Network File System Version 4
Internet-Draft
Intended status: Informational
Expires: September 26, 2019

C. Lever
Oracle

March 25, 2019

Network File System Requirements for Computational Storage
draft-cel-nfsv4-comp-stor-reqs-00

Abstract

This document introduces an architecture for supporting Computational Storage on Network File System version 4 (NFS) servers and clients.

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 https://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 September 26, 2019.

1. Introduction

Computational storage is more than providing compute offload. True computational storage conforms to one or both of the following criteria:

- Compute resources co-located with data storage leverages a high bandwidth link between storage and local compute.
- Compute resources co-located with data storage reduces interrupt or data bandwidth needed between storage and host.

For NFS, the focus of computational storage techniques is on reducing network utilization between a server and its clients. NFSv4.2 [3] already applies this approach: new features include copy offload and file initialization (ALLOCATE).

There are two broad types of computation offloaded to storage:

Search: Examples include SQL offload, or performing a "find" operation without pulling a filesystem's data to a client.

Filtering: Also known as data transformation. Examples include compression, transcoding, encryption, or integrity checking.

The purpose of the current document is to provide a framework for reasoning about computational storage relative to the NFS protocol.

1.1. 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 [1] [2] when, and only when, they appear in all capitals, as shown here.

2. Parameters

For various reasons, we do not want to require changes to the NFS protocol to expose computational resources. Instead, an NFS server host can advertise alternate RPC programs which allow NFS clients access to the server's computational services in a structured fashion. The underlying assumption is that such computation runs faster on a host that can access file data directly rather than via NFS.

An important class of input and output parameters for these remote procedures are objects (e.g. files and directories) that exist in a

filesystem that is shared via NFS. Such objects are referenced by filehandle and optionally a range of bytes.

Serialization is necessary to prevent an offload agent from colliding with access by NFS clients. Open state or a delegation might be appropriate for this purpose.

3. Security Considerations

A trust relationship must exist between clients and servers. For example, how would clients be certain that the server has actually encrypted a file's content?

There will need to be a mechanism for authorizing offload agents to access file data.

4. IANA Considerations

This document requests no action from IANA.

5. References

5.1. Normative References

[1] 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>.

[2] 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>.

5.2. Informative References

[3] Haynes, T., "Network File System (NFS) Version 4 Minor Version 2 Protocol", RFC 7862, DOI 10.17487/RFC7862, November 2016, <https://www.rfc-editor.org/info/rfc7862>.

Acknowledgments

Special thanks go to Transport Area Director Magnus Westerlund, NFSV4 Working Group Chairs Spencer Shepler and Brian Pawlowski, and NFSV4 Working Group Secretary Thomas Haynes for their support.

Author's Address

Charles Lever
Oracle Corporation
United States of America

Email: chuck.lever@oracle.com