

PRACTICAL-1

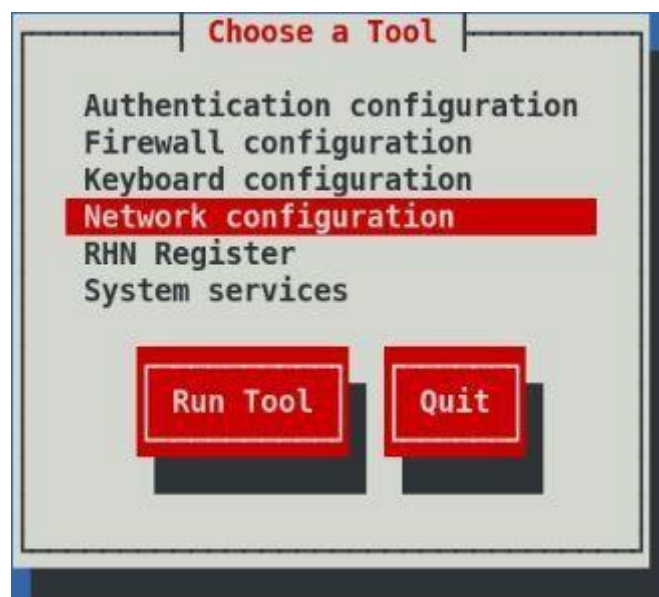
Aim: Install DHCP Server and Configure DHCP Server, Configure DHCP (Dynamic Host Configuration Protocol) Server.

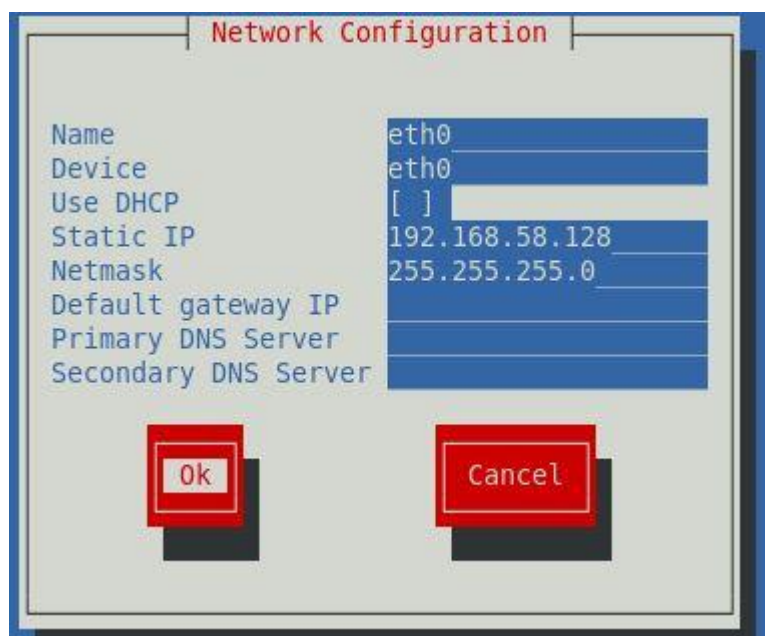
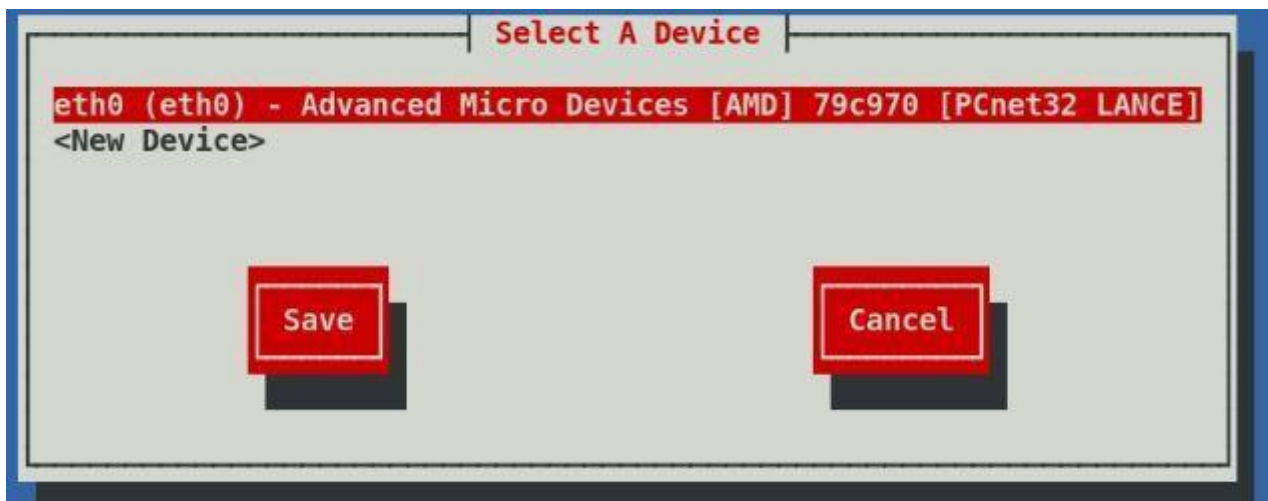
Server Side:

Step 1:

First, we will setup a static IP address by using the setup command and going into Network Configuration → Device Configuration → eth0... → deselect DHCP and set a static IP address → OK → save → Save and Quit.

```
[root@localhost Packages]# setup
```





Step 2:

We check if DHCP package is installed in our system using the yum command with list option to list if DHCP package is installed or not.

```
[root@ldapl Desktop]# yum list dhcp
Loaded plugins: product-id, refresh-packagekit, subscription-manager
Updating Red Hat repositories.
Installed Packages
dhcp.i686                               12:4.1.1-19.P1.el6                installed
```

Step 3:

Now we will verify the proper installation using the rpm command with -qa (query all) option

```
[root@localhost Packages]# rpm -qa dhcp
dhcp-4.1.1-19.P1.el6.i686
```

Step 4:

We will check where the DHCP configuration files are stored in the system using rpm command with -ql (query list) option.

```
[root@ldapl Desktop]# rpm -ql dhcp
/etc/dhcp
/etc/dhcp/dhcpd.conf
/etc/dhcp/dhcpd6.conf
/etc/openssl/schema/dhcp.schema
/etc/rc.d/init.d/dhcpd
/etc/rc.d/init.d/dhcpd6
/etc/rc.d/init.d/dhcrelay
/etc/sysconfig/dhcpd
/etc/sysconfig/dhcpd6
/etc/sysconfig/dhcrelay
/usr/bin/omshell
/usr/sbin/dhcpd
/usr/sbin/dhcrelay
/usr/share/doc/dhcp-4.1.1
/usr/share/doc/dhcp-4.1.1/3.0b1-lease-convert
/usr/share/doc/dhcp-4.1.1/IANA-arp-parameters
/usr/share/doc/dhcp-4.1.1/LICENSE
/usr/share/doc/dhcp-4.1.1/README
/usr/share/doc/dhcp-4.1.1/README.ldap
/usr/share/doc/dhcp-4.1.1/RELNOTES
```

Step 5:

Now we will manually configure the DHCP by writing into dhcpd.conf file which is located in the /etc/dhcp directory.

```
[root@ldapl Desktop]# vim /etc/dhcp/dhcpd.conf
```

```
#
# DHCP Server Configuration file.
#   see /usr/share/doc/dhcp*/dhcpd.conf.sample
#   see 'man 5 dhcpd.conf'
#

subnet 192.168.58.0 netmask 255.255.255.0{
range 192.168.58.10 192.168.58.30;
default-lease-time 600;
max-lease-time 4800;
}

~
```

Step 6:

Now we will test the DHCP configuration file using the service command with configtest option

```
[root@localhost Packages]# service dhcpd configtest
Syntax: OK
```

Step 7:

We will now start the DHCP service using the service command with start option.

```
[root@localhost Packages]# service dhcpd start
Starting dhcpd: [ OK ]
```

Step 8:

To start the DHCP service on boot time we will use the chkconfig command with on option.

```
[root@localhost Packages]# chkconfig dhcpd on
```

Step 9:

Before we go to the client side we will check if the DHCP has provided a IP address to any client system by reading the DHCP leases information file i.e. dhcpd.leases that is located in the /var/lib/dhcpd directory using the cat command.

```
[root@localhost Packages]# cat /var/lib/dhcpd/dhcpd.leases
# The format of this file is documented in the dhcpd.leases(5) manual page.
# This lease file was written by isc-dhcp-4.1.1-P1

server-uid "\000\001\000\001'\012&9\000\014)\363\033\316";
```

Step 10:

We will restart the network service for the changes to take effect using the service command with restart option.

```
[root@localhost Packages]# service network restart
Shutting down interface eth0: Device state: 3 (disconnected)
[ OK ]
Shutting down loopback interface:
[ OK ]
Bringing up loopback interface:
[ OK ]
Bringing up interface eth0: Active connection state: activating
Active connection path: /org/freedesktop/NetworkManager/ActiveConnection/3
state: activated
Connection activated
[ OK ]
```

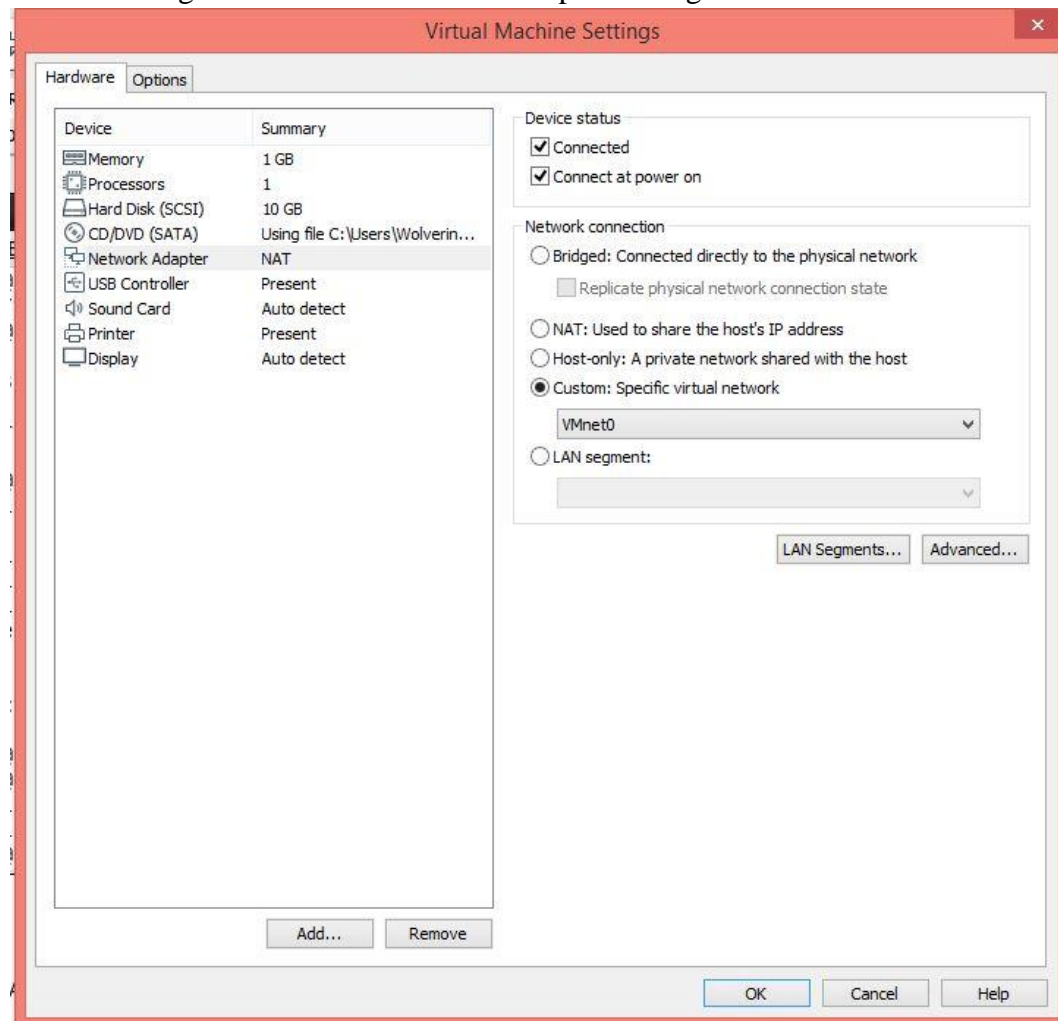
Step 11:

We will also restart the DHCP service using the service command with restart option.

```
[root@localhost Packages]# service dhcpd restart
Shutting down dhcpd: [ OK ]
Starting dhcpd: [ OK ]
[root@localhost Packages]#
```

Step 12:

We will change the VMware Network Adapter setting to VMnet0 for both client and server.



Client Side:

Step 1:

First, we will restart the Network Manager service for the DHCP server to assign the IP using the service command with restart option.

```
[root@localhost ~]# service NetworkManager restart
Stopping NetworkManager daemon: [ OK ]
Setting network parameters... [ OK ]
Starting NetworkManager daemon: [ OK ]
```

Step 2:

Now will check if the DHCP server has assigned IP address to this machine by using the ifconfig command.

```
[root@localhost ~]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0C:29:F3:1B:CE
          inet6 addr: fe80::20c:29ff:fef3:1bce/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:189 errors:0 dropped:0 overruns:0 frame:0
          TX packets:250 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:23836 (23.2 KiB)  TX bytes:28085 (27.4 KiB)
          Interrupt:19 Base address:0x2000
```

Server Side:

Step 13:

Now we will check lease file for the entry of the client-side machine by checking the hardware address.

```
[root@localhost ~]# cat /var/lib/dhcpd/dhcpd.leases
# The format of this file is documented in the dhcpd.leases(5) manual page.
# This lease file was written by isc-dhcp-4.1.1-P1

lease 192.168.0.100 {
    starts 3 2020/09/30 08:16:05;
    ends 3 2020/09/30 08:26:05;
    tstp 3 2020/09/30 08:26:05;
    cltt 3 2020/09/30 08:16:05;
    binding state active;
    next binding state free;
    hardware ethernet 00:0c:29:d1:51:c7;
}
server-uid "\000\001\000\001'\006\372\266\000\014)\325\370e";
```


PRACTICAL-2

Aim: Initial settings: Add a User, Network Settings, change to static IP address, Disable IPv6 if not needed, Configure Services, display the list of services which are running, Stop and turn OFF auto-start setting for a service if you don't need it, Sudo Settings.

Step 1:

Adding user 'user0' and setting password 'user0' for this user using useradd command with sudo.

```
[root@localhost Desktop]# sudo useradd user7563
```

Step 2:

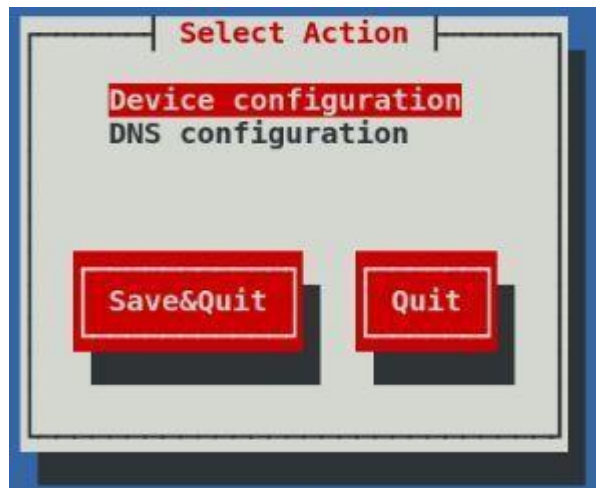
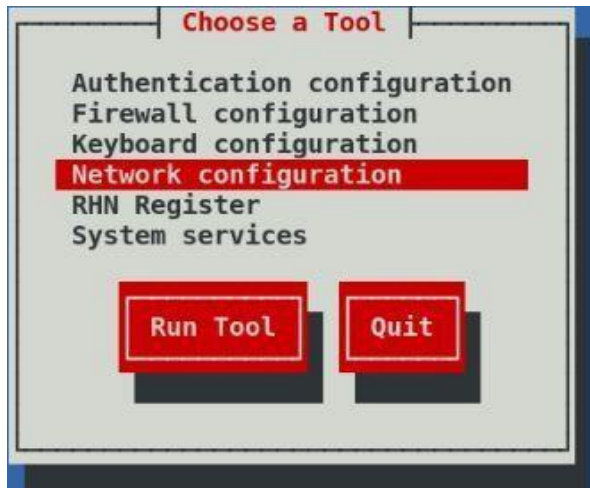
Setting password 'user0' for this user using passwd command with sudo.

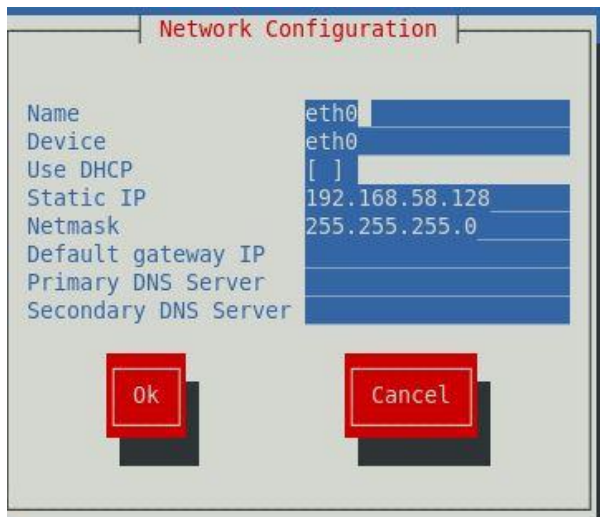
```
[root@localhost Desktop]# sudo passwd user7563
Changing password for user user7563.
New password:
BAD PASSWORD: it is too simplistic/systematic
Retype new password:
passwd: all authentication tokens updated successfully.
```

Step 3:

Setting static IP address using setup command and going into Network Configuration → Device Configuration → eth0... → deselect DHCP and set a static IP address → OK → save → Save and Quit.

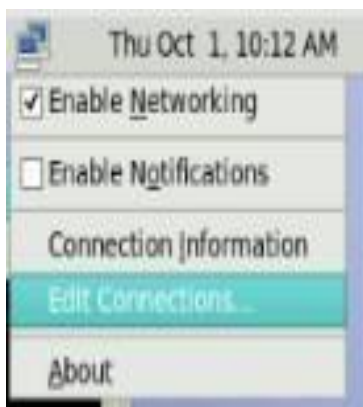
```
[root@localhost Desktop]# sudo setup
```





Step 4:

Disabling IPv6 by right click on NetworkManager on status bar of your desktop → Edit Connections → select network card which in case is eth0 → edit → go to IPv6 tab → Method: Ignore → Apply → close.



Step 5:

Displaying all running service in the machine using service command with `--status-all` option.

```
[root@localhost Desktop]# sudo service --status-all
abrttd (pid 1650) is running...
acpid (pid 1483) is running...
atd (pid 1688) is running...
auditd (pid 1342) is running...
avahi-daemon (pid 1441) is running...
Usage: /etc/init.d/bluetooth {start|stop}
cpuspeed is stopped
crond (pid 1658) is running...
cupsd (pid 1458) is running...
dhcpd (pid 3268) is running...
dhcpd is stopped
dhcrelay is stopped
dnsmasq is stopped
Usage: /etc/init.d/firstboot {start|stop}
hald (pid 1492) is running...
httpd is stopped
Table: filter
Chain INPUT (policy ACCEPT)
num  target      prot opt source                destination
1    ACCEPT      all  opt  ::/0                  ::/0          state RELATED,
ESTABLISHED
2    ACCEPT      icmpv6  ::/0                  ::/0
```

PRACTICAL -6

Aim: Configure NFS Server to share directories on your Network, Configure NFS Client.

Step 1:

Before we start the practical, we will check the necessary NFS packages are installed in our system using the rpm command with -q(query) option.

```
[root@ldapl Desktop]# rpm -q nfs-utils  
nfs-utils-1.2.3-7.el6.i686
```

Step 2:

After verifying that the NFS packages are installed we will create a shared directory called “/data” in server and give user, groups and other the read, write and execute permission.

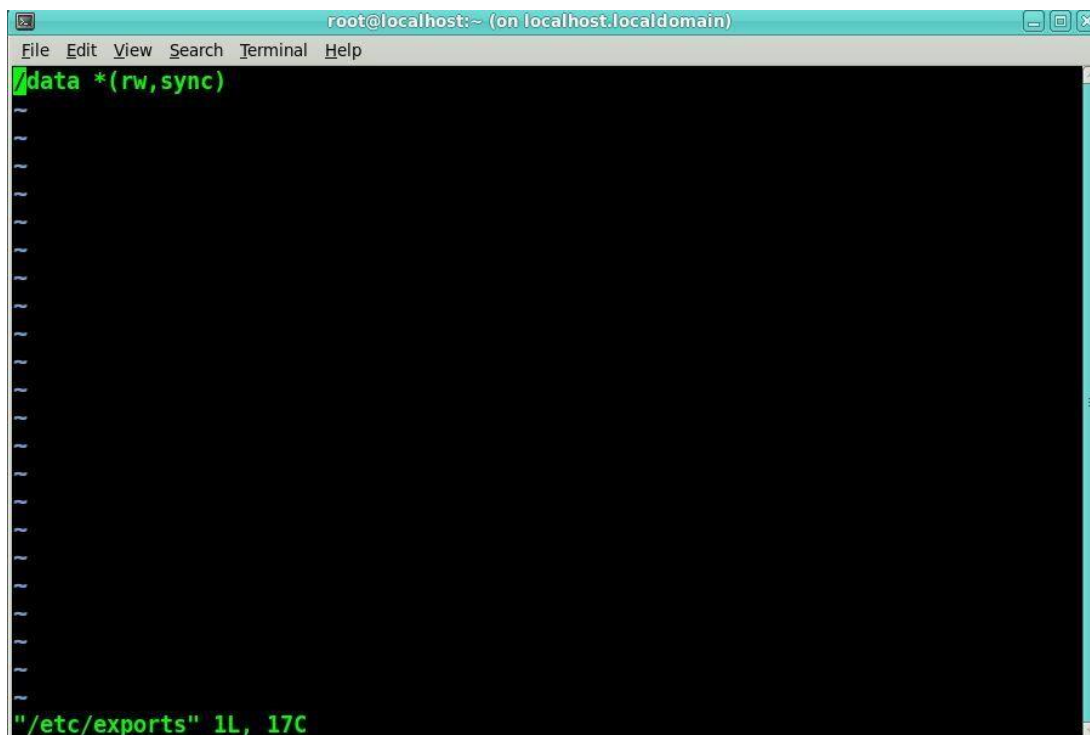
```
[root@localhost Packages]# mkdir /redhat  
[root@localhost Packages]# chmod 777 /redhat
```

Step 3:

Now edit the **export shared directory on NFS Server i.e. /etc/exports** using the vi editor.

```
[root@localhost Packages]# vim /etc/exports
```

Now add the following line in the file save and exit.



Step 4:

To allow NFS server to access from outbound we will stop the iptables service using the service command with stop option.

```
[root@localhost ~]# service iptables stop
iptables: Flushing firewall rules:      [ OK ]
iptables: Setting chains to policy ACCEPT: filter [ OK ]
iptables: Unloading modules:           [ OK ]
```

Step 5:

We will restart the rpcbind service using the service command with restart option.

```
[root@localhost ~]# service rpcbind restart
Stopping rpcbind:                       [ OK ]
Starting rpcbind:                       [ OK ]
```

Step 6:

We will restart the nfs service using the service command with restart option.

```
[root@localhost Packages]# service nfs restart
Shutting down NFS mountd:               [ OK ]
Shutting down NFS daemon:               [ OK ]
Shutting down NFS quotas:               [ OK ]
Shutting down NFS services:             [ OK ]
Starting NFS services: exportfs: No options for /redhat 192.168.10.0: suggest 1
92.168.10.0(sync) to avoid warning
exportfs: No host name given with /redhat (rw,sync), suggest *(rw,sync) to avoid
warning
Starting NFS quotas:                   [ OK ]
Starting NFS daemon:                   [ OK ]
Starting NFS mountd:                   [ OK ]
```

Step 7: To start the nfs service at boot time we use the chkconfig command with on option.

```
[root@localhost Packages]# chkconfig nfs on
```

Step 8:

We will use the showmount command with -e(export) option to check the list of files on localhost.

```
[root@localhost Packages]# showmount -e
Export list for localhost.localdomain:
/redhat (everyone)
```

Step 9:

We will use the mount command to mount the “/data” directory from server machine.

```
[root@localhost Packages]# mount 192.168.58.128:/redhat
```

Step 10:

We will use the showmount command with -e(export) option to check the list of files on localhost.

```
[root@MYSYSTEM Desktop]# showmount -e localhost
Export list for localhost:
/data *
```

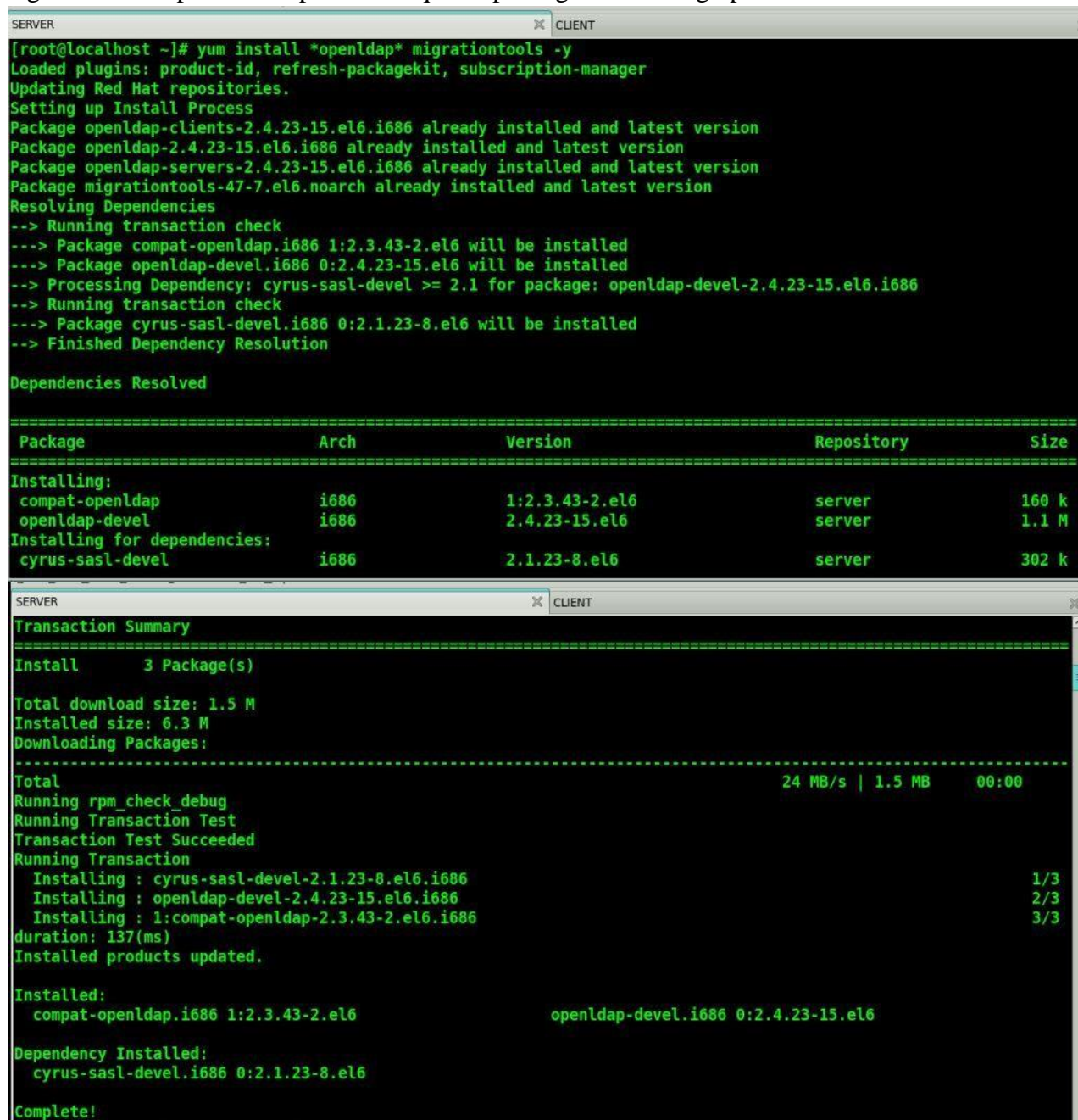
PRACTICAL 7

Aim: Configure LDAP Server, Configure LDAP Server in order to share users' accounts in your local networks, Add LDAP User Accounts in the OpenLDAP Server, Configure LDAP Client in order to share users' accounts in your local networks.

Server Side:

Step 1:

Before we start, we will install the openldap package using the yum command with install option and migrationtools option this option all required packages for setting up LDAP.



```
[root@localhost ~]# yum install *openldap* migrationtools -y
Loaded plugins: product-id, refresh-packagekit, subscription-manager
Updating Red Hat repositories.
Setting up Install Process
Package openldap-clients-2.4.23-15.el6.i686 already installed and latest version
Package openldap-2.4.23-15.el6.i686 already installed and latest version
Package openldap-servers-2.4.23-15.el6.i686 already installed and latest version
Package migrationtools-47-7.el6.noarch already installed and latest version
Resolving Dependencies
--> Running transaction check
--> Package compat-openldap.i686 1:2.3.43-2.el6 will be installed
--> Package openldap-devel.i686 0:2.4.23-15.el6 will be installed
--> Processing Dependency: cyrus-sasl-devel >= 2.1 for package: openldap-devel-2.4.23-15.el6.i686
--> Running transaction check
--> Package cyrus-sasl-devel.i686 0:2.1.23-8.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                                Arch      Version              Repository            Size
=====
Installing:
compat-openldap                        i686      1:2.3.43-2.el6       server                160 k
openldap-devel                         i686      2.4.23-15.el6        server                1.1 M
Installing for dependencies:
cyrus-sasl-devel                       i686      2.1.23-8.el6         server                302 k
=====

Transaction Summary
=====
Install      3 Package(s)

Total download size: 1.5 M
Installed size: 6.3 M
Downloading Packages:
-----
Total                                          24 MB/s | 1.5 MB    00:00
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : cyrus-sasl-devel-2.1.23-8.el6.i686                1/3
  Installing : openldap-devel-2.4.23-15.el6.i686                2/3
  Installing : 1:compat-openldap-2.3.43-2.el6.i686              3/3
duration: 137(ms)
Installed products updated.

Installed:
  compat-openldap.i686 1:2.3.43-2.el6                openldap-devel.i686 0:2.4.23-15.el6

Dependency Installed:
  cyrus-sasl-devel.i686 0:2.1.23-8.el6

Complete!
```

Step 2:

We will set password for the LDAP using the `slappasswd` command.

```
[root@localhost cn=config]# slappasswd
New password:
Re-enter new password:
{SSHA}b1kINpoP43zyDQg68YPfauinKT7FJoRk
```

After setting the password we will copy the encrypted password.

Step 3:

Now we will copy the encrypted password the we copied into the `/etc/pass` file; we will use vim editor to open the `/etc/pass` file.

```
[root@localhost cn=config]# vim /etc/pass
```

Copy the encrypted password into the file.

[illegible]

Step 4:

Now change directory to /etc/openldap/slapd.d/cn=config using the cd option.

```
[root@localhost ~]# cd /etc/openldap/slapd.d/cn=config
```

Step 5:

We will show file or directory, size, modified date and time, file or folder name and owner of file and its permission of the `cn=config` directory using the `ls` command with `-l` option.

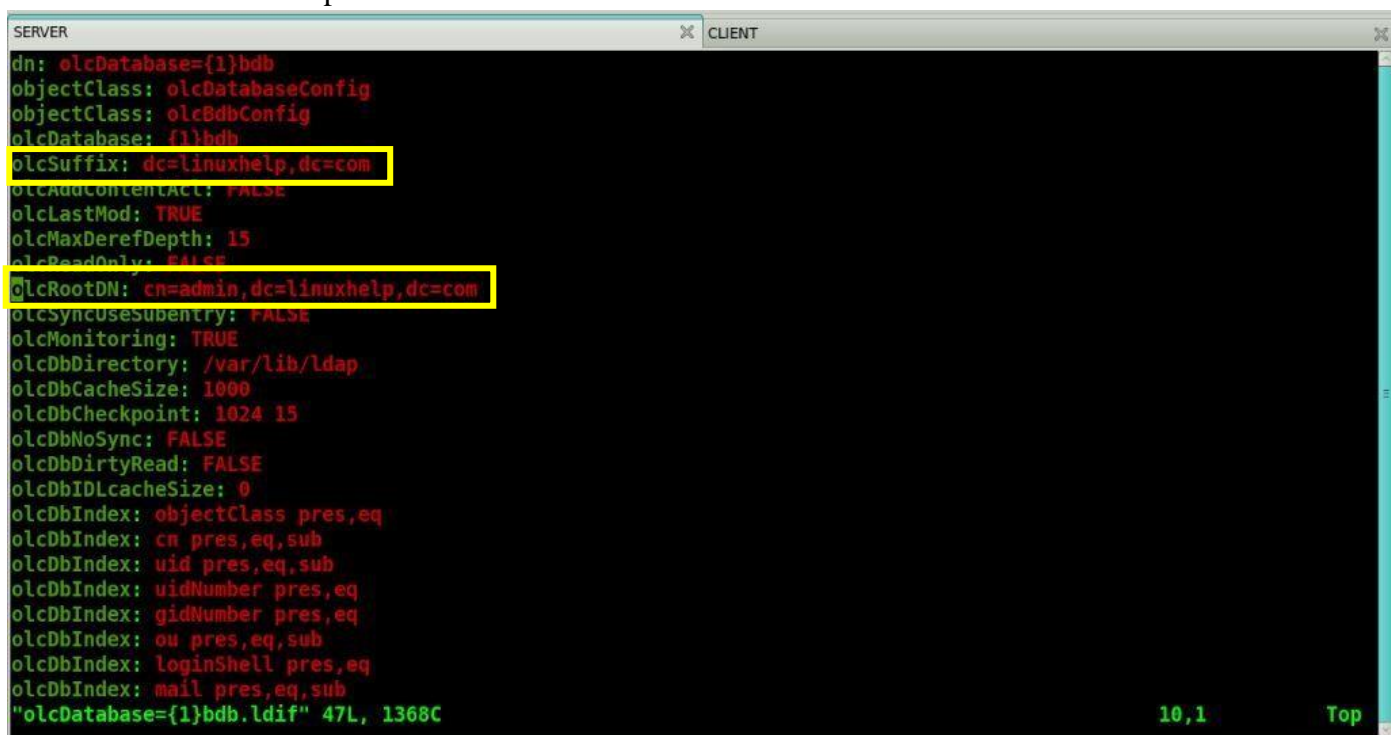

```
[root@localhost cn=config]# ls -l
total 72
drwx-----. 2 ldap ldap 4096 Sep 30 01:36 cn=schema
-rw-----. 1 ldap ldap 51896 Sep 30 01:36 cn=schema.ldif
-rw-----. 1 ldap ldap 522 Sep 30 01:36 olcDatabase={0}config.ldif
-rw-----. 1 ldap ldap 1370 Sep 30 01:48 olcDatabase={1}bdb.ldif
-rw-----. 1 ldap ldap 525 Sep 30 01:36 olcDatabase={-1}frontend.ldif
-rw-----. 1 ldap ldap 547 Sep 30 01:49 olcDatabase={2}monitor.ldif
```

Step 6:

We will edit the olcDatabase={1}bdb.ldif configuration file and replace value of olcSuffix and olcRootDN domain name, also add few lines; we will edit the file using the vim editor.

```
[root@MYSYSTEM cn=config]# vim olcDatabase={1}bdb.ldif
```

Alter values of olcSuffix: dc=example and olcRootDN: dc=example to olcSuffix: dc=linuxhelp and olcRootDN: dc=linuxhelp.



```
SERVER CLIENT
dn: olcDatabase={1}bdb
objectClass: olcDatabaseConfig
objectClass: olcBdbConfig
olcDatabase: {1}bdb
olcSuffix: dc=linuxhelp,dc=com
olcAddContentAcc: FALSE
olcLastMod: TRUE
olcMaxDerefDepth: 15
olcReadOnly: FALSE
olcRootDN: cn=admin,dc=linuxhelp,dc=com
olcSyncUsesSubentry: FALSE
olcMonitoring: TRUE
olcDbDirectory: /var/lib/ldap
olcDbCacheSize: 1000
olcDbCheckpoint: 1024 15
olcDbNoSync: FALSE
olcDbDirtyRead: FALSE
olcDbIDLcacheSize: 0
olcDbIndex: objectClass pres,eq
olcDbIndex: cn pres,eq,sub
olcDbIndex: uid pres,eq,sub
olcDbIndex: uidNumber pres,eq
olcDbIndex: gidNumber pres,eq
olcDbIndex: ou pres,eq,sub
olcDbIndex: loginShell pres,eq
olcDbIndex: mail pres,eq,sub
"olcDatabase={1}bdb.ldif" 47L, 1368C
10,1 Top
```

Add the highlighted lines at the end of the file.


```
SERVER CLIENT
olcDbIndex: uidNumber pres,eq
olcDbIndex: gidNumber pres,eq
olcDbIndex: ou pres,eq,sub
olcDbIndex: loginShell pres,eq
olcDbIndex: mail pres,eq,sub
olcDbIndex: sn pres,eq,sub
olcDbIndex: givenName pres,eq,sub
olcDbIndex: memberUid pres,eq,sub
olcDbIndex: nisMapName pres,eq,sub
olcDbIndex: nisMapEntry pres,eq,sub
olcDbLinearIndex: FALSE
olcDbMode: 0600
olcDbSearchStack: 16
olcDbShmKey: 0
olcDbCacheFree: 1
olcDbDNcacheSize: 0
structuralObjectClass: olcBdbConfig
entryUUID: c0cc54ae-9743-103a-8cd9-c9f078107619
creatorsName: cn=config
createTimestamp: 20200930083617Z
entryCSN: 20200930083617.673717Z#000000#000#000000
modifiersName: cn=config
modifyTimestamp: 20200930083617Z
olcRootPW: {SSHA}b1kINpoP43zyDQg68YPfauINKT7FJoRk
olcTLSCertificateFile: /etc/pki/tls/certs/linuxhelp.pem
olcTLSCertificateKeyFile: /etc/pki/tls/certs/linuxhelpkey.pem

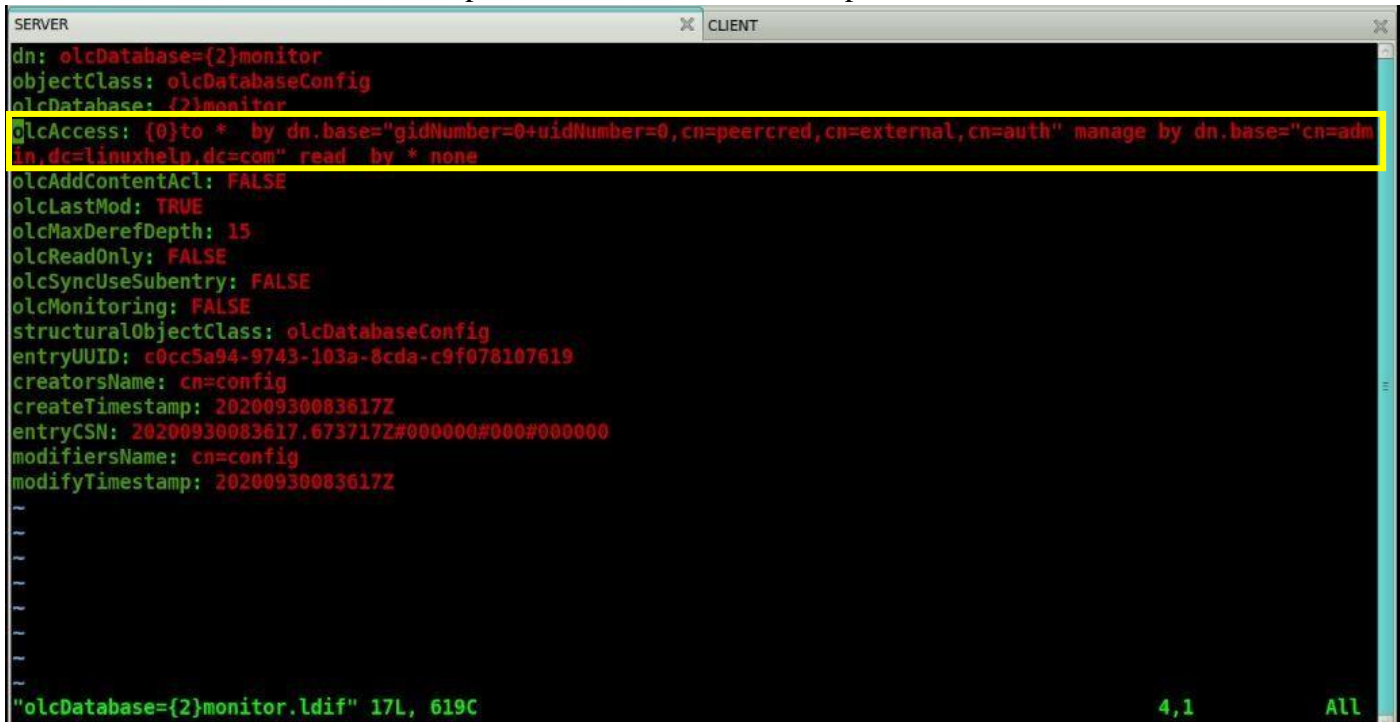
47,1 Bot
```

Step 7:

We will edit the `olcDatabase={2}monitor.ldif` configuration file and replace value of `olcAccess` domain name; we will edit the file using the vim editor.

```
[root@MYSYSTEM cn=config]# vim olcDatabase={2}monitor.ldif
```

Alter values of `olcAccess`: `dc=example` to `olcAccess`: `dc=linuxhelp`.



```
SERVER CLIENT
dn: olcDatabase={2}monitor
objectClass: olcDatabaseConfig
olcDatabase: {2}monitor
olcAccess: {0}to * by dn.base="gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth" manage by dn.base="cn=admin,dc=linuxhelp,dc=com" read by * none
olcAddContentAcl: FALSE
olcLastMod: TRUE
olcMaxDerefDepth: 15
olcReadOnly: FALSE
olcSyncUseSubentry: FALSE
olcMonitoring: FALSE
structuralObjectClass: olcDatabaseConfig
entryUUID: c0cc5a94-9743-103a-8cda-c9f078107619
creatorsName: cn=config
createTimestamp: 20200930083617Z
entryCSN: 20200930083617.673717Z#000000#000#000000
modifiersName: cn=config
modifyTimestamp: 20200930083617Z
-
-
-
-
-
-
-
-
"olcDatabase={2}monitor.ldif" 17L, 619C 4,1 All
```

Step 8:

We will verify the configuration file using `slaptest` command with `-u` option.

```
[root@localhost cn=config]# slaptest -u
config file testing succeeded
```

Step 9:

Now the `slaptest` in succeeded ignore the Checksum error, Start the `slapd` service using the `service` command with `restart` option.

```
[root@localhost cn=config]# service slapd restart
Stopping slapd: [ OK ]
Starting slapd: [ OK ]
```

Step 10:

Now to start the `slapd` service at boot we will use `chkconfig` command with `on` option.

```
[root@localhost cn=config]# chkconfig slapd on
```

Step 11:

To check the status of `slapd` service we will use the `service` command with `status` option.

```
[root@localhost cn=config]# service slapd status
slapd (pid 15136) is running...
```

Step 12:

Now to configure the database for LDAP we will copy the contents of DB_CONFIG.example which is located in /usr/share/openldap-servers/ directory to DB_CONFIG file which is located in /var/lib/ldap/ directory to do that we will use the cp command.

```
[root@localhost cn=config]# cp /usr/share/openldap-servers/DB_CONFIG.example /var/lib/ldap/DB_CONFIG
```

Step 13:

Now to change the file permissions to ldap user and ldap group of /var/lib/ldap we will use the chown command with -R option.

```
[root@localhost cn=config]# chown -R ldap:ldap /var/lib/ldap/
```

Step 14:

Now add the following LDAP Schemas.

```
[root@localhost cn=config]# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/cosine.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=cosine,cn=schema,cn=config"
```

```
[root@localhost cn=config]# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/nis.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=nis,cn=schema,cn=config"
```

```
[root@localhost cn=config]# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/inetorgperson.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=inetorgperson,cn=schema,cn=config"
```

Step 15:

Now generate the certificate file that we have mentioned in the configuration file for LDAP but while generating it will ask for some information. For generating the certificate, we will use the openssl command with req option, -new option, -x509 option, -nodes option, -out option, -days option and also provide the location for saving the certificate and the key.

```
[root@localhost cn=config]# openssl req -new -x509 -nodes -out /etc/pki/tls/certs/linuxhelp.pem -keyout /etc/pki/tls/certs/linuxhelpkey.pem -days 365
Generating a 2048 bit RSA private key
.....+++
.....+++
writing new private key to '/etc/pki/tls/certs/linuxhelpkey.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:IN
State or Province Name (full name) [:]:Maharashtra
Locality Name (eg, city) [Default City]:Navi Mumbai
Organization Name (eg, company) [Default Company Ltd]:TYCS
Organizational Unit Name (eg, section) [:]:IT
Common Name (eg, your name or your server's hostname) [:]:server.linuxhelp.com
Email Address [:]:admin@linuxhelp.com
```


Step 16:

Now to verify about the generated certificate we use the ls command with -l option and location of the certificate.

```
[root@localhost cn=config]# ls -l /etc/pki/tls/certs/*.pem
-rw-r--r--. 1 root root 1704 Sep 30 09:24 /etc/pki/tls/certs/linuxhelpkey.pem
-rw-r--r--. 1 root root 1456 Sep 30 09:24 /etc/pki/tls/certs/linuxhelp.pem
-rw-r-----. 1 root ldap 3246 Sep 30 01:36 /etc/pki/tls/certs/slappedd.pem
```

Step 17:

Now navigate to this directory /usr/share/migrationtools/ using the cd command.

```
[root@localhost cn=config]# cd /usr/share/migrationtools/
```

Step 18:

Once in the migrationtools directory we will list the content of the file using the ls command with -l option and edit the migrate-commons.ph file using the vim command.

```
[root@localhost migrationtools]# ls -l
total 128
-rwxr-xr-x. 1 root root 2652 Apr 27 2010 migrate_aliases.pl
-rwxr-xr-x. 1 root root 2950 Apr 27 2010 migrate_all_netinfo_offline.sh
-rwxr-xr-x. 1 root root 2946 Apr 27 2010 migrate_all_netinfo_online.sh
-rwxr-xr-x. 1 root root 3011 Apr 27 2010 migrate_all_nis_offline.sh
-rwxr-xr-x. 1 root root 3006 Apr 27 2010 migrate_all_nis_online.sh
-rwxr-xr-x. 1 root root 3164 Apr 27 2010 migrate_all_nisplus_offline.sh
-rwxr-xr-x. 1 root root 3146 Apr 27 2010 migrate_all_nisplus_online.sh
-rwxr-xr-x. 1 root root 5267 Apr 27 2010 migrate_all_offline.sh
-rwxr-xr-x. 1 root root 7468 Apr 27 2010 migrate_all_online.sh
-rwxr-xr-x. 1 root root 3278 Apr 27 2010 migrate_automount.pl
-rwxr-xr-x. 1 root root 2608 Apr 27 2010 migrate_base.pl
-rw-r--r--. 1 root root 8880 Apr 27 2010 migrate_common.ph
-rwxr-xr-x. 1 root root 2952 Apr 27 2010 migrate_rstab.pl
-rwxr-xr-x. 1 root root 2714 Apr 27 2010 migrate_group.pl
-rwxr-xr-x. 1 root root 2751 Apr 27 2010 migrate_hosts.pl
-rwxr-xr-x. 1 root root 2856 Apr 27 2010 migrate_netgroup_byhost.pl
-rwxr-xr-x. 1 root root 2856 Apr 27 2010 migrate_netgroup_byuser.pl
-rwxr-xr-x. 1 root root 3879 Apr 27 2010 migrate_netgroup.pl
-rwxr-xr-x. 1 root root 2840 Apr 27 2010 migrate_networks.pl
-rwxr-xr-x. 1 root root 5635 Apr 27 2010 migrate_passwd.pl
-rwxr-xr-x. 1 root root 2428 Apr 27 2010 migrate_profile.pl
-rwxr-xr-x. 1 root root 2873 Apr 27 2010 migrate_protocols.pl
-rwxr-xr-x. 1 root root 2854 Apr 27 2010 migrate_rpc.pl
-rwxr-xr-x. 1 root root 10248 Apr 27 2010 migrate_services.pl
-rwxr-xr-x. 1 root root 3419 Apr 27 2010 migrate_slappedd_conf.pl
```

Step 19:

Now go to line number 71 and edit domain name as required also edit the base name in line number 74.

```
# Default DNS domain
$DEFAULT_MAIL_DOMAIN = "linuxhelp.com";

# Default base
$DEFAULT_BASE = "dc=linuxhelp,dc=com";
```

Step 20:

Now go to line number 90 and change the EXTENDED_SCHEMA value to "1"

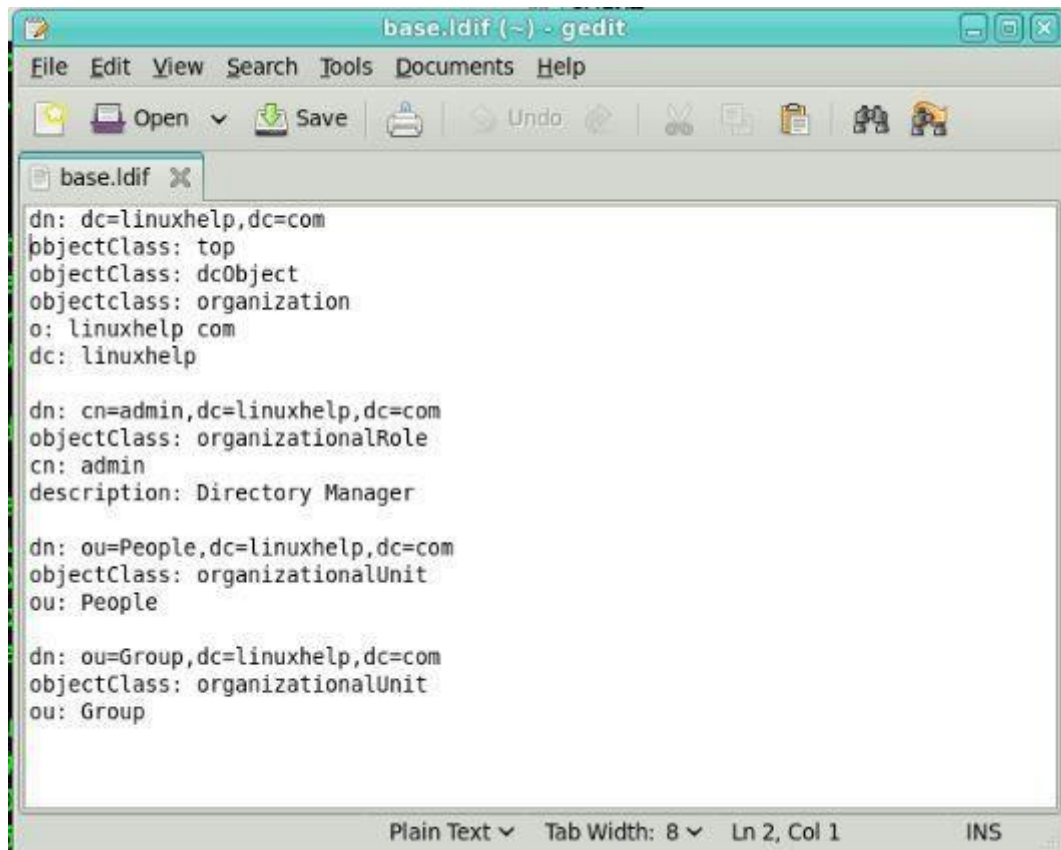
```
$EXTENDED_SCHEMA = 1;
```

Step 21:

Now create the base.ldif file for domain in /root directory using touch command.

```
[root@localhost migrationtools]# touch /root/base.ldif  
[root@localhost migrationtools]# gedit /root/base.ldif
```

Open it using gedit editor and add the following lines, save and quit.



Step 22:

Now we will read the file using the cat command

```
[root@localhost migrationtools]# cat /root/base.ldif
dn: dc=linuxhelp,dc=com
objectClass: top
objectClass: dcObject
objectClass: organization
o: linuxhelp com
dc: linuxhelp

dn: cn=admin,dc=linuxhelp,dc=com
objectClass: organizationalRole
cn: admin
description: Directory Manager

dn: ou=People,dc=linuxhelp,dc=com
objectClass: organizationalUnit
ou: People

dn: ou=Group,dc=linuxhelp,dc=com
objectClass: organizationalUnit
ou: Group
```

Step 23:

Now create two LDAP users ldapuser1 and ldapuser2 using useradd command, also set password for the users using the passwd command.

```
[root@localhost migrationtools]# useradd ldapuser1
[root@localhost migrationtools]# useradd ldapuser2
[root@localhost migrationtools]# passwd ldapuser1
Changing password for user ldapuser1.
New password:
BAD PASSWORD: it is based on a dictionary word
Retype new password:
passwd: all authentication tokens updated successfully.
[root@localhost migrationtools]# passwd ldapuser2
Changing password for user ldapuser2.
New password:
BAD PASSWORD: it is based on a dictionary word
Retype new password:
passwd: all authentication tokens updated successfully.
```

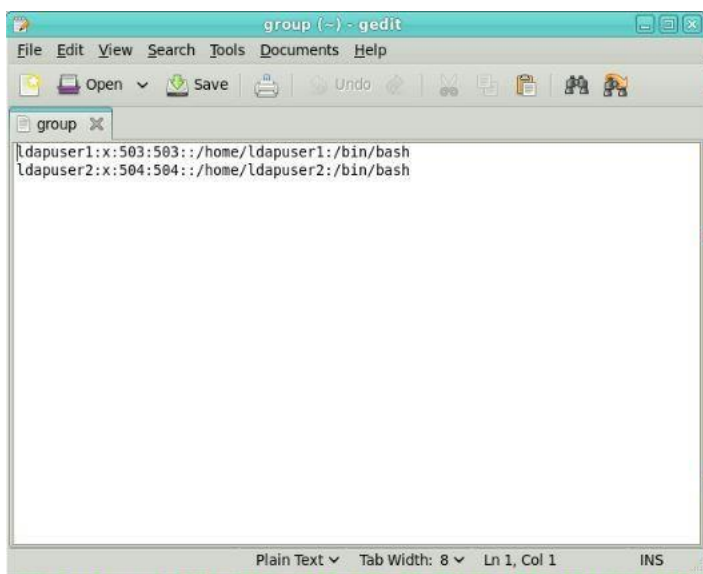
Step 24:

Now copy the user information for LDAP users from the file passwd file in /etc directory using the tail command to read the last 10 lines of the file.


```
[root@localhost migrationtools]# tail /etc/passwd
harsh:x:500:500:Harsh Pawani:/home/harsh:/bin/bash
lucy:x:501:501:./home/lucy:/bin/bash
rpc:x:32:32:Rpcbind Daemon:/var/cache/rpcbind:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
user4:x:502:502:./rhome/user4:/bin/bash
ldap:x:55:55:LDAP User:/var/lib/ldap:/sbin/nologin
named:x:25:25:Named:/var/named:/sbin/nologin
ldapuser1:x:503:503:./home/ldapuser1:/bin/bash
ldapuser2:x:504:504:./home/ldapuser2:/bin/bash
```

Step 25 :

Now paste the information to a group file in /root directory using the gedit editor.



Step 26:

To verify the contents of the group file in /root directory using the cat command.

```
[root@localhost migrationtools]# cat /root/group
ldapuser1:x:503:503:./home/ldapuser1:/bin/bash
ldapuser2:x:504:504:./home/ldapuser2:/bin/bash
```

Step 27:

Now to convert the passwd file into ldif (LDAP Data Interchange Format) format.

```
[root@localhost migrationtools]# ./migrate_passwd.pl /root/passwd /root/users.ldif
```

Step 28:

Now to convert the group file into ldif (LDAP Data Interchange Format) format.

```
[root@localhost migrationtools]# ./migrate_group.pl /root/group /root/grops.ldif
```

Step 29:

Now list the contents of /root directory using the ls command with -l option.

```
[root@localhost migrationtools]# ls -l /root/
total 116
-rw-----, 1 root root 3350 Sep 22 04:27 anaconda-ks.cfg
-rw-r--r--, 1 root root 381 Sep 30 08:20 base.ldif
drwxr-xr-x, 2 root root 4096 Sep 25 11:45 Desktop
drwxr-xr-x, 2 root root 4096 Sep 25 11:45 Documents
drwxr-xr-x, 2 root root 4096 Sep 25 11:45 Downloads
-rw-r--r--, 1 root root 406 Sep 30 08:26 grops.ldif
-rw-r--r--, 1 root root 95 Sep 30 08:23 group
drwxr-xr-x, 2 root root 4096 Sep 28 23:16 harsh
drwxr-xr-x, 2 root root 4096 Sep 28 23:11 hello world
-rw-r--r--, 1 root root 38978 Sep 22 04:27 install.log
-rw-r--r--, 1 root root 10337 Sep 22 04:26 install.log.syslog
drwxr-xr-x, 2 root root 4096 Sep 27 07:33 media
drwxr-xr-x, 2 root root 4096 Sep 25 11:45 Music
drwxr-xr-x, 2 root root 4096 Sep 25 11:45 Pictures
drwxr-xr-x, 2 root root 4096 Sep 25 11:45 Public
drwxr-xr-x, 2 root root 4096 Sep 25 11:45 Templates
-rw-r--r--, 1 root root 0 Sep 30 08:25 users.ldif
drwxr-xr-x, 2 root root 4096 Sep 25 11:45 Videos
```

Step 30:

Now to import the all ldif files in the root directory created for base and groups into the LDAP database use the ldapadd command with -x option, -W option and -D option.

```
[root@localhost migrationtools]# ldapadd -x -W -D "cn=admin,dc=linuxhelp,dc=com" -f /root/base.ldif
Enter LDAP Password:
adding new entry "dc=linuxhelp,dc=com"

adding new entry "cn=admin,dc=linuxhelp,dc=com"

adding new entry "ou=People,dc=linuxhelp,dc=com"

adding new entry "ou=Group,dc=linuxhelp,dc=com"
```

```
[root@localhost migrationtools]# ldapadd -x -W -D "cn=admin,dc=linuxhelp,dc=com" -f /root/grops.ldif
Enter LDAP Password:
adding new entry "cn=ldapuser1,ou=Group,dc=linuxhelp,dc=com"

adding new entry "cn=ldapuser2,ou=Group,dc=linuxhelp,dc=com"

adding new entry "cn=,ou=Group,dc=linuxhelp,dc=com"
```

Step 31:

Now verify whether the information is imported to the database using the ldapsearch command using with -x option.

```
[root@localhost migrationtools]# ldapsearch -x cn=ldapuser1 -b dc=linuxhelp,dc=com
# extended LDIF
#
# LDAPv3
# base <dc=linuxhelp,dc=com> with scope subtree
# filter: cn=ldapuser1
# requesting: ALL
#
# ldapuser1, Group, linuxhelp.com
dn: cn=ldapuser1,ou=Group,dc=linuxhelp,dc=com
objectClass: posixGroup
objectClass: top
cn: ldapuser1
userPassword:: e2NyeXB0fXg=
gidNumber: 503
memberUid: 503

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

Step 32:

Now stop the firewall by stopping iptables service using the service command with stop option.

```
[root@localhost migrationtools]# service iptables stop
```

33:

Before proceeding further install the NFS and rpcbind package using the yum command using the install option.

```
[root@localhost migrationtools]# yum install nfs* rpcbind -y
Loaded plugins: product-id, refresh-packagekit, subscription-manager
Updating Red Hat repositories.
Setting up Install Process
Package nfs-utils-lib-1.1.5-3.el6.i686 already installed and latest version
Package 1:nfs-utils-1.2.3-7.el6.i686 already installed and latest version
Package nfs4-acl-tools-0.3.3-5.el6.i686 already installed and latest version
Package rpcbind-0.2.0-8.el6.i686 already installed and latest version
Nothing to do
```

Step 34:

Now open the /etc/exports file to share the home directory with client using the vim editor

```
[root@localhost migrationtools]# vim /etc/exports
```

```
/home *(rw,sync)
```

Step 35:

Restart the nfs service use the service command with restart option.

```
[root@localhost migrationtools]# service nfs restart
Shutting down NFS mountd: [ OK ]
Shutting down NFS daemon: [ OK ]
Shutting down NFS quotas: [ OK ]
Shutting down NFS services: [ OK ]
Starting NFS services: [ OK ]
Starting NFS quotas: [ OK ]
Starting NFS daemon: [ OK ]
Starting NFS mountd: [ OK ]
```

Step 36:

Restart the rpcbind service use the service command with restart option.

```
[root@localhost migrationtools]# service rpcbind restart
Stopping rpcbind: [ OK ]
Starting rpcbind: [ OK ]
```

Client Side:

Step 1:

To use the LDAP service, install openldap-client package using the yum command with install option.

```
[root@MYSYSTEM /]# yum install openldap-clients
Loaded plugins: product-id, refresh-packagekit, subscription-manager
Updating Red Hat repositories.
Setting up Install Process
Package openldap-clients-2.4.23-15.el6.i686 already installed and latest version
Nothing to do
```

Step 2:

To use the LDAP service, also install the nss-pam-ldapd package using the yum command with install option.

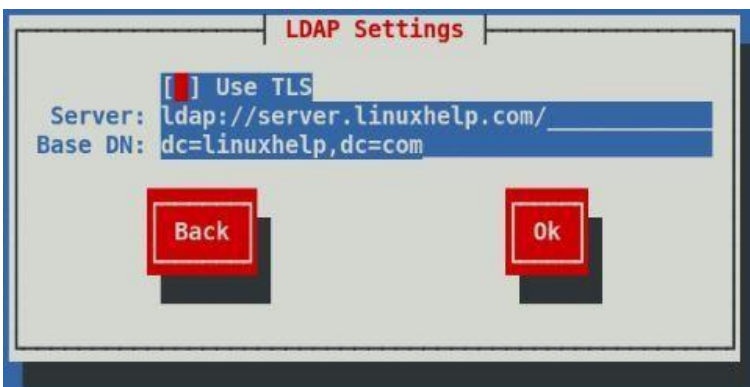
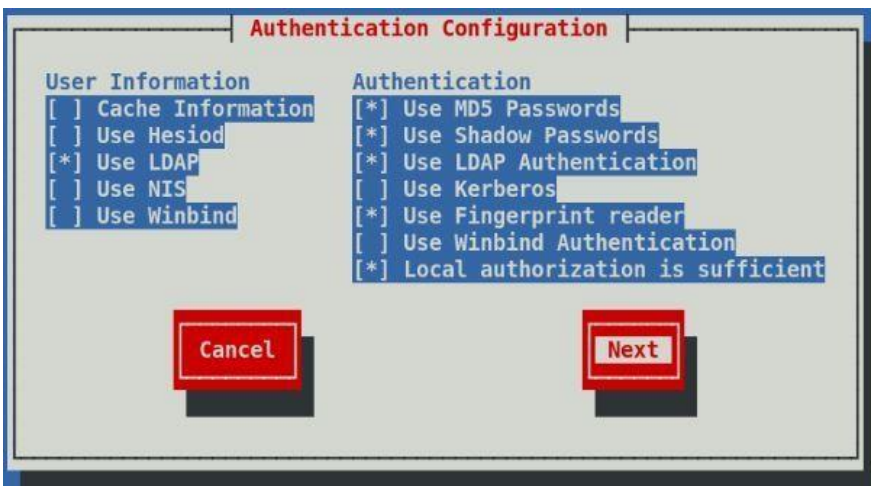
```
SERVER CLIENT
[root@MYSYSTEM /]# yum install nss-pam-ldapd
Loaded plugins: product-id, refresh-packagekit, subscription-manager
Updating Red Hat repositories.
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package nss-pam-ldapd.i686 0:0.7.5-7.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Size Repository
=====
Installing:
nss-pam-ldapd i686 0.7.5-7.el6 147 k server
Transaction Summary
=====
Install 1 Package(s)

Total download size: 147 k
Installed size: 452 k
Is this ok [y/N]: y
Downloading Packages:
Running rpm_check_debug
Running Transaction Test
```

```
[root@MYSYSTEM /]# authconfig-tui
```



```
Starting rpcbind: [ OK ]
```

```
Starting nslcd: [ OK ]
```

Step 4:

Now to test client-side configuration is properly working use the getent command.

```
[root@MYSYSTEM /]# getent passwd ldapuser1  
ldapuser1:x:503:503:./home/ldapuser1:/bin/bash
```

Step 5:

Now we can login for user ldapuser1 using the su command with - option.

```
[root@MYSYSTEM /]# su - ldapuser1  
[ldapuser1@MYSYSTEM ~]$
```

Step 6:

To exit use the exit command.

```
[ldapuser1@MYSYSTEM ~]$ exit  
logout
```

PRACTICAL-8

Aim: Configure NIS Server in order to share users' accounts in your local networks, Configure NIS Client to bind NIS Server

Server Side:

Step 1:

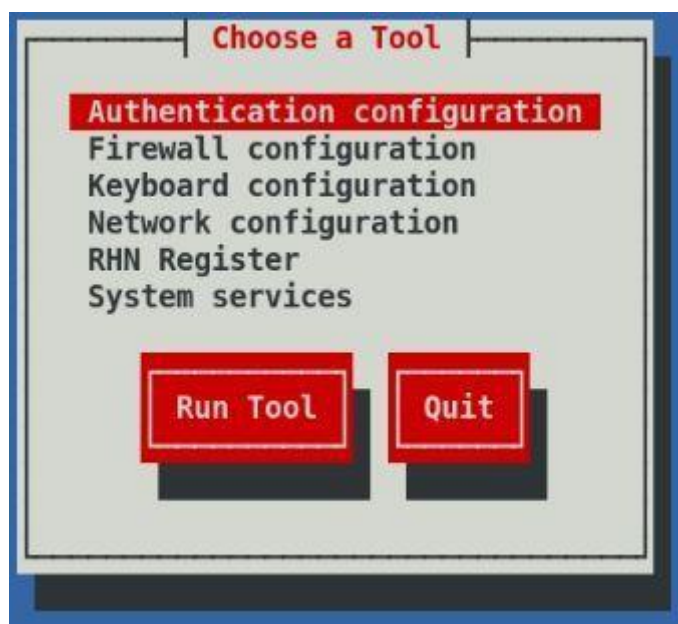
Before we start, we will check if we have installed the ypserv, ypbind, nfs-utils, make, xinetd and cachefilesd packages in the system using the rpm command with -qa(query all) option.

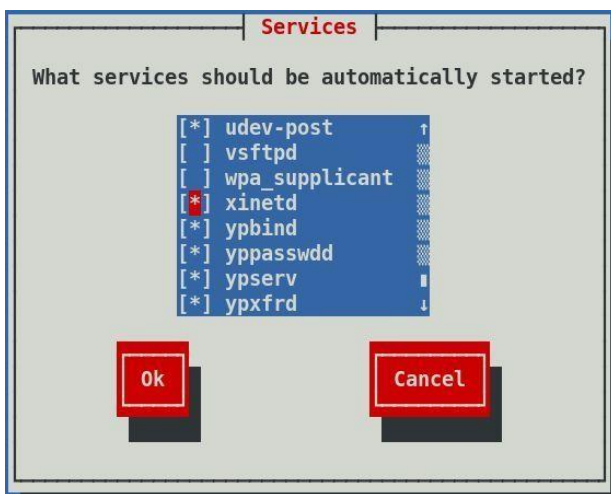
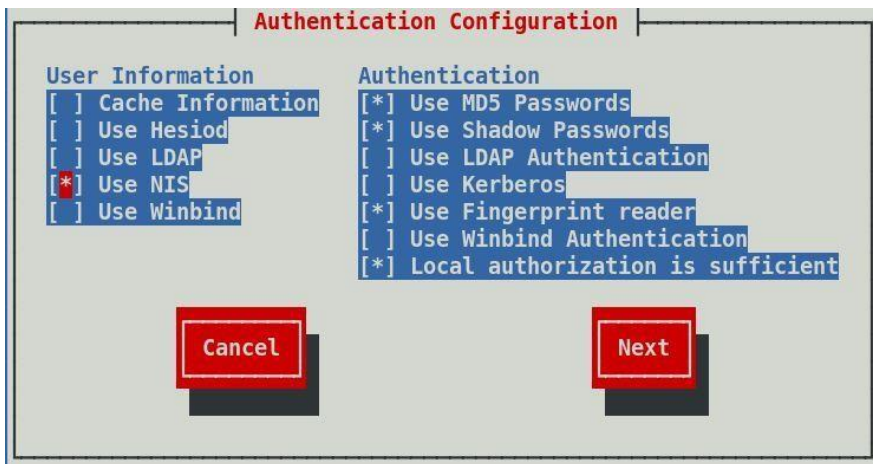
```
[root@localhost Desktop]# rpm -qa ypserv
ypserv-2.19-18.el6.i686
[root@localhost Desktop]# rpm -qa ypbind
ypbind-1.20.4-29.el6.i686
[root@localhost Desktop]# rpm -qa ypbind
ypbind-1.20.4-29.el6.i686
[root@localhost Desktop]# rpm -qa nfs-utils
nfs-utils-1.2.3-7.el6.i686
[root@localhost Desktop]# rpm -qa make
make-3.81-19.el6.i686
[root@localhost Desktop]# rpm -qa xinetd
xinetd-2.3.14-31.el6.i686
[root@localhost Desktop]# rpm -qa xinetd
xinetd-2.3.14-31.el6.i686
[root@localhost Desktop]# rpm -qa cachefilesd
cachefilesd-0.10.1-2.el6.i686
```

Step 2:

Now setup LDAP Authentication using setup command → Authentication configuration → turn on Use NIS → next → start xinetd, ypbind, ypserv and ypxfrd service → ok

```
[root@localhost Desktop]# setup
```





Step 3:

Now open the network file in /etc/sysconfig directory using gedit editor

```
[root@localhost Desktop]# gedit /etc/sysconfig/network
```

Edit the hostname to any name you prefer and add "NISDOMAIN=mydomain.com" line at the end.

```
HOSTNAME=ldapl.mydomain.com|
NETWORKING=yes
NISDOMAIN=mydomain.com
```

Step 4:

Now create a new directory rhome in "/" directory.

```
[root@localhost Desktop]# mkdir /rhome
```

Step 5:

Now create user "user1" in /rhome directory using useradd command with -d(to directory) option.

```
[root@localhost Desktop]# useradd -d /rhome/user1 user1
```

Setup password for the user1.

```
[root@localhost Desktop]# passwd user1
Changing password for user user1.
New password:
BAD PASSWORD: it is too short
BAD PASSWORD: is too simple
Retype new password:
Sorry, passwords do not match.
New password:
BAD PASSWORD: it is too short
BAD PASSWORD: is too simple
Retype new password:
passwd: all authentication tokens updated successfully.
```

Step 6:

Now give users, groups and other the read, write and execute permission to the file using chmod command with 777 option.

```
[root@localhost Desktop]# chmod 777 /rhome/user1
```

Step 7:

Now edit the export shared directory on NIS Server i.e. /etc/exports using the vi editor.

```
[root@localhost Desktop]# vi /etc/exports
```

Now add the following line in the file save and exit.

```
/rhome/user1 *(rw, sync)
```

Step 8:

Now open the /var/yp/Makefile using vi editor and locate the line 117 using “:117”.

```
[root@localhost Desktop]# vi /var/yp/MakeFile
```

Step 9:

Now restart xinetd, ypserv, nfs, yppasswdd service using service command with restart option.

```
[root@localhost yp]# service xinetd restart
Stopping xinetd: [ OK ]
Starting xinetd: [ OK ]
[root@localhost yp]# service ypserv restart
Stopping YP server services: [ OK ]
Starting YP server services: [ OK ]
[root@localhost yp]# service nfs restart
Shutting down NFS mountd: [ OK ]
Shutting down NFS daemon: [ OK ]
Shutting down NFS quotas: [ OK ]
Shutting down NFS services: [ OK ]
Starting NFS services: [ OK ]
Starting NFS quotas: [ OK ]
Starting NFS daemon: [ OK ]
Starting NFS mountd: [ OK ]
[root@localhost yp]# service yppasswdd restart
Stopping YP passwd service: [ OK ]
Starting YP passwd service: [ OK ]
```

Step 10:

Now change path to /var/yp directory and execute the make command to create database.

```
[root@localhost Desktop]# cd /var/yp
[root@localhost yp]# make
gmake[1]: Entering directory `/var/yp/mydomain.com'
Updating passwd.byname...
Updating passwd.byuid...
Updating group.byname...
```

Step 11:

Now update the database by running the following details.

```
[root@localhost yp]# /usr/lib/yp/ypinit -m

At this point, we have to construct a list of the hosts which will run NIS
servers. ldapl.mydomain.com is in the list of NIS server hosts. Please continu
e to add
the names for the other hosts, one per line. When you are done with the
list, type a <control D>.
    next host to add: ldapl.mydomain.com
    next host to add: PVR
    next host to add:
    next host to add: ^C
[root@localhost yp]# /usr/lib/yp/ypinit -m
```

```
Is this correct? [y/n: y] y
We need a few minutes to build the databases...
Building /var/yp/rhce/ypservers...
gethostbyname(): Resource temporarily unavailable
Running /var/yp/Makefile...
gmake[1]: Entering directory `/var/yp/rhce'
Updating passwd.byname...
Updating passwd.byuid...
Updating group.byname...
Updating group.bygid...
Updating hosts.byname...
Updating hosts.byaddr...
Updating rpc.byname...
Updating rpc.bynumber...
Updating services.byname...
Updating services.byservicename...
Updating netid.byname...
Updating protocols.bynumber...
Updating protocols.byname...
Updating mail.aliases...
gmake[1]: Leaving directory `/var/yp/rhce'

MYSYSTEM has been set up as a NIS master server.
```

Now you can run `ypinit -s MYSYSTEM` on all slave server.

Step 12:

Once you have updated all services, once again restart all the services to take effect using service command with restart option and also to make all the services online upon next reboot with the help of chkconfig command with “on” option.

```
[root@localhost yp]# service xinetd restart
Stopping xinetd: [ OK ]
Starting xinetd: [ OK ]
[root@localhost yp]# service ypserv restart
Stopping YP server services: [ OK ]
Starting YP server services: [ OK ]
[root@localhost yp]# service nfs restart
Shutting down NFS mountd: [ OK ]
Shutting down NFS daemon: [ OK ]
Shutting down NFS quotas: [ OK ]
Shutting down NFS services: [ OK ]
Starting NFS services: [ OK ]
Starting NFS quotas: [ OK ]
Starting NFS daemon: [ OK ]
Starting NFS mountd: [ OK ]
[root@localhost yp]# service yppasswdd restart
Stopping YP passwd service: [ OK ]
Starting YP passwd service: [ OK ]
[root@localhost yp]# chkconfig xinetd on
[root@localhost yp]# chkconfig nfs on
[root@localhost yp]# chkconfig ypserv on
[root@localhost yp]# chkconfig ypbind on
[root@localhost yp]# chkconfig yppasswdd on
[root@localhost yp]# █
```

Client Side:

Step 1:

Before we start, we will check if we have installed the yp-tools and ypbind packages in the system using the rpm command with -qa(query all) option.

```
[root@localhost Desktop]# rpm -qa yp-tools
yp-tools-2.9-10.el6.i686
[root@localhost Desktop]# rpm -qa ypbind
ypbind-1.20.4-29.el6.i686
```

Step 2:

Now we open the network file which is located in /etc/sysconfig directory using the vi editor.

```
[root@localhost Desktop]# vi /etc/sysconfig/network
```

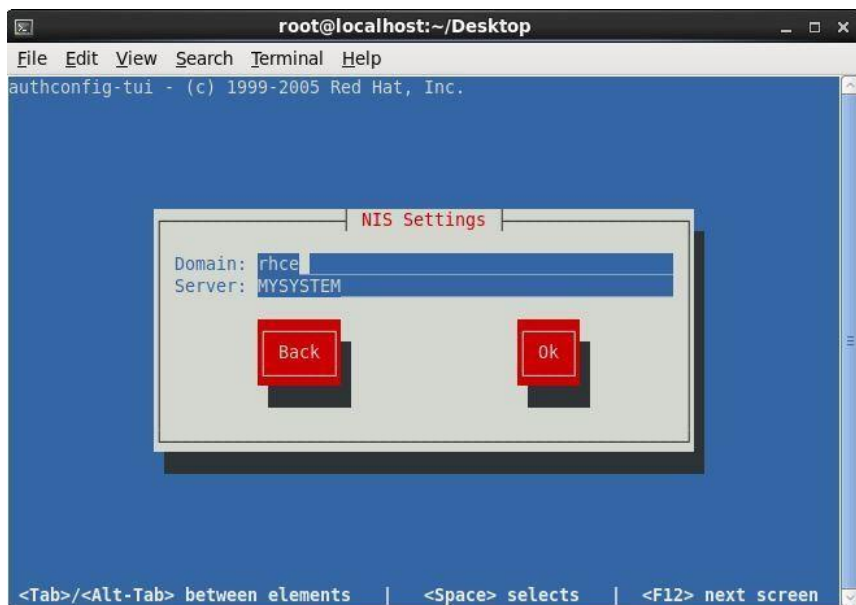
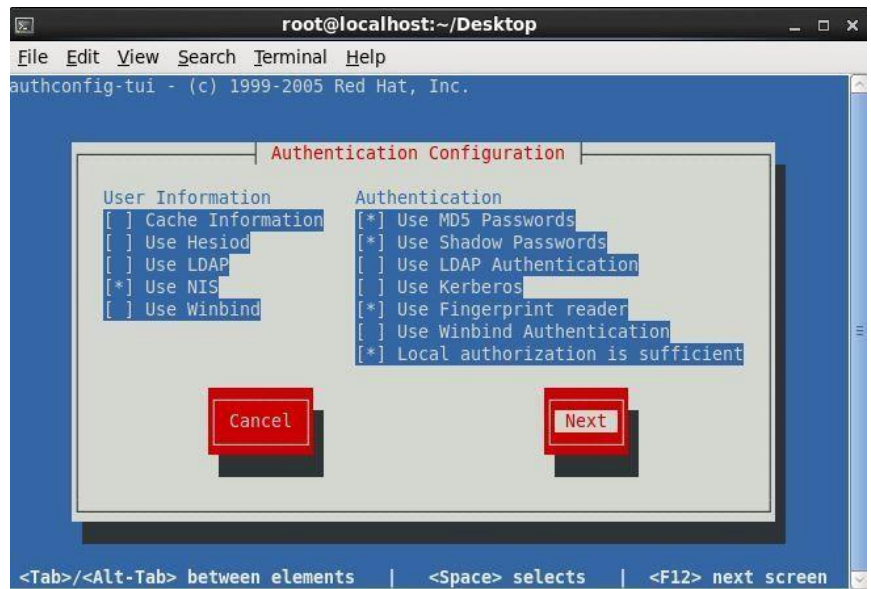
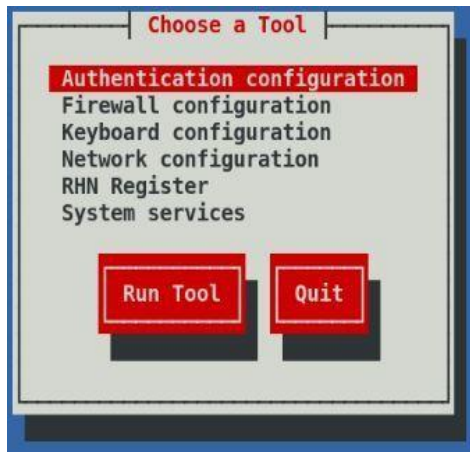
Add the "NISDOMAIN=mydomain.com" at the end of the file as shown below.

```
HOSTNAME=ldapl.mydomain.com
NETWORKING=yes
NISDOMAIN=mydomain.com
```

Step 3:

Now we use setup command → Authentication configuration → turn on Use NIS → next → enter the server name or IP address → ok.

```
[root@localhost Desktop]# setup
Starting rpcbind: [ OK ]
Starting NIS service: [ OK ]
Binding NIS service: ..... [ OK ]
```



PRACTICAL-10

Aim: Install Samba to share folders or files between Windows and Linux.

Server (RHEL 6) Side:

Step 1:

Before starting check if samba, samba-common and samba-client packages are installed in the system using the rpm command with -qa(query all) option.

```
[root@ldapl Desktop]# rpm -qa samba samba-common samba-client
samba-3.5.6-86.el6.i686
samba-common-3.5.6-86.el6.i686
samba-client-3.5.6-86.el6.i686
```

Step 2:

Now create a new directory in “/” directory using mkdir command, give user, group and other the permission to read, write and execute in the directory using the chmod command with 777 option and also list the security context of the file using the ls command with -ldZ option.

```
[root@ldapl Desktop]# mkdir /sambafile
[root@ldapl Desktop]# chmod 777 /sambafile
[root@ldapl Desktop]# ls -ldZ /sambafile
drwxrwxrwx. root root unconfined_u:object_r:default_t:s0 /sambafile
```

Step 3:

Now open the samba configuration file i.e. smb.conf located in “/etc/samba/” directory

```
[root@ldapl Desktop]# vim /etc/samba/smb.conf
```

Add the below lines at the end of the file.

```
[sambafile]
comment = My Data
path = /sambafile
public = no
writable = yes
printable = no
valid users = diksha
host allow = 192.168.58.128
```

Step 4:

Restart the smb service using the service command with restart option and to start the service on reboot use the chkconfig command with “on” option.

```
[root@ldapl Desktop]# service smb restart
Shutting down SMB services: [ OK ]
Starting SMB services: [ OK ]
[root@ldapl Desktop]# chkconfig smb on
```

Step 5:

Now setup the samba password for root using the smbpasswd command with -a option.

```
[root@ldapl Desktop]# smbpasswd -a root
New SMB password:
Retype new SMB password:
Added user root.
```

Step 6:

Now to look at what services are available on a server smbclient command with -L option to show login prompt.

```
[root@ldapl Desktop]# smbclient -L //localhost
Enter root's password:
Domain=[MYGROUP] OS=[Unix] Server=[Samba 3.5.6-86.el6]

    Sharename      Type            Comment
    -----
    sambafile      Disk           My Data
    IPC$           IPC           IPC Service (Samba Server Version 3.5.6-86.el6)
    root           Disk           Home Directories
Domain=[MYGROUP] OS=[Unix] Server=[Samba 3.5.6-86.el6]
```

Step 7:

Now create a new directory to share files in root directory with client using the mkdir command and also give user, group and other the permission to read, write and execute in the directory using the chmod command with 777 option and also list the security context of the file using the ls command with -ldZ option.

```
[root@ldapl Desktop]# mkdir /share
[root@ldapl Desktop]# chmod 777 /share
[root@ldapl Desktop]# ls -ldZ /share/
drwxrwxrwx. root root unconfined_u:object_r:default_t:s0 /share/
```

Step 8:

Now setting the type part of the security context to samba_share_t for the /share file using the chcon command with -t option.

```
[root@ldapl Desktop]# chcon -t samba_share_t /share
```

Step 9:

Now to the users accounts stored in the SAM database (Database of Samba Users) use pdbedit command with -L(to show all users) option.

```
[root@ldapl Desktop]# pdbedit -L
diksha:501:
root:0:root
```

Step 10:

Now open the samba configuration file i.e. smb.conf located in “/etc/samba/” directory and add the following lines at the bottom of the file.

```
[smbafale]
comment = My Data
path = /smbafale
public = no
writable = yes
printable = no
valid users = diksha,root
host allow = 192.168.58.128
```

Also change the work group name according to work group present on the client side.

```
# Hosts Allow/Hosts Deny lets you restrict who can connect, and you can
# specify it as a per share option as well
#
    workgroup = WORKGROUP
    server string = Samba Server Version %v
```

Step 11:

Now restart the smb and nmb service using the service command with “restart” option.

```
[root@ldapl Desktop]# service smb restart
Shutting down SMB services: [ OK ]
Starting SMB services: [ OK ]
```

```
[root@ldapl Desktop]# service nmb restart
Shutting down NMB services: [ OK ]
Starting NMB services: [ OK ]
```

To boot the services during reboot, use the chkconfig command with “on” option.

```
[root@ldapl Desktop]# chkconfig smb on
[root@ldapl Desktop]# chkconfig nmb on
```

Stop the firewall by stopping iptables service to do so use the service command with “stop” option.

```
[root@ldapl Desktop]# service iptables stop
iptables: Flushing firewall rules: [ OK ]
iptables: Setting chains to policy ACCEPT: filter [ OK ]
iptables: Unloading modules: [ OK ]
```

Step 12:

Now copy the IP address of the server machine, to check IP address use ifconfig command.

```
[root@ldapl Desktop]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0C:29:F3:1B:CE
          inet addr:192.168.58.128  Bcast:192.168.58.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fef3:1bce/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:379 errors:0 dropped:0 overruns:0 frame:0
          TX packets:136 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:50687 (49.4 KiB)  TX bytes:19773 (19.3 KiB)
          Interrupt:19 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:742 errors:0 dropped:0 overruns:0 frame:0
          TX packets:742 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:56791 (55.4 KiB)  TX bytes:56791 (55.4 KiB)
```

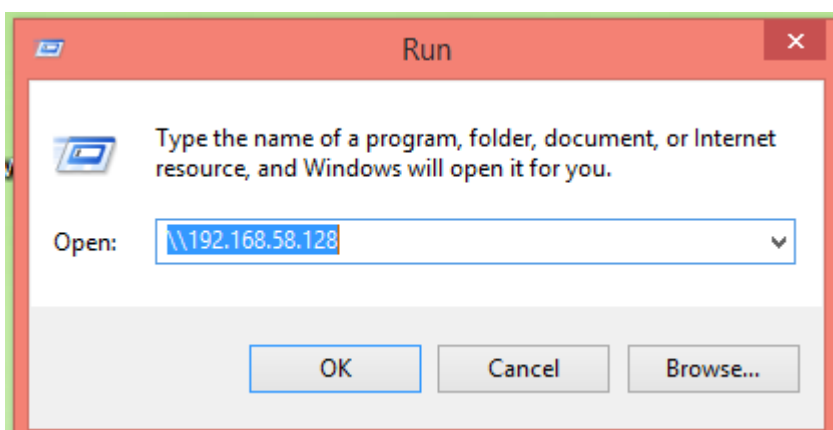
Now verify that smb service is on use service command with “status” option.

```
[root@ldapl Desktop]# service smb status
smbd (pid 2477) is running...
```

Client (Windows 10) Side:

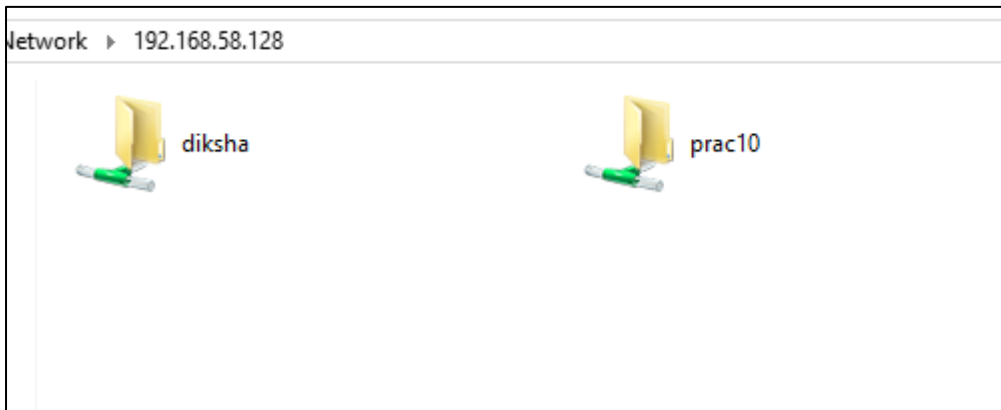
Step 1:

Windows start search for “run” click the first option, now in “Open:” enter the server IP address in manner as below.

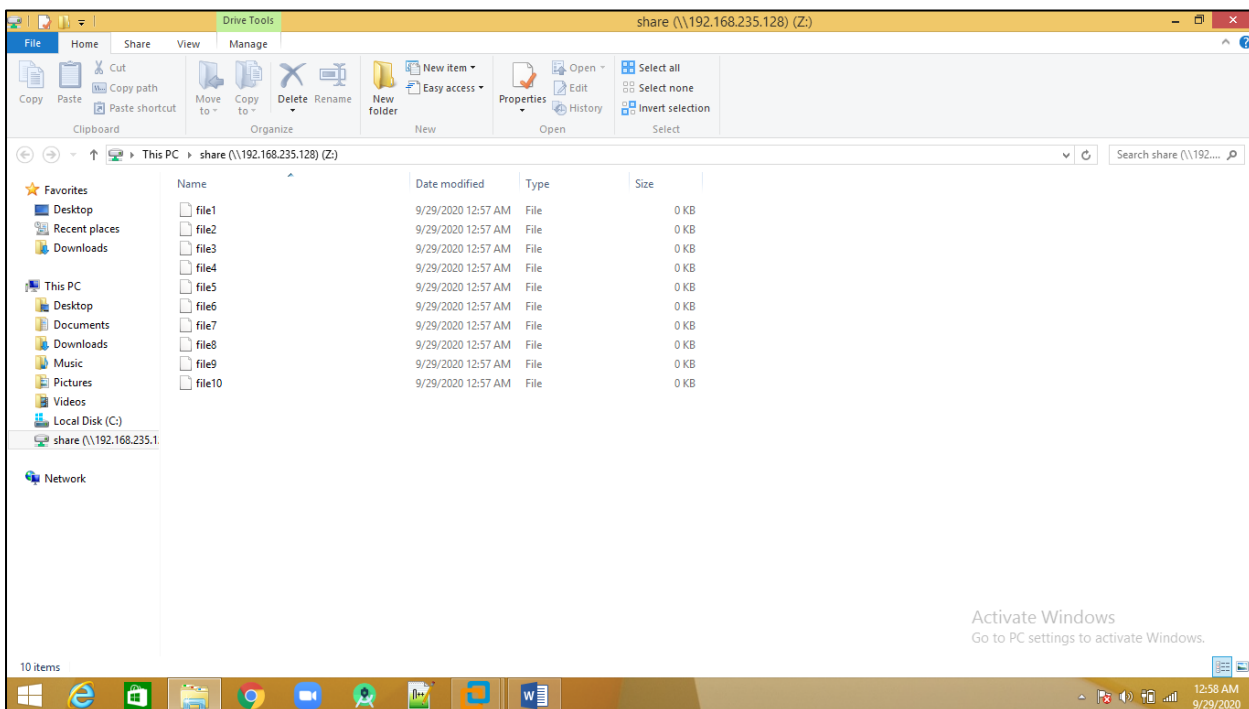


2:

It will take you to file explorer with files from server one is shared file and other is root.



Now create a new file named “Test” to verify the connection.



Server (RHEL 6) Side:

Now to verify list the files in shared directory using the ls command.

```
[root@localhost Desktop]# ls /share
Test
```