

**NFS v4 on cDOT**

**Authors:** Ian Daniel,Brett Truhler

**Contributors:**  David Ng, Craig Goettig

**Document Version:** V0.8

**Date:** August 2017

Status: Released

**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](#_Toc440289205)

[1.1 Management Summary 3](#_Toc440289206)

[1.2 Change History 3](#_Toc440289207)

[1.3 Distribution List 4](#_Toc440289208)

[1.4 Glossary 4](#_Toc440289209)

[2 NFS v4 Configuration 5](#_Toc440289210)

[2.1 Pre-requisites 5](#_Toc440289211)

[2.2 Vserver and LIF creation 5](#_Toc440289212)

[2.2.1 Create Vserver (replace hyphen with underscore in vserver rootvolume name) 5](#_Toc440289213)

[2.2.2 Create LIF with default route and failover group 5](#_Toc440289214)

[2.3 Volume creation 6](#_Toc440289215)

[2.3.1 Create Volume 6](#_Toc440289216)

[2.3.2 Create export policies (repeat this step for each volume) 6](#_Toc440289217)

[2.3.3 Create the default export-policy rule (repeat this command for every nfs client) 6](#_Toc440289218)

[2.3.4 Create volume export policy rule (repeat this command for each volume and nfs client) 6](#_Toc440289219)

[2.4 Qtree Creation 6](#_Toc440289220)

[2.4.1 Create Qtree 6](#_Toc440289221)

[2.4.2 Create Quota 6](#_Toc440289222)

[2.4.3 Apply Quota 6](#_Toc440289223)

[volume quota on -vserver <vsname> -volume <volume\_name> 6](#_Toc440289224)

[2.5 Primary job schedule and snapshot policy configuration 6](#_Toc440289225)

[2.5.1 Create job schedule and snapshot policy: 6](#_Toc440289226)

[2.5.2 Setup snap autodelete on volumes (run for each SNAP volume) 6](#_Toc440289227)

[2.6 Add volumes to export & snapshot policies 7](#_Toc440289228)

[2.7 DNS configuration 7](#_Toc440289229)

[2.7.1 Setup DNS on a Vserver 7](#_Toc440289230)

[2.8 Showmount script user 7](#_Toc440289231)

[2.9 NFS v4 Configuration 7](#_Toc440289232)

[2.9.1 Enable NFSv4 7](#_Toc440289233)

[2.9.2 Ensure Numeric ID option is enabled 7](#_Toc440289234)

[2.10 QoS policy group creation 8](#_Toc440289235)

[2.10.1 Create QoS policy group per volume and apply 8](#_Toc440289236)

[2.11 Cluster and Vserver Peering 8](#_Toc440289237)

[2.11.1 Confirm that cluster peering has been enabled 8](#_Toc440289238)

[2.11.2 Create the cluster peer (skip this step if cluster peering has been configured) 8](#_Toc440289239)

[2.11.3 Confirm if vserver peering has been configured 8](#_Toc440289240)

[2.11.4 Create vserver peering on the destination system (skip this step if vserver peering has been configured) 8](#_Toc440289241)

[2.11.5 Accept the vserver peering on the source system 8](#_Toc440289242)

[2.12 SnapVault configuration 9](#_Toc440289243)

[2.12.1 Create secondary volumes for SnapVault as type “DP” on the destination cluster 9](#_Toc440289244)

[2.12.2 Create a cron job schedule if it does not exist in the destination 9](#_Toc440289245)

[2.12.3 Configure a snapmirror policy on the destination 9](#_Toc440289246)

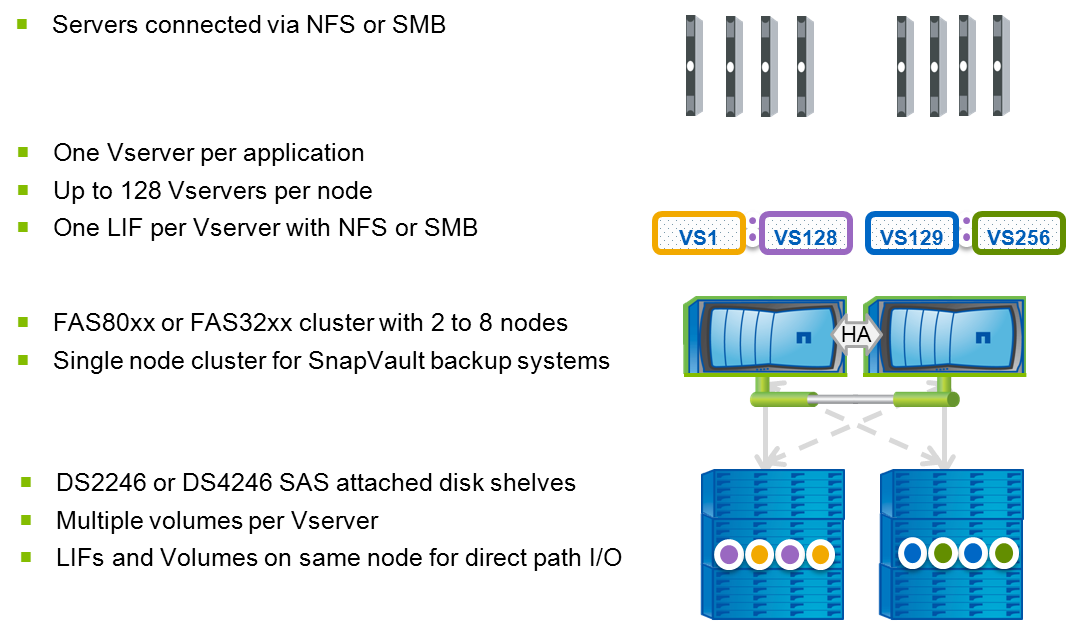
[2.12.1 Initialize SnapVault relationship on the destination 9](#_Toc440289247)

# Introduction

## Management Summary

This document details the process used to configure NFS v4 in cDOT. At this time the use of NFS v4 is a standard only for use with MQ deployments in Thomson Reuters.

The diagram below shows an overview of the current cDOT architecture.



## Change History

|  |  |  |  |
| --- | --- | --- | --- |
| **Ver** | **Date** | **Author** | **Key Changes** |
| 0.1 | January 2016 | Ian Daniel | Initial Version |
| 0.2 | January 2016 | Ian Daniel | Updated UID/GID text |
| 0.3 | January 2016 | Ian Daniel | Snapshot schedules edited |
| 0.4 | January 2016 | Ian Daniel | Modified to use Numeric ID mapping |
| 0.5 | January 2016 | Ian Daniel | Removed duplicated command |
| 0.6 | January 2016 | Ian Daniel | Added qtree and quota |
| 0.7 | February 2016 | Ian Daniel | Added MQ note to management summary. |
| 0.8 | August 2017 | Ian Daniel | Updated default export policy |
|  |  |  |  |

## Distribution List

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

## Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| cDOT | clustered Data ONTAP |
| Vserver | A logical storage virtual server, also known as a Storage Virtual Machine (SVM), which contains LIFs, Volumes, and configuration information such as access control details. |
| LIF | Logical Interface – a cDOT logical network interface with an IP address, assigned to a single Vserver. |

# NFS v4 Configuration

## Pre-requisites

* Confirm the target aggregate has sufficient capacity
* Confirm the target cluster has an NFS license
* Make a note of required accounts on the client systems so they can be created on the vserver.
* Make a note of required groups on the client systems so they can be created on the vserver.
* Confirm there is network connectivity between the client and the cluster.
* Make a note of the required NFS v4 Domain.

## ****Vserver and LIF creation****

### Create Vserver (replace hyphen with underscore in vserver rootvolume name)

vserver create -vserver <vsname> -rootvolume <vsname>\_root -aggregate <aggrname> -ns-switch file -nm-switch file -rootvolume-security-style unix -language <language>

vserver show

**Note:** vserver language is inherited by volumes and should be set to en\_US

### Create LIF with default route and failover group

network interface create -vserver <vsname> -lif <vsname>-lif-<lif#> -role data -data-protocol nfs -home-node <node> -home-port <port> -address <ip> -netmask <netmask> -status-admin up -firewall-policy mgmt -failover-group <group>

network routing-groups route create -vserver <vsname> -routing-group d<network>/<mask> -destination 0.0.0.0/0 -gateway <gateway>

vserver show

network interface show

network interface show -failover

network routing-groups route show –vserver <vsname>

## ****Volume creation****

### Create Volume

volume create -vserver <vserver> -volume <volume\_name> -aggregate <aggregate\_name> -size <size\_of\_volume> -junction-path /<volume\_name>

volume mount -vserver <vserver> -volume <volume\_name> -junction-path /<volume\_name>

volume show -vserver <vserver> -junction

### Create export policies (repeat this step for each volume)

vserver export-policy create –vserver <vsname> –policyname <volume\_name>

### Create the default export-policy rule (repeat this command for every nfs client)

vserver export-policy rule create -vserver <vsname> -policyname default -clientmatch <nfsclient> -rorule sys -rwrule never -superuser never

### Create volume export policy rule (repeat this command for each volume and nfs client)

vserver export-policy rule create -vserver <vsname> -policyname <volume\_name> -clientmatch <nfsclient> -rorule sys -rwrule sys -superuser sys -protocol nfs4

## ****Qtree Creation****

### ****Create Qtree****

volume qtree create -vserver <vsname> -volume <volume\_name> -qtree <qtree> -security-style unix

### ****Create Quota****

volume quota policy rule create -vserver <vsname> -policy-name default -volume <volume\_name> -type tree -target <qtree> -disk-limit <quota\_limit>

### ****Apply Quota****

### volume quota on -vserver <vsname> -volume <volume\_name>

## ****Primary job schedule and snapshot policy configuration****

### Create job schedule and snapshot policy:

job schedule cron create -name <volume\_name> -minute <min> -hour <hour>

volume snapshot policy create -vserver <vsname> -policy <volume\_name> -enabled true -schedule1 <volume\_name> -count1 7 -snapmirror-label1 snapvault -prefix1 sv\_<volume\_name>

job schedule cron show -name <vol\_name>

volume snapshot policy show –vserver <vsname>

### Setup snap autodelete on volumes (run for each SNAP volume)

volume modify -vserver <vsname> -volume <volname> -space-mgmt-try-first snap\_delete

volume snapshot autodelete modify -vserver <vsname> -volume <vol\_name> -enabled true

volume snapshot autodelete modify -vserver <vsname> -volume <vol\_name> -trigger snap\_reserve

volume show -vserver <vsname> -fields space-mgmt-try-first

volume snapshot autodelete show -vserver <vsname>

## Add volumes to export & snapshot policies

volume modify -volume <volume\_name> -policy <volume\_name> -snapshot-policy <volume\_name> -vserver <vserver>

## ****DNS configuration****

### Setup DNS on a Vserver

vserver services dns create -vserver <vsname> -domains <domainname> -name-servers <comma\_separate\_name\_server\_list>

vserver services dns show

## Showmount script user

security login role create -role showmount -cmddirname "vserver export-policy" -access readonly -vserver <vserver>

security login role create -role showmount -cmddirname volume -access readonly -vserver <vserver>

security login role create -role showmount -cmddirname "version" -access all -vserver <vsname>

security login create -username shwmnt -application ontapi -authmethod password -role showmount -vserver <vserver>

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

### Enable NFSv4

vserver nfs create -vserver <vsname> -access true –v3 disabled -v4.0 enabled -v4-id-domain <NFS\_V4\_DOMAIN>

vserver nfs show

**Note:** On the client the NFS V4 Domain can be found using: grep -i "domain =" /etc/idmapd.conf

**Example**

# grep -i "domain =" /etc/idmapd.conf

Domain = int.westgroup.com

### Ensure Numeric ID option is enabled

This option will ensure the v4 mount behaves the same way as a v3 mount in that it returns UID/GID to the client and the client displays the file ownership by mapping them to users/groups. This requires diag privilege.

cluster01::\*> nfs server show -vserver nfsv4-testvs01 -fields v4-numeric-ids

vserver        v4-numeric-ids

-------------- --------------

nfsv4-testvs01 enabled

This option can be set using but is normally enabled by default:

nfs server modify -vserver nfsv4-testvs01 -v4-numeric-ids  enabled

## ****QoS policy group creation****

### Create QoS policy group per volume and apply

qos policy-group create -policy-group <volname> -vserver <vsname> -max-throughput 6000iops

volume modify -vserver <vsname> -volume <volume> -qos-policy-group <volname>

qos policy-group show

volume show –vserver <vsname> -fields qos-policy-groupSnapvault Configuration

## Cluster and Vserver Peering

### Confirm that cluster peering has been enabled

cluster peer show

### Create the cluster peer (skip this step if cluster peering has been configured)

cluster peer create -peer-addrs <remote\_ICL\_IP1,remote\_ICL\_IP2> -username admin

cluster peer show

### Confirm if vserver peering has been configured

vserver peer show

### Create vserver peering on the destination system (skip this step if vserver peering has been configured)

vserver peer create -vserver <destination\_vserver> -peer-cluster <source\_cluster> -peer-vserver <source\_vserver> -applications snapmirror

vserver peer show

### Accept the vserver peering on the source system

vserver peer accept -vserver <source\_vserver> -peer-vserver <destination\_vserver>

vserver peer show

## SnapVault configuration

### Create secondary volumes for SnapVault as type “DP” on the destination cluster

volume create -vserver <vserver> -volume <volume\_name> -aggregate <aggr\_name> -size <size> -security-style unix -space-guarantee none -percent-snapshot-space 0 -language <vol\_language> -type DP

volume show

### Create a cron job schedule if it does not exist in the destination

job schedule cron create -name xdp\_<hour> -minute 00 -hour <hour>

job schedule show

### Configure a snapmirror policy on the destination

* **7 day retention will have a snapshot count of 7 on the secondary**
* **14 day retention will have a snapshot count of 14 on the secondary**
* **30 day retention will have a snapshot count of 30 on the secondary**
* **45 day retention will have a snapshot count of 45 on the secondary**

snapmirror policy create -vserver <vserver> -policy <volume>

snapmirror policy add-rule -vserver <vserver> -policy <volume> -snapmirror-label snapvault -keep <retention#>

snapmirror show -destination-path \* -fields Schedule

snapmirror policy show

### Initialize SnapVault relationship on the destination

snapmirror create -source-path <source\_vserver>:<source\_volume> -destination-path <destination\_vserver>:<destination\_volume> -type XDP -schedule <schedule\_name> -policy <policy\_name>

snapmirror initialize -destination-path <destination\_vserver>:<destination\_volume>

snapmirror show