IT601P Finalterm Past Papers

[BY Mah jabeen]

[mahjabeen9786@gmail.com]

[7/22, 1:09 PM]

- MCQs Were conceptual and commands base the
- Or
- Long mai
- LDAP configuration mai users home directory use kese kren ge. Isk steps are likhne the
- |, >,>> use for
- 2 questions the lab mai se inki commands files mai jo code sir ne likhe the

[7/22, 1:09 PM]

Dhcp bhht achi trha ready kr len un k bhi long question tha

[7/29, 6:25 PM] IT601p

- 1.Post fix and email say related command pochi thi
- 2. Write A and PTR bind9 configuration with host name "ldap" domain name "vusna.com" and ip address "192.168.19.7"
- 3. Sna@vusna on kld server . Enlist all steps
- 4. Wich file contain and configure on ubtan server . Enlist parameters.
- 5. Aur ak question.hatcess say related tha

[7/30, 3:00 PM] IT601P

- Just app lms wali lab ready kry sary McQ meray usi m Sy thy
- Question
- - 16 labe Sy the 5 comnd likhni thi emails Sy related
- 2- ip addresses given tha pocha hua tha knsa protocol ka he idress
- 3- ek command di hui thi uski output find krni thi
- 4-ek replace ki command likhni thi second m Sy
- 5 - labe 9 m Sy tha server ka neme aur edress given tha bqi implemented server KY name aur edress write krny thy
- Vusna vala aik qs smjh m aya kuch thora bot r aik tha records vala lab se blkl change tha
- Full of commands tha mcqs me b r long me be
- Mne 13 labs prepare ki thi mgr ksm se blkl smj ni a ri ti kia kru kia ni
- Baqi output kuch ajeeb tra se diy oy ty ke iski command kia o g vo b lab me ni ty vcy
- Mcqs pta ni kis jgah ke ty soch soch ke
- Kuch 2 sy 3 file sy hu gy Vo jo file snd ki ti self made notes wali
- R kuch 1 sy 2 lab se
- Bot mushkil tha paper bot ziada

[7/30, 9:43 PM]

- Aik qs tha steps ty vusna ko kerberos me configure ke
- [7/31, 12:01 PM] G directories aur command Sy five McQ meray ay thy
- [7/31, 3:45 PM] | > >> ka use in linux
- Ldap related tha
- Write ouput of ldap@vusna.com with ip address....

[8/1, 1:21 PM] IT601P

- Mcqs are easy and from lab
- ✍question
- Enable ldap step
- Linux
- List the step to configure sshd
- |, >, >> use
- Wite the output
- Ldap@vusna.com with ip address

[8/3, 11:03 AM]

- Sachi Mera tou bahot bura hwa😢
- Question aik tou Linux ki comman thi

- Or aik output btani thi
- CrURL ki command thi
- Aik scenerio tha uski command likhni thi
- Scenario diya hwa tha
- Ldap k enable k step

Installing LDAP

- sudo apt install slapd ldap-utils
- sudo dpkg-reconfigure slapd
- configure the OpenLDAP libraries with certain defaults in /etc/ldap/ldap.conf

```
BASE dc=vusna,dc=com
URI ldap://ldap.vusna.com
```

After installation, you will get two databases, or suffixes:

1. For data, based on your host's domain (dc=vusna,dc=com),
2. For configuration, with its root at cn=config.

To change the data on each we need different credentials and access methods:

dc=example,dc=com

The administrative user for this suffix is cn=admin,dc=example,dc=com and its password is the one selected during the installation of the slapd package.

cn=config

The configuration of slapd itself is stored under this suffix.

Configure TLS certificate and private key

Restart postfix service

```
sudo systemctl restart postfix
```

Confirm active ports

```
sudo ss -lnpt | grep master
```

we required three types of dependencies

1) cs-certificates

2) curl

3) gnupg

There are two types keys

1) public key

2) private key

Inko ham authentication or encryption k Liya use kar sakty

Ca certificates are electronic passwords jinko ham authentication and or encryption k Liya use karty hy

ca certificates k Liya Jo certified authority hoti hy os k public certificates ko manage karny k Liya ca-certificate ka package use hota hy isko different server communication ya identity verification k Kiya b use kar sakty hy

4. install ca-certificate
   - it601p@labserver: ~$ sudo apt install ca-certificate

curl package is a client url Jo k diff web server k sath communication k Liya use hota hy eg file download ya server sy response retrieve karny k liya

5. install curl package
   - it601p@labserver: ~$ sudo apt install curl

next package is pretty good privacy gnu pgp isko ham different files ko encrypt karny k Liya use Kary gy ya fr hmari communication ko sign ya is sy ham apni or remote server ki identity ko verify kar saky gy isko ham ssh k sath b use kar sakty

6. install gpg
   - it601p@labserver: ~$ sudo apt install gnupg

These are some of our initial requirements before installing docker

Now we have to install the keys of docker in our system

0755 mean world/ globally readable

7. To change the permission of directory
   - it601p@labserver: ~$ sudo install-m 0755 -Id/etc/apt/keyrings

Installation and Configuration

- **Step 1: Set hostname for the Ubuntu server**
- **Step 2: Install OpenLDAP Server**

```
sudo apt update
sudo apt -y install slapd ldap-utils
```

Verify installation with command slapcat to output SLAPD database contents.

- **Step 3: Add base dn for Users and Groups**
  - Add a base DN for users and groups. Create a file named basedn.ldif and the file by running t
    ldapadd
- **Step 4: Add User Accounts and Groups**
  - Generate a password for the user account to add.
  - Create ldif file for adding users.
  - When done with edit, add account by running ldapadd command.
  - Create ldif file for groups and add groups with command ldapadd

Install LDAP Account Manager

- **Step 5: Install LDAP Account Manager**
  - Install Apache Web server & PHP

```
sudo apt -y install apache2 php php-cgi libapache2-mod-php php-mbstring php-common php
```

  - Then enable php-cgi PHP extension.

```
sudo a2enconf php*-cgi
sudo systemctl reload apache2
```


  - Step 3: Install LDAP Account Manager

```
sudo apt -y install ldap-account-manager
```

```
sudo vim /etc/apache2/conf-enabled/ldap-account-manager.conf
```

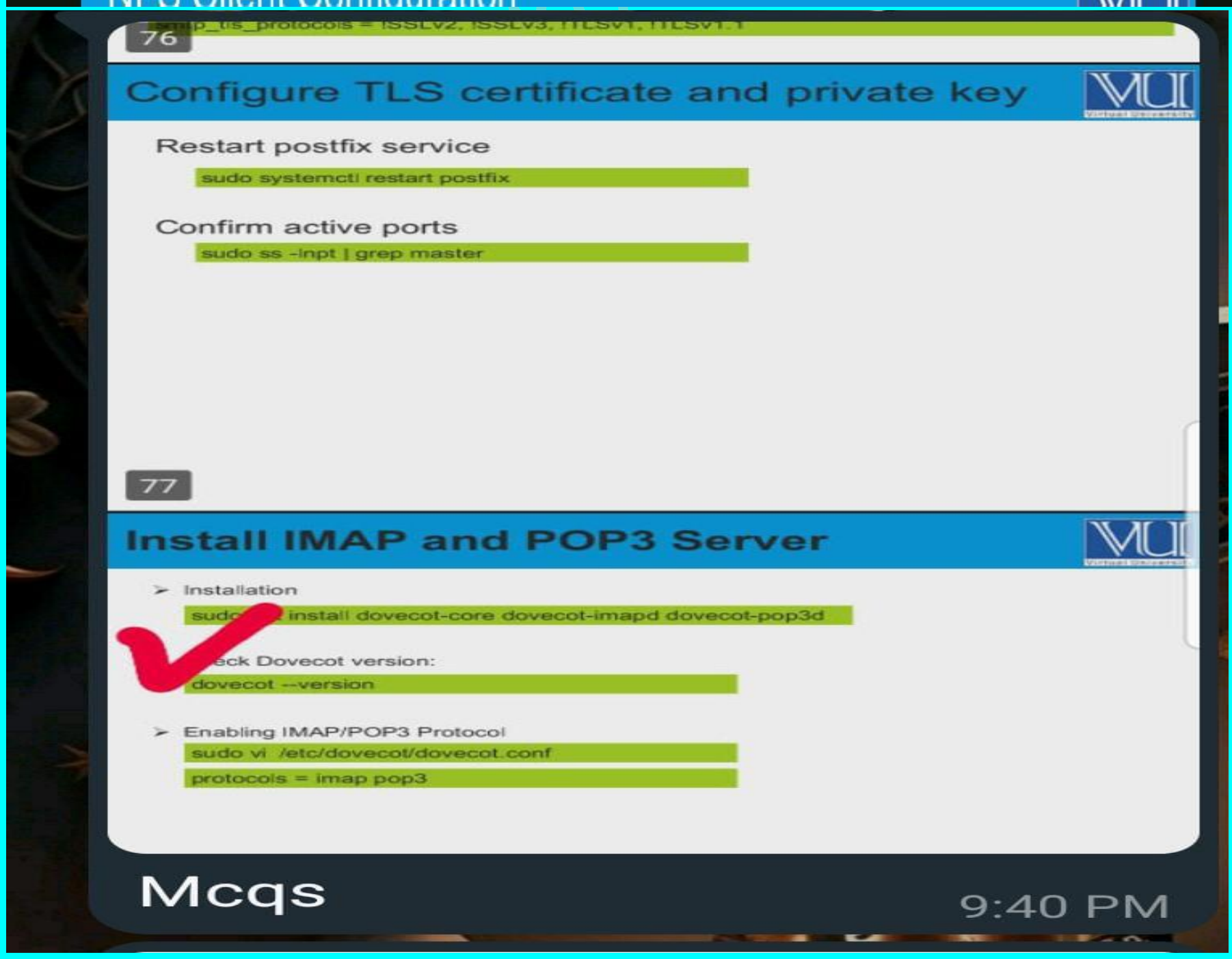
  - comment the line Require all granted and add subnet(s) allowed to access LDAP Account Mana
    administration interface.

Installing NFS on Linux Server

- **Install the nfs-kernel-server software package.**
  - sudo apt install nfs-kernel-server
- **Start the NFS server**
  - sudo systemctl start nfs-kernel-server
- **Configure the directories to be exported by adding them to the /etc/exports file.**
  - /srv *(ro,sync,subtree_check)
  - /cshare *.vusna.com(rw,sync,no_subtree_check)
  - /scratch *(rw,async,no_subtree_check,no_root_squash)
- **Apply the new config via**
  - sudo exportfs -a
- **Check status or restart service**
  - sudo systemctl restart nfs-kernel-server

- **rw**
  - It gives the client computer both read and write access to the volume.
- **sync**
  - It forces NFS to write changes to disk before replying. This results in a more stable and consistent environment but reduces the speed of file operations.
- **no_subtree_check**
  - It prevents subtree checking, which is a process where the host must check whether the file is actually still available in the exported tree for every request. This can cause many problems when a file is renamed while the client has it opened. In almost all cases, it is better to disable subtree checking.
- **no_root_squash**
  - By default, NFS translates requests from a root user remotely into a non-privileged user on the server. This was intended as security feature to prevent a root account on the client from using the file system of the host as root. no_root_squash disables this behavior for certain shares.

NFS Client Configuration



```
sudo ldapsearch -Q -LLL -Y EXTERNAL -H ldapi:/// -b cn=config d
```

Q : SASL Quite mode
LLL : Print responses in LDIF format without comments
-Y : SASL mechanism
-H : LDAP URL
-b : basedn

```
dn: cn=config
dn: cn=module{0},cn=config
dn: cn=schema,cn=config
dn: cn={0}core,cn=schema,cn=config
dn: cn={1}cosine,cn=schema,cn=config
dn: cn={2}nis,cn=schema,cn=config
dn: cn={3}inetorgperson,cn=schema,cn=config
dn: olcDatabase={-1}frontend,cn=config
dn: olcDatabase={0}config,cn=config
dn: olcDatabase={1}mdb,cn=config
```

View Information

This is what the dc=vusna,dc=com DIT looks like:

```
$ ldapsearch -x -LLL -H ldap:/// -b
dc=vusna,dc=com dn

dn: dc=vusna,dc=com

dn: cn=admin,dc=vusna,dc=com
```

verify the authentication with ldapwhoami

Blkl asa ni tha milta julta much tha qs

9:41 PM

Authenticating as principal root/admin@vusna.com with password.

kadmin.local: addprinc sna/admin

WARNING: no policy specified for sna/admin@vusna.com; defaulting to no policy

Enter password for principal " sna/admin@vusna.com ":

Re-enter password for principal " sna/admin@vusna.com ":

Principal "sna/admin@vusna.com" created.

kadmin.local: quit

Create Admi Principal

6 - Set admin p cipal with appropriate Access Control List (ACL) permissions. The are nfigur n the /etc/krb5kdc/kadm5.acl file:

```
sna/admin@vusna.com      *
```

The a e will grant all privileges to any admin instance of a principal. See the kadm deta

- R art the krb5-admin-server for the new ACL to take affect:

```
sudo systemctl restart krb5-admin-server.service
```

8 - The new user principal can be tested using the kinit utility:

```
$ kinit sna/admin
Password for sna/admin@vusna.com:
```

Verifying and Completing kerberos

9 - Verify information about the Ticket Granting Ticket (TGT):

9:41 PM

```
$ klist
```

KDC Installation

1 - Install the Kerberos Packages

- krb5-kdc
- krb5-admin-server
- sudo apt install krb5-kdc krb5-admin-server

2 - Create the new realm with the kdb5_newrealm utility

- sudo krb5_newrealm

3 - you can reconfigure Kerberos from scratch, if need to change the realm name. Following commands can be used for this purpose.

- sudo dpkg-reconfigure krb5-kdc

KDC Configuration

4 - Create the first principal.

```
sudo kadmin.local
Authenticating as principal root/admin@vusna.com with password.
kadmin.local: addprinc sna
WARNING: no policy specified for sna@vusna.com; defaulting to no policy
Enter password for principal "sna@vusna.com":
Re-enter password for principal " sna@vusna.com ":
Principal "sna@ vusna.com" created.
kadmin.local: quit
```

Create Admin Principal

5 - Create an admin principal and assign permissions.

```
$ sudo kadmin.local
Authenticating as principal root/admin@vusna.com with password.
kadmin.local: addprinc sna/admin
WARNING: no policy specified for sna/admin@vusna.com; defaulting to no policy
Enter password for principal " sna/admin@vusna.com ":
Re-enter password for principal " sna/admin@vusna.com ":
Principal "sna/admin@vusna.com" created.
kadmin.local: quit
```

Create Admin Principal

6 - Set admin principal with appropriate Access Control List (ACL) permissions. The permissions are configured in the /etc/krb5kdc/kadm5.acl file:

```
sna/admin@vusna.com      *
```

The above will grant all privileges to any admin instance of a principal. See the kadm5.acl manpage for details.

7 - Restart the krb5-admin-server for the new ACL to take affect:

```
sudo systemctl restart krb5-admin-server.service
```

8 - The new user principal can be tested using the kinit utility:

```
$ kinit sna/admin
Password for sna/admin@vusna.com:
```

named-checkconf /etc/bind/named.conf.options

1) mkdir /etc/bind/zones
2) cp /etc/bind/db.local /etc/bind/zones/vusna.com
3) vi /etc/bind/zones/vusna.com

VU A.COM Zone Configuration

```
$TTL    604800
@       IN      SOA     vusna.com. root.vusna.com. (3; 604800; 86400; 2419200
@       IN      NS      bindserver.vusna.com.
bindserver      IN      A       192.168.2.2
web     IN      A       192.168.2.3
dhcp    IN      A       192.168.2.4
db      IN      A       192.168.2.5
elk     IN      A       192.168.2.6
```

Qs ty change vle ke sth

9:42 PM

...reate a di... ...e dockerfile will be stored.

- mkdir c1 && cd c1

➢ Create dockerfile

... vi dockerfile

```
####Contents of Dockerfile####

FROM ubuntu:22.04

MAINTAINER Virtual_University_of_Pakistan

ENV DEBIAN_FRONTEND noninteractive

RUN apt-get update && apt-get install -y ubuntu-server

EXPOSE 2022
#expose the port 2022

CMD ["echo", "Welcome to it601p"]
#above command will print the message and container will exit.

#To keep it running  run bash and keep it running

CMD ["PROGRAM_TO_RUN"]
```

Build a Docker Image with Dockerfile

➢ **Build the image using above docker file**

- docker build <options>
- docker build --help

➢ **Some Options**

--file, -f : docker file name

--label : set meta data for image

--quite -q : suppress output

--rm : Remove intermediate containers

--tag, -t : set name or tag

--add-host : Add custom host to IP mapping

Option
--build-arg
--cache-fro...
--cgroup-pa...
--compress
--cpu-perio...
--cpu-quota
--cpu-shares
--cpuset-cpu...
--cpuset-me...
--disable-co...
--iidfile
--isolation
--memory , -...
--memory-sw...
--no-cache

Mcqs

9:42 PM

- `ldapsearch`: The command to query the LDAP server.
- `-x`: Use simple authentication.
- `-H ldap://192.168.1.1`: Specifies the LDAP server IP address.
- `-D "cn=admin,dc=vusna,dc=com"`: The bind DN (Distinguished Name) to use for the query.
- `-w password`: The password for the bind DN.
- `-b "dc=vusna,dc=com"`: The search base (the point in the directory tree at which the search is started).
- `"(mail=ldap@vusna.com)"`: The search filter (in this case, searching for an entry with the email `ldap@vusna.com`).
- `> ldap_output.txt`: Redirects the output to a file named `ldap_output.txt`.

Regarding LDAP (Lightweight Directory Access Protocol) and your request to write the output of `ldap@vusna.com` with an IP address, here's an example of how you might do this in a Linux environment. This assumes you want to query an LDAP server and save the results to a file:

Sh Copy code

```
ldapsearch -x -H
ldap://192.168.1.1 -D
"cn=admin,dc=vusna,dc=com" -w
password -b "dc=vusna,dc=com"
"(mail=ldap@vusna.com)" >
ldap_output.txt
```

Here's what each part of the command does:

1. Pipe (|)

- **Function:** The pipe symbol is used to take the output of one command and pass it as input to another command. This is useful for chaining multiple commands together, allowing for complex data processing in a single line.
- **Example:**

Bash Copy code

```
cat file.txt | grep "search_term"
```

  - Here, `cat file.txt` outputs the content of `file.txt`.
  - The pipe `|` sends that content to `grep`, which then searches for "search_term" within the text.

2. *Redirect Output (>)*

- **Function:** The > symbol redirects the standard output (stdout) of a command to a file. If the file already exists, it will be overwritten with the new data. This is commonly used to save command results to a file.
- **Example:**

Bash Copy code

```
ls -l > directory_list.txt
```

  - This command lists the contents of the current directory in long format (`ls -l`) and saves the output to `directory_list.txt`.
  - If `directory_list.txt` exists, it will be replaced with the new content.

3. Append Output (>>)

- **Function:** The >> symbol also redirects the output of a command to a file, but instead of overwriting the file, it appends the new data to the end of the file. This is useful for adding data to an existing file without losing the previous content.
- **Example:**

```bash
echo "New line" >> notes.txt
```

  - This command adds the text "New line" to the end of `notes.txt`.
  - If `notes.txt` doesn't exist, it will be created; if it does exist, the text will be added after the existing content.

Key Points:

- **Pipes** (|) are used for chaining commands together, allowing for complex operations.
- **Redirection** (>) is used for sending output to a file, replacing any existing content.
- **Appending** (>>) is used to add output to a file without overwriting it.

These symbols are essential for effective command-line usage, enabling powerful manipulation of data and files in Linux.

Best of luck