

iu-ne-lab-01-Iskander_Nafikov

- **Name:** Iskander Nafikov
- **E-mail:** i.nafikov@innopolis.university
- **Username:** [iskanred](#)
- **Hostname:** lenovo

Overview

In this lab, you will set up your initial basic network for INR course. Then you will make a small switched network, get familiar with subnets and configure IPs and then test the connectivity between your machines. Once you have the network running, you will configure a router as a gateway.

Task 1 - Tools

To make it easier to prototype network topologies and troubleshoot them, you will be using GNS3 software to emulate networks.

1. Install the needed dependencies for GNS3: QEMU/KVM, Docker and Wireshark.

I have installed the necessary software (**Docker** and **QEMU** for virtualisation and **WireShark** for traffic sniffing):

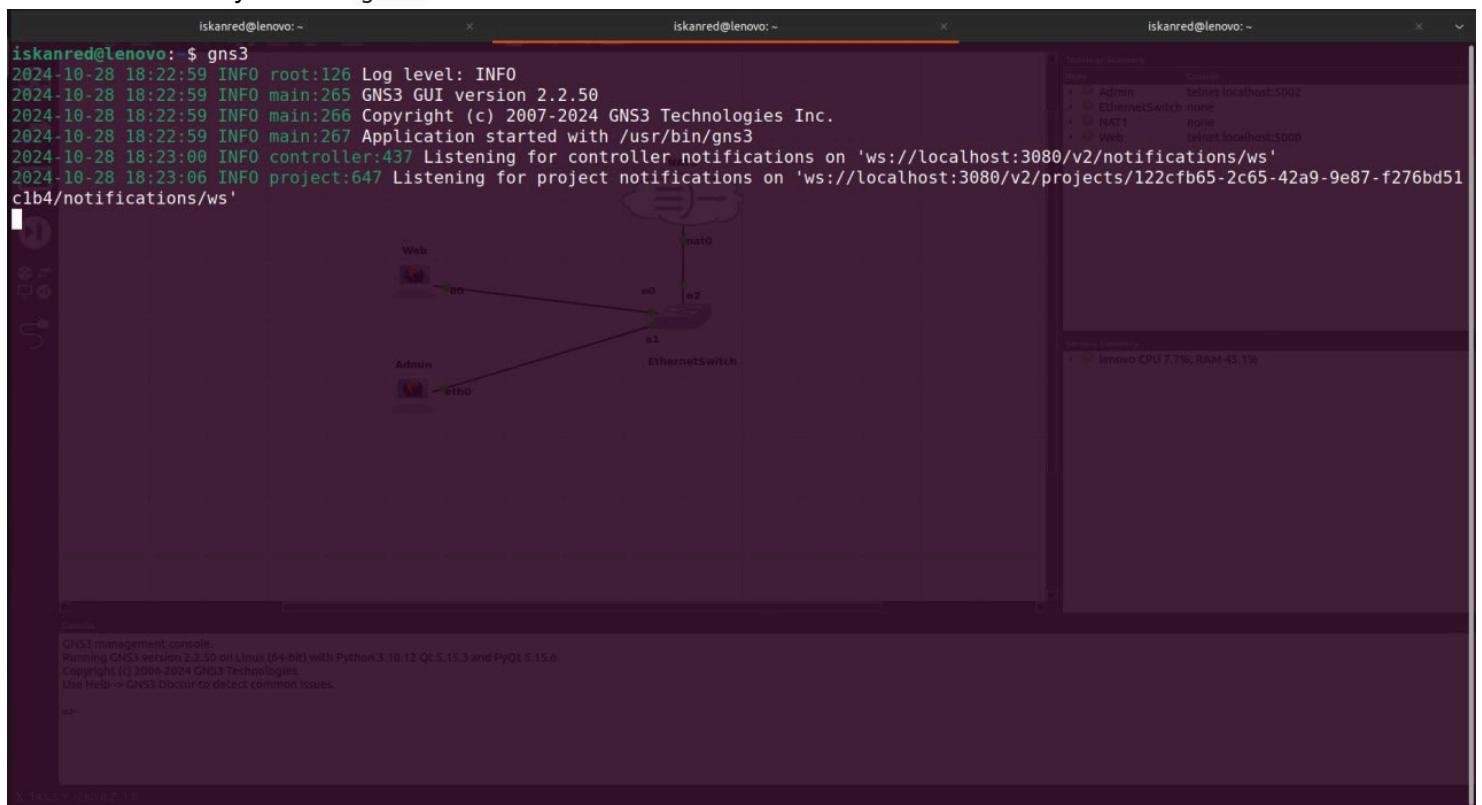
- Here I successfully started gns3server

The screenshot shows three windows. The leftmost window is a terminal session for user 'iskanred' on 'lenovo' at the prompt '\$'. It displays the log output of the 'gns3server' command, which starts with 'INFO run.py:218 GNS3 server version 2.2.50' and continues through various startup messages including configuration loading and controller initialization. The middle window is a 'Topology Summary' dialog from GNS3, showing a network diagram with nodes labeled 'Admin', 'EthernetSwitch', 'NAT1', and 'Web'. The rightmost window is a Wireshark capture window showing network traffic on interface 'eth0'. The traffic includes several 'GET /v2/version HTTP/1.1' requests from '127.0.0.1' to '192.168.1.10' (the GNS3 server IP), indicating successful communication.

```
iskanred@lenovo: ~
iskanred@lenovo: ~
iskanred@lenovo: ~
```

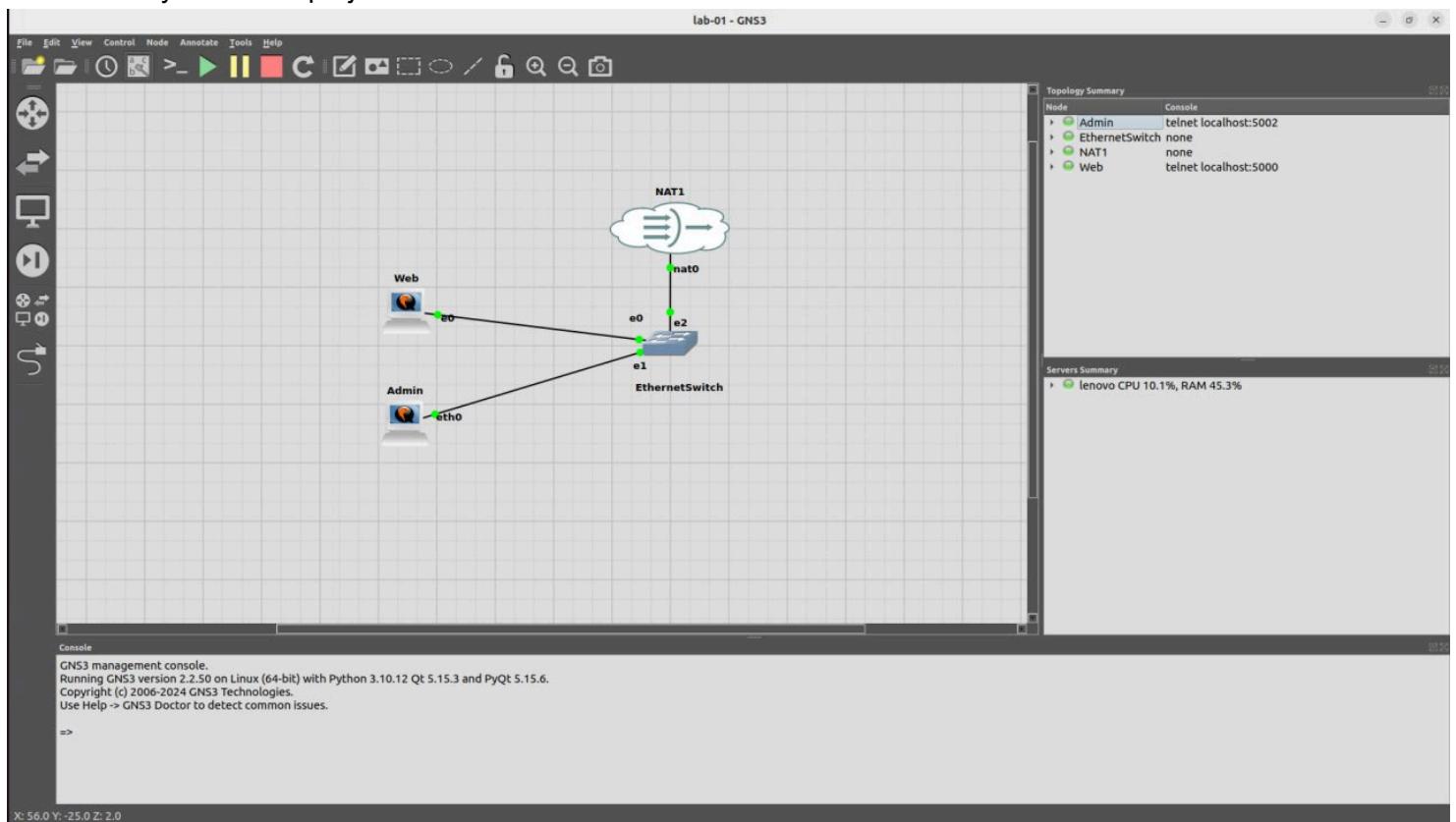
```
iskanred@lenovo: $ gns3server
2024-10-28 18:22:55 INFO run.py:218 GNS3 server version 2.2.50
2024-10-28 18:22:55 INFO run.py:220 Copyright (c) 2007-2024 GNS3 Technologies Inc.
2024-10-28 18:22:55 INFO run.py:223 Config file /home/iskanred/.config/GNS3/2.2/gns3_server.conf loaded
2024-10-28 18:22:55 INFO run.py:236 HTTP authentication is enabled with username 'admin'
2024-10-28 18:22:55 INFO run.py:242 Running with Python 3.10.12 and has PID 3171
2024-10-28 18:22:55 INFO run.py:248 Using system certificate store for SSL connections
2024-10-28 18:22:55 INFO run.py:78 Current locale is en_US.UTF-8
2024-10-28 18:22:55 INFO web_server.py:338 Starting server on localhost:3080
2024-10-28 18:22:55 INFO __init__.py:70 Load controller configuration file /home/iskanred/.config/GNS3/2.2/gns3_controller.conf
2024-10-28 18:22:55 INFO __init__.py:74 Controller is starting
2024-10-28 18:22:55 INFO __init__.py:320 Installing base configs in '/home/iskanred/GNS3/configs'
2024-10-28 18:22:55 INFO __init__.py:333 Installing built-in disks in '/home/iskanred/GNS3/images/QEMU'
2024-10-28 18:22:55 INFO __init__.py:255 Comparing controller version 2.2.50 with config version 2.2.50
2024-10-28 18:22:55 INFO __init__.py:263 Built-in appliances are installed in '/home/iskanred/.local/share/GNS3/appliances'
2024-10-28 18:22:55 INFO compute.py:64 Create compute local
2024-10-28 18:22:55 INFO compute.py:364 Connecting to compute 'local'
2024-10-28 18:22:55 INFO web_log.py:211 127.0.0.1 [28/Oct/2024:18:22:55 +0300] "GET /v2/compute/capabilities HTTP/1.1" 200 553 "-" "Python/3.10 aiohttp/3.10.10"
2024-10-28 18:22:55 INFO web_server.py:247 Computing image checksums...
2024-10-28 18:22:55 INFO notification_handler.py:50 New client has connected to compute WebSocket
2024-10-28 18:22:55 INFO compute.py:431 Connected to compute 'local' WebSocket 'http://localhost:3080/v2/compute/notifications/ws'
2024-10-28 18:22:55 INFO web_server.py:249 Finished computing image checksums
2024-10-28 18:22:59 INFO web_log.py:211 127.0.0.1 [28/Oct/2024:18:22:59 +0300] "GET /v2/version HTTP/1.1" 200 229 "-" "GNS3 QT Client v2.2.50"
2024-10-28 18:23:00 INFO web_log.py:211 127.0.0.1 [28/Oct/2024:18:23:00 +0300] "GET /v2/version HTTP/1.1" 200 229 "-" "GNS3 QT Client v2.2.50"
2024-10-28 18:23:00 INFO web_log.py:211 127.0.0.1 [28/Oct/2024:18:23:00 +0300] "GET /v2/version HTTP/1.1" 200 229 "-" "GNS3 QT Client v2.2.50"
2024-10-28 18:23:00 INFO web_log.py:211 127.0.0.1 [28/Oct/2024:18:23:00 +0300] "GET /v2/version HTTP/1.1" 200 229 "-" "GNS3 QT Client v2.2.50"
2024-10-28 18:23:00 INFO web_log.py:211 127.0.0.1 [28/Oct/2024:18:23:00 +0300] "GET /v2/version HTTP/1.1" 200 229 "-" "GNS3 QT Client v2.2.50"
2024-10-28 18:23:00 INFO notification_handler.py:73 New client has connected to controller WebSocket
2024-10-28 18:23:00 INFO web_log.py:211 127.0.0.1 [28/Oct/2024:18:23:00 +0300] "GET /v2/templates HTTP/1.1" 200 4921 "-" "GNS3 QT Client v2.2.50"
2024-10-28 18:23:00 INFO web_log.py:211 127.0.0.1 [28/Oct/2024:18:23:00 +0300] "GET /v2/computes HTTP/1.1" 200 1022 "-" "GNS3 QT Client v2.2.50"
2024-10-28 18:23:00 INFO web_log.py:211 127.0.0.1 [28/Oct/2024:18:23:00 +0300] "GET /v2/computes HTTP/1.1" 200 1022 "-" "GNS3 QT Client v2.2.50"
```

- Here I successfully started gns3 client



2. Start a GNS3 project, configure the pre-installed Ubuntu Cloud Guest template. Check that you can start it.

- I successfully created a project:



- Here is the telnet connection to the Web machine (Ubuntu Cloud Guest) console:

```

Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

ubuntu-cloud login: ubuntu
Password:
Welcome to Ubuntu 24.10 (GNU/Linux 6.11.0-8-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://Landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Mon Oct 28 16:11:08 UTC 2024

System load: 0.0          Processes: 94
Usage of /: 81.5% of 2.25GB Users logged in: 0
Memory usage: 16%          IPv4 address for e0: 192.168.122.2
Swap usage: 0%             EthernetSwitch

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

14 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Failed to connect to https://changelogs.ubuntu.com/meta-release. Check your Internet connection or proxy settings

```

```

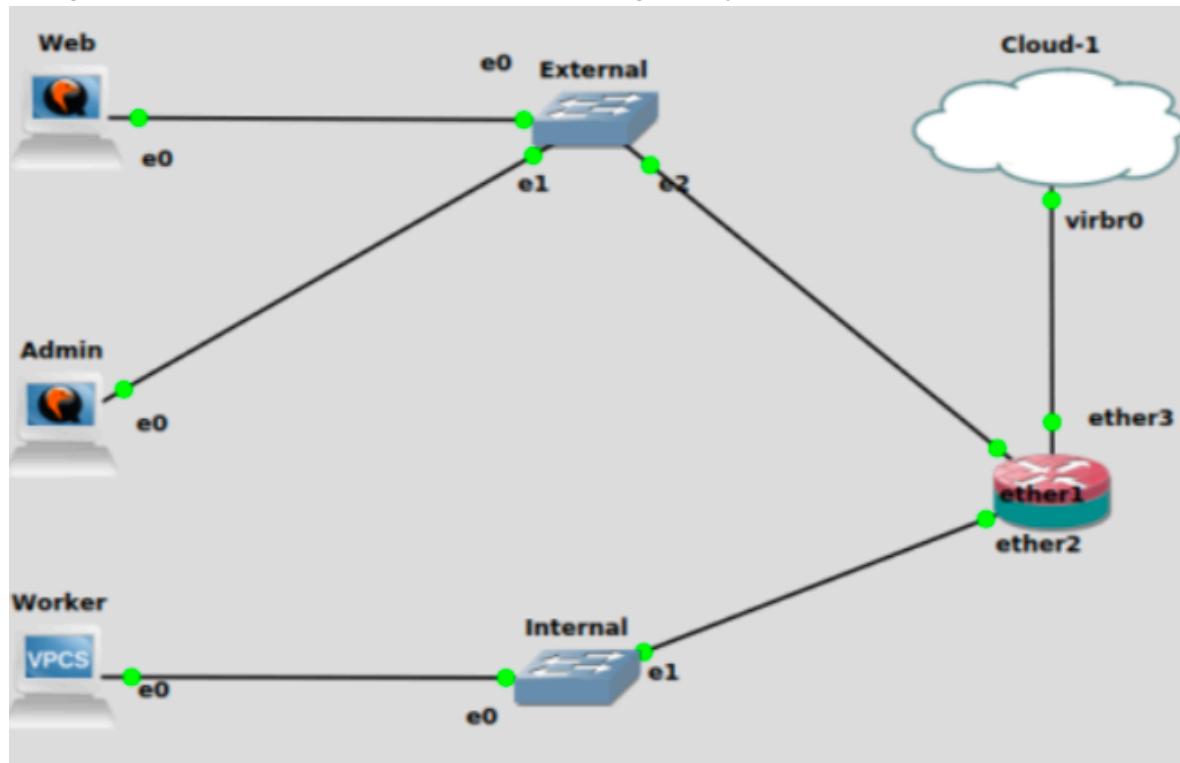
GNS3 management console.
Running GNS3 version 2.2.50 on Linux (64-bit) with Python 3.10.12-QT 5.15.3 and PyQt 5.15.6.
ubuntu@ubuntu-cloud:~$ pwd
/home/ubuntu
ubuntu@ubuntu-cloud:~$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: en
   Active: active (running) since Mon 2024-10-28 15:23:50 UTC; 47min ago
     Invocation: fe7b2e53e5484189af15e4b0a7469972

```

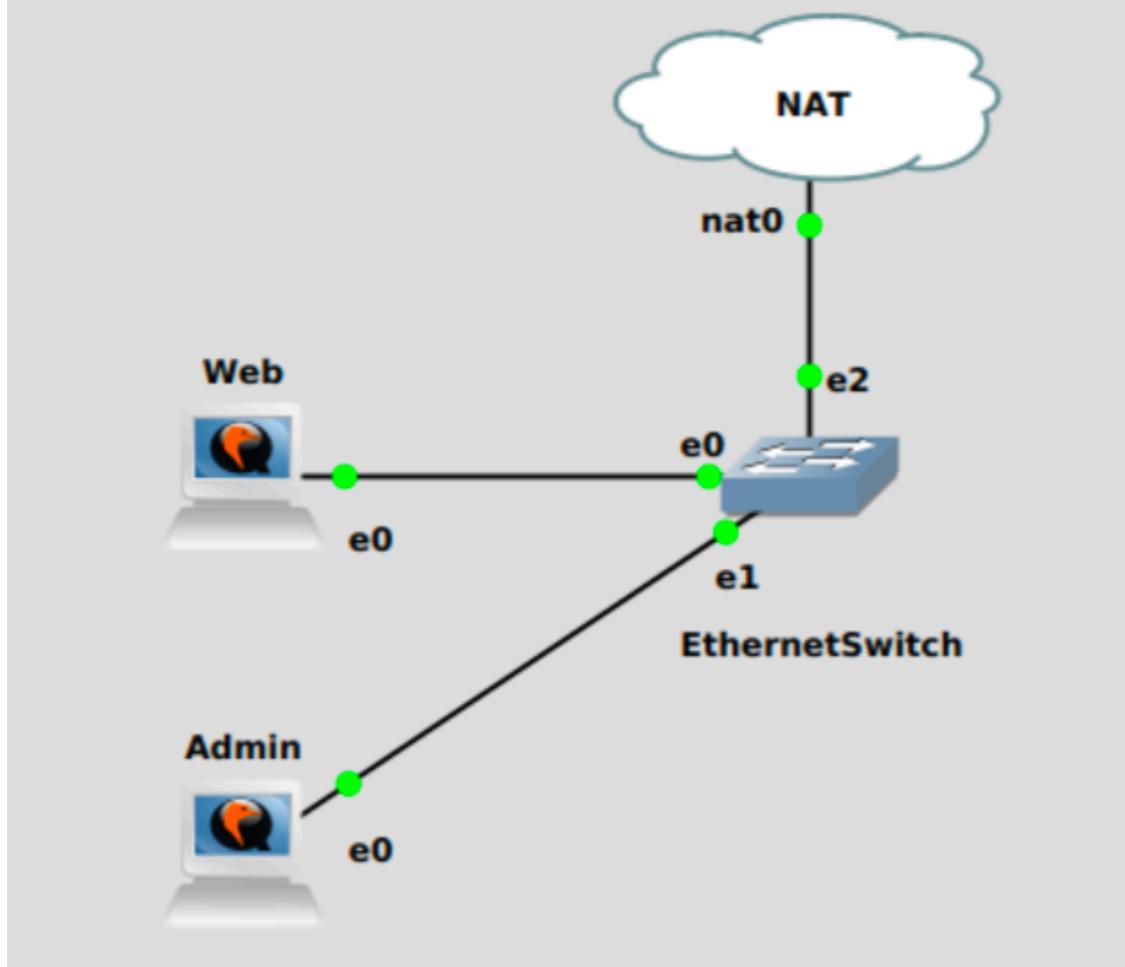
3. What are the different ways you can configure internet access in GNS3?

I found three convenient ways to configure internet access in GNS3 for a node:

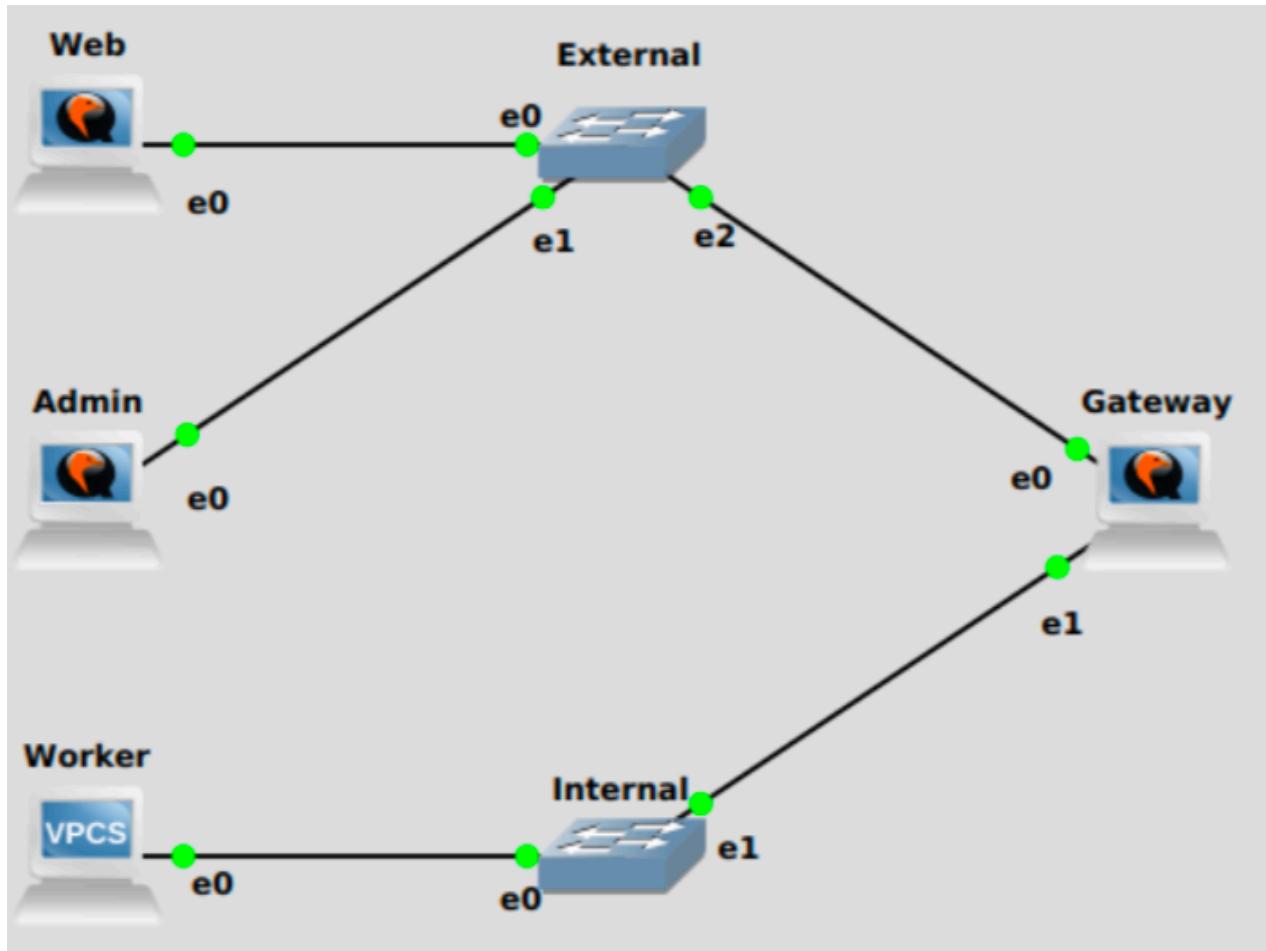
- Using a "Router" node that can represent itself a gateway to the internet, a "Cloud" node actually.



- Using a "NAT" node that can also represent itself a gateway to the internet. By default, it runs a DHCP server with a predefined pool in the 192.168.122.0/24 range.



- Another method is to create your own gateway node using some PC instead of a simple router. The usage is the same as for a router



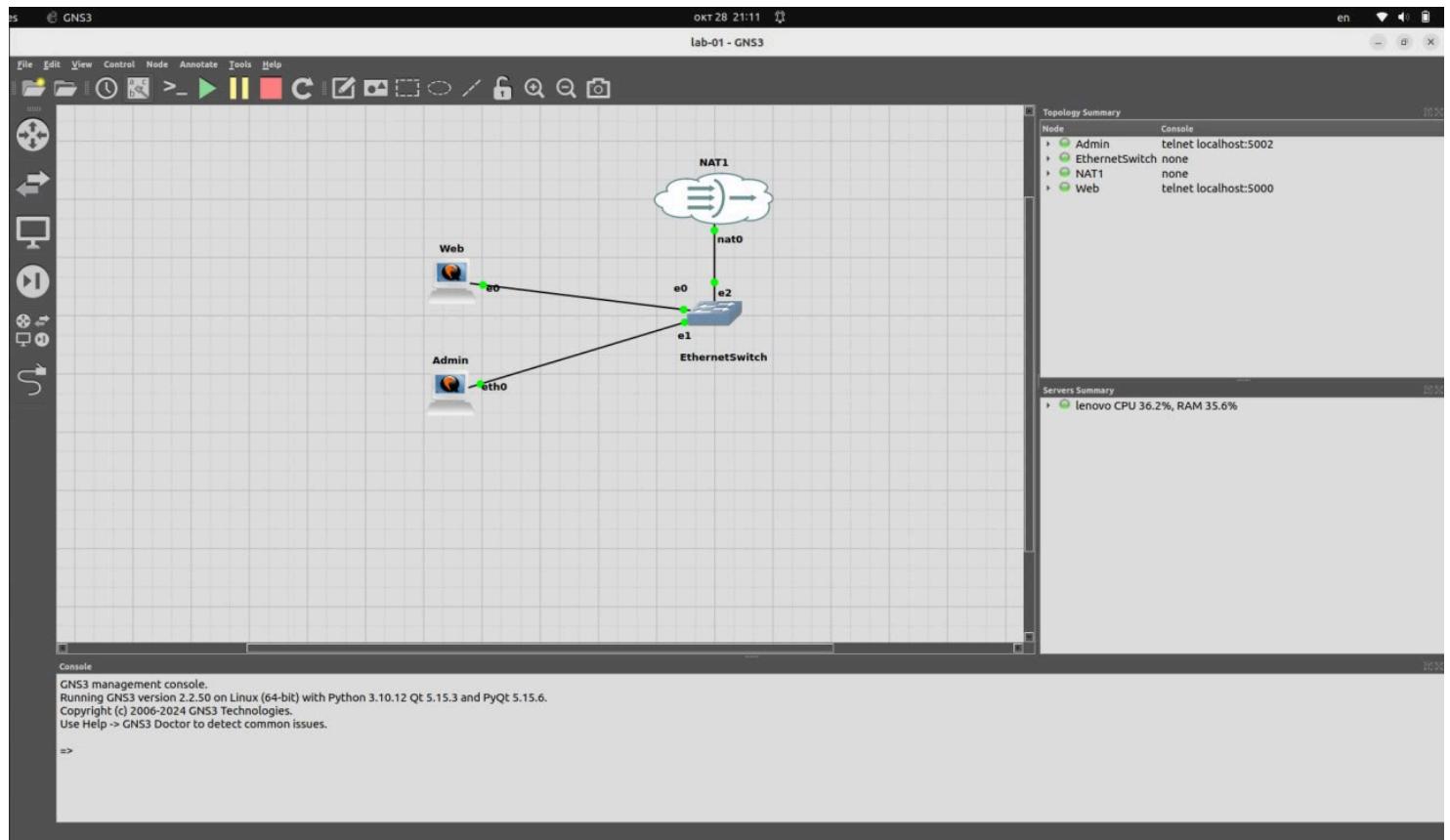
Overall, the first and the last approaches are similar, and the first one is easier and more convenient to use.

The first and the second ones differ. NAT is easier, but is limited to some network with only 256 available nodes (192.168.122.0/24). A router is more flexible but more difficult approach.

Task 2 - Switching

1. Make the following network topology:

- I made up the following topology where Web is the Ubuntu Cloud Guest node and the Admin is the Ubuntu Docker Guest node



- For example, here is the Web node's console:

```

NoCloud [seed=/dev/sr0]. Up 12.09 seconds
[ OK ] Finished cloud-final.service - Cloud-init: Final Stage.
[ 12.114771] sh[804]: Completed socket interaction for boot stage final
[ OK ] Reached target cloud-init.target - Cloud-init target.

Ubuntu 24.10 ubuntu-cloud ttyS0

ubuntu-cloud login: ubuntu
Password:
Welcome to Ubuntu 24.10 (GNU/Linux 6.11.0-8-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Mon Oct 28 18:15:56 UTC 2024

System load: 0.01 Processes: 96
Usage of /: 82.1% of 2.25GB Users logged in: 0
Memory usage: 16% IPv4 address for e0: 192.168.122.2
Swap usage: 0%

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

14 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Failed to connect to https://changelogs.ubuntu.com/meta-release. Check your Internet connection or proxy settings

```

- To work it well I configured a subnet 192.168.122.0/28 and set the static IP addresses (192.168.122.2 and 192.168.122.3) for both PC nodes and set 192.168.122.1 as the default gateway which is represented by NAT node.

- First, I configured the local network for the Web node with the help of ` inside the etc/netplan/50-cloud-init.yaml:

```

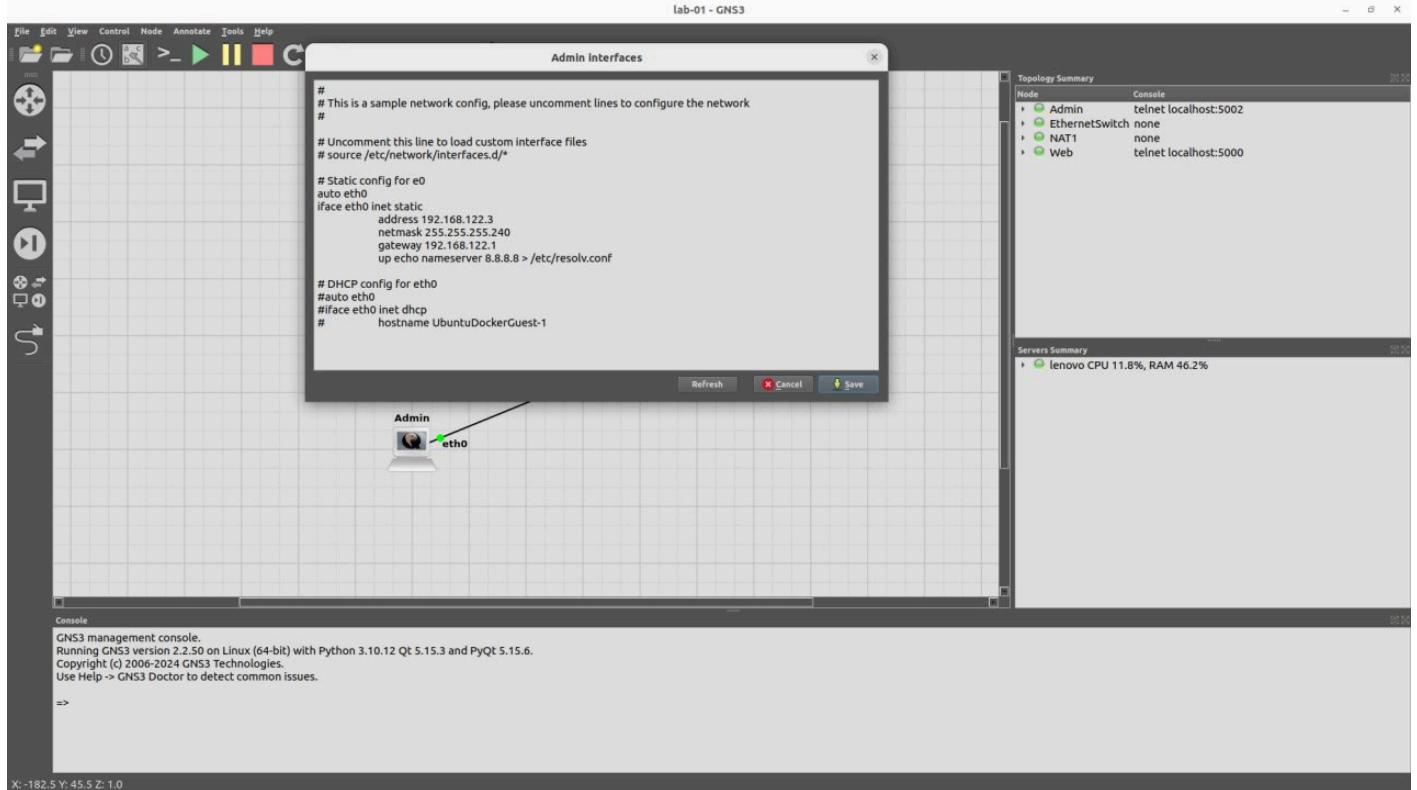
network:
  version: 2
  ethernets:
    e0:
      match:
        macaddress: "0c:eb:82:e3:00:00"
      dhcp4: no
      dhcp6: no
      addresses: [192.168.122.2/28]
      gateway4: 192.168.122.1
      set-name: "e0"
      nameservers:
        addresses: [1.1.1.1, 8.8.8.8]

```

/etc/netplan/50-cloud-init.yaml" 13L, 272B 13,36 All

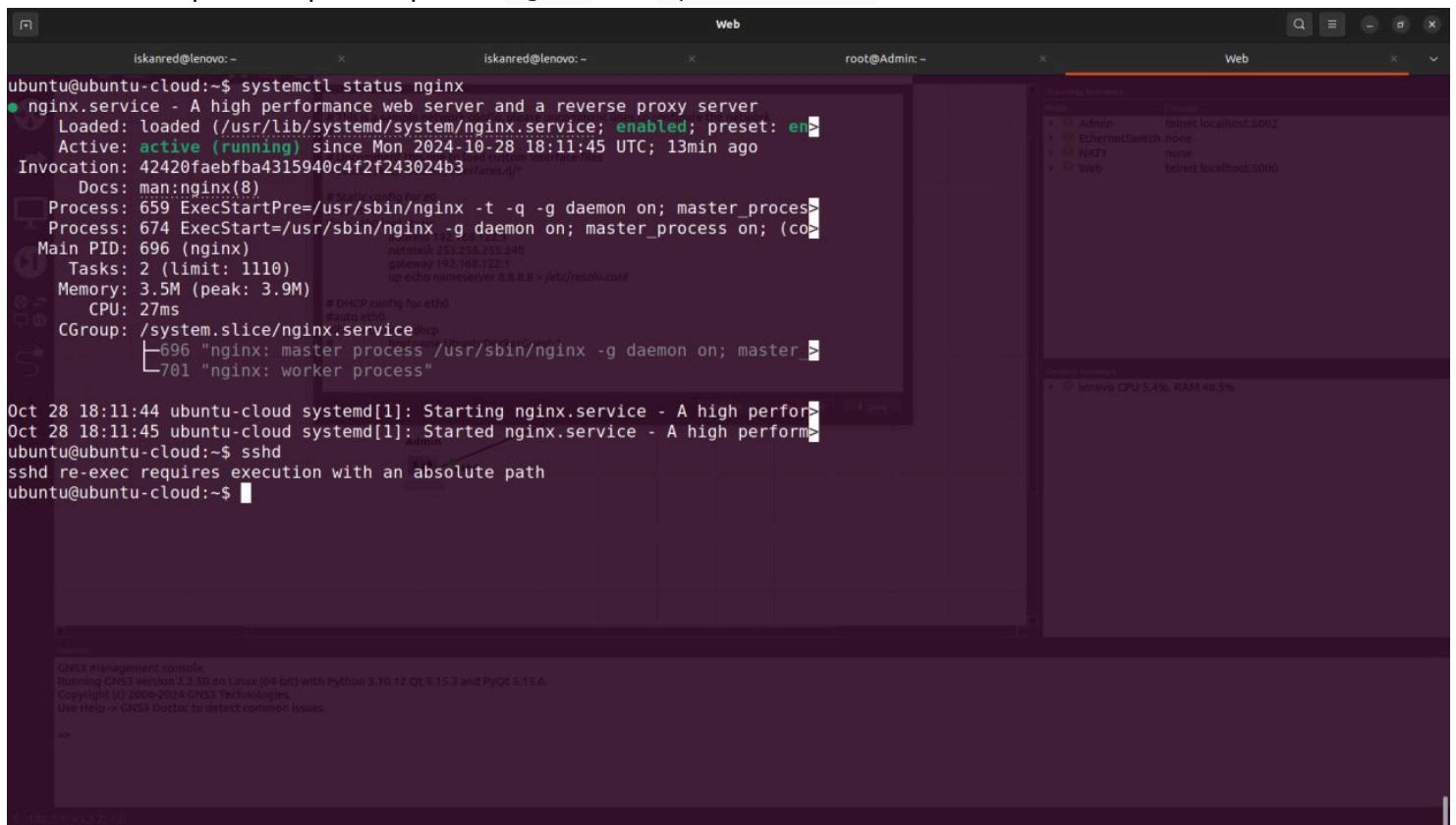
CN3 management sample
Running CN3 version 2.2.50 on Linux (64-bit) with Python 3.10.12, Qt 5.15.3 and PyQt 5.15.6.
Copyright (C) 2006-2024 CN3 Technologies.
Use Help-> CN3 Doctor to detect common issues.

- Then I did similar for the Admin node inside its config in GNS since it is a Docker node:



2. Install openssh-server on both VMs and nginx web server on the Web VM.

- Here the example of output that proves nginx and openssh-server are installed:



3. What is the IP of the mask corresponding to /28?

The subnet mask is 255.255.255.240 or in the binary form 11111111.11111111.11111111.11110000 .

We see that there are only 4 free bits at the end of the mask => we can assign only 2^4 addresses in our subnet = 16.

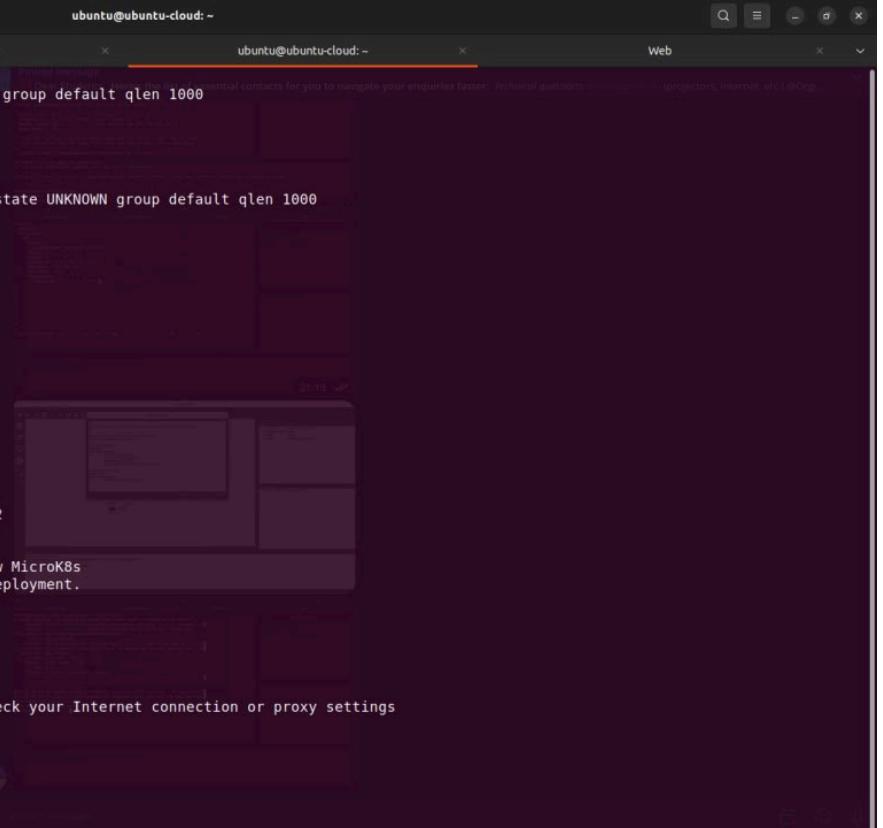
What is more, two IP addresses are reserved for broadcast (255.255.255.255) and network address 255.255.255.240 => only 14 IP-addresses can be actually assigned to some hosts in our local network.

4. Configure the VMs with private static IPs under a /28 subnet.

Demonstrated in the 1st clause

5. Check that you have connectivity between them.

- Here is the example of ssh connection from Admin node to the Web node:



```
root@Admin:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> 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
8: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 1000
    link/ether 02:42:cc:32:98:00 brd ff:ff:ff:ff:ff:ff
        inet 192.168.122.3/28 scope global eth0
            valid_lft forever preferred_lft forever
        inet6 fe80::42:cff:fe32:9800/64 scope link
            valid_lft forever preferred_lft forever
root@Admin:~# ssh ubuntu@192.168.122.2
ubuntu@192.168.122.2's password:
Welcome to Ubuntu 24.10 (GNU/Linux 6.11.0-8-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Mon Oct 28 18:34:35 UTC 2024

System load:  0.16      Processes:         97
Usage of /:   82.1% of 2.25GB  Users logged in:     0
Memory usage: 17%          IPv4 address for eth0: 192.168.122.2
Swap usage:   0%          

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

14 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Failed to connect to https://changelogs.ubuntu.com/meta-release. Check your Internet connection or proxy settings

Last login: Sun Oct 27 22:35:02 2024 from 192.168.122.3
ubuntu@ubuntu-cloud:~$ pwd
/home/ubuntu
ubuntu@ubuntu-cloud:~$
```

- Here is the example of pinging the Admin node from the Web node:

```

root@Admin:~# systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/etc/systemd/system/nginx.service)
   Active: active (running) since Sun 2017-01-01 12:00:00 UTC
     Docs: man:nginx(8)
     Tasks: 1 (limit: 4900)
    CGroup: /system.slice/nginx.service
            └─nginx[nginx: master process]

root@Admin:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 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 noprefixroute
        valid_lft forever preferred_lft forever
2: e0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 0c:eb:82:e3:00:00 brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet 192.168.122.2/28 brd 192.168.122.15 scope global e0
        valid_lft forever preferred_lft forever
    inet6 fe80::fe80:82ff:fe3:0/64 scope link proto kernel ll
        valid_lft forever preferred_lft forever
ubuntu@ubuntu-cloud:~$ ping 192.168.122.3
PING 192.168.122.3 (192.168.122.3) 56(84) bytes of data.
64 bytes from 192.168.122.3: icmp_seq=1 ttl=64 time=0.812 ms
64 bytes from 192.168.122.3: icmp_seq=2 ttl=64 time=0.894 ms
64 bytes from 192.168.122.3: icmp_seq=3 ttl=64 time=0.871 ms
64 bytes from 192.168.122.3: icmp_seq=4 ttl=64 time=0.851 ms
64 bytes from 192.168.122.3: icmp_seq=5 ttl=64 time=0.853 ms
64 bytes from 192.168.122.3: icmp_seq=6 ttl=64 time=0.785 ms
^C
--- 192.168.122.3 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5010ms
rtt min/avg/max/mdev = 0.785/0.844/0.894/0.036 ms
ubuntu@ubuntu-cloud:~$
```

6. Make sure your web server is accessible from the Admin VM.

```

root@Admin:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 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
8: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 1000
    link/ether 02:42:cc:32:98:00 brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.3/28 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::42:ccff:fe32:9800/64 scope link
        valid_lft forever preferred_lft forever
root@Admin:~# curl 192.168.122.2:80
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>
<p>For online documentation and support please refer to
<a href="http://nginx.org">nginx.org</a>.<br>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>
root@Admin:~#
```

Task 3 - Routing

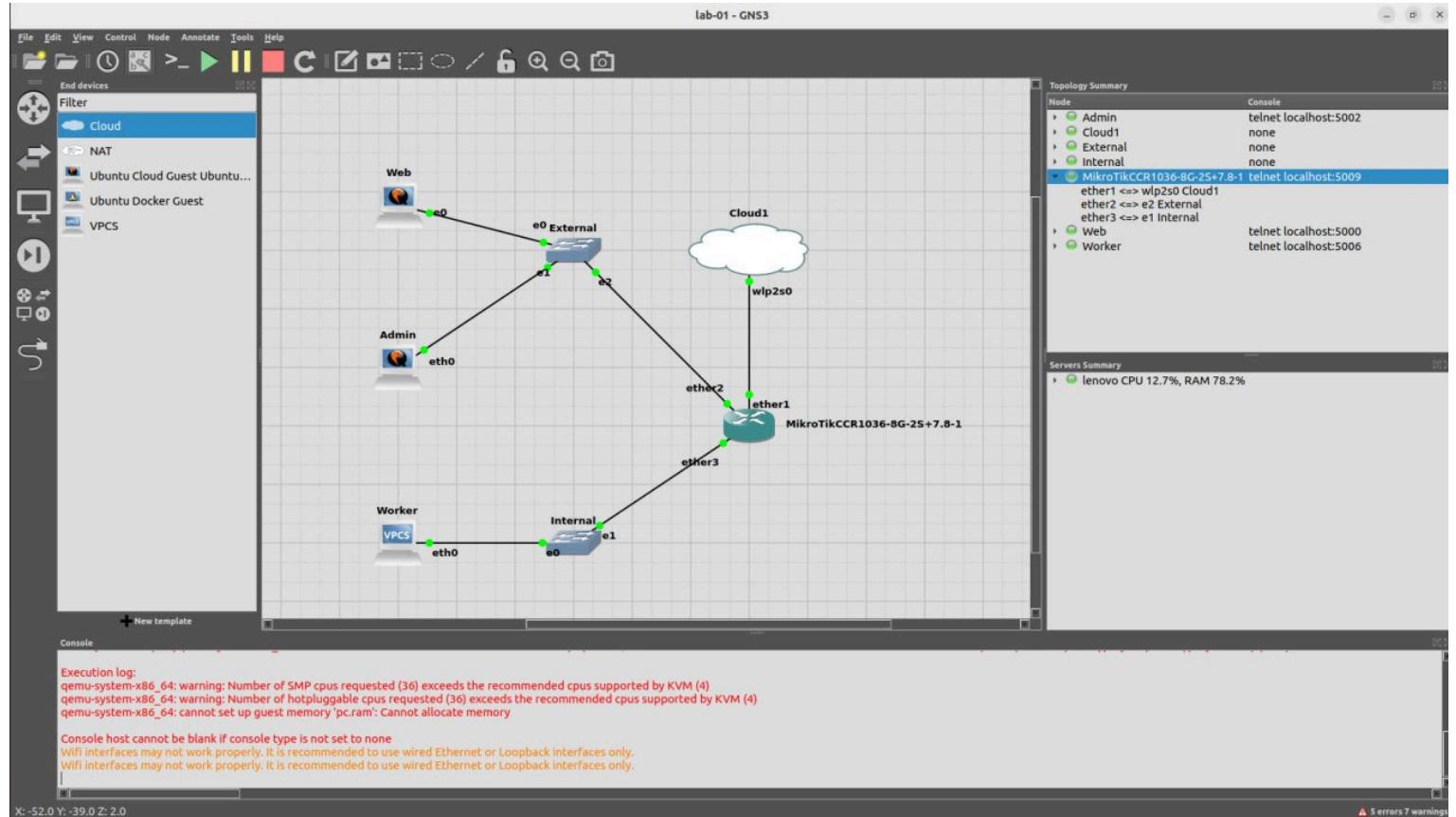
Now that you have a small network running, it is time to have it properly routed. Delete the NAT device because you will make your own gateway now.

1. Select a virtual Routing solution (Gateway) such as Mikrotik (recommended default choice), PfSense, VyOS, Untangle NG, OpenWrt, Cumulus VX.

I selected MikrotikCCR1038-8G-2S+7.8-1

2. Create Internal network for Worker instance

Created:



3. Connect your Gateway to the internet and to your workstation/host

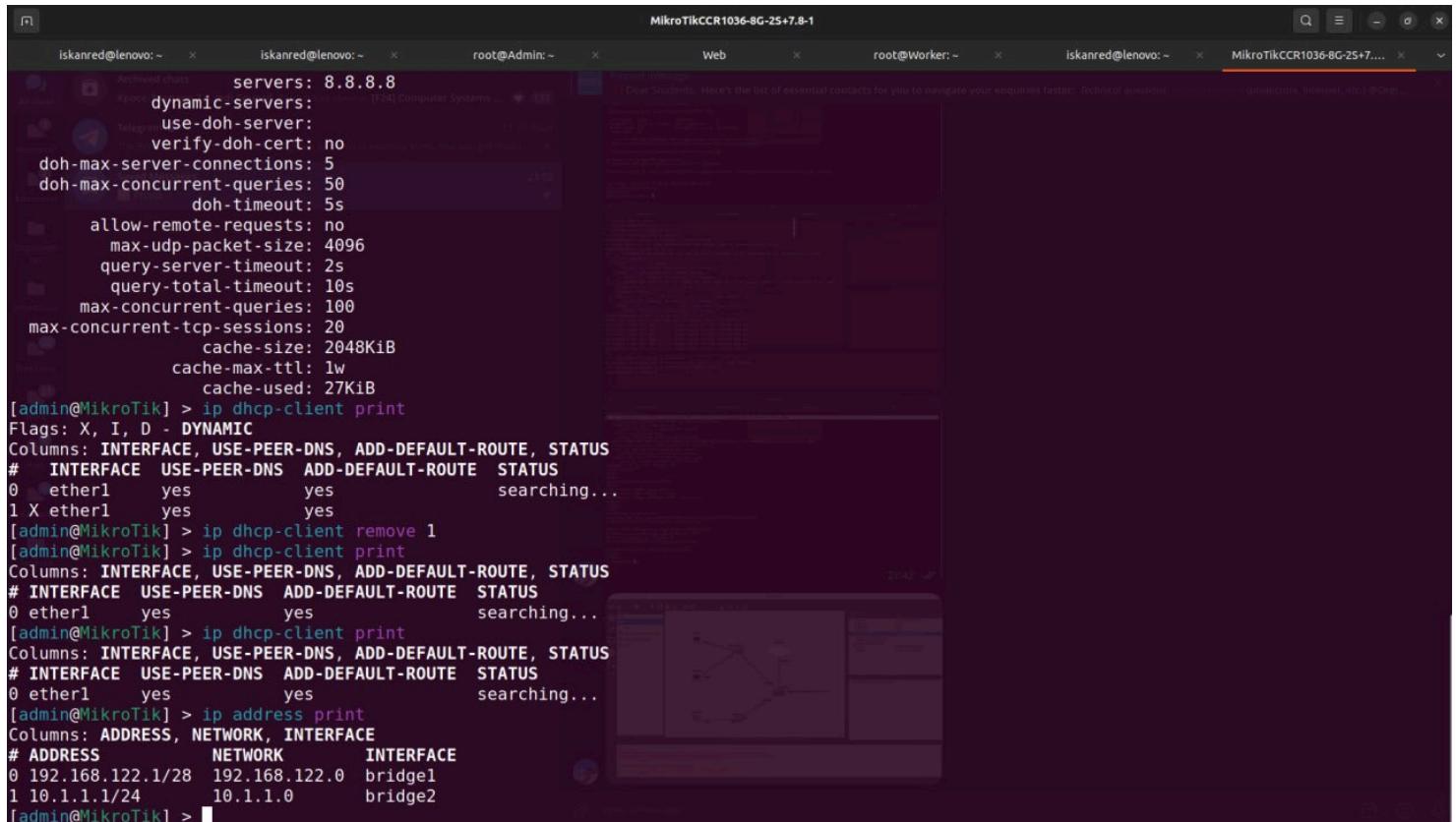
This task is failed because I have only WiFi adapter since my laptop has no right Ethernet adapter :(

With WiFi I couldn't connect to the internet

4. Setup the gateway for Admin, Web and Worker, then check their connectivity.

However, I were able to configure the gateway for every end device node:

- The router configuration:

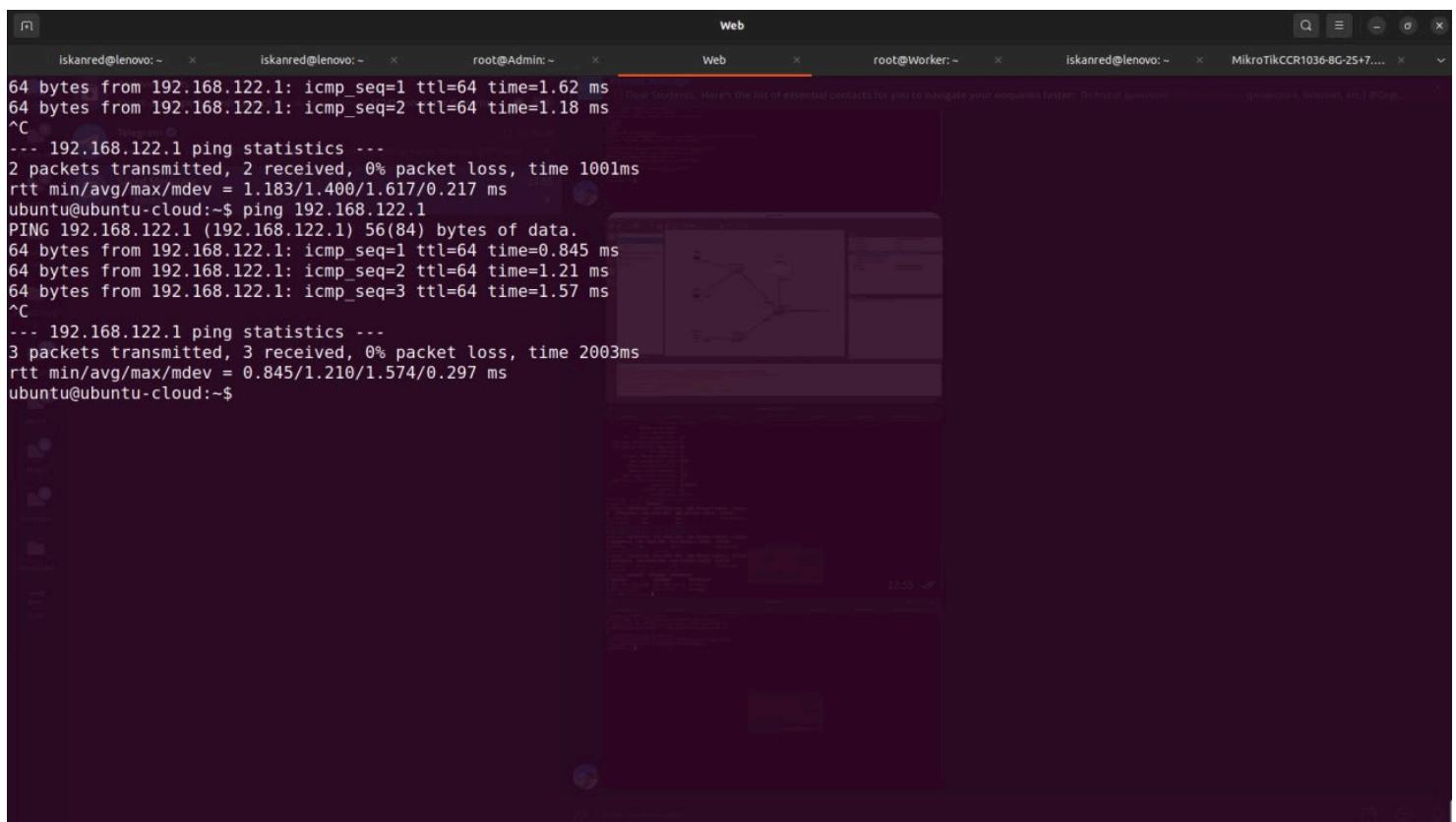


```

servers: 8.8.8.8
dynamic-servers:
use-doh-server:
verify-doh-cert: no
doh-max-server-connections: 5
doh-max-concurrent-queries: 50
doh-timeout: 5s
allow-remote-requests: no
max-udp-packet-size: 4096
query-server-timeout: 2s
query-total-timeout: 10s
max-concurrent-queries: 100
max-concurrent-tcp-sessions: 20
cache-size: 2048KiB
cache-max-ttl: 1w
cache-used: 27KiB
[admin@MikroTik] > ip dhcp-client print
Flags: X, I, D - DYNAMIC
Columns: INTERFACE, USE-PEER-DNS, ADD-DEFAULT-ROUTE, STATUS
# INTERFACE USE-PEER-DNS ADD-DEFAULT-ROUTE STATUS
0 ether1 yes yes searching...
1 X ether1 yes yes
[admin@MikroTik] > ip dhcp-client remove 1
[admin@MikroTik] > ip dhcp-client print
Columns: INTERFACE, USE-PEER-DNS, ADD-DEFAULT-ROUTE, STATUS
# INTERFACE USE-PEER-DNS ADD-DEFAULT-ROUTE STATUS
0 ether1 yes yes searching...
[admin@MikroTik] > ip dhcp-client print
Columns: INTERFACE, USE-PEER-DNS, ADD-DEFAULT-ROUTE, STATUS
# INTERFACE USE-PEER-DNS ADD-DEFAULT-ROUTE STATUS
0 ether1 yes yes searching...
[admin@MikroTik] > ip address print
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS NETWORK INTERFACE
0 192.168.122.1/28 192.168.122.0 bridgel
1 10.1.1.1/24 10.1.1.0 bridge2
[admin@MikroTik] >

```

- Web



```

64 bytes from 192.168.122.1: icmp_seq=1 ttl=64 time=1.62 ms
64 bytes from 192.168.122.1: icmp_seq=2 ttl=64 time=1.18 ms
^C
--- 192.168.122.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.183/1.400/1.617/0.217 ms
ubuntu@ubuntu-cloud:~$ ping 192.168.122.1
PING 192.168.122.1 (192.168.122.1) 56(84) bytes of data.
64 bytes from 192.168.122.1: icmp_seq=1 ttl=64 time=0.845 ms
64 bytes from 192.168.122.1: icmp_seq=2 ttl=64 time=1.21 ms
64 bytes from 192.168.122.1: icmp_seq=3 ttl=64 time=1.57 ms
^C
--- 192.168.122.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.845/1.210/1.574/0.297 ms
ubuntu@ubuntu-cloud:~$

```

- Admin

```
root@Admin:~# ping 192.168.122.1
PING 192.168.122.1 (192.168.122.1) 56(84) bytes of data.
64 bytes from 192.168.122.1: icmp_seq=1 ttl=64 time=0.858 ms
64 bytes from 192.168.122.1: icmp_seq=2 ttl=64 time=0.876 ms
^C
--- 192.168.122.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1057ms
rtt min/avg/max/mdev = 0.858/0.867/0.876/0.009 ms
root@Admin:~#
```

- Worker (its ip is 10.1.1.2/24)

```
root@Worker:~# ping 10.1.1.1
PING 10.1.1.1 (10.1.1.1) 56(84) bytes of data.
64 bytes from 10.1.1.1: icmp_seq=1 ttl=64 time=0.765 ms
64 bytes from 10.1.1.1: icmp_seq=2 ttl=64 time=0.858 ms
^C
--- 10.1.1.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1041ms
rtt min/avg/max/mdev = 0.765/0.811/0.858/0.046 ms
root@Worker:~#
```