

exp 8 Configure SSH server and establish SSH communication on port 41101

SERVER

- ① yum install openssh-server
- ② systemctl status sshd → (initially status must be inactive)
- ③ systemctl start sshd
- ④ systemctl status sshd → (now status is active)
- ⑤ ifconfig
- ⑥ systemctl stopfirewalld

This is normal configure of SSH server client.

- ① ssh centos@server_ip
password: centos

CLIENT

To establish communication on port 41101

- ① do same process as given in Server →

SERVER

- Then
- ② cp /etc/ssh/sshd-config /etc/ssh/sshd-config-original
 - ③ mv /etc/ssh/sshd-config-original /etc/ssh/sshd-config
 - ④ nano /etc/ssh/sshd-config

PermitRootlogin = YES } remove #

- ⑤ Port 45010

cd /home create txt file pol.txt.

CLIENT

- ① Login :- ssh centos @ serverIP -p 45010
pass :- centos.
- ② cd /home.
- ③ ls
(you will see pol.txt on client side.)
- ④ exit.

Exp 9 Configure Telnet-server-client and establish Telnet configuration on port 43897.

SERVER

① Configuring Telnet server client

- ① rpm -qa | grep telnet. → check if telnet packet is already installed.
- ② yum install telnet-server telnet.
- ③ systemctl status telnet.socket → (initially must be inactive)
- ④ systemctl start telnet.socket
- ⑤ " enable telnet.socket.
- ⑥ " status telnet.socket → (now status will be active)
- ⑦ " ~~stop firewall~~
- ⑦ firewall-cmd --add-service=telnet --zone=public --permanent
- ⑧ " " " " " " " "
- ⑧ ifconfig (for server IP)
- ⑨

CLIENT

- ① yum install telnet.
- ② telnet server_IP
- ③ ~~login~~ login: centos
password: centos
- ④ exit (to logout)

Create file in server using touch command and write text in it and login into telnet through client and access that file which you created if you have saved txt file in directory home then in client first change directory to home by cd/home. then do ls. you will see the file created by server.

Telnet configuration on port 43897

① do same process of server.

② then, open config file

nano /usr/lib/systemd/system/telnet.socket.

③ change ListenStream= 43897

④ systemctl daemon-reload
 " restart telnet.socket
 " enable "
 " status "

⑤ firewall-cmd --permanent --add-port=43897/tcp
firewall-cmd --reload.

CLIENT

① telnet server_ip portno

ie telnet 192.168.1.11 43897

② login: centos

password: centos

exp 10 Apache Web Server

- ① yum install httpd
- ② systemctl status httpd → (initially service must be inactive).
- ③ systemctl start httpd
- ④ systemctl enable httpd
- ⑤ systemctl status httpd → (now service must be active)
systemctl stop firewalld
- ⑥ open firefox browser
→ type in url :- localhost:port ie localhost:80

⑦ cd /var/www/html

⑧ ls

⑨ touch parag.html.
nano parag.html.

```
<html>
<title> parag </title>
<body>
  parag extc a
</body>
</html>
```

find IP
using ifconfig
enp0s3 inet

⑩ go to browser type ~~server IP~~ IP/ parag.html

Now changing port no to 8080

nano /etc/httpd/conf/httpd.conf

listen port = 8080 → change to 8080

now ~~systemctl~~
systemctl restart httpd
" enable httpd.

go to firefox

write IP : 8080 /parag.html.

Exp 11 configure NFS server client.

for Server

- ① yum install nfs-utils → network file system utilities
- ② yum install libnfsidmap → library for NFS identifying mapping
- ③ systemctl enable rpcbind
- ④ systemctl enable nfs-server
- ⑤ systemctl start rpcbind → remote procedure call binders.
" start nfs-server
" start rpc-statd → remote procedure call status daemon
" start nfs-idmapd → identity mapper daemon
- ⑥ cd / → root directory.
- ⑦ mkdir parag → make directory
- ⑧ ls -l | grep parag → list files and directories in long format
- ⑨ chmod a+rwX parag → change file permissions (1) → pipe
- ⑩ ls -l | grep parag
- ⑪ cd parag
- ⑫ touch p01.txt
- ⑬ echo "My name is parag" > p01.txt. [a+rwX → all users Add read, write, execute permission]
- ⑭ cd ..
- ⑮ nano /etc/exports → #/DIR-NAME CLIENT-IP (PERMISSION)
- ⑯ cat /etc/exports → /parag 192.168.1.1(rw,sync,no-root_squash)
- ⑰ exportfs -rv → -r exports all directories.
v → verbose mode.
- ⑱ systemctl stop firewalld.

Client

- ① yum install nfs-utils
- ② yum install rpcbind
- ③ systemctl start rpcbind
- ④ systemctl ~~start~~ enable rpcbind
- ⑤ systemctl stop firewalld.
- ⑥ showmount -e server-ip
- ⑦ mkdir /mnt/anabhavane.
- ⑧ mount server-ip : /server_directory_name /mnt/directory_name
- ⑨ df -h

for server

- ① nano /etc/exports
#/DIR_NAME CLIENTIP (PERMISSION)
/parag 192.168.1.1 (rw, sync, no-root-squash)
/parag * (rw, sync, no-root-squash)
cat /etc/exports
- ② exportfs -rv

for client.

- ① showmount -e serverIP
- ② step ⑧ of client.
- ③ cd /mnt/anabhavane
- ④ ls
- ⑤ cat textfile_.
- ⑥ cd ..
- ⑦ cd /
- ⑧ ~~umount~~ umount /mnt/anabhavane.
df -h.

- * experiment 12 for server
- ① `rpm -qa | grep vsftpd` → checks if vsftpd is already installed (very secure file transfer protocol daemon)
 - ② `yum install vsftpd`
 - ③ `systemctl start vsftpd`
 - ④ `systemctl enable vsftpd`
 - ⑤ `nano /etc/vsftpd/vsftpd.conf`

`systemctl` → system control used to control & manage system services.

- make changes
- i) `anonymous_enable = YES` to `anonymous_enable = NO`
 - remove #
 - ii) `ascii_upload_enable = YES` & `ascii_download_enable = YES`
 - remove #
 - iii) Welcome to blah ftp service, also edit blah if needed (optional) but remove # compulsory
 - add at end of file
 - iv) `use_localtime = YES`

- ⑥ `systemctl start vsftpd`
- ⑦ `systemctl enable vsftpd`
- ⑧ `systemctl stop firewalld`
- ⑨ `ifconfig` for servers IP `enp0s3 (inet)`

for client (Another terminal)

- ① `yum install ftp`
- ② `cd /home` → change directory to home
- ③ `touch p01.txt` → create text file
- ④ `echo "my name is XYZ" > p01.txt`
- ⑤ `cat p01.txt`
- ⑥ `touch p02.txt`
`touch p03.txt`
`touch p04.txt` } create txt files.
- ⑦ `ftp servers_IP`
- ⑧ name: centos, password centos
- ⑨ `ftp > bin` → only this to write.
`ftp > hash`
- ⑩ `put p01.txt` → put (uploads file to server)
- ⑪ `bye`

(> this is output redirection)

for server .

- ① cd /home
- ② ls
- ③ cat p01.txt → (this file will be seen on server o/p as it was uploaded by client .
also p02.txt , p03.txt , p04.txt will be seen.)
- ~~④ touch~~
- ④ cd centos
- ⑤ touch down1.txt .
echo " thx for downloading " > down1.txt .
- ⑥ touch down2.txt
touch down8.txt

- client
- ① again login using ftp server_IP .
then bin
hash
 - ② get ~~pi~~ down1.txt
mget down02.txt down03.txt [download multiple files
 - ③ bye