

# NFS

## NFS      NFS Server

NFS

[1]      NFS

```
[root@dlp ~]# dnf -y install nfs-utils
[root@dlp ~]# vi /etc/idmapd.conf
# line 5 : uncomment and change to your domain name
Domain = srv.world
[root@dlp ~]# vi /etc/exports
# create new
# for example, set [/home/nfsshare] as NFS share
/home/nfsshare 10.0.0.0/24(rw,no_root_squash)
[root@dlp ~]# mkdir /home/nfsshare
[root@dlp ~]# systemctl enable --now rpcbind nfs-server
```

[2]      Firewalld                  NFS

```
[root@dlp ~]# firewall-cmd --add-service=nfs
success
# if use NFSv3, allow follows, too
[root@dlp ~]# firewall-cmd --add-service={nfs3,mountd,rpc-bind}
success
[root@dlp ~]# firewall-cmd --runtime-to-permanent
success
```

	NFS
	NFS
	NFS      NFS
	IPPORT_RESERVED (1024)    Internet

no_wdelay	NFS
no_subtree_check	
root_squash	uid/gid 0      uid/gid      uid    gid
no_root_squash	
all_squash	uid    gid      NFS      FTP
no_all_squash	
anonuid=UID	uid    gid      PC/NFS
anongid=GID	annuid=UID

NFS      NFS

NFS      NFS      NFS

+-----+ | +-----+  
| [NFS    ] |10.0.0.30 | 10.0.0.51| [NFS    ] |  
| dlp.srv.world +-----+-----+ node01.srv.world |  
| | | |  
+-----+ +-----+

[1]    NFS

```
[root@node01 ~]# dnf -y install nfs-utils
[root@node01 ~]# vi /etc/idmapd.conf
# line 5 : uncomment and change to your domain name
Domain = srv.world
[root@node01 ~]# mount -t nfs dlp.srv.world: /home/nfsshare /mnt
[root@node01 ~]# df -hT
Filesystem                                Type      Size  Used Avail Use% Mounted on
devtmpfs                                devtmpfs  1.9G   0    1.9G   0% /dev
tmpfs                                    tmpfs     2.0G   0    2.0G   0% /dev/shm
tmpfs                                    tmpfs     783M  968K  782M   1% /run
/dev/mapper/fedora_fedora-root          xfs       15G   1.6G   14G  11% /
tmpfs                                    tmpfs     2.0G   4.0K   2.0G   1% /tmp
/dev/vda1                                xfs      1014M  195M   820M  20% /boot
tmpfs                                    tmpfs     392M   0    392M   0% /run/user/0
dlp.srv.world: /home/nfsshare            nfs4      15G   1.6G   14G  11% /mnt
# NFS share is mounted
```

```
# if mount with NFSv3, add [-o vers=3] option
[root@node01 ~]# mount -t nfs -o vers=3 dlp.srv.world: /home/nfsshare /mnt
[root@node01 ~]# df -hT /mnt
```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
dlp.srv.world: /home/nfsshare	nfs	15G	1.5G	14G	10%	/mnt

## [2] [/etc/fstab]

```
[root@node01 ~]# vi /etc/fstab
/dev/mapper/fedora_fedora-root / xfs defaults 0 0
UUID=7a32c4aa-4536-4a53-9098-d8fce81050e6 /boot xfs defaults 0 0
# add to the end : set NFS share
dlp.srv.world: /home/nfsshare /mnt nfs defaults 0 0
```

## [3] NFS AutoFS

```
[root@node01 ~]# dnf -y install autofs
[root@node01 ~]# vi /etc/auto.master
# add to the end
/- /etc/auto.mount

[root@node01 ~]# vi /etc/auto.mount
# create new : [mount point] [option] [location]
/mnt -fstype=nfs,rw dlp.srv.world: /home/nfsshare

[root@node01 ~]# systemctl enable --now autofs
# move to the mount point to verify mounting
[root@node01 ~]# cd /mnt
[root@node01 mnt]# ll
total 4
drwxr-xr-x. 2 root root 6 Nov 9 14:13 testdir
-rw-r--r--. 1 root root 10 Nov 9 14:13 testfile.txt

[root@node01 mnt]# grep /mnt /proc/mounts
/etc/auto.mount /mnt autofs
rw,relatime,fd=17,pgrp=24684,timeout=300,minproto=5,maxproto=5,direct,pipe_ino=50098 0 0
dlp.srv.world: /home/nfsshare /mnt nfs4
rw,relatime,vers=4.2,rsiz=524288,wsiz=524288,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,clientaddr=10.0.0.51,local_lock=none,addr=10.0.0.30 0 0
```

# NFS NFS 4 ACL

NFS(v4)      ACL      NFS 4 ACL  
POSIX ACL Tool

[1]      NFSv4      NFS      NFS      NFS 4 ACL

```
[root@node01 ~]# dnf -y install nfs4-acl-tools
```

[2]

```
[root@node01 ~]# df -hT /mnt
Filesystem                                Type  Size  Used Avail Use% Mounted on
dlp.srv.world: /home/nfsshare nfs4   15G   1.6G   14G   11% /mnt

[root@node01 ~]# ll /mnt
total 4
drwx-----. 2 root root  6 Nov  9 17:45 testdir
-rw-----. 1 root root 10 Nov  9 17:44 testfile.txt
```

[3]      NFSv4      ACL

```
[root@node01 ~]# nfs4_getfacl /mnt/testfile.txt

# file: /mnt/testfile.txt
A::OWNER@:rwatTcCy
A::GROUP@:tcy
A::EVERYONE@:tcy

[root@node01 ~]# nfs4_getfacl /mnt/testdir

# file: /mnt/testdir
A::OWNER@:rwaDxtTcCy
A::GROUP@:tcy
A::EVERYONE@:tcy

# each entry means like follows
# ACE = Access Control Entry
# ( ACE Type):( ACE Flags):( ACE Principal):( ACE Permissions)
```

	A = Allow
D	D = Deny
d	Directory-Inherit                      ACE
F	File-Inherit                      ACE
n	No-Propagate-Inherit                      ACE
	Inherit-Only      /                      ACE                      ACE
(USER)@(NFSD )	[NFSDomain]                      [idmapd.conf]                      [Domain]
(GROUP)@(NFSD )	[g]      ⇒ A:g:GROUP@NFSDomain:rxtncy
@	
@	
@	
ACE	
r	/
w	/
	/
X	/
d	
D	
n	
ñ	
C	ACL
C	ACL
○	
ACE	nfs4_setfacl                      ACE
R	R = rntcy
W	W = watTNcCy
X	X = xtcy

```
[root@node01 ~]# ll /mnt
total 4
drwx-----. 2 root root  6 Nov  9 17:45 testdir
-rw-----. 1 root root 10 Nov  9 17:44 testfile.txt

[root@node01 ~]# nfs4_getfacl /mnt/testfile.txt

# file: /mnt/testfile.txt
A::OWNER@:rwatTcCy
A::GROUP@:tcy
A::EVERYONE@:tcy

# add generic read/execute for [fedora] user to [/mnt/testfile.txt] file
[root@node01 ~]# nfs4_setfacl -a A::fedora@srv.world:rxtncy /mnt/testfile.txt
[root@node01 ~]# nfs4_getfacl /mnt/testfile.txt

# file: /mnt/testfile.txt
D::OWNER@:x
A::OWNER@:rwatTcCy
A::1000:rxtcy
A::GROUP@:tcy
A::EVERYONE@:tcy

# verify with [fedora] user
[fedora@node01 ~]$ ll /mnt
total 4
drwx-----. 2 root root  6 Nov  9 17:45 testdir
-rw-r-x---. 1 root root 10 Nov  9 17:44 testfile.txt

[fedora@node01 ~]$ cat /mnt/testfile.txt
test file

# delete generic read/execute for [fedora] user from [/mnt/testfile.txt] file
[root@node01 ~]# nfs4_setfacl -x A::1000:rxtcy /mnt/testfile.txt
[root@node01 ~]# nfs4_getfacl /mnt/testfile.txt

# file: /mnt/testfile.txt
```

```
A: : OWNER@: rwaTcCy
A: : GROUP@: tcy
A: : EVERYONE@: tcy
```

## [5] ACL

```
[root@node01 ~]# nfs4_setfacl -e /mnt/testfile.txt

# $EDITOR is run and enter to ACL editing
# default $EDITOR on Fedora 34 is [nano], if $EDITOR=null, default is set to [vi]
## Editing NFSv4 ACL for file: /mnt/testfile.txt
A: : OWNER@: rwaTcCy
A: : GROUP@: tcy
A: : EVERYONE@: tcy
```

## [6] ACE

```
# create ACL list
[root@node01 ~]# vi acl.txt
A: : fedora@srv.world: RX
A: : redhat@srv.world: RWX

# add ACL from the file
[root@node01 ~]# nfs4_setfacl -A acl.txt /mnt/testfile.txt
[root@node01 ~]# nfs4_getfacl /mnt/testfile.txt

# file: /mnt/testfile.txt
D: : OWNER@: x
A: : OWNER@: rwaTcCy
A: : 1000: rxtcy
A: : 1001: rwxtcy
A: : GROUP@: tcy
A: : EVERYONE@: tcy
```

## [7] ACE ACE

```
# create ACL list
[root@node01 ~]# vi acl.txt
A: : OWNER@: rwxtTcCy
A: : GROUP@: tcy
A: : EVERYONE@: tcy
```

```
# replace ACL from the file
[root@node01 ~]# nfs4_setfacl -S acl.txt /mnt/testfile.txt
[root@node01 ~]# nfs4_getfacl /mnt/testfile.txt

# file: /mnt/testfile.txt
A::OWNER@:rwaxtTcCy
A::GROUP@:tcy
A::EVERYONE@:tcy
```

[8]      ACE      ACE

```
[root@node01 ~]# nfs4_getfacl /mnt/testfile.txt

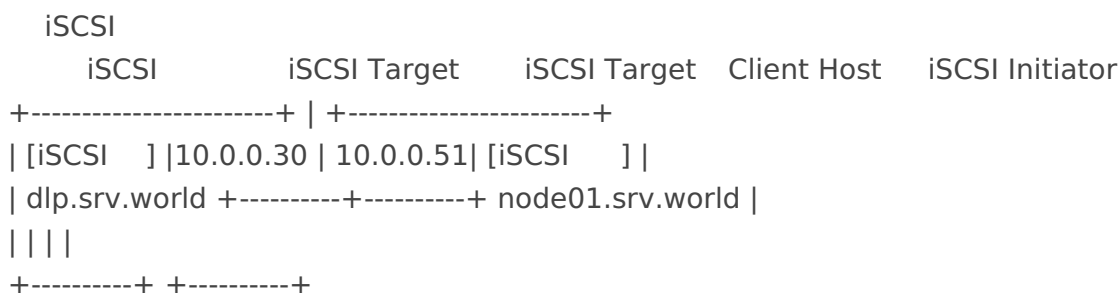
# file: /mnt/testfile.txt
A::OWNER@:rwaxtTcCy
A::GROUP@:tcy
A::EVERYONE@:tcy

# replace EVERYONE's ACE to read/execute
[root@node01 ~]# nfs4_setfacl -m A::EVERYONE@:tcy A::EVERYONE@:RX /mnt/testfile.txt
[root@node01 ~]# nfs4_getfacl /mnt/testfile.txt

# file: /mnt/testfile.txt
A::OWNER@:rwaxtTcCy
A::GROUP@:rxtcy
A::EVERYONE@:rxtcy
```

# iSCSI

## iSCSI      Targetcli





[1]

```
[root@dlp ~]# dnf -y install targetcli
```

[2] iSCSI

[/var/lib/iscsi\_disks]

SCSI

```
# create a directory
[root@dlp ~]# mkdir /var/lib/iscsi_disks
# enter the admin console
[root@dlp ~]# targetcli
targetcli shell version 2.1.54
Copyright 2011-2013 by Datera, Inc and others.
For help on commands, type 'help'.

/> cd backstores/fileio

# create a disk-image with the name [disk01] on [/var/lib/iscsi_disks/disk01.img] with 10G
/backstores/fileio> create disk01 /var/lib/iscsi_disks/disk01.img 10G
Created fileio disk01 with size 10737418240
/backstores/fileio> cd /iscsi

# create a target
# naming rule : [ iqn.(year)-(month).(reverse of domain name):(any name you like) ]
/iscsi> create iqn.2021-11.world.srv:dlp.target01
Created target iqn.2021-11.world.srv:dlp.target01.
Created TPG 1.
Global pref auto_add_default_portal=true
Created default portal listening on all IPs (0.0.0.0), port 3260.
/iscsi> cd iqn.2021-11.world.srv:dlp.target01/tpg1/luns

# set LUN
/iscsi/iqn.20...t01/tpg1/luns> create /backstores/fileio/disk01
Created LUN 0.
/iscsi/iqn.20...t01/tpg1/luns> cd ../acls

# set ACL (it's the IQN of an initiator you permit to connect)
/iscsi/iqn.20...t01/tpg1/acls> create iqn.2021-11.world.srv:node01.initiator01
Created Node ACL for iqn.2021-11.world.srv:node01.initiator01
Created mapped LUN 0.
```

```

/iscsi/iqn.20...t01/tpgl/acls> cd iqn.2021-11.world.srv:node01.initiator01

# set UserID and Password for authentication
/iscsi/iqn.20...w.initiator01> set auth userid=username
Parameter userid is now 'username'.
/iscsi/iqn.20...w.initiator01> set auth password=password
Parameter password is now 'password'.
/iscsi/iqn.20...w.initiator01> exit
Global pref auto_save_on_exit=true
Configuration saved to /etc/target/saveconfig.json

# after configuration above, the target enters in listening like follows
[root@dlp ~]# ss -napt | grep 3260
LISTEN 0      256          0.0.0.0:3260      0.0.0.0:*

[root@dlp ~]# systemctl enable target

```

### [3] Firewalld iSCSI Target

```

[root@dlp ~]# firewall-cmd --add-service=iscsi-target
success
[root@dlp ~]# firewall-cmd --runtime-to-permanent
success

```

## iSCSI (tgt)

iSCSI

scsi-target-utils iSCSI Target (tgt)

iSCSI iSCSI Target iSCSI Target Client Host iSCSI Initiator

```

+-----+ | +-----+
| [iSCSI ] | 10.0.0.30 | 10.0.0.51 | [iSCSI ] |
| dlp.srv.world +-----+-----+ node01.srv.world |
| | | |
+-----+ +-----+

```

### [1]

```

[root@dlp ~]# dnf -y install scsi-target-utils

```

### [2] iSCSI

[/var/lib/iscsi\_disks]

SCSI

```
[root@dlp ~]# systemctl enable --now tgt
# show status
[root@dlp ~]# tgtadm --mode target --op show
Target 1: iqn.2021-11.world.srv:dlp.target01
  System information:
    Driver: iscsi
    State: ready
  I_T nexus information:
  LUN information:
    LUN: 0
      Type: controller
      SCSI ID: IET      00010000
      SCSI SN: beaf10
      Size: 0 MB, Block size: 1
      Online: Yes
      Removable media: No
      Prevent removal: No
      Readonly: No
      SWP: No
      Thin-provisioning: No
      Backing store type: null
      Backing store path: None
      Backing store flags:
    LUN: 1
      Type: disk
      SCSI ID: IET      00010001
      SCSI SN: beaf11
      Size: 10737 MB, Block size: 512
      Online: Yes
      Removable media: No
      Prevent removal: No
      Readonly: No
      SWP: No
      Thin-provisioning: No
      Backing store type: rdwr
      Backing store path: /var/lib/iscsi_disks/disk01.img
      Backing store flags:
  Account information:
    username
  ACL information:
```

ALL

iqn.2021-11.world.srv:node01.initiator01

```
# create a disk image
[root@dlp ~]# mkdir /var/lib/iscsi_disks
[root@dlp ~]# dd if=/dev/zero of=/var/lib/iscsi_disks/disk01.img count=0 bs=1 seek=10G
[root@dlp ~]# vi /etc/tgt/conf.d/target01.conf
# create new
# if you set some devices, add <target>-</target> and set the same way with follows
# naming rule : [ iqn.(year)-(month).(reverse of domain name):(any name you like) ]
<target iqn.2021-11.world.srv:dlp.target01>
    # provided device as a iSCSI target
    backing-store /var/lib/iscsi_disks/disk01.img
    # iSCSI Initiator's IQN you allow to connect
    initiator-name iqn.2021-11.world.srv:node01.initiator01
    # authentication info ( set anyone you like for "username", "password" )
    incominguser username password
</target>
```

### [3] SELinux SELinux

```
[root@dlp ~]# dnf -y install policycoreutils-python-utils
[root@dlp ~]# chcon -R -t tgtd_var_lib_t /var/lib/iscsi_disks
[root@dlp ~]# semanage fcontext -a -t tgtd_var_lib_t /var/lib/iscsi_disks
```

### [4] Firewalld iSCSI Target

```
[root@dlp ~]# firewall-cmd --add-service=iscsi-target
success
[root@dlp ~]# firewall-cmd --runtime-to-permanent
success
```

### [5] tgtd

```
[root@dlp ~]# systemctl enable --now tgtd
# show status
[root@dlp ~]# tgtadm --mode target --op show
Target 1: iqn.2021-11.world.srv:dlp.target01
    System information:
        Driver: iscsi
```

```
State: ready
I_T nexus information:
LUN information:
  LUN: 0
    Type: controller
    SCSI ID: IET      00010000
    SCSI SN: beaf10
    Size: 0 MB, Block size: 1
    Online: Yes
    Removable media: No
    Prevent removal: No
    Readonly: No
    SWP: No
    Thin-provisioning: No
    Backing store type: null
    Backing store path: None
    Backing store flags:
  LUN: 1
    Type: disk
    SCSI ID: IET      00010001
    SCSI SN: beaf11
    Size: 10737 MB, Block size: 512
    Online: Yes
    Removable media: No
    Prevent removal: No
    Readonly: No
    SWP: No
    Thin-provisioning: No
    Backing store type: rdwr
    Backing store path: /var/lib/iscsi_disks/disk01.img
    Backing store flags:
Account information:
  username
ACL information:
  ALL
  iqn.2021-11.world.srv:node01.initiator01
```

## iSCSI

## iSCSI

```
+-----+ | +-----+
|[iSCSI  ]|10.0.0.30|10.0.0.51|[iSCSI  ]|
|dlp.srv.world +-----+-----+ node01.srv.world |
||||
+-----+ +-----+
```

### [1] iSCSI Initiator iSCSI Target

```
[root@node01 ~]# dnf -y install iscsi-initiator-utils
[root@node01 ~]# vi /etc/iscsi/initiatorname.iscsi
# change to the same IQN you set on the iSCSI target server
InitiatorName=iqn.2021-11.world.srv:node01.initiator01
[root@node01 ~]# vi /etc/iscsi/iscsid.conf
# line 58 : uncomment
node.session.auth.authmethod = CHAP
# line 69,70 : uncomment and specify the username and password you set on the iSCSI target
server
node.session.auth.username = username
node.session.auth.password = password
# discover target
[root@node01 ~]# iscsiadm -m discovery -t sendtargets -p 10.0.0.30
10.0.0.30:3260,1 iqn.2021-11.world.srv:dlp.target01

# confirm status after discovery
[root@node01 ~]# iscsiadm -m node -o show
# BEGIN RECORD 2.1.4
node.name = iqn.2021-11.world.srv:dlp.target01
node.tpgt = 1
node.startup = automatic
node.leading_login = No
iface.iscsi_ifacename = default
.....
.....
node.conn[0].iscsi.HeaderDigest = None
node.conn[0].iscsi.DataDigest = None
node.conn[0].iscsi.IFMarker = No
node.conn[0].iscsi.OFMarker = No
# END RECORD
```

```
# login to the target # if logout => iscsiadm --mode node --logoutall=all
[root@node01 ~]# iscsiadm -m node --login
Logging in to [iface: default, target: iqn.2021-11.world.srv:dlp.target01, portal:
10.0.0.30,3260]
Login to [iface: default, target: iqn.2021-11.world.srv:dlp.target01, portal: 10.0.0.30,3260]
successful.

# confirm the established session
[root@node01 ~]# iscsiadm -m session -o show
tcp: [1] 10.0.0.30:3260,1 iqn.2021-11.world.srv:dlp.target01 (non-flash)
# confirm the partitions
[root@node01 ~]# cat /proc/partitions
major minor #blocks name

11          0    1048575 sr0
252         0    31457280 sda
252         1     1048576 sda1
252         2    30407680 sda2
253         0     15728640 dm-0
251         0     4007936 zram0
8           0     10485760 sdb

# added new device provided from the target server as [sdb]
```

## [2] iSCSI Initiator

```
# create label
[root@node01 ~]# parted --script /dev/sdb "mklabel gpt"
# create partition
[root@node01 ~]# parted --script /dev/sdb "mkpart primary 0% 100%"
# format with XFS
[root@node01 ~]# mkfs.xfs -i size=1024 -s size=4096 /dev/sdb1
meta-data=/dev/sdb1            isize=1024   agcount=4, agsize=654336 blks
      =                       sectsz=4096   attr=2, projid32bit=1
      =                       crc=1         finobt=1, sparse=1, rmapbt=0
      =                       reflink=1      bigtime=0
data      =                       bsize=4096   blocks=2617344, imaxpct=25
      =                       sunit=0       swidth=0 blks
naming    =version 2           bsize=4096   ascii-ci=0, ftype=1
log       =internal log       bsize=4096   blocks=2560, version=2
      =                       sectsz=4096   sunit=1 blks, lazy-count=1
```

```
realtime =none                extsz=4096   blocks=0, rtextents=0
```

```
[root@node01 ~]# mount /dev/sdb1 /mnt
```

```
[root@node01 ~]# df -hT
```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
devtmpfs	devtmpfs	1.9G	0	1.9G	0%	/dev
tmpfs	tmpfs	2.0G	0	2.0G	0%	/dev/shm
tmpfs	tmpfs	786M	1.0M	785M	1%	/run
/dev/mapper/fedora_fedora-root	xfs	15G	1.6G	14G	11%	/
tmpfs	tmpfs	2.0G	4.0K	2.0G	1%	/tmp
/dev/sda1	xfs	1014M	186M	829M	19%	/boot
tmpfs	tmpfs	393M	0	393M	0%	/run/user/0
/dev/sdb1	xfs	10G	99M	9.9G	1%	/mnt

# GlusterFS 9

## GlusterFS 9

GlusterFS

GlusterFS /

[sdb1]

[/glusterfs]

[1] GlusterFS

```
[root@node01 ~]# dnf -y install glusterfs-server
```

```
[root@node01 ~]# systemctl enable --now glusterd
```

```
[root@node01 ~]# gluster --version
```

glusterfs 9.4

Repository revision: git://git.gluster.org/glusterfs.git

Copyright (c) 2006-2016 Red Hat, Inc. <<https://www.gluster.org/>>

GlusterFS comes with ABSOLUTELY NO WARRANTY.

It is licensed to you under your choice of the GNU Lesser

General Public License, version 3 or any later version (LGPLv3

or later), or the GNU General Public License, version 2 (GPLv2),

in all cases as published by the Free Software Foundation.

[2] Firewalld GlusterFS

```
[root@node01 ~]# firewall-cmd --add-service=glusterfs
```

success



```
[root@node01 ~]# firewall-cmd --runtime-to-permanent
success
```

## GlusterFS 9

```
GlusterFS
  2
  2      3
|
+-----+ | +-----+
| [GlusterFS #1] | 10.0.0.51 | 10.0.0.52 | [GlusterFS #2] |
| node01.srv.world +-----+-----+ node02.srv.world |
| | | |
+-----+ +-----+
↑ ↑
  1   3 ...   2   4 ...
```

```
GlusterFS      /
                [sdb1]          [/glusterfs]
```

[1] GlusterFS

[2] GlusterFS

```
[root@node01 ~]# mkdir -p /glusterfs/distributed
```

[3]

```
# probe nodes
[root@node01 ~]# gluster peer probe node02
peer probe: success.

# confirm status
[root@node01 ~]# gluster peer status
Number of Peers: 1

Hostname: node02
Uuid: 447dedcb-fe9b-4743-851c-a7c2adef0043
State: Peer in Cluster (Connected)

# create volume
[root@node01 ~]# gluster volume create vol_distributed transport tcp \
node01:/glusterfs/distributed \
```

```

node02: /glusterfs/distributed
volume create: vol_distributed: success: please start the volume to access data
# start volume
[root@node01 ~]# gluster volume start vol_distributed
volume start: vol_distributed: success
# confirm volume info
[root@node01 ~]# gluster volume info

Volume Name: vol_distributed
Type: Distribute
Volume ID: 3a671a01-2a6c-4c4d-858c-4c8e401bc23c
Status: Started
Snapshot Count: 0
Number of Bricks: 2
Transport-type: tcp
Bricks:
Brick1: node01: /glusterfs/distributed
Brick2: node02: /glusterfs/distributed
Options Reconfigured:
storage.fips-mode-rchecksum: on
transport.address-family: inet
nfs.disable: on

```

## GlusterFS 9 GlusterFS + NFS-Ganesha

NFS-Ganesha	GlusterFS	NFS	Gluster Volume
NFS-Ganesha	NFS	v3 v4.0 v4.1 pNFS	
[1]	Gluster	NFS	
	Gluster	NFS	
	NFS		

```

# OK if [nfs.disable: on] (default setting)
[root@node01 ~]# gluster volume get vol_distributed nfs.disable
Option                                     Value
-----                                     -
nfs.disable                                on

# if [nfs.disable: off], turn to disable
[root@node01 ~]# gluster volume set vol_distributed nfs.disable on
volume set: success

```

```
# if NFS server is running, disable it
[root@node01 ~]# systemctl disable --now nfs-server
```

## [2] GlusterFS                      NFS-Ganesha

```
[root@node01 ~]# dnf -y install nfs-ganesha-gluster
[root@node01 ~]# mv /etc/ganesha/ganesha.conf /etc/ganesha/ganesha.conf.org
[root@node01 ~]# vi /etc/ganesha/ganesha.conf

# create new
NFS_CORE_PARAM {
    # possible to mount with NFSv3 to NFSv4 Pseudo path
    mount_path_pseudo = true;

    # NFS protocol
    Protocols = 3,4;
}

EXPORT_DEFAULTS {
    # default access mode
    Access_Type = RW;
}

EXPORT {
    # uniq ID
    Export_Id = 101;

    # mount path of Gluster Volume
    Path = "/vol_distributed";

    FSAL {
        [# any name
        name = GLUSTER;

        # hostname or IP address of this Node
        hostname="10.0.0.51";

        # Gluster volume name
        volume="vol_distributed";

        }

    # config for root Squash
    Squash="No_root_squash";

    # NFSv4 Pseudo path
    Pseudo="/vfs_distributed";

    # allowed security options
    SecType = "sys";
}

LOG {
    # default log level
```

```

    Default_Log_Level = WARN;
}

[root@node01 ~]# systemctl enable --now nfs-ganesha
# verify mount
[root@node01 ~]# showmount -e localhost
Export list for localhost:
/vfs_distributed (everyone)

```

### [3] Firewalld NFS

```

[root@node01 ~]# firewall-cmd --add-service=nfs
success
[root@node01 ~]# firewall-cmd --runtime-to-permanent
success

```

### [4] NFS

```

[root@client ~]# dnf -y install nfs-utils
# specify Pseudo path set on [Pseudo=***] in ganesha.conf
[root@client ~]# mount -t nfs4 node01.srv.world:/vfs_distributed /mnt
[root@client ~]# df -hT

```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
devtmpfs	devtmpfs	1.9G	0	1.9G	0%	/dev
tmpfs	tmpfs	2.0G	0	2.0G	0%	/dev/shm
tmpfs	tmpfs	783M	996K	782M	1%	/run
/dev/mapper/fedora_fedora-root	xfs	15G	1.6G	14G	11%	/
tmpfs	tmpfs	2.0G	4.0K	2.0G	1%	/tmp
/dev/vda1	xfs	1014M	195M	820M	20%	/boot
tmpfs	tmpfs	392M	0	392M	0%	/run/user/0
node01.srv.world:/vfs_distributed	nfs4	30G	3.5G	27G	12%	/mnt

```

# verify reading and writing
[root@client ~]# echo "Gluster NFS write test" > /mnt/testfile.txt
[root@client ~]# cat /mnt/testfile.txt
Gluster NFS write test

```

## GlusterFS 9 GlusterFS + SMB

GlusterFS      SMB

```

[root@node01 ~]# dnf -y install samba ctdb samba-vfs-glusterfs
# stop the target Gluster volume and change settings
[root@node01 ~]# gluster volume stop vol_distributed
Stopping volume will make its data inaccessible. Do you want to continue? (y/n) y
volume stop: vol_distributed: success
[root@node01 ~]# gluster volume set vol_distributed user.smb enable
volume set: success
[root@node01 ~]# gluster volume set vol_distributed performance.write-behind off
volume set: success
[root@node01 ~]# gluster volume set vol_distributed group samba
volume set: success
[root@node01 ~]# vi /var/lib/glusterd/hooks/1/start/post/S29CTDBsetup.sh
# line 25 : change to the target Gluster volume name
META="vol_distributed"
[root@node01 ~]# vi /var/lib/glusterd/hooks/1/stop/pre/S29CTDB-teardown.sh
# line 13 : change to the target Gluster volume name
META="vol_distributed"
# start Gluster volume
[root@node01 ~]# gluster volume start vol_distributed
volume start: vol_distributed: success
# with the settings above, following mounting is done automatically
[root@node01 ~]# df -h /gluster/lock

```

Filesystem	Size	Used	Avail	Use%	Mounted on
node01.srv.world:/vol_distributed.tcp	30G	3.5G	27G	12%	/gluster/lock

```

[root@node01 ~]# tail -1 /etc/fstab
node01.srv.world:/vol_distributed /gluster/lock glusterfs _netdev,transport=tcp,xlator-
option=*client*.ping-timeout=10 0 0

[root@node01 ~]# vi /etc/ctdb/nodes
# create new
# write all Nodes that configure target Gluster volume
10.0.0.51
10.0.0.52
[root@node01 ~]# vi /etc/ctdb/public_addresses
# create new
# set virtual IP address for SMB access
# [enp1s0] means network interface name => replace to your environment

```

```

10.0.0.59/24 enp1s0
[root@node01 ~]# systemctl enable --now ctdb
# confirm status
[root@node01 ~]# ctdb status
Number of nodes: 2
pnn: 0 10.0.0.51          OK ( THIS NODE)
pnn: 1 10.0.0.52          DISCONNECTED| UNHEALTHY| INACTIVE
Generation: 1113695787
Size: 1
hash: 0 lmaster: 0
Recovery mode: NORMAL ( 0)
Recovery master: 0

[root@node01 ~]# ctdb ip
Public IPs on node 0
10.0.0.59 0

```

## [2] Samba

[smbgroup]

[smbshare]

```

# mount Gluster volume with GlusterFS Native and create a shared folder for SMB access
[root@node01 ~]# mount -t glusterfs node01.srv.world:/vol_distributed /mnt
[root@node01 ~]# mkdir /mnt/smbshare
[root@node01 ~]# groupadd smbgroup
[root@node01 ~]# chgrp smbgroup /mnt/smbshare
[root@node01 ~]# chmod 770 /mnt/smbshare
[root@node01 ~]# umount /mnt
[root@node01 ~]# vi /etc/samba/smb.conf
[global]
    workgroup = SAMBA
    security = user

    passdb backend = tdbsam

    printing = cups
    printcap name = cups
    load printers = yes
    cups options = raw
    # add follows
    clustering = yes

```

```

    kernel share modes = no
    kernel oplocks = no
    map archive = no
    map hidden = no
    map read only = no
    map system = no
    store dos attributes = yes

# following 9 lines are configured automatically
[gluster-vol_distributed]
comment = For samba share of volume vol_distributed
vfs objects = glusterfs
glusterfs: volume = vol_distributed
glusterfs: logfile = /var/log/samba/glusterfs-vol_distributed.%M.log
glusterfs: loglevel = 7
path = /
read only = no
kernel share modes = no
# add follows
writable = yes
valid users = @smbgroup
force create mode = 777
force directory mode = 777
inherit permissions = yes

[root@node01 ~]# systemctl enable --now smb
# add Samba user
[root@node01 ~]# useradd fedora
[root@node01 ~]# smbpasswd -a fedora
New SMB password:      # set any SMB password
Retype new SMB password:
Added user fedora.
[root@node01 ~]# usermod -aG smbgroup fedora

```

### [3] SELinux

```

[root@node01 ~]# setsebool -P use_fusefs_home_dirs on
[root@node01 ~]# setsebool -P samba_load_libgfapi on
[root@node01 ~]# setsebool -P domain_kernel_load_modules on

```

#### [4] Firewall

```
[root@node01 ~]# firewall-cmd --add-service={samba,ctdb}
success
[root@node01 ~]# firewall-cmd --runtime-to-permanent
success
```

#### [5] Linux SMB Linux Windows

```
# verify with [smbclient]
[root@client ~]# smbclient //node01.srv.world/gluster-vol_distributed -U fedora
Enter SAMBA\fedora's password:
Try "help" to get a list of possible commands.

# verify writable to move to shared folder
smb: \> cd smbshare

smb: \smbshare\> mkdir testdir
smb: \smbshare\> ls
.                D            0  Tue Nov  9 15:13:16 2021
..               D            0  Tue Nov  9 15:09:06 2021
anaconda-ks.cfg  A          872  Tue Nov  9 15:13:17 2021
testdir         D            0  Tue Nov  9 15:12:38 2021

31436800 blocks of size 1024. 27701820 blocks available

smb: \smbshare\> exit
```



---

Revision #6

Created 3 March 2022 02:43:15 by

Updated 9 July 2022 19:08:53 by