5.5 [RFC4291].

5. Handling of link-local addresses in multi-homed hosts

[RFC4007] describes link-local IPv6 addresses.

There may be environments where hosts operate only with auto-

configured (link-local) addresses. NFS implementations SHOULD support link-local addresses, so they can operate in such environments. For example, hosts booting over the network, via NFS. However, since link-local addresses are link-scoped, they can cause ambiguity on multi-homed hosts.

An NFS implementation on a multi-homed host MUST keep track of the local interface (zone) when communicating with a link-local address of another host. Alternately, such hosts can support a default zone, which the network layer can use when no interface info is specified explicitly. See the 'Scope Zones' section of RFC 4007 [RFC4007] for more on (scope) zones and their implementation.

While making a callback to an address received in a NLM LOCK call or a NFSv4 SETCLIENTID call, a server MUST specify the local interface via which the call needs to be made (or let the default zone be selected, if supported).

An NFS implementation on multi-homed hosts MUST also make sure that a link-local address of any one of it's (local) interfaces is not advertised out in any way, via any of it's other (local) interfaces. For instance, the address list that a NFS server returns in a RPCBPROC_GETADDRLIST response, MUST NOT contain a link-local address any interface other than the one on which the request was received (which will be same as the one which the response is being sent out).

6. Acknowledgments

The authors would like to acknowledge Mike Eisler for reviews of the various early versions of the draft.

7. IANA Considerations

This memo includes no request to IANA.

8. Security Considerations

All considerations from RFC 3530 Section 16 [RFC3530]

9. References

9.1. Normative References

[RFC1813] Callaghan, B., Pawlowski, B., and P. Staubach, "NFS Version 3 Protocol Specification", RFC 1813, June 1995.

[RFC1833] Srinivasan, R., "Binding Protocols for ONC RPC Version 2", RFC 1833, August 1995.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://xml.resource.org/public/rfc/html/rfc2119.html>.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC3530] Shepler, S., Callaghan, B., Robinson, D., Thurlow, R., Beame, C., Eisler, M., and D. Noveck, "Network File System (NFS) version 4 Protocol", RFC 3530, April 2003.

[RFC4007] Deering, S., Haberman, B., Jinmei, T., Nordmark, E., and B. Zill, "IPv6 Scoped Address Architecture", RFC 4007, March 2005.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[RFC4506] Eisler, M., "XDR: External Data Representation Standard", STD 67, RFC 4506, May 2006.

[RFC5378] Bradner, S. and J. Contreras, "Rights Contributors Provide to the IETF Trust", BCP 78, RFC 5378, November 2008.

[RFC5531] Thurlow, R., "RPC: Remote Procedure Call Protocol Specification Version 2", RFC 5531, May 2009.

[netid_ID] Eisler, M., "IANA Considerations for RPC Net Identifiers and Universal Address Formats", draft-ietf-nfsv4-rpc-netid-04 (work in progress), December 2008.

9.2. Informative References

[RFC1094] Nowicki, B., "NFS: Network File System Protocol specification", RFC 1094, March 1989.

[RFC2624] Shepler, S., "NFS Version 4 Design Considerations", RFC 2624, June 1999.

[RFC2663] Srisuresh, P. and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations", RFC 2663, August 1999.

[RFC3493] Gilligan, R., Thomson, S., Bound, J., McCann, J., and W. Stevens, "Basic Socket Interface Extensions for IPv6", RFC 3493, February 2003.

[RFC3542] Stevens, W., Thomas, M., Nordmark, E., and T. Jinmei, "Advanced Sockets Application Program Interface (API) for IPv6", RFC 3542, May 2003.

[RFC3593] Tesink, K., "Textual Conventions for MIB Modules Using Performance History Based on 15 Minute Intervals", RFC 3593, September 2003.

[RFC4620] Crawford, M. and B. Haberman, "IPv6 Node Information Queries", RFC 4620, August 2006.

[RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

Authors' Addresses

Alex RN (editor)
NetApp
3rd Floor, Fair Winds Block, EGL Software Park,
Bangalore, Karnataka 560071
IN

Phone: +91-80-41843352
Email: rnalex@netapp.com

Dhawal Bhagwat (editor)
NetApp
3rd Floor, Fair Winds Block, EGL Software Park,
Bangalore, Karnataka 560071
IN

Phone: +91-80-41843134
Email: dhawal@netapp.com

Dipankar Roy
NetApp
3rd Floor, Fair Winds Block, EGL Software Park,
Bangalore, Karnataka 560071
IN

Phone: +91-80-41843303
Email: dipankar@netapp.com

Rishikesh Barooah
NetApp
3rd Floor, Fair Winds Block, EGL Software Park,
Bangalore, Karnataka 560071
IN

Email: rbarooah@netapp.com