

Network File System Version 4
Internet-Draft
Updates: [7530](#) (if approved)
Intended status: Standards Track
Expires: May 17, 2018

C. Lever, Ed.
Oracle
D. Noveck
NetApp
November 13, 2017

NFS version 4.0 Trunking Update
draft-cel-nfsv4-mv0-trunking-update-00

Abstract

Location-related attributes in NFS version 4.0 are used to support the migration and replication of server file systems. In this document, we describe an additional use for these attributes as a mechanism to enable client discovery of an NFS version 4.0 server's trunking capabilities. The interaction of trunking with migration and replication is also clarified. This document updates [RFC 7530](#).

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 May 17, 2018.

Copyright Notice

Copyright (c) 2017 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

Internet-Draft

NFSv4.0 Trunking Update

November 2017

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
2.	Requirements Language	3
3.	Preliminaries	3
3.1.	Terminology	3
3.2.	Document Organization	4
3.3.	Document Goals	5
4.	Overview of changes in RFC7530 Section 8	5
5.	Location Attributes (as Updated)	6
6.	Updates to RFC7530 Section 8.4 (Uses of Location Information)	7
6.1.	Introduction to uses of Location Information (as updated)	7
6.2.	Trunking Discovery and Detection (to be added)	8
6.3.	File System Replication and Trunking (as updated)	9
6.4.	File System Migration (as updated)	10
6.5.	Interaction of Trunking, Migration, and Replication (to be added)	10
7.	Location Entries and Server Identity Update (as updated)	12
8.	Updates to RFC7530 Outside Section Eight	12
9.	Security Considerations	12
10.	IANA Considerations	14
11.	References	14
11.1.	Normative References	14
11.2.	Informative References	15
	Appendix A. Section Classification	15
	Acknowledgments	16
	Authors' Addresses	16

[1.](#) Introduction

The NFS version 4.0 specification [[RFC7530](#)] defines a migration feature which enables the transfer of a file system from one server to another without disruption of client activity. There were a number of issues with the original definition of this feature, which are described in [[I-D.ietf-nfsv4-migration-issues](#)], and are resolved with the publication of [[RFC7931](#)].

The latter document introduces into NFS version 4.0 a means of trunking detection as a means to determine whether two network addresses are connected to the same NFS version 4.0 server instance.

Even though migration recovery is closely related to handling trunking, the NFS version 4.0 specification remains without a complete discussion of trunking.

File system migration, replication, and trunking discovery are distinct protocol features. However, it is not appropriate to treat each of these features in isolation. For example, client migration recovery processing needs to deal with the possibility of multiple server addresses in `fs_location` attributes. In addition, `fs_location` attributes, which both provide trunking-related and replication information, may change over repeated retrievals, requiring an integrated description of how clients are to deal with such changes.

In addition, the NFS version 4.0 specification needs clarification as to how the client is to respond to changes in trunking arrangements when migration occurs, as well as in some other important cases. All of the issues discussed in the current document relate to the interpretation of the `fs_locations` attribute and to the proper client and server handling of changes in `fs_location` attribute values.

[2.](#) 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 [BCP 14](#) [[RFC2119](#)] [[RFC8174](#)] when, and only when, they appear in all capitals, as shown here.

[3.](#) Preliminaries

[3.1.](#) Terminology

Most of the terms related to handling location attributes are appropriately defined in [Section 5](#) below. However, there are a few terms used outside that context that require further elucidation. Particularly important is the distinction between trunking detection and trunking discovery. The definitions we present are applicable to all minor versions of NFSv4, but we put particular emphasis on how these terms apply to NFS version 4.0.

- o Trunking detection refers to ways of determining whether two unique network addresses are associated with the same NFSv4 server instance. The means available to make this determination depends on the protocol version and, in some cases, on the client implementation.

In the case of NFS version 4.0, the means to be used are described in [[RFC7931](#)] and require use of the Uniform Client String approach to be effective. This is in contrast to later minor versions for which the means of trunking detection is described by [[RFC5661](#)] and is available to every client.

- o Trunking discovery is a process by which a client, accessing one server network address, can obtain other addresses that are associated with the same server instance. Typically it builds on a trunking detection facility by providing one or more methods by which candidate addresses are made available to the client, who then uses trunking detection to appropriately filter them.

Trunking discovery is not described in [[RFC7530](#)] and no description of it is provided in [[RFC7931](#)].

[3.2](#). Document Organization

The sections of the current document are divided into four types based on how they relate to the eventual updating of the NFS version 4.0 specification. Once this update is published, NFS version 4.0 will be specified by multiple documents that need to be read together, until such time as a consolidated replacement specification is produced.

- o The base specification [[RFC7530](#)].
- o The migration-related update [[RFC7931](#)].
- o An eventual RFC based on the current document.

The section types are as follows. See [Appendix A](#) for a classification of each section of the current document.

- o An explanatory section does not contain any material that is meant

to update the specification of NFS version 4.0. Such sections may contain explanation about why and how changes are to be done, but do not include any text that is to update [[RFC7530](#)] or appear in an eventual consolidated document.

- o A replacement section contains text that is to replace and thus supersede text within [[RFC7530](#)] and then appear in an eventual consolidated document.
- o An additional section contains text which, although not replacing anything in [[RFC7530](#)], will be part of the specification of NFS version 4.0 and will be expected to be part of an eventual consolidated document.
- o An editing section contains some text that replaces text within [[RFC7530](#)], although the entire section will not consist of such text and will include other text as well. Such sections make relatively minor adjustments in the existing NFS version 4.0 specification which are expected to be reflected in an eventual

consolidated document. Generally such replacement text appears as a quotation, possibly taking the form of an indented set of paragraphs.

[3.3.](#) Document Goals

The goals of this document are as follows:

- o To provide NFS version 4.0 with a means of trunking discovery, compatible with the means of trunking detection introduced by [[RFC7931](#)].
- o To describe how NFS version 4.0 clients are to handle the presence of multiple network addresses associated to the same server, when recovering from a replication and migration event.
- o To describe how NFS version 4.0 clients are to handle changes in the location attributes returned, including those that indicate changes in the responding NFS version 4.0 server's trunking configuration.

The current document pursues these goals by presenting a set of

updates to [\[RFC7530\]](#) as summarized in [Section 4](#) below.

4. Overview of changes in [RFC7530 Section 8](#)

With a few small exceptions (see below), all of the updates to [\[RFC7530\]](#) to provide support for trunking using the `fs_locations` attribute apply to [Section 8](#) of that document, entitled "Multi-Server Namespace".

- o [Section 5](#) replaces [Section 8.1 of \[RFC7530\]](#), entitled "Location Attributes". This section has been reorganized and extended to explicitly allow the use of `fs_locations` to provide trunking-related information that appropriately interacts with the migration, replication and referral features of `fs_location`. Terminology used to describe the interactions is added.
- o [Section 6](#) updates [Section 8.4 of \[RFC7530\]](#), entitled "Uses of Location Information". This section comprises the bulk of the updates. Each paragraph of [Section 8.4](#) and its sub-sections has been reviewed to clarify the provision of trunking-related information using the `fs_locations` attribute.
 - * [Section 6.1](#) replaces the introductory material within [Section 8.4 of \[RFC7530\]](#).

- * [Section 6.2](#) is to be added after the introductory material within [Section 8.4 of \[RFC7530\]](#).
 - * [Section 6.3](#) replaces [Section 8.4.1 of \[RFC7530\]](#), entitled "File System Replication".
 - * [Section 6.4](#) replaces [Section 8.4.2 of \[RFC7530\]](#), entitled "File System Migration".
 - * [Section 6.5](#) is to be added after the updated [Section 8.4.2 of \[RFC7530\]](#).
-
- o [Section 7](#) replaces [Section 8.5 of \[RFC7530\]](#), entitled "Location Entries and Server Identity". The last paragraph of the existing section has been removed.

- o A small set of updates outside [Section 8 of \[RFC7530\]](#) are presented in [Section 8](#).
- o [Section 9](#) introduces additional security considerations that are to be added to those within [Section 19 of \[RFC7530\]](#), entitled "Security Considerations".

5. Location Attributes (as Updated)

The `fs_locations` RECOMMENDED attribute allows specification of file system locations where the data corresponding to a given file system may be accessed. This attribute represents such file system instances as a server address target (as either a DNS name representing one or more IP addresses, or a literal IP address) together with the path of that file system within the associated single-server namespace. Individual `fs_location` entries can express trunkable addresses, locations of file system replicas on other servers, migration targets, or pure referrals.

We introduce the following terminology:

- o Two network addresses connected to the same server are said to be server-trunkable.
- o Trunking detection refers to ways of deciding whether two specific network addresses are connected to the same NFSv4 server.
- o Trunking discovery is a process by which a client using one network address can obtain other addresses that are server-trunkable with it.

Regarding terminology relating to attributes used in trunking discovery and other multi-server namespace features:

- o Location entries (`fs_location4`, defined in [\[RFC7530\] Section 2.2.6](#)) are the individual file system locations in the `fs_locations` attribute (defined in [\[RFC7530\] Section 2.2.7](#)).
- o Location elements are derived from location entries. If a

location entry specifies an IP address there is only a single corresponding location element. Location entries that contain a host name are resolved by the client, and may result in one or more location elements.

- o All location elements consist of a location address, which is the IP address of an interface to a server, and an fs name, which is the location of the file system within the server's pseudo-fs.
- o The fs name is empty if the server has no pseudo-fs and only a single exported file system at the root filehandle.

6. Updates to [RFC7530 Section 8.4](#) (Uses of Location Information)

The subsections below provide replacement sections for existing sections within [Section 8.4 of \[RFC7530\]](#) or new sub-sections to be added to that section.

6.1. Introduction to uses of Location Information (as updated)

The location-bearing attribute `fs_locations` provides, together with the possibility of absent file systems, a number of important facilities in providing reliable, manageable, and scalable data access.

When a file system is present, these attributes can provide alternative locations, to be used to access the same data, in the event of server failures, communications problems, or other difficulties that make continued access to the current file system impossible or otherwise impractical. Provision of such alternative locations is referred to as "replication".

One type of replication is trunking, where the location entries do not in fact reside on different servers, but are instead different network paths to the same server. A client may use location elements simultaneously to provide higher-performance access to the target file system. The client utilizes trunking detection and/or discovery (see [Section 6.2](#)) to determine if two location elements are server-trunkable.

When a file system is present and subsequently becomes absent,

clients can be given the opportunity to have continued access to their data, at an alternative location. Transfer of the file system contents to the new location is referred to as "migration". See [Section 6.4](#) and [Section 6.5](#) (of the current document) for details.

Alternative locations may be physical replicas of the file system data or, in the case of various forms of server clustering, another server providing access to the same physical file system. The client's responsibilities in dealing with this transition depend on the specific nature of the new access path as well as how and whether data was in fact migrated. These issues will be discussed in detail below.

Where a file system was not previously present, specification of file system location provides a means by which file systems located on one server can be associated with a namespace defined by another server, thus enabling the creation of a multi-server namespace. A designation of such a location, in place of an absent file system, is called a "referral". A particularly important case is that of a "pure referral", in which the absent file system has never been present on the source server.

Because client support for location-related attributes is OPTIONAL, a server may (but is not required to) take action to hide migration and referral events from such clients, by acting as a proxy, for example.

[6.2.](#) Trunking Discovery and Detection (to be added)

Trunking detection refers to a way for an NFSv4 client to determine whether two independently acquired network addresses are connected to the same NFSv4 server. [Section 5.8 of \[RFC7931\]](#) describes an OPTIONAL means by which it can be determined if two server network addresses correspond to the same server instance. Without trunking detection, a client has no way to determine that two network addresses are server-trunkable.

In the context of NFS version 4.0, trunking detection requires that the client support the Uniform Client ID String approach (UCS), described in [Section 5.6 of \[RFC7931\]](#). Any NFS version 4.0 client that supports migration or trunking detection needs to present a Uniform Client ID String to all servers. If it does not do so, it will be unable to perform trunking detection.

Trunking discovery is the process by which an NFSv4 client using one server network address can obtain other server addresses that are trunkable with it; i.e., the set of addresses connected to the same server instance. Location entries that specify a server host name

that resolves via DNS into multiple addresses provide a list of server-trunkable addresses.

An NFS version 4.0 client can discover a set of server-trunkable network addresses in a number of ways:

1. If the client is accessing a server using its host name, that host name can be resolved to one or more IP addresses using DNS. If multiple addresses are present in the DNS query result, these addresses are server-trunkable and can be used together to access the server.
2. A client connected to a server without knowledge of its host name can obtain the value of a location attribute (i.e., `fs_locations`). Where a location entry within that attribute specifies a server host name, DNS can be used to obtain one or more network addresses corresponding to that host name. In cases in which one of those addresses is the address being used, the other addresses corresponding to that host name are server-trunkable and can be used to access the server.
3. A client can obtain the value of an `fs_location` attribute and use location entries that specify network addresses. When there is a means of trunking detection available all of addresses that are determined to correspond to the same server can be used to access that server.

[6.3](#). File System Replication and Trunking (as updated)

On first access to a file system, the client should obtain the value of the set of alternative locations by interrogating the `fs_locations` attribute. Trunking discovery and/or detection can then be applied to the location entries to separate the potential server-trunkable addresses from the replica addresses that provide alternative locations of the file system. Server-trunkable addresses may be used simultaneously to provide higher performance through the exploitation of multiple paths between client and target file system.

In the event that server failures, communications problems, or other difficulties make continued access to the current file system impossible or otherwise impractical, the client can use the alternative locations as a way to get continued access to its data. See [Section 6.5](#) (of the current document) for more detail.

[6.4.](#) File System Migration (as updated)

When a file system is present and becomes absent, clients can be given the opportunity to have continued access to their data, at an alternative location, as specified by the `fs_locations` attribute. Typically, a client will be accessing the file system in question, get an `NFS4ERR_MOVED` error, and then use the `fs_locations` attribute to determine the new location of the data. See [Section 6.5](#) (of the current document) for more detail.

Such migration can be helpful in providing load balancing or general resource reallocation. The protocol does not specify how the file system will be moved between servers. It is anticipated that a number of different server-to-server transfer mechanisms might be used, with the choice left to the server implementer. The NFSv4 protocol specifies the method used to communicate the migration event between client and server.

When an alternative location is designated as the target for migration, it must designate the same data. Where file systems are writable, a change made on the original file system must be visible on all migration targets. Where a file system is not writable but represents a read-only copy (possibly periodically updated) of a writable file system, similar requirements apply to the propagation of updates. Any change visible in the original file system must already be effected on all migration targets, to avoid any possibility that a client, in effecting a transition to the migration target, will see any reversion in file system state.

[6.5.](#) Interaction of Trunking, Migration, and Replication (to be added)

When the set of network addresses designated by a location attribute changes, `NFS4ERR_MOVED` might or might not result. In some of the cases in which `NFS4ERR_MOVED` is returned migration has occurred, while in others there is a shift in the network addresses used to access a particular file system (no migration occurred).

1. When the list of network addresses is a superset of that previously in effect, there is no need for migration or any other

sort of client adjustment. Nevertheless, the client is free to use an additional address in the replacement list if that address provides another path to the same server. Or, the client may use an additional address in the replacement list if server addresses it is currently using become unavailable without warning.

2. When the list of networks addresses is a subset of that previously in effect, immediate action is not needed if an address missing in the replacement list is not currently in use

by the client. The client should avoid using it in the future, whether the address is for a replica or a potential additional path to the server being used.

3. When an address being removed is one of a number of paths to the current server, the client may continue to use it until NFS4ERR_MOVED is received. This is not considered a migration event unless the last available path to the server has become unusable.

When migration does occur, multiple addresses may be in use on the server previous to migration and multiple addresses may be available for use on the destination server.

With regard to the server in use, it may be that return of NFS4ERR_MOVED indicates that a particular network address is no longer to be used, without implying that migration of the file system to a different server is needed. In light of this possibility, clients are best off not concluding that migration has occurred until concluding that all the network addresses known to be associated with the server are not usable.

It should be noted that the need to defer this determination is not absolute. If a client is not aware of all network addresses for any reason, it may conclude that migration has occurred when it has not and treat a switch to a different server address as if it were a migration event. This is generally harmless since the use of the same server via a new address will appear as a successful Transparent State Migration.

While significant harm will not arise from this misapprehension, it can give rise to disconcerting situations. For example, if a lock

has been revoked during the address shift, it will appear to the client as if the lock has been lost during migration, normally calling for it to be recoverable via an fs-specific grace period associated with the migration event.

With regard to the destination server, it is desirable for the client to be aware of all the valid network addresses that can be used to access the destination server. However, there is no need for this to be done immediately. Implementations can process the additional location elements in parallel with normal use of the first valid location entry found to access the destination.

[7.](#) Location Entries and Server Identity Update (as updated)

As mentioned above, a single location entry may have a server address target in the form of a DNS name that may represent multiple IP addresses, while multiple location entries may have their own server address targets that reference the same server.

When server-trunkable addresses for a server exist, the client may assume that for each file system in the namespace of a given server network address, there exist file systems at corresponding namespace locations for each of the other server network addresses. It may do this even in the absence of explicit listing in `fs_locations`. Such corresponding file system locations can be used as alternative locations, just as those explicitly specified via the `fs_locations` attribute.

[8.](#) Updates to [RFC7530](#) Outside Section Eight

Since the existing description of `NFS4ERR_MOVED` (in [Section 13.1.2.4 of \[RFC7530\]](#)) does not take proper account of trunking, it needs to be modified by replacing the first two sentences of the description with the following material:

The file system that contains the current filehandle object cannot be accessed using the current network address. It may be

accessible using other network addresses connected to the same server, it may have been relocated to another server, or it may never have been present.

9. Security Considerations

The Security Considerations section of [\[RFC7530\]](#) needs the additions below to properly address some aspects of trunking discovery, referral, migration and replication.

The possibility that requests to determine the set of network addresses corresponding to a given server might be interfered with or have their responses corrupted needs to be taken into account.

- o When DNS is used to convert NFS server host names to network addresses and DNSSEC [\[RFC4033\]](#) is not available, the validity of the network addresses returned cannot be relied upon. However, when the client uses RPCSEC_GSS [\[RFC7861\]](#) to access NFS servers, it is possible for mutual authentication to detect invalid server addresses. Other forms of transport layer security (e.g., [\[RFC5246\]](#)) can also offer strong authentication of NFS servers.

- o Fetching location information SHOULD be performed using RPCSEC_GSS with integrity protection, as previously explained in the Security Considerations section of [\[RFC7530\]](#). Making a request of this sort without using strong integrity protection permits corruption during transit of returned location information. The client implementer needs to recognize that using such information to access an NFS server without use of RPCSEC_GSS (e.g., by using AUTH_SYS) can result in the client interacting with an unverified network address that is posing as an NFS server.
- o Despite the fact that it is a REQUIREMENT of [\[RFC7530\]](#) that "implementations" provide "support" for use of RPCSEC_GSS, it cannot be assumed that use of RPCSEC_GSS is always available between any particular client-server pair.
- o Returning only network addresses to a client with no trusted DNS resolution service can hamper its ability to use

RPCSEC_GSS.

Therefore an NFS server SHOULD present location entries that correspond to file systems on other servers using only host names. This enables the client to interrogate the fs_locations on the destination server to obtain trunking information (as well as replica information) using RPCSEC_GSS with integrity, validating the name provided while assuring that the response has not been corrupted.

When RPCSEC_GSS is not available on an NFS server, returned location information is subject to corruption during transit and cannot be relied upon. In the case of a client being directed to another server after NFS4ERR_MOVED, this could vitiate the authentication provided by the use of RPCSEC_GSS, since the destination server can represent itself as the server to which the client was erroneously directed. [cel: this is still confusing.]

When a location attribute is fetched upon connecting with an NFS server, it is best for the client to ignore trunking and replica information when RPCSEC_GSS with integrity protection cannot be used. [cel: why then fetch location information in this case?]
[cel: should this be normative advice?]

When location information cannot be verified, it can be subjected to additional filtering to prevent the client from being inappropriately directed. [cel: why can't filtering be used in the previous paragraph?]

To summarize considerations regarding the use of RPCSEC_GSS in fetching location information, consider the following possibilities for requests to interrogate location information, with interrogation approaches on the referring and destination servers arrived at separately:

- o The use of RPCSEC_GSS with integrity protection is RECOMMENDED in all cases, since the absence of integrity protection exposes the client to the possibility of the results being modified in transit.

- o The use of RPCSEC_GSS without integrity protection to fetch location information SHOULD NOT be attempted. In cases of migration or referral, this applies both to the referring and destination servers. [cel: how is this normatively different than the first bullet?]
- o The use of requests issued without RPCSEC_GSS (e.g., using AUTH_SYS), while undesirable, might be unavoidable in some cases. Unprotected returned location information should be subject to filtering to eliminate the possibility that the client would treat an invalid address as if it were a trusted NFSv4 server. The specifics will vary depending on the degree of network isolation and whether the request is to the referring or destination servers.

10. IANA Considerations

This document does not require actions by IANA.

11. References

11.1. 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>>.
- [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>>.
- [RFC7931] Noveck, D., Ed., Shivam, P., Lever, C., and B. Baker, "NFSv4.0 Migration: Specification Update", [RFC 7931](#), DOI 10.17487/RFC7931, July 2016, <<https://www.rfc-editor.org/info/rfc7931>>.

- [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>>.

11.2. Informative References

- [I-D.ietf-nfsv4-migration-issues]
Noveck, D., Shivam, P., Lever, C., and B. Baker, "NFSv4 Migration and Trunking: Implementation and Specification Issues", [draft-ietf-nfsv4-migration-issues-13](#) (work in progress), May 2017.
- [RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", [RFC 4033](#), DOI 10.17487/RFC4033, March 2005, <<https://www.rfc-editor.org/info/rfc4033>>.
- [RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", [RFC 5246](#), DOI 10.17487/RFC5246, August 2008, <<https://www.rfc-editor.org/info/rfc5246>>.
- [RFC5661] Shepler, S., Ed., Eisler, M., Ed., and D. Noveck, Ed., "Network File System (NFS) Version 4 Minor Version 1 Protocol", [RFC 5661](#), DOI 10.17487/RFC5661, January 2010, <<https://www.rfc-editor.org/info/rfc5661>>.
- [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>>.

Appendix A. Section Classification

All sections of this document are considered explanatory with the following exceptions.

- o Sections [5](#) and [6.1](#) are replacement sections.
- o [Section 6.2](#) is an additional section.
- o Sections [6.3](#) and [6.4](#) are replacement sections.
- o [Section 6.5](#) is an additional section.
- o [Section 7](#) is a replacement section.
- o [Section 8](#) is an editing section.

- o [Section 9](#) is an additional section.

Acknowledgments

The authors wish to thank Andy Adamson, who wrote the original version of this document. All the innovation in this document is the result of Andy's work, while mistakes are best ascribed to the current authors.

The editor wishes to thank Greg Marsden of Oracle for his support of this work, and Rob Thurlow of Oracle for review and suggestions.

Special thanks go to Transport Area Director Spencer Dawkins, NFSV4 Working Group Chair Spencer Shepler, and NFSV4 Working Group Secretary Thomas Haynes for their support.

Authors' Addresses

Charles Lever (editor)
Oracle Corporation
1015 Granger Avenue
Ann Arbor, MI 48104
United States of America

Phone: +1 248 816 6463
Email: chuck.lever@oracle.com

David Noveck
NetApp
1601 Trapelo Road
Waltham, MA 02451
United States of America

Phone: +1 781 572 8038
Email: davenoveck@gmail.com

