

Network File Shares

Network file sharing (Linux to Linux)

- A client's mounted NFS export appears as a local directory

NFS - Exam Scenario

1. install nfs utilities
2. enable the nfs server service
3. add a correct line to /etc/exports
4. apply it with exportfs -arv
5. allow the nfs service in the firewall
6. mount on the client
7. mount persistent in /etc/fstab with options like _netdev & vers=4.2

NFS - Server 192.168.17.185

dnf install nfs-utils -y	install nfs utilities
systemctl enable --now nfs-server	start and enable nfs service
mkdir -p /srv/nfsshare	create NFS share directory
chmod -R 0777 /srv/nfsshare	open permissions (0777 for lab)
semanage fcontext -a -t public_content_rw_t "/srv/nfsshare(/.*)"?	assign selinux context
restorecon -Rv /srv/nfsshare	apply selinux context
ls -Zd /srv/nfsshare	verify selinux context
nano /etc/exports ... /srv/nfsshare 192.168.17.0/24(rw,sync,no_root_squash)	export share to subnet with read-write access
exportfs -arv	reload and verify exports
firewall-cmd --add-service=nfs --permanent	allow nfs through firewall
firewall-cmd --add-service=rpc-bind --permanent	NFSv4 alone works over port 2049, but legacy tools need rpc-bind and mountd
firewall-cmd --add-service=mountd --permanent	
firewall-cmd --reload	reload firewall rules
exportfs -v grep /srv/nfsshare	verify export configuration

NFS - Client 192.168.17.208

ip a	confirm subnet
showmount -e 192.168.17.185	verify server is exporting
sudo dnf install nfs-utils -y	install nfs utilities

<code>sudo mkdir -p /mnt/nfsshare</code>	create nfs mount directory
<code>sudo mount -t nfs -o vers=4.2 192.168.17.185:/srv/nfsshare /mnt/nfsshare</code>	mount nfs share
<code>df -h grep nfsshare</code>	verify
<code>echo "Created from client" sudo tee /mnt/nfsshare/client.txt >/dev/null</code>	test write access
<code>sudo nano /etc/fstab</code> ... <code>192.168.17.185:/srv/nfsshare /mnt/nfsshare nfs _netdev,vers=4.2,rw 0 0</code>	persistent nfs mount entry
<code>sudo systemctl daemon-reload</code>	reload systemd config
<code>sudo mount -a</code>	test fstab config
<code>reboot</code>	verify mount survives

Samba (Linux to Windows)

- Implements SMB/CIFS (windows file sharing)
- Allows Linux to share a folder that Windows clients can map
- Linux clients can connect using CIFS
- Define a share in `/etc/samba/smb.conf`
- With SELinux, you need to label the path you are sharing
- Set proper file permissions and ACLs
- Ensure SELinux is enforcing with right context on shared directories

Samba - Exam Scenario

1. install Samba
2. add a share block in `smb.conf` and verify with `testparm`
3. label the directory using `semanage fcontext` and `restorecon`
4. allow the samba service through the firewall
5. start and enable the `smb` daemon
6. test with `smbclient`
7. mount using CIFS
8. make persistent in `/etc/fstab`

Samba - Server 192.168.17.185

<code>dnf install -y samba policycoreutils-python-utils</code>	install samba server and selinux tools
<code>mkdir -p /srv/sambashare</code>	create samba share directory
<code>echo "This is a samba file" > /srv/sambashare/readme.txt</code>	create test file
<code>chmod -R 0777 /srv/sambashare</code>	open permissions (0777 for lab)

nano /etc/samba/smb.conf ... [smbashare] path = /srv/smbashare browsable = yes writable = yes guest ok = yes read only = no	define samba share configuration
testparm -s	validate samba config
semanage fcontext -a -t samba_share_t "/srv/smbashare(/.*)?"	assign selinux context
restorecon -Rv /srv/smbashare	apply selinux context
ls -Zd /srv/smbashare	verify selinux context
firewall-cmd --add-service=samba --permanent	allow samba through firewall
firewall-cmd --reload	reload firewall rules
systemctl enable --now smb	start and enable samba service
systemctl status smb --no-pager	verify status

Samba - Client 192.168.17.208

sudo dnf install -y samba-client cifs-utils	install samba client tools
smbclient -L //192.168.17.185 -N	list available samba shares
smbclient //192.168.17.185/smbashare -N	connect to samba share
ls	list files in share
get readme.txt	download file from share
quit	exit smbclient session
sudo mkdir -p /mnt/smbashare	create mount directory
sudo mount -t cifs //192.168.17.185/smbashare /mnt/smbashare -o guest	mount samba share
ls -l /mnt/smbashare	verify share contents
echo "Client wrote this via Samba" sudo tee /mnt/smbashare/client.txt >/dev/null	create file on share
sudo nano /etc/fstab ... //192.168.17.185/smbashare /mnt/smbashare cifs _netdev,guest 0 0	configure persistent mount
sudo mount -a	test fstab config
reboot	verify mount survives