Workgroup: Network File System Version 4
Internet-Draft: draft-haynes-nfsv4-layrec-00
Updates: 8435 (if approved)
Published: 12 March 2023
Intended Status: Standards Track
Expires: 13 September 2023
Authors: T. Haynes
Hammerspace
T. Myklebust
Hammerspace

Reporting of Errors via LAYOUTRETURN in NFSv4.2

Abstract

The Parallel Network File System (pNFS) allows for a file's metadata (MDS) and data (DS) to be on different servers. When the metadata server is restarted, the client can still modify the data file component. During the recovery phase of startup, the metadata server and the data servers work together to recover state (which files are open, last modification time, size, etc). A problem with servers which do client side mirroring there is no means by which the client can report errors to the metadata server. As such, the metadata server has to assume that file needs resilvering. This document presents a refinement to RFC8435 to allow the client to update the metadata

This note is to be removed before publishing as an RFC.

Discussion of this draft takes place on the NFSv4 working group mailing list (nfsv4@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/nfsv4/. Working Group information can be found at https://datatracker.ietf.org/wg/nfsv4/about/.

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 13 September 2023.

Copyright Notice

Table of Contents

1. Introduction

In the Network File System version4 (NFSv4) with a Parallel NFS (pNFS) Flexible File Layout ([RFC8435]) server, during file recovery after a restart, there is no mechanism for the client to inform the metadata servers for when an error occurred during a WRITE operation to the data servers.

Using the process detailed in [RFC8178], the revisions in this document become an extension of NFSv4.2 [RFC7862]. They are built on top of the external data representation (XDR) [RFC4506] generated from [RFC7863].

1.1. Definitions

See Section 1.1 of [RFC8435] for a more complete set of definitions.

(file) data:
that part of the file system object that contains the data to be read or written. It is the contents of the object rather than the attributes of the object.

data server (DS): a pNFS server that provides the file's data when the file system object is accessed over a file-based protocol.

(file) metadata: the part of the file system object that contains various descriptive data relevant to the file object, as opposed to the file data itself. This could include the time of last modification, access time, EOF position, etc.

metadata server (MDS): the pNFS server that provides metadata information for a file system object.

1.2. 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.

2. File Recovery

When a metadata server restarts, clients are provided a grace period where they are allowed to recover any state that they had established. With open files, the client can send an OPEN (see Section 18.16 of [RFC8881]) operation with a claim type of CLAIM_PREVIOUS (see Section 9.11 of [RFC8881]). The client uses the RECLAIM_COMPLETE (see Section 18.51 of [RFC8881]) operation to notify the metadata server that it is done reclaiming state.

The NFSv4 Flexible File Layout Type allows for the client to mirror files (see Section 8 of [RFC8435]). With client side mirroring, it is important for the client to inform the metadata server of any I/O errors encountered with one of the mirrors. This is the only way for the metadata server to determine one or more of the mirrors is corrupt and then repair the mirrors via resilvering. The client can use LAYOUTRETURN and the ff_ioerr4 structure to inform the metadata server of I/O errors.

A problem is that if the metadata server restarts and the client has errors it needs to report, it can not do so. The LAYOUTRETURN needs a layout stateid to proceed and there is no way for the client to recover layout state. As such, clients have no choice but to not recover files with I/O errors. In turn, the metadata server **MUST** assume that the mirrors are inconsistent and pick one for resilvering. It is a **MUST** because as there is no control protocol

between the metadata server and the data servers, the metadata server has to assume that the client could have written data whilst it held a layout of iomode LAYOUTIOMODE4_RW.

If the server were to allow the client to use the anonymous stateid of all zeros (see Section 8.2.3 of [RFC8881]) for lrf_stateid in LAYOUTRETURN (see Section 18.44.1 of [RFC8881]), then the client could inform the metadata server of errors encountered. That in turn would allow the metadata server to accurately resilver the file by picking the correct mirror(s).

There are two error scenarios that can occur:

During the grace period: If the client were to send any lrf_stateid in the LAYOUTRETURN other than the anonymous stateid of all zeros, then the metadata server would respond with an error of NFS4ERR_GRACE.

After the grace period: If the client were to send any lrf_stateid in the LAYOUTRETURN with the anonymous stateid of all zeros, then the metadata server would respond with an error of NFS4ERR_NO_GRACE.

Also, when the metadata server builds the reply to the LAYOUTRETURN, it **MUST NOT** bump the seqid of the lorr_stateid.

The metadata server **MUST NOT** have been resilvering the file such that it has a different layout (set of mirror instances) than the client before the restart of the metadata server. Further, the metadata server **MUST NOT** start a new resilvering of the file during the grace period. If the metadata server is tracking write intents (the number of outstanding layouts with iomode of LAYOUTIOMODE4_RW), then it can relax this constraint and start a resilvering once all write intents have been recovered for that file.

If the metadata server detects that the layout being returned in the LAYOUTRETURN does not match the current mirror instances found for the file, then it should ignore the LAYOUTRETURN and resilver the file in question.

Finally, the metadata server **MAY** assume that any files which are neither explicitly recovered with a CLAIM_PREVIOUS nor have a reported error via a LAYOUTRETURN, do not need to be resilvered. The client is most likely using the forgetful model of returning layouts (see Section 12.5.5.1 of [RFC8881]).

3. Security Considerations

There are no new security considerations beyond those in [RFC7862].

4. IANA Considerations

IANA should use the current document (RFC-TBD) as the reference for the new entries.

5. References

5.1. Normative References

[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>.

[RFC7862] 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>.

[RFC7863] Haynes, T., "Network File System (NFS) Version 4 Minor Version 2 External Data Representation Standard (XDR) Description", RFC 7863, DOI 10.17487/RFC7863, November 2016, <https://www.rfc-editor.org/info/rfc7863>.

[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>.

[RFC8178] Noveck, D., "Rules for NFSv4 Extensions and Minor Versions", RFC 8178, DOI 10.17487/RFC8178, July 2017, <https://www.rfc-editor.org/info/rfc8178>.

[RFC8435] Halevy, B. and T. Haynes, "Parallel NFS (pNFS) Flexible File Layout", RFC 8435, DOI 10.17487/RFC8435, August 2018, <https://www.rfc-editor.org/info/rfc8435>.

[RFC8881] Noveck, D., Ed. and C. Lever, "Network File System (NFS) Version 4 Minor Version 1 Protocol", RFC 8881, DOI 10.17487/RFC8881, August 2020, <https://www.rfc-editor.org/info/rfc8881>.

Appendix A. Acknowledgments

None yet...

Authors' Addresses

Thomas Haynes
Hammerspace

Email: loghyr@hammerspace.com

Trond Myklebust
Hammerspace

Email: trondmy@hammerspace.com