Internal Use Only

THOMSON REUTERS

MySQL Deployment Guide

Synopsis: Deployment guide for MySQL

Segment: Storage D&E (NAS)

Authors: David Ng

Contributors:

Document Version: V4
Date: 12/11/2014
Document Status: Draft

Internal Use Only

Contents

1 Introduction

1.1 Management Summary

This document details the NetApp clustered Data ONTAP (cDOT) solution used for MySQL environments at Thomson Reuters. The storage system configuration of Vservers, Networking, Volumes, and Storage Efficiency will be covered in detail. The following drawing gives a high level overview of the cDOT shared storage environment solution at Thomson Reuters.
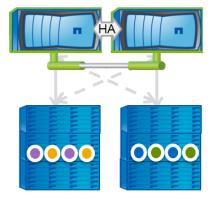
- Servers connected via NFS or SMB

- One Vserver per application
- Up to 128 Vservers per node
- One LIF per Vserver with NFS or SMB

- FAS80xx or FAS32xx cluster with 2 to 8 nodes
- Single node cluster for SnapVault backup systems

- DS2246 or DS4246 SAS attached disk shelves
- Multiple volumes per Vserver
- LIFs and Volumes on same node for direct path I/O

1.2 Assumptions

It is assumed the person(s) reading this document are conversant with NetApp hardware and software. They will also be conversant with the Linux operating systems, NFS protocol, and database systems at a high level.

1.3 References

	Document	Version	Date	Author
1				

1.4 Change History

Ver	Date	Author	Key Changes
1	11/20/2014	David Ng	initial

1.5 Distribution List

Name	Role
Storage Delivery	Storage provisioning team
Storage Support	Support of storage systems

1.6 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.
QoS	Quality of Service – introduced in cDOT 8.2 to provide workload monitoring and throughput rate limiting as desired.
WFA	OnCommand Workflow Automater – An automation framework application from NetApp, used for storage provisioning.

2 MySQL Deployment

2.1 Volume and Qtree layout

MySQL snap deployment will typically have 2 flexvols configured on a NetApp primary storage system, as show in the following table. The following table gives a high level description of the volumes and qtrees used in a configuration for TR. Archive logs setup is covered in a separate document and can be created via WFA.

Storage Volume	Storage Qtree	Mount Point	Description	Snap Reserve	Snap Vault
<cb#>_<bu>_<app>_ n01mysql1_nosnap	n01mysqldata1	/n01/mysqldata1	Binary logs	0%	No
	n01mysqltemp1	/n01/mysqltemp1	Temp – 10% of mysqldata1_snap		
	n01mysqladmin1	/n01/mysqladmin1	gen.log, slow.log, mysqld log, other admin files		
<cb#>_<bu>_<app>_ s01mysql1_snap	mysqldata1	/s01/mysqldata1	Datafiles, Write-ahead trx logs, config file, bin.log (only required if there is a snap retention)	20%	Yes

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

vserver create -vserver <vserver> -rootvolume <vserver>_root -aggregate <aggregate> -ns-switch file -nm-switch file -rootvolume-security-style unix -language C.UTF-8

vserver show

2.3 5.1.2 Create LIF with default route and failover group

network interface create -vserver <vserver> -lif <vserver>-lif-<lif#> -role data -data-protocol nfs - home-node <home_node> -home-port <home_port> -address <vserver_IP> -netmask <vserver_netmask> -status-admin up -firewall-policy mgmt -failover-group <failover_group>

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

vserver show

network interface show

network interface show –failover

Internal Use Only

network routing-groups route show –vserver <vsname>

2.4 Setup DNS on a Vserver

vserver services dns create -vserver <vserver>-domains <domain> -name-servers <nameserver1, nameserver2, etc.>

vserver services dns show

2.5 Backup user role, user, and SSH publickey configuration

security login role create -role db_backup_user -cmddirname "volume snapshot" -access all -vserver <vserver>

security login role create -role db_backup_user -cmddirname "set" -access all -vserver <vserver>

security login role create -role db_backup_user -cmddirname "job show" -access readonly -vserver <vserver>

security login role create -role db_backup_user -cmddirname "version" -access readonly -vserver <vserver>

security login role create -role oracle -cmddirname "df" -access readonly -vserver <vsname>

security login role create -role db_backup_user -cmddirname "snapmirror list-destinations" -access readonly -vserver <vserver>

security login create -username db_backup_user -application ssh -authmethod publickey -role db_backup_user -vserver <vserver>

security login publickey create -username db_backup_user -vserver <vserver> -publickey "ssh-dss AAAAB3NzaC1kc3MAAACBAMfLbRNLcOf3/D1rvwCLS6Xr93+eOxgYgrtYCcVu7v6k0UrDGXG/Qmkt OCiwuEZKazgajkzH5CWmbRH/8tg4xxdnwW6QGdVIj5asZNB4urU+yOGDfxGePC7GHvYd/qVjdJLOj 1HiZ+b3WZ55//ZVF3NYqoy9XtZieTemHuNEupcpAAAAFQDOYD7ccpRKDVIZnJP9kBhuGcql8QAAA IEAr22DSBBnG+gu8LeMAXs8rxaJuYvHgUgxyTJHYq78oR2wXiVNUgBUrSWf4HuzL4VtFnSW7iBA/il m0XRYN7xus78z8C9Jocc20evQkbsQNTWtM8VSznovp9E7ErS2U7rgwJiAgoezl+S8abdqZgpPY7Q6 NIqSetIV6KUyO1LHDUcAAACAPDiywxnJUTfodjIGPnV0XZ1nGNEQTpa17SJpn1WKati0NJOgrouara qK5zA2fRv8igqQHpLxPxdm1d1mQgt87LA2Cs9QL4Tbc7IwFBa8G/BFA5Fg5tvrHxU0E+24484Z/af7T RWxA8+l72OhOpWVumaIxQEZnI9BLgRORBKN2Tk= Snapvault backup key for Netapp db_backup_user account"

2.6 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>

2.7 Enable NFSv3

vserver nfs create -vserver <vserver>-access true -v3 enabled

vserver nfs show

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

vserver export-policy create –vserver <vsname> –policyname <volume_name>

volume show -vserver <vserver>

vserver export-policy rule show

2.9 Create the default export policy (repeat this step for each nfs client)

vserver export-policy rule create -vserver <vsname> -policyname default -clientmatch <nfsclient> -rorule sys -rwrule sys -superuser sys

2.10 Create export policy rule (repeat this step for each volume and nfs client)

vserver export-policy rule create -vserver <vsname> -policyname <volume_name> -clientmatch <nfsclient> -rorule sys -rwrule sys -superuser sys

2.11 Create the primary NOSNAP volume

volume create -vserver <vserver> -volume <nosnap_vol_name>_nosnap -aggregate <aggregate> -size <size> -policy default -snapshot-policy none -junction-path /<nosnap_vol_name>_nosnap -security-style unix -space-guarantee none -percent-snapshot-space 0 -language C.UTF-8

2.12 Create the primary SNAP volume (This volume is only required if a snap retention has been specified)

volume create -vserver <vserver>-volume <snap_vol_name>_snap -aggregate <aggregate> -size <size> -policy default -snapshot-policy none -junction-path /<snap_vol_name>_snap -security-style unix -space-guarantee none -percent-snapshot-space 20 -language C.UTF-8

2.13 Setup snap autodelete on volumes (repeat step for each SNAP volume)

volume modify -vserver <vserver>-volume <snap_vol_name>_snap -space-mgmt-try-first snap_delete

volume snapshot autodelete modify -vserver <vserver>-volume <snap_vol_name>_snap -enabled true

volume snapshot autodelete modify -vserver <vsname> -volume <vol_name> -trigger snap_reserve

volume snapshot autodelete show -vserver <vsname>

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

volume snapshot autodelete show -vserver <vserver>

2.14 Create primary qtrees with quotas

volume qtree create -vserver <vserver> -volume <nosnap_vol_name>_nosnap -qtree n01mysqldata1 -security-style unix

volume qtree create -vserver <vserver> -volume <nosnap_vol_name>_nosnap -qtree n01mysqltemp1 -security-style unix

volume qtree create -vserver <vserver> -volume <nosnap_vol_name>_nosnap -qtree n01mysqladmin1 -security-style unix

volume qtree create -vserver <vserver>-volume <snap_vol_name>_snap -qtree s01mysqldata1 -security-style unix

volume quota policy rule create -vserver <vserver>-policy-name default -volume <nosnap_vol_name>_nosnap -type tree -target n01mysqldata1 -disk-limit <size>

volume quota policy rule create -vserver <vserver> -policy-name default -volume <nosnap_vol_name>_nosnap -type tree -target n01mysqltemp1 -disk-limit <size>

volume quota policy rule create -vserver <vserver>-policy-name default -volume <nosnap_vol_name>_nosnap -type tree -target n01mysqladmin1 -disk-limit <size>

volume quota policy rule create -vserver <vserver> -policy-name default -volume <snap_vol_name>_snap -type tree -target s01mysqldata1

volume quota on -vserver <vserver>-volume <nosnap_vol_name>_nosnap

2.15 Create QoS policy group and apply it at the volume level (repeat for each volume)

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-group

3 Snapvault Configuration (Steps below are only required if this is a SNAP deployment)

3.1 Confirm that cluster and Vserver Peering

cluster peer show

3.2 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

3.3 Confirm if vserver peering has been configured

vserver peer show

3.4 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

3.5 Accept the vserver peering on the source system

vserver peer accept -vserver <source_vserver> -peer-vserver <destination_vserver>
vserver peer show

3.6 SnapVault configuration

There are two data protection configurations for MySQL: SNAP & NOSNAP. The 's01mysql1_snap' volume is vaulted in a SNAP config, but not a NOSNAP configuration.

3.7 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 C.UTF-8 -type DP

volume show

3.8 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

3.9 Configure a snapmirror policy on the destination (execute these command for s01mysql1_snap)

- **7 day retention will have a snapshot count of 14 on the secondary**
- **14 day retention will have a snapshot count of 28 on the secondary**
- **30 day retention will have a snapshot count of 60 on the secondary**
- **45 day retention will have a snapshot count of 90 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

3.10 Initialize SnapVault relationship on the destination (execute these command for s01mysql1_snap)

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