NETWORKING ASSIGNMENT 2

* We will try to create a web server first,

sumaM@GQNTJM3:~$ nc -lp 5000

|

* We initially execute this command nc -lp <port>, Here l stands for listening, So, this port listens for some data as well. p stands for port and we have provided the port as 5000
* We do not get any response initially and just can see the blinking cursor in the next line as shown above.

------------------------------------------------

sumaM@GQNTJM3:~$ nc -lp 5000

GET / HTTP/1.1

Host: localhost:5000

Connection: keep-alive

sec-ch-ua: " Not A;Brand";v="99", "Chromium";v="99", "Google Chrome";v="99"

sec-ch-ua-mobile: ?0

sec-ch-ua-platform: "Linux"

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (X11; Linux x86\_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/99.0.4844.82 Safari/537.36

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b3;q=0.9

Sec-Fetch-Site: none

Sec-Fetch-Mode: navigate

Sec-Fetch-User: ?1

Sec-Fetch-Dest: document

Accept-Encoding: gzip, deflate, br

Accept-Language: en-GB,en-US;q=0.9,en;q=0.8

Cookie: Idea-bcc0b7a3=01ee2591-7f6d-4951-98e6-d6ae13341306

* We now need to open the url : localhost:5000 in the browser and immediately after opening the url in browser, we get the response in the terminal as shown above.
* Then we need to provide the below 3 lines to say about the status of connection that we have given 200 OK, the type of content we want to display on the webserver, that is html and we also need to give a server name that we have given nooblinux.

HTTP/1.1 200 OK

Content-Type: text/html; charset=UTF-8

Server: nooblinux

* Then we can type the HTML page and we can se that the browser is showing up the content in real time. I have added the below 3 lines to check is it is reflecting in the browser

<!doctype html>

<title>NoobLinux</title>

<h1>Sumanth Mysore</h1>

sumaM@GQNTJM3:~$ nc -lp 5000

GET / HTTP/1.1

Host: localhost:5000

Connection: keep-alive

sec-ch-ua: " Not A;Brand";v="99", "Chromium";v="99", "Google Chrome";v="99"

sec-ch-ua-mobile: ?0

sec-ch-ua-platform: "Linux"

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (X11; Linux x86\_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/99.0.4844.82 Safari/537.36

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b3;q=0.9

Sec-Fetch-Site: none

Sec-Fetch-Mode: navigate

Sec-Fetch-User: ?1

Sec-Fetch-Dest: document

Accept-Encoding: gzip, deflate, br

Accept-Language: en-GB,en-US;q=0.9,en;q=0.8

Cookie: Idea-bcc0b7a3=01ee2591-7f6d-4951-98e6-d6ae13341306

HTTP/1.1 200 OK

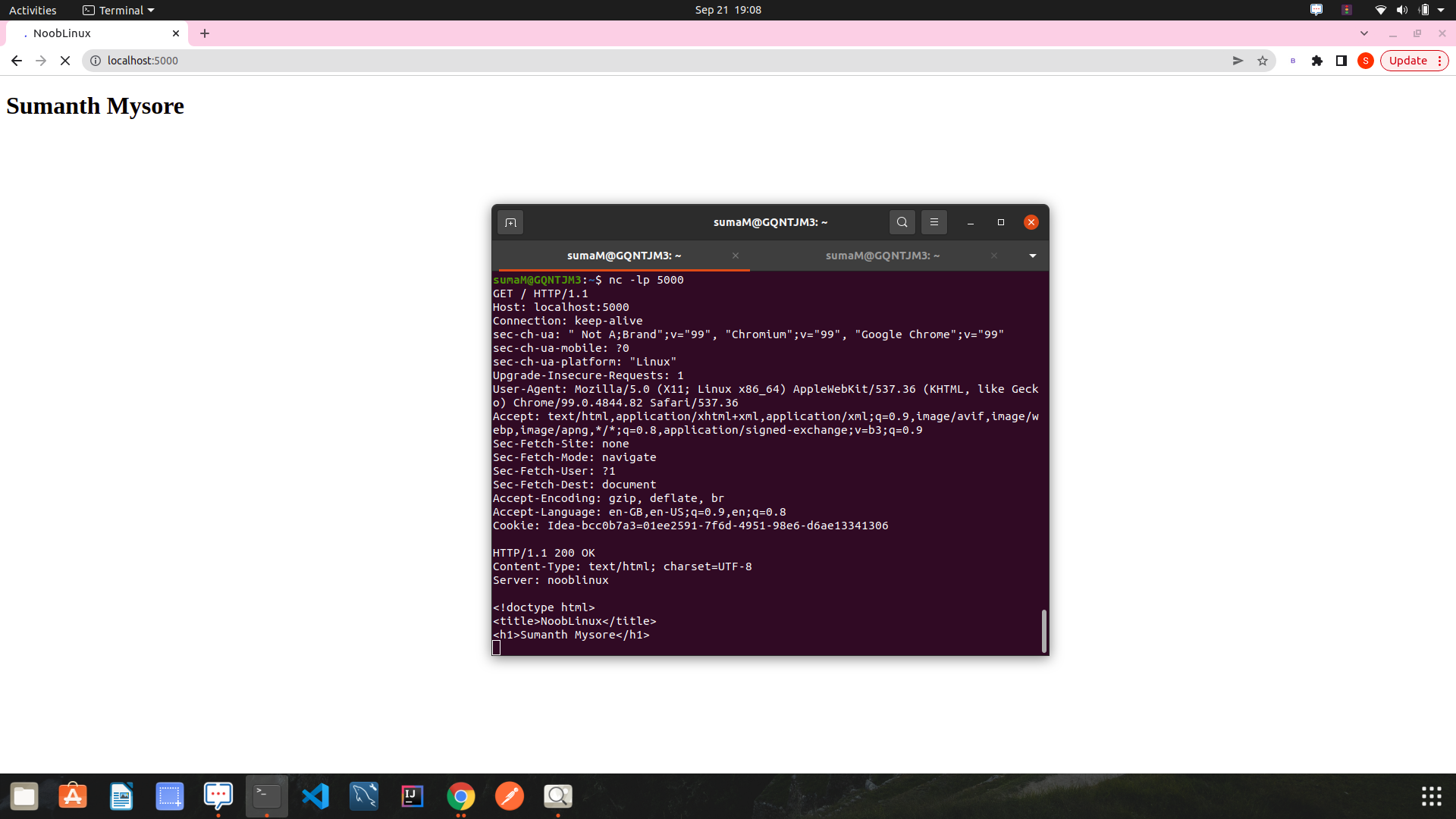
Content-Type: text/html; charset=UTF-8

Server: nooblinux

<!doctype html>

<title>NoobLinux</title>

<h1>Sumanth Mysore</h1>



* We can see the content enclosed in the h1 tags (Sumanth Mysore) on the webpage. Thus we are successful creating the web server and we got to know how to create a local webserver.

Now we will create a simple chat using NETCAT

We can do this over remote network machines or within your local network. I have done it creating a local network. We’ll just need two computers that can run Netcat.

* On the first machine we’ll just run the command to create a server and listen on a port, in our case 4000.
* On the second machine we’ll run the command to connect to the first machine’s IP and port, thereby establishing the connection. From there we can just write messages from one machine and they’ll instantly appear on the other.

As the first step, We need to find the PRIVATE IP address of the first system.

* Your private IP address is different than your public IP address [which is the IP address most of us are familiar with].
* A private IP address is an IP address used within a private network, such as your home network (unlike the public IP address which you would use to access the internet).
* Typically, *a private IP address is assigned to each device connected to your local network by your router*. Say you have multiple computers and phones, a printer and a smart TV – all of them are assigned a private IP address.
* On Linux, you can determine your private IP address using command such as ip addr,

$ ip addr

1: lo: &lt;LOOPBACK,UP,LOWER\_UP&gt; mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00

inet 127.0.0.1/8 scope host lo

valid\_lft forever preferred\_lft forever

inet6 ::1/128 scope host

valid\_lft forever preferred\_lft forever

2: ens33: &lt;BROADCAST,MULTICAST,UP,LOWER\_UP&gt; mtu 1500 qdisc fq\_codel state UNKNOWN group default qlen 1000

link/ether 00:0c:29:12:e9:70 brd ff:ff:ff:ff:ff:ff

altname enp2s1

inet **192.168.29.199/24** brd 192.168.145.255 scope global dynamic noprefixroute ens33

valid\_lft 992sec preferred\_lft 992sec

inet6 fe80::c567:c033:897f:58ea/64 scope link noprefixroute

valid\_lft forever preferred\_lft forever

* what comes after inet in the details for the network interface that private ip address and that i have highlighted in the above response.
* So the private IP address of the machine1 is 192.168.145.131
* Now, assuming that you’ve found your private IP address for machine1, create a server on it, listening on any port (I’ll use 4000).
* To do this run:

$ nc -vlp 4000

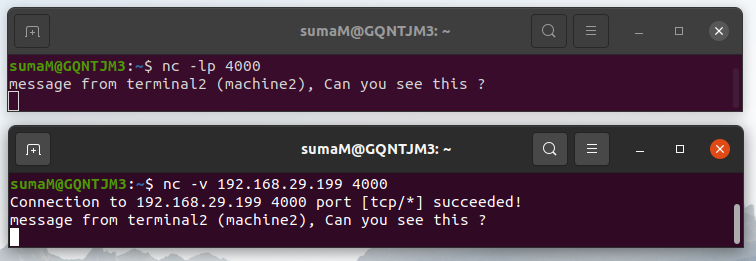
* Now Netcat will be listening on machine1 which has the IP address of 192.168.145.131. *This is our server*.
* Now let’s connect to this server from another device within our local network which is machine2. We’ll use the server’s IP address and port to connect to it.
* Run the following command :

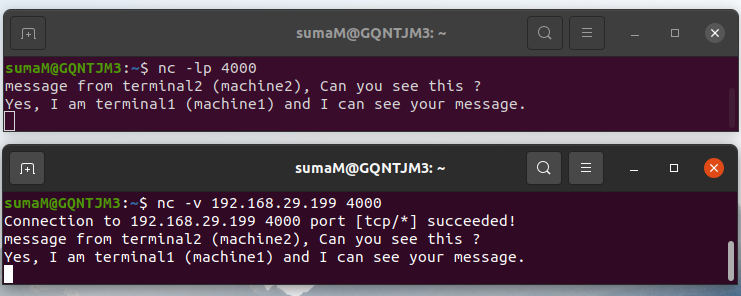
$ nc -v 192.168.29.199 4000

* Now the client-server connection has been established. You can type in anything in any of the machines and you’ll see the message instantly on the other machine. Here are the commands and outputs of each machine:

### Within a single computer,

* If you do not have access to another computer in your local network, you can also try this on your computer with two terminals.
* Open two terminals and just follow the same procedure with the nc command.
* Create a server and a client and you can send text from one to the other terminal and communicate between them in real-time.
* I have done in this way and attaching few screenshots of the obtained responses.



* First, we have sent a message from terminal2 (bottom terminal) and it immediately got shown up in the terminal1 (top terminal).
* Second, we have replied to that message by terminal2 from terminal1 and that reply also gor immediately shown up in terminal2.

Thus the connection between 2 terminals as server and client has been successful and thus a mini chat has been set up between 2 devices on a network.