

# SQUID

## Experiment: 5

### Aim:

To create and configure Squid -proxy server

### Description:

#### SQUID – PROXY SERVER

Squid is a full-featured web proxy cache server application which provides proxy and cache services for HyperText Transport Protocol (HTTP), File Transfer Protocol (FTP), and other popular network protocols. Squid can implement caching and proxying of Secure Sockets Layer (SSL) requests and caching of Domain Name Server (DNS) lookups, and perform transparent caching. Squid also supports a wide variety of caching protocols, such as Internet Cache Protocol (ICP), the HyperText Caching Protocol (HTCP), the Cache Array Routing Protocol (CARP), and the Web Cache Coordination Protocol (WCCP).

The Squid proxy cache server is an excellent solution to various proxy and caching server needs, and scales from the branch office to enterprise-level networks while providing extensive, granular access control mechanisms, and monitoring of critical parameters via the Simple Network Management Protocol (SNMP). When selecting a computer system for use as a dedicated Squid caching proxy server for many users ensure it is configured with a large amount of physical memory as Squid maintains an in-memory cache for increased performance.

**Port No:** 3128

**Package name:** squid

**Configuration file:** /etc/squid/squid.conf

### Procedure:

1. At a terminal prompt, enter the following command to install the Squid server:

```
$sudo apt install squid
```

2. Squid is configured by editing the directives contained within the /etc/squid/squid.conf configuration file.
3. Change the access as shown below:

```
acl localnet src 192.168.234.139(your ip address)
```

```
acl blocksite dstdomain &quot;/etc/squid/blocksite&quot;;
```

```
http_access deny blocksite
```

```
http_access allow localnet
```

```
#http_access deny all
```

```
http_access allow all
```

4. To block access to the website we must configure using  
"etc/squid/blocksite"

we edit the file by running:

```
$cd /etc/squid
```

```
$sudo gedit blocksite
```

5. Add the websites to block:

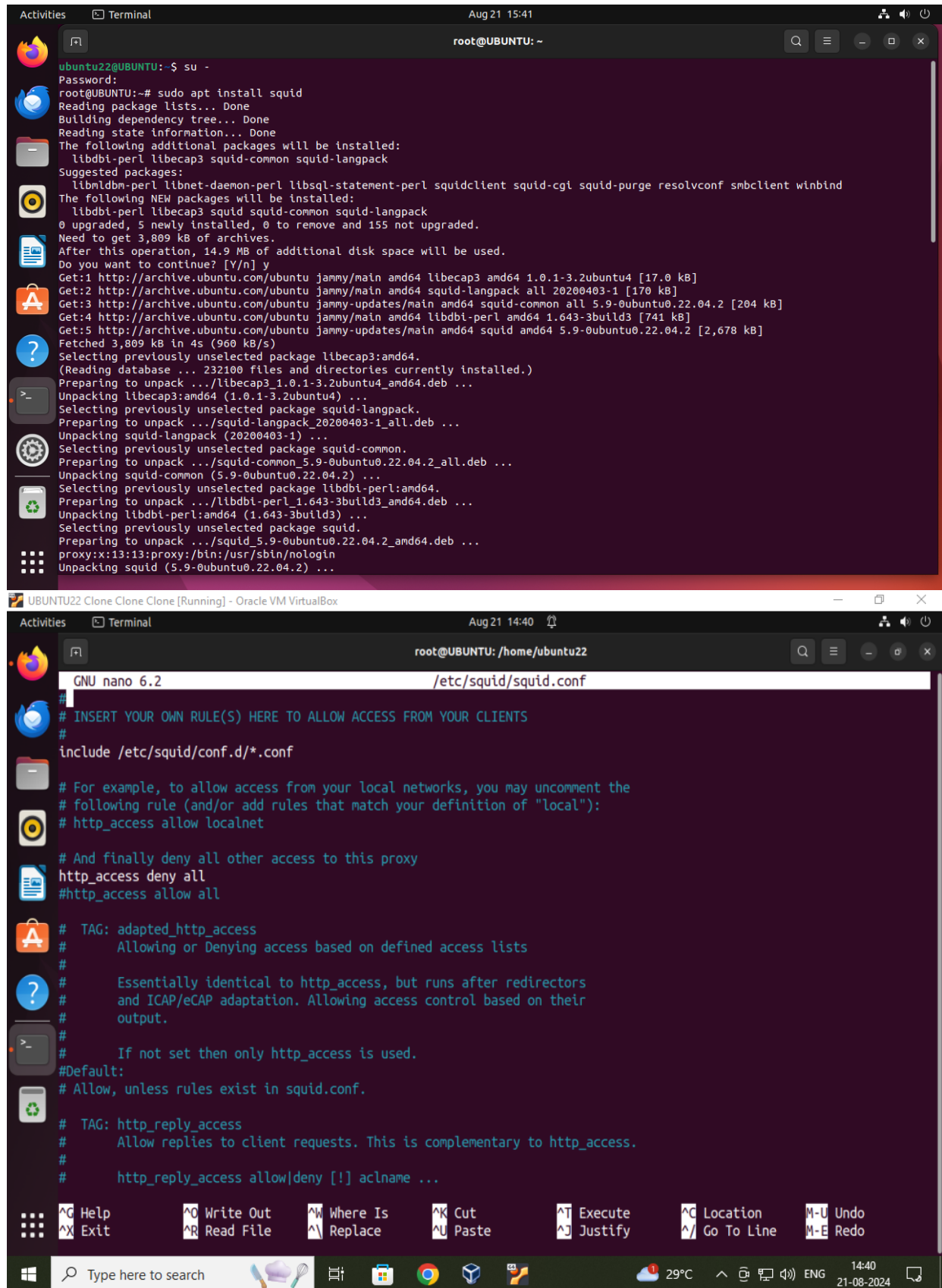
in this case, I am blocking youtube, facebook, google

6. To check the actual functioning of the proxy server go to the browser and click settings, search proxy in connection settings.
7. To configure Proxy access to the internet
8. Select Manual Proxy configuration
9. Type your HTTP Proxy(IP Address) and Port number as 3128.
10. Select SOCKS v5

#### CONNECTING TO WEBSITE

11. Search for the blocked websites
12. Access is denied to the above websites

## Result:



The first screenshot shows a terminal window with the command `sudo apt install squid` being executed. The output displays the package lists, dependency tree, and state information. It lists additional packages to be installed: `libdbi-perl libcap3 squid-common squid-langpack`. The terminal shows the progress of downloading and installing these packages, including the disk space requirements and the list of files to be unpacked.

```
ubuntu22@UBUNTU:~$ su -
Password:
root@UBUNTU:~# sudo apt install squid
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libdbi-perl libcap3 squid-common squid-langpack
Suggested packages:
  libmldbm-perl libnet-daemon-perl libsql-statement-perl squidclient squid-cgi squid-purge resolvconf smbclient winbind
The following NEW packages will be installed:
  libdbi-perl libcap3 squid squid-common squid-langpack
0 upgraded, 5 newly installed, 0 to remove and 155 not upgraded.
Need to get 3,809 kB of archives.
After this operation, 14.9 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://archive.ubuntu.com/ubuntu jammy/main amd64 libcap3 amd64 1.0.1-3.2ubuntu4 [17.0 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy/main amd64 squid-langpack all 20200403-1 [170 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 squid-common all 5.9-0ubuntu0.22.04.2 [204 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy/main amd64 libdbi-perl amd64 1.643-3build3 [741 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 squid amd64 5.9-0ubuntu0.22.04.2 [2,678 kB]
Fetched 3,809 kB in 4s (960 kB/s)
Selecting previously unselected package libcap3:amd64.
(Reading database ... 232100 files and directories currently installed.)
Preparing to unpack .../libcap3_1.0.1-3.2ubuntu4_amd64.deb ...
Unpacking libcap3:amd64 (1.0.1-3.2ubuntu4) ...
Selecting previously unselected package squid-langpack.
Preparing to unpack .../squid-langpack_20200403-1_all.deb ...
Unpacking squid-langpack (20200403-1) ...
Selecting previously unselected package squid-common.
Preparing to unpack .../squid-common_5.9-0ubuntu0.22.04.2_all.deb ...
Unpacking squid-common (5.9-0ubuntu0.22.04.2) ...
Selecting previously unselected package libdbi-perl:amd64.
Preparing to unpack .../libdbi-perl_1.643-3build3_amd64.deb ...
Unpacking libdbi-perl:amd64 (1.643-3build3) ...
Selecting previously unselected package squid.
Preparing to unpack .../squid_5.9-0ubuntu0.22.04.2_amd64.deb ...
Unpacking squid (5.9-0ubuntu0.22.04.2) ...
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
Unpacking squid (5.9-0ubuntu0.22.04.2) ...
```

The second screenshot shows the `/etc/squid/squid.conf` file being edited in the `nano` text editor. The file contains comments and configuration options for allowing or denying access from clients. The configuration includes a rule to allow access from the local network and a default rule to deny all other access.

```
GNU nano 6.2 /etc/squid/squid.conf
#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#
include /etc/squid/conf.d/*.conf

# For example, to allow access from your local networks, you may uncomment the
# following rule (and/or add rules that match your definition of "local"):
# http_access allow localnet

# And finally deny all other access to this proxy
http_access deny all
#http_access allow all

# TAG: adapted_http_access
#
#   Allowing or Denying access based on defined access lists
#
#   Essentially identical to http_access, but runs after redirectors
#   and ICAP/eCAP adaptation. Allowing access control based on their
#   output.
#
#   If not set then only http_access is used.
#Default:
# Allow, unless rules exist in squid.conf.

# TAG: http_reply_access
#   Allow replies to client requests. This is complementary to http_access.
#
#   http_reply_access allow|deny [!] aclname ...
```

```
UBUNTU22 Clone Clone Clone [Running] - Oracle VM VirtualBox
Activities Terminal Aug 21 14:40
root@UBUNTU: /home/ubuntu22

Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
squid is already the newest version (5.9-0ubuntu0.22.04.2).
0 upgraded, 0 newly installed, 0 to remove and 116 not upgraded.
root@UBUNTU:/home/ubuntu22# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::baf:b255:e5ba:c37c prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:0e:3c:4f txqueuelen 1000 (Ethernet)
    RX packets 450 bytes 569708 (569.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 368 bytes 36173 (36.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 362 bytes 39896 (39.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 362 bytes 39896 (39.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@UBUNTU:/home/ubuntu22# nano /etc/squid/squid.conf
root@UBUNTU:/home/ubuntu22# sudo systemctl restart squid
^C
root@UBUNTU:/home/ubuntu22# sudo systemctl restart squid
root@UBUNTU:/home/ubuntu22# nano /etc/squid/squid.conf
root@UBUNTU:/home/ubuntu22# sudo systemctl restart squid
root@UBUNTU:/home/ubuntu22# nano /etc/squid/squid.conf
root@UBUNTU:/home/ubuntu22#
```

```
UBUNTU22 Clone Clone Clone [Running] - Oracle VM VirtualBox
Activities Terminal Aug 21 14:47
root@UBUNTU: /home/ubuntu22

GNU nano 6.2 /etc/squid/squid.conf *
# their server via certain well-known link-local (a.k.a. APIPA) addresses.
#http_access deny to_linklocal
#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#
include /etc/squid/conf.d/*.conf

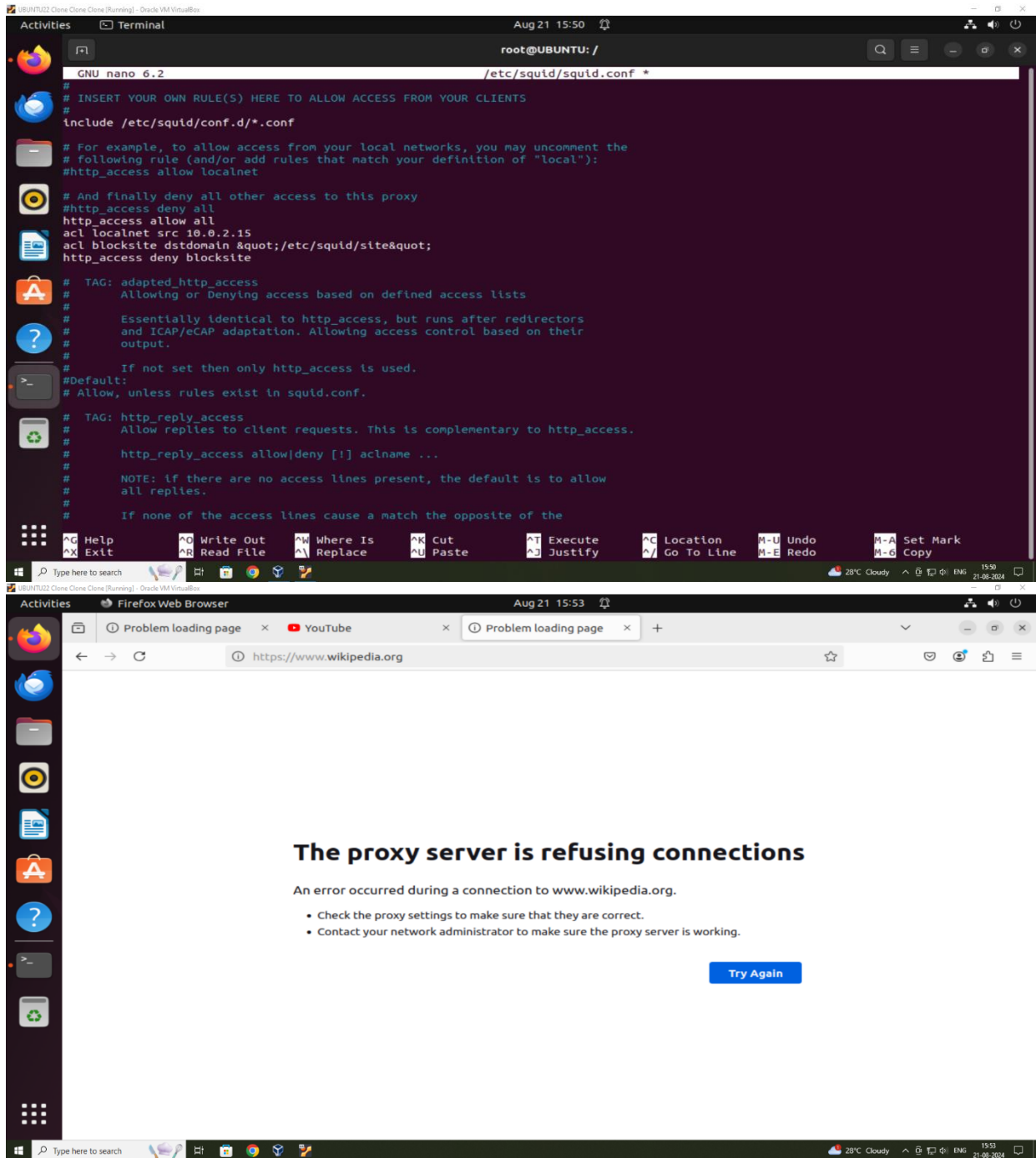
# For example, to allow access from your local networks, you may uncomment the
# following rule (and/or add rules that match your definition of "local"):
# http_access allow localnet

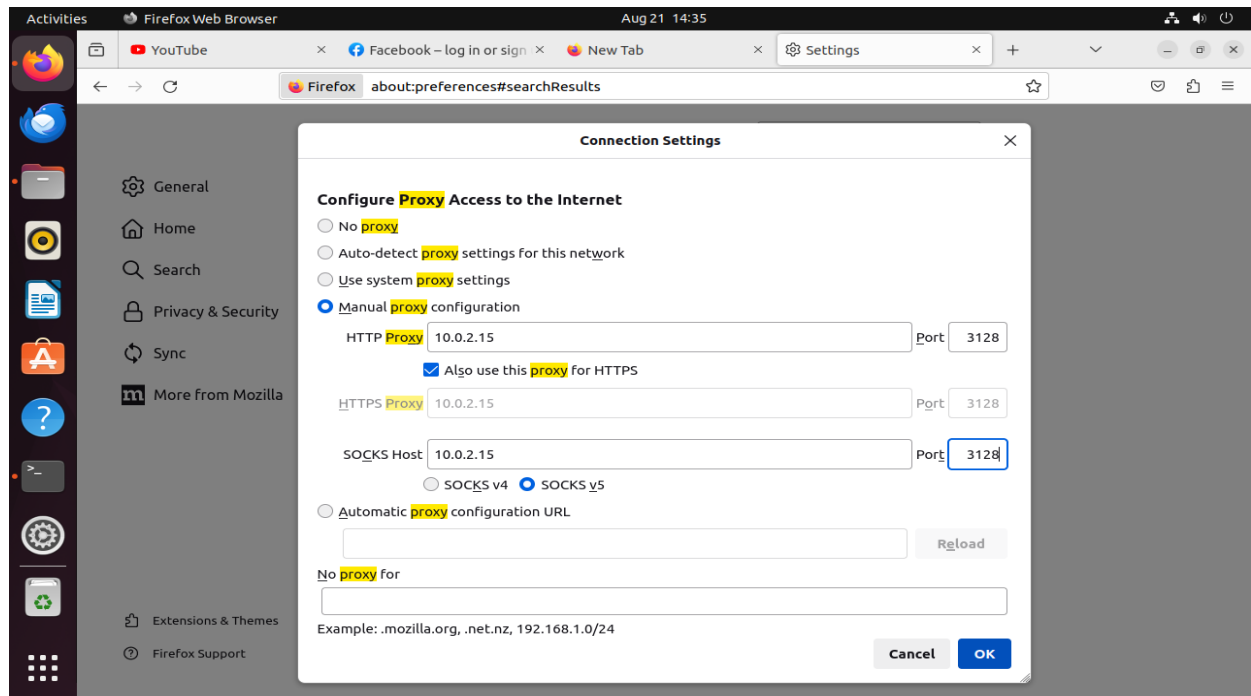
# And finally deny all other access to this proxy
#http_access deny all
http_access allow all
acl localhost src 10.0.2.15
acl blocksite dstdomain "/etc/squid/site"
http_access deny blocksite

# TAG: adapted_http_access
#
#     Allowing or Denying access based on defined access lists
#
#     Essentially identical to http_access, but runs after redirectors
#     and ICAP/eCAP adaptation. Allowing access control based on their
#     output.
#
#     If not set then only http_access is used.
#Default:

^O Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify
^_          ^L Location   ^M-U Undo
^_          ^_ Go To Line ^M-E Redo

Type here to search Match 29°C 14:40 21-08-2024
```





All the commands have been executed and the output has been obtained successfully.