

**NFS and MQ**

**Authors:** Ian Daniel

**Contributors:**  Stewart Bird, Brett Truhler

**Document Version:** V0.2

**Date:** February 2016

Status: Draft

**CONFIDENTIAL INFORMATION**

This document contains information proprietary to Thomson Reuters and may not be reproduced, disclosed or used in whole or part without express permission of Thomson Reuters.

© Thomson Reuters 2016

Contents

[1 Introduction 3](#_Toc443850729)

[1.1 Management Summary 3](#_Toc443850730)

[1.2 Change History 3](#_Toc443850731)

[1.3 Distribution List 4](#_Toc443850732)

[1.4 Glossary 4](#_Toc443850733)

[2 Overview 5](#_Toc443850734)

[2.1 NFS v3 5](#_Toc443850735)

[2.2 NFS v4 5](#_Toc443850736)

[3 MQ Series and NFS 5](#_Toc443850737)

[3.1 HA Configuration 5](#_Toc443850738)

[3.2 NFS Locking in a HA Configuration 5](#_Toc443850739)

[4 TR NFS Standards 5](#_Toc443850740)

[4.1 NFS v3 5](#_Toc443850741)

[4.2 NFS v4 5](#_Toc443850742)

[5 TR NFS v3/4 Mixed Deployments 6](#_Toc443850743)

# Introduction

## Management Summary

This document details the reasons for using NFS v4 in the TR environment for MQ Series applications as an exception to the standard NFS v3 shares normally provided.

## Change History

|  |  |  |  |
| --- | --- | --- | --- |
| **Ver** | **Date** | **Author** | **Key Changes** |
| 0.1 | February 2016 | Ian Daniel | Initial Version |
| 0.2 | February 2016 | Ian Daniel | Updated wording |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Distribution List

|  |  |
| --- | --- |
| **Name** | **Role** |
| Storage Engineering | Reviewer |
| Storage Delivery | Reviewer |
| Storage Architecture | Reviewer |

## Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| NFS | A distributed file system protocol originally developed by Sun Microsystems in 1984, allowing a user on a client computer to access files over a computer network much like local storage is accessed. |
| vfiler | A logical storage virtual server, also known as a Storage Virtual Machine (SVM), which contains interfaces, Volumes, and configuration information such as access control details. |
| MQ | Message Queue software enabling multiple clients and servers to interact. |

# Overview

## ****NFS v3****

NFS Versions 2 and 3 are stateless protocols. The NFS (Versions 2 and 3) protocol does not support file locking, but the NFS environment supports an ancillary protocol called NLM, which originally stood for "Network Lock Manager." When an NFS filesystem on an NFS client gets a request to lock a file, instead of an NFS remote procedure call, it generates an NLM remote procedure call.

## ****NFS v4****

NFS Version 4 introduces state. An NFS Version 4 client uses state to notify an NFS Version 4 server of its intentions on a file: locking, reading, writing and so on.

An NFS Version 4 server can return information to a client about what other clients have intentions on a file to allow a client to cache file data more aggressively via delegation. To help keep state consistent more sophisticated client and server reboot recovery mechanisms are built in to the NFS Version 4 protocol. NFS Version 4 introduces support for byte-range locking and share reservation. Locking in NFS Version 4 is lease-based so an NFS Version 4 client must maintain contact with an NFS Version 4 server to continue extending its open and lock leases.

# MQ Series and NFS

## ****HA Configuration****

MQ can be configured to use a NFS share (or shared SAN LUN) to provide HA capabilities. Basically you can run as many brokers as you wish from the same shared file system directory. The first broker to get an exclusive lock on the file is the master broker. If that broker dies and releases the lock then another broker takes over. The slave brokers sit in a loop trying to grab the lock from the master broker.

## ****NFS Locking in a HA Configuration****

The ActiveMQ documentation makes the following statement regarding NFS v3 locking.

**NFSv3 Warning**

In the event of an abnormal NFSv3 client termination (i.e., the ActiveMQ master broker), the NFSv3 server will not timeout the lock that is held by that client. This effectively renders the ActiveMQ data directory inaccessible because the ActiveMQ slave broker can't acquire the lock and therefore cannot start up. The only solution to this predicament with NFSv3 is to reboot all ActiveMQ instances to reset everything.

Use of NFSv4 is another solution because its design includes timeouts for locks. When using NFSv4 and the client holding the lock experiences an abnormal termination, by design, the lock is released after 30 seconds, allowing another client to grab the lock.

# TR NFS Standards

## ****NFS v3****

This is our standard for all file shares and as such is what should normally be deployed and used within TR.

## ****NFS v4****

The only supported use of NFS v4 within TR is for queuing services such as IBM MQ, Rabbit MQ etc.

# TR NFS v3/4 Mixed Deployments

In situations where there is a mixture of applications on a server and MQ is present a vfiler/vserver should be dedicated to MQ and have only those mount points available via the NFS v4 protocol. This ensures that only MQ related mounts can be mounted using NFS v4.