

### III. APPLICATION LAYER PROTOKOL PRACTICUM MODULE

#### COMPETENCE:

- ❖ Students are able to use multiple application layer protocols in the network

#### TOOLS AND MATERIALS:

- Software Simulator GNS3
- Stable Internet Connection
- OpenVPN app and profile

#### THEORY REVIEW:

##### I. DNS

In data networks, devices are labeled with numeric IP addresses, so they can participate in transmitting and receiving data in a computer network. After all, most people have difficulty remembering those numeric addresses. Thus, domain names are created to represent numeric addresses so that they become simple and easily recognizable names.

On the internet, these domain names, such as [www.cisco.com](http://www.cisco.com), are easier for people to remember than the number 198.132.219.25, which is the actual numeric address of cisco servers. Later, if Cisco decides to change the numeric address, the change will not be known by the user, because the domain name remains [www.cisco.com](http://www.cisco.com). This new address will be linked to an existing domain name and connectivity will be maintained. If the computer network is small, managing the mapping between the domain name and the real IP address is quite easy. However, as computer networks grow and the number of connected network devices increases, manual systems for mapping is not adequate.

DNS was created to translate domain names into the numeric IP addresses. DNS uses a number of distributed servers to do that. The DNS protocol defines an automated service that matches the name of a network resource (*domainname*) with the network's numeric address. It includes formats for queries, responses, and data formats. DNS protocol communication uses a

single format called message. This message format is used for all types of client queries and server responses, error messages, and the transfer of information between other DNS servers.

There are tools that operating systems provide to manually request information from DNS services. The tools are *dig* and *nslookup* command. We can use those tools to request IP information and hostnames from DNS. Both tools are covered in a software package in Linux system that is called **dnsutils**. So if those command / tools wont run, it may be the **dnsutils** package has not installed yet on that computer. Both dig and nslookup commands are executed from *the command prompt or console of the operating system*. Followed by a paramater that we want to know the DNS information from. Examples of execution and output from dig and nslookup commands can be seen in the image below.

```
root@debian-0731140009:/home/sofyan# dig google.com

; <<>> DiG 9.10.3-P4-Debian <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 64092
;; flags: qr rd ra; QUERY: 1, ANSWER: 6, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:: MBZ: 0005 , udp: 4096
;; QUESTION SECTION:
;google.com.                IN      A

;; ANSWER SECTION:
google.com.                 5       IN      A       74.125.200.113
google.com.                 5       IN      A       74.125.200.101
google.com.                 5       IN      A       74.125.200.102
google.com.                 5       IN      A       74.125.200.138
google.com.                 5       IN      A       74.125.200.139
google.com.                 5       IN      A       74.125.200.100

;; Query time: 311 msec
;; SERVER: 172.16.215.2#53(172.16.215.2)
;; WHEN: Wed Oct 11 05:16:06 WIB 2017
;; MSG SIZE rcvd: 135

root@debian-0731140009:/home/sofyan#
```

Figure 4.1. Command and output command dig.

In Figure 4.1 we can see the result of the excecution of the dig command followed by the paramater “polinema.ac.id”. The output of the dig command has the following parts:

- Header: displays the version of the dig command, the global options that the dig command uses, and other additional header information.

- Question section: displays the questions we ask to DNS service, i.e. the inputs/parameters we add in this case are **polinema.ac.id**. By default it is interpreted by the dig command, that we search for an A record information or alias address of the domain address polinema.ac.id.
- Answer section: displays the answers from the DNS server.
- Authority section: displays a DNS server that has the authority to answer our questions. Authority DNS is usually the DNS server of the domain we are looking for. We can see on the result, that the name of the DNS server of Polinema is **ns1.polinema.ac.id**
- Additional Section: shows the IP address of the name server - name server in the Authority Section. In this example, there is only one name server (DNS Server) in the domain of polinema.ac.id
- The final section shows additional information such as the DNS server that answers our query, the time it takes to answer and others.

Next in figure 4.2 is an example of a simple nslookup command with no other parameters than the domain name/host.

```
root@debian-0731140009:/home/sofyan# nslookup google.com
Server:          172.16.215.2
Address:         172.16.215.2#53

Non-authoritative answer:
Name:   google.com
Address: 74.125.200.139
Name:   google.com
Address: 74.125.200.100
Name:   google.com
Address: 74.125.200.113
Name:   google.com
Address: 74.125.200.101
Name:   google.com
Address: 74.125.200.102
Name:   google.com
Address: 74.125.200.138

root@debian-0731140009:/home/sofyan# _
```

Figure 4.2. Command and output nslookup command.


We can add additional parameters to the dig and nslookup commands. On the dig and nslookup commands we can add what *record* parameters we want to know from the domain / host that we are looking for information on. Example:

- dig polinema.ac.id MX --> displays *the mail server* on the polinema.ac.id domain
- dig polinema.ac.id NS --> displays *nameserver* information on polinema.ac.id domain
- nslookup -type=NS polinema.ac.id --> display *nameserver* information on *the domain* polinema.ac.id
- nslookup -type=MX polinema.ac.id --> displays *mail servers* on polinema.ac.id domain

## II. SSH

Secure Shell (abbreviated as SSH ) is one of the application layer protocols that provides terminals to control one computer from other computer connected to the network. SSH replaces the telnet protocol, because telnet has no security function in data communication. SSH provides cryptographic functions on every data transmitted over the network. By using telnet and SSH, we can execute every command as if we were using directly the PC / computer / device that we control. Telnet and SSH require the *username* and *password* of the PC / Computer / device that we want to control.

```
root@debian-0731140009:/home/sofyan# ssh engine-home.ddns.net -l coba
The authenticity of host 'engine-home.ddns.net (36.79.244.237)' can't be established.
ECDSA key fingerprint is SHA256:tkuZaPb2mq5N+L/e+4jeEsUrLjBKA9sETShuFclobaU.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'engine-home.ddns.net,36.79.244.237' (ECDSA) to the list of known hosts.
coba@engine-home.ddns.net's password:
```



```
Welcome to ARMBIAN 5.31 stable Debian GNU/Linux 8 (jessie) 3.10.105-s500
System load:   0.02 0.03 0.05    Up time:         26 days
Memory usage:  17 % of 2004MB    IP:            192.168.1.2
CPU temp:      51°C
Usage of /:    54% of 29G

[ 0 security updates available, 87 updates total: apt upgrade ]
Last check: 2017-10-11 00:00

[ General system configuration: armbian-config ]

coba@roseapple:~$ _
```

Figure 4.3. Use of ssh command with additional parameters -l and *username*

We can run the ssh command from *the command prompt* / shell of the operating system as well as using applications such as **putty**. The standard parameter that we need is the IP address or hostname of the device that we are going to control. An example of a ssh execution is in figure 4.3.

### III. FTP

File Transfer Protocol (FTP) is an application layer protocol used to control and transfer files between computers connected to a network. FTP was one of the earliest Internet protocols developed, and is still used today to download and upload computer files between ftp clients and FTP servers. An FTP client is an application that can issue FTP commands to an FTP server, while an FTP server is a Windows Service or *daemon* that runs on a computer that responds to commands from an FTP client. FTP commands can be used to change directories, change the mode of delivery between binary and ASCII, upload computer files to ftp servers, and download files from FTP servers.

FTP only uses the standard authentication method, which uses *usernames* and *passwords* sent in an unencrypted form. Registered users can use their *username* and *password* to access, download, and upload the files they want. Generally, registered users have full access to multiple directories, so they can create files, create directories, and even delete files.

To use FTP from the linux terminal, the **ftp** package must first be installed in the computer to be used. In addition, the use of FTP from command prompt / shell to FTP server requires a minimum host name or IP address from the FTP server. Example: ftp 192.168.1.155. Then we enter the *username* and *password*. Once connected to the service from the FTP server, we can give ftp commands.

```
root@debian-0731140009:/home/sofyan# ftp engine-home.ddns.net
Connected to engine-home.ddns.net.
220 Welcome to Engine-Home FTP service.
Name (engine-home.ddns.net:sofyan): coba
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Figure 4.4. FTP service call to FTP server with address 192.168.1.155

Common shell commands/command prompts can be used on FTP, such as deleting files, renaming files, viewing active directories, changing directories, creating directories, etc. At the FTP prompt, you can also give executed commands to your local computer, by adding an exclamation point ("!") at the beginning of the command you want to execute.

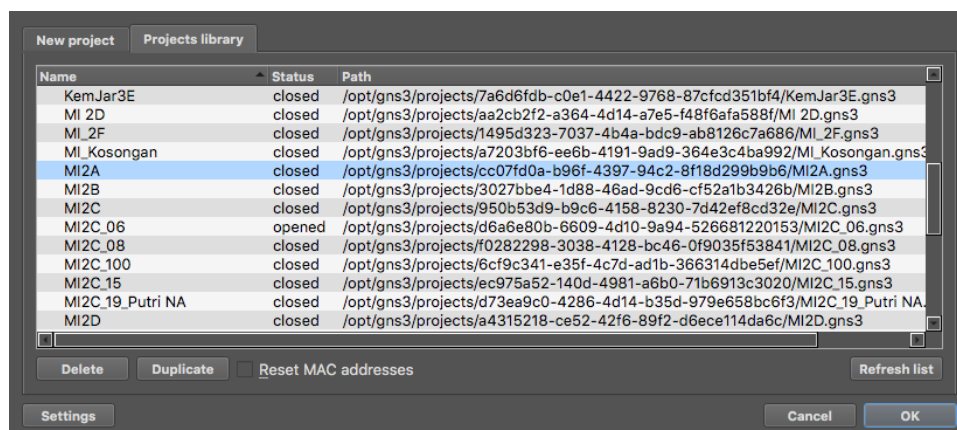
```
ftp> pwd
257 "/home/coba"
ftp> !pwd
/home/sofyan
ftp> _
```

Figure 4.5. Commands executed on FTP servers and FTP clients.

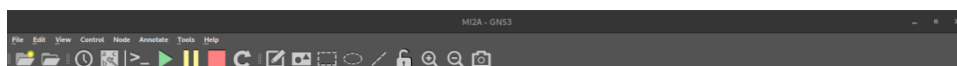
In figure 4.5, it can be seen that the FTP user change the active directory on the FTP server into an *ti2a* directory, then displays the FTP server's active directory ("/home/jarkom/ti2a"). Then it is followed by displaying the active directory that is on the FTP client ("/home/user9"). The command to *upload* a file on FTP is the **put** command, while to *download* the file on FTP is the **get** command. Each is followed by the file name parameter to be transferred. The transfer will be made to the active directory (client or server).

## PRACTICUM PREPARATION

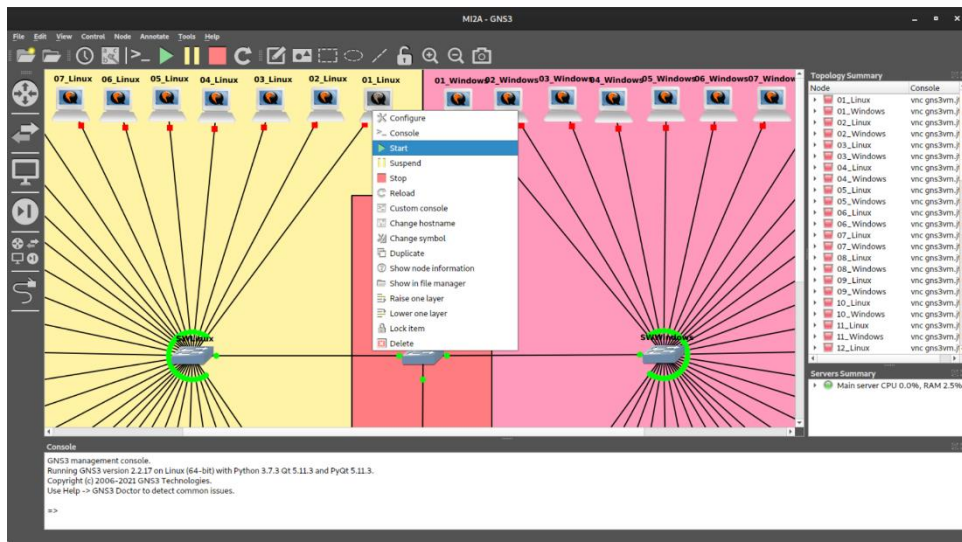
1. Connect your computer to the internet network.
2. Connect your computer to the Information Technology Department's VPN server using the OpenVPN Connect app. Use the profile, username and password you've obtained at previous meetings.
3. Once connected to an OpenVPN server, open the GNS3 app on your computer.
4. In the initial view of the GNS3 application window, select the Project library tab. Then select the project that has been set up for your class (e.g. Jarkom\_TI2I). Then **remove the check mark on the Reset MAC Address option**. Then press the OK button.



5. After the project opens in the main window of the GNS3 application, you can adjust the zoom on the project display as you see fit by pressing the positive magnifying glass button (to enlarge) or the negatif magnifying glass button (to minimize) on the toolbar at the top of the window.



6. Then you can turn on the computer you are going to use. Right-click on the logo of the computer you want to use, then select the Start option.

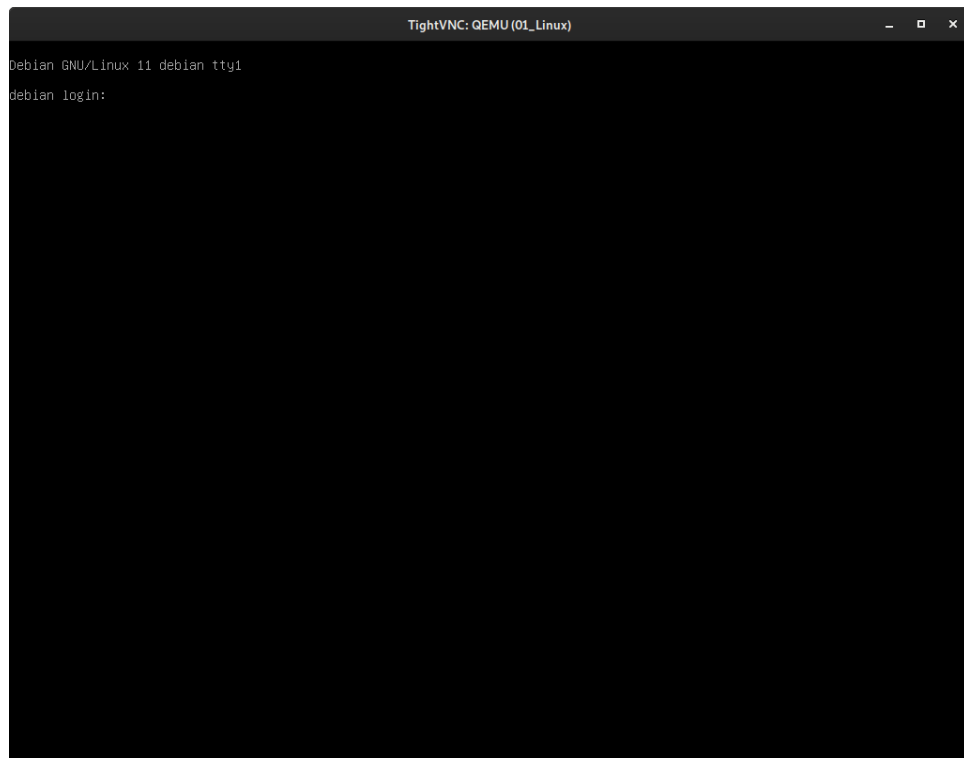


7. Wait a while and you can check the status of your computer on or off in the Topology Summary sidebar to the right of the window.

Node	Console
01_Linux	vnc gns3vm.j
01_Window	vnc gns3vm.j
02_Linux	vnc gns3vm.j
02_Window	vnc gns3vm.j
03_Linux	vnc gns3vm.j
03_Window	vnc gns3vm.j
04_Linux	vnc gns3vm.j

8. Once your computer is on, access your computer by double-clicking (2x) on your computer logo. Then a new window will appear, which is the appearance of your computer like the picture below.





9. You can use the computer for practicum according to the next steps.

## PRACTICUM STEPS

### I. dig dan nslookup

1. Access your Linux computer in your class GNS project.
2. Make sure your computer (in your GNS Project) connection is connected to the internet, by running ping commands to [www.google.com](http://www.google.com). Make sure there are replay words on the output of the command. Stop the ping utility by pressing the keyboard key combination ctrl+c.

```
debian@debian:~$ ping google.com
PING google.com (216.239.38.120) 56(84) bytes of data.
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=1 ttl=113 time=31.4 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=2 ttl=113 time=28.6 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=3 ttl=113 time=28.5 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=4 ttl=113 time=28.5 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=5 ttl=113 time=28.8 ms
^C
--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 28.468/29.157/31.357/1.106 ms
```

3. If not, ask the lecturer / instructor in order to get an internet connection.
4. If your computer has internet connection, type in the *console* the command : **dig polinema.ac.id**. Observe and take a *screenshot* of the results.

```
debian@debian:~$ dig polinema.ac.id

; <<> DiG 9.16.15-Debian <<> polinema.ac.id
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 5424
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;polinema.ac.id.                IN      A

;; ANSWER SECTION:
polinema.ac.id.                7200    IN      A      172.16.250.125

;; Query time: 4 msec
;; SERVER: 10.10.10.1#53(10.10.10.1)
;; WHEN: Mon Sep 13 10:22:10 WIB 2021
;; MSG SIZE rcvd: 48
```

5. Type in *the console* the command : **nslookup polinema.ac.id** . Observe and take a screenshot of the results.

```
debian@debian:~$ nslookup polinema.ac.id
Server:          10.10.10.1
Address:         10.10.10.1#53

Non-authoritative answer:
Name:   polinema.ac.id
Address: 172.16.250.125
```

6. Search the IP address and *name server* (DNS server) of the google.com domain and polinema.ac.id by using the dig and nslookup commands.

a. To use the dig command, type it into *the console*: **dig google.com NS**

```
debian@debian:~$ dig google.com NS

; <<> DiG 9.16.15-Debian <<> google.com NS
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 17545
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 4, ADDITIONAL: 0

;; QUESTION SECTION:
;google.com.                IN      NS

;; ANSWER SECTION:
google.com.                 7193    IN      NS      ns3.google.com.
google.com.                 7193    IN      NS      ns2.google.com.
google.com.                 7193    IN      NS      ns1.google.com.
google.com.                 7193    IN      NS      ns4.google.com.

;; AUTHORITY SECTION:
google.com.                 7193    IN      NS      ns2.google.com.
google.com.                 7193    IN      NS      ns1.google.com.
google.com.                 7193    IN      NS      ns4.google.com.
google.com.                 7193    IN      NS      ns3.google.com.

;; Query time: 4 msec
;; SERVER: 10.10.10.1#53(10.10.10.1)
;; WHEN: Mon Sep 13 10:25:42 WIB 2021
;; MSG SIZE  rcvd: 156
```

b. To use the nslookup command, type it in *the console*: **nslookup -type=NS**

**google.com**

```
debian@debian:~$ nslookup -type=NS google.com
Server:          10.10.10.1
Address:         10.10.10.1#53

Non-authoritative answer:
google.com       nameserver = ns1.google.com.
google.com       nameserver = ns4.google.com.
google.com       nameserver = ns3.google.com.
google.com       nameserver = ns2.google.com.

Authoritative answers can be found from:
google.com       nameserver = ns4.google.com.
google.com       nameserver = ns3.google.com.
google.com       nameserver = ns2.google.com.
google.com       nameserver = ns1.google.com.
```

- c. Do the same for polinema.ac.id domain. Take a *screenshot* of each output and analyze it.
7. Look for the IP address and mail server name of the google.com domain and polinema.ac.id by using the dig and nslookup commands.

a. To use the dig command, type it into *the console*: **dig google.com MX**

```

debian@debian:~$ dig google.com MX

; <<> DiG 9.16.15-Debian <<> google.com MX
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 39040
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 4, ADDITIONAL: 0

;; QUESTION SECTION:
;google.com.                IN      MX

;; ANSWER SECTION:
google.com.                570     IN      MX      40 alt3.aspmx.l.google.com.
google.com.                570     IN      MX      50 alt4.aspmx.l.google.com.
google.com.                570     IN      MX      10 aspmx.l.google.com.
google.com.                570     IN      MX      20 alt1.aspmx.l.google.com.
google.com.                570     IN      MX      30 alt2.aspmx.l.google.com.

;; AUTHORITY SECTION:
google.com.                6361    IN      NS      ns3.google.com.
google.com.                6361    IN      NS      ns2.google.com.
google.com.                6361    IN      NS      ns1.google.com.
google.com.                6361    IN      NS      ns4.google.com.

;; Query time: 40 msec
;; SERVER: 10.10.10.1#53(10.10.10.1)
;; WHEN: Mon Sep 13 10:39:35 WIB 2021
;; MSG SIZE rcvd: 208

```

- b. To use the nslookup command, type in *the console*: nslookup -type=MX  
google.com

```

debian@debian:~$ nslookup -type=MX google.com
Server:          10.10.10.1
Address:         10.10.10.1#53

Non-authoritative answer:
google.com       mail exchanger = 50 alt4.aspmx.l.google.com.
google.com       mail exchanger = 10 aspmx.l.google.com.
google.com       mail exchanger = 20 alt1.aspmx.l.google.com.
google.com       mail exchanger = 30 alt2.aspmx.l.google.com.
google.com       mail exchanger = 40 alt3.aspmx.l.google.com.

Authoritative answers can be found from:
google.com       nameserver = ns2.google.com.
google.com       nameserver = ns1.google.com.
google.com       nameserver = ns4.google.com.
google.com       nameserver = ns3.google.com.

```

- c. Do the same for polinema.ac.id as domain.  
d. Take *a screenshot* of each output and analyze it.

## II. SSH

1. The IP address of the server to be used in this practicum is 10.10.10.5.
2. Use your class as your username and password. For example: Mi2A class then the username and password is mi2a.
3. Perform ssh to the server by typing on the console: ssh IPserver -l username. And then enter the password.

```
debian@debian:~$ ssh 10.10.10.5 -lmi2a
The authenticity of host '10.10.10.5 (10.10.10.5)' can't be established.
ECDSA key fingerprint is SHA256:6z/slmRQyr4cn0EG6XQQC0q85rJxYom8jDCyektfFyA.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.10.5' (ECDSA) to the list of known hosts.
mi2a@10.10.10.5's password:
Linux debian 5.10.0-8-amd64 #1 SMP Debian 5.10.46-4 (2021-08-03) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
mi2a@debian:~$ _
```

4. Once you successfully log into the server, you will be directed to your account home folder (/home/kelas\_anda).
5. Look at the list of folders in your directory now using the **ls -la** command.

```
mi2a@debian:~$ ls -la
total 24
drwxr-xr-x 3 mi2a mi2a 4096 Sep 13 11:19 .
drwxr-xr-x 4 root root 4096 Sep 13 10:15 ..
-rw-r--r-- 1 mi2a mi2a 220 Sep 13 10:15 .bash_logout
-rw-r--r-- 1 mi2a mi2a 3526 Sep 13 10:15 .bashrc
-rw-r--r-- 1 mi2a mi2a 0 Sep 13 11:19 file_to_edit.txt
-rw-r--r-- 1 mi2a mi2a 807 Sep 13 10:15 .profile
drwxr-xr-x 3 mi2a mi2a 4096 Sep 13 11:17 ssh
mi2a@debian:~$ _
```

6. Move into the **ssh** folder using the command: **cd ssh**

```
mi2a@debian:~$ cd ssh
mi2a@debian:~/ssh$
```

7. Create new folder using this format: NIM\_firstname. Use the command: **mkdir folder\_name**.

```
mi2a@debian:~/ssh$ mkdir 0731140009_Sofyan
```

8. Move into your new folder. Use command: **cd folder\_name**

```
mi2a@debian:~/ssh$ cd 0731140009_Sofyan
mi2a@debian:~/ssh/0731140009_Sofyan$ _
```

9. Copy *the* file\_to\_edit.txt file in your account's home folder (/home/kelas\_anda) to the directory you've created in step 7. Use the command:

```
cp full_path_with_file_name full_path_tujuan
```

```
mi2a@debian:~/ssh/0731140009_Sofyan$ cp /home/mi2a/file_to_edit.txt ./
mi2a@debian:~/ssh/0731140009_Sofyan$ _
```

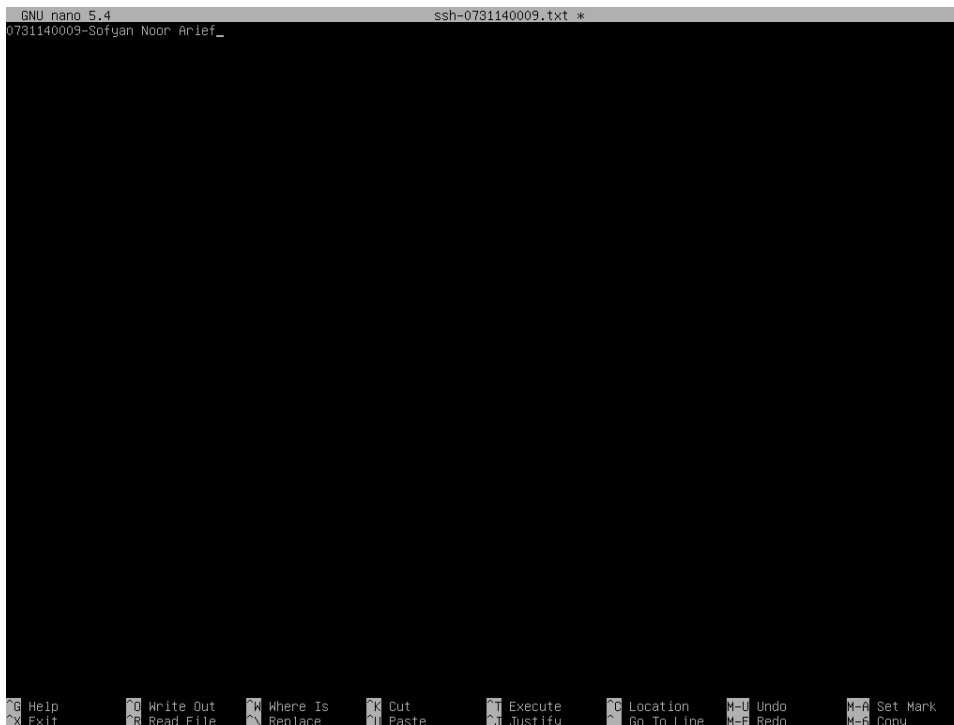
10. Then change the name of the file you have copied (*copy*) to ssh-your\_NIM.txt.

Use the command: mv nama\_file\_lama nama\_file\_baru

```
mi2a@debian:~/ssh/0731140009_Sofyan$ mv file_to_edit.txt ssh-0731140009.txt
mi2a@debian:~/ssh/0731140009_Sofyan$
```

11. Change the contents of the file you copied in step 10. Replace the contents with nim and your full name is separated by the "-" sign. Please use Nano text editor as in the previous practicum.

```
mi2a@debian:~/ssh/0731140009_Sofyan$ nano ssh-0731140009.txt
```



12. Logout from the remote server's ssh by executing the exit command.

```
mi2a@debian:~/ssh/0731140009_Sofyan$ exit
logout
Connection to 10.10.10.5 closed.
debian@debian:~$ _
```

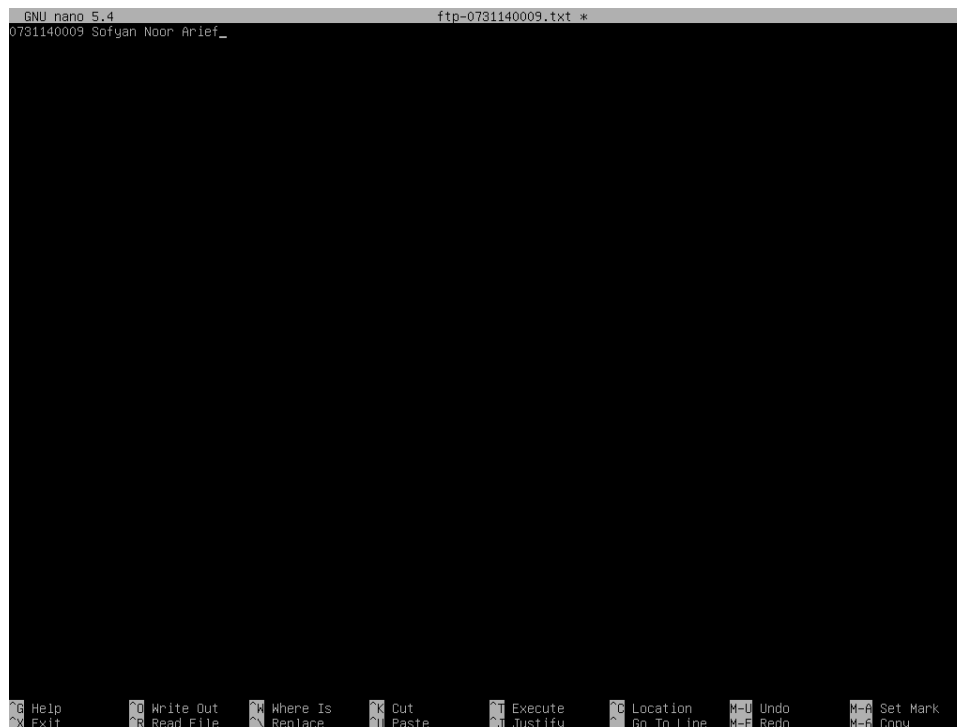
13. Take screenshots of each of the steps you take.

### III. FTP

1. The IP address of the server to be used in this practicum is 10.10.10.5.
2. Use your class as your username and password. For example: for Mi2A class then the username and password is mi2a.
3. Create a text file on your Linux computer with the file name ftp-your\_NIM.txt. You can use nano text editors as in previous practicums.

```
debian@debian:~$ nano ftp-0731140009.txt
```

4. Fill in the file with your full name and your full name.



5. Before connecting FTP to a server computer, make sure your linux computer has an ftp application installed. To check it, you can run the following commands on your Linux computer console.

```
debian@debian:~$ apt search ftp | grep installed
WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

apt-utils/stable,now 2.2.4 amd64 [installed]
doc-debian/stable,now 6.5 all [installed]
libapt-pkg6.0/stable,now 2.2.4 amd64 [installed]
libcurl3-gnutls/stable,now 7.74.0-1.3+b1 amd64 [installed,automatic]
libssh2-1/stable,now 1.9.0-2 amd64 [installed,automatic]
openssh-client/stable,now 1:8.4p1-5 amd64 [installed]
wget/stable,now 1.21-1+b1 amd64 [installed]
debian@debian:~$
```

6. In the image above, it appears that the ftp application has not been installed on the linux computer I am using. This is marked by the absence of ftp packages that appear

on the list of applications that have been installed on the computer. If this also happens in your Linux computer, then you can install ftp application packages using the following commands:

```
debian@debian:~$ sudo apt install ftp
[sudo] password for debian:
```

7. Then enter your password (it's **debian**). Then the installation process will continue and after the installation process is complete, you will return to the console as shown below.

```
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  ftp
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 59.9 kB of archives.
After this operation, 140 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian bullseye/main amd64 ftp amd64 0.17-34.1.1 [59.9 kB]
Fetched 59.9 kB in 0s (347 kB/s)
Selecting previously unselected package ftp.
(Reading database ... 28155 files and directories currently installed.)
Preparing to unpack .../ftp_0.17-34.1.1_amd64.deb ...
Unpacking ftp (0.17-34.1.1) ...
Setting up ftp (0.17-34.1.1) ...
update-alternatives: using /usr/bin/netkit-ftp to provide /usr/bin/ftp (ftp) in auto mode
Processing triggers for man-db (2.9.4-2) ...
debian@debian:~$ _
```

8. And after completing the ftp application installation process, use FTP connection to the server computer. Use the following command's format on *the console*: ftp IPserver

```
debian@debian:~$ ftp 10.10.10.5
Connected to 10.10.10.5.
220 Welcome to Dosen Jaringan FTP service.
Name (10.10.10.5:debian): _
```

9. Then input the *username* and *password* that has been described in step no.2.

```
Name (10.10.10.5:debian): mi2a
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> _
```

10. Send or transfer ftp\_yourNIM.txt you have created previously on your linux computer to the server computer in the **ftp** directory on the ftp server.
11. To do this, you must first check where you are at the moment (what folder you are currently in). This is how to use the **pwd** command.

```
ftp> pwd
257 "/" is the current directory
```



12. Then you can see the contents of the folder you are in now using the **ls** command.

```
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r--  1 1001      1001          0 Sep 13 11:19 file_to_edit.txt
drwx-----  2 1001      1001      4096 Sep 13 12:59 ftp
drwxr-xr-x  3 1001      1001      4096 Sep 13 11:17 ssh
226 Directory send OK.
ftp> _
```

13. To move from the folder where you are now to the ftp folder, use the command **cd folder\_name**.

```
ftp> cd ftp
250 Directory successfully changed.
ftp>
```

14. Upload the file in step 10 by using the **put file\_name** command.

```
ftp> put ftp-0731140009.txt
local: ftp-0731140009.txt remote: ftp-0731140009.txt
200 PORT command successful. Consider using PASV.
150 Ok to send data.
226 Transfer complete.
29 bytes sent in 0.00 secs (13.6090 kB/s)
ftp>
```

15. When you're done, exit the FTP server using the **bye** or **exit** command. Then you will exit the server and return to your Linux computer.

```
ftp> bye
221 Goodbye.
debian@debian:~$ _
```

16. Take screenshots of each steps you take.

## ASSIGNMENT

1. Make a report that contains *screenshot* and *step-by-step* explanations of the three practicum you have done in this module.
2. You can also do the practicum steps part I, II and III using the windows computer (command prompt / cmd) in your GNS class project and document the steps as well.
3. Especially for the part II practicum step that is done using a windows computer, you will need additional applications to do the practicum. The app is called Putty. You can download the app through the <http://repolinux.jti.polinema.ac.id/iso/> page. Then do the installation. Search in the internet, how to use Putty to connect to the same FTP server in the Linux practicum.
4. Submit the reports in pdf by uploading them to the lms server as in previous practices.
5. Do the practicum only during practicum hours. Outside of the paktikum clock, the server may not be accessible. So make the best use of your time.
6. Good luck.