Lab 7 Notes

Step 0

- Install GNS3 on your system with your package manager

```
sudo dnf install gns3-server gns3-gui
```

```
sudo add-apt-repository ppa:gns3/ppa
sudo apt update
sudo apt install gns3-gui gns3-**server**
```

```
sudo pacman -Syu yay
sudo yay -S qemu docker vpcs dynamips libvirt ubridge inetutils
sudo yay -S gns3-server gns3-gui
```

- Start working on the lab. Spend a half of an hour doing that
- Realize that lab requires some old version that none of the distros package anymore. Spend another half of an hour realizing that.
- Download correct version for a Windows VM. Spend another half of an hour waiting for Windows to install updates, download all dependencies, and install this program.
- Start working on the lab from scratch.

Step 1

- Create 3 computers (`PC1`, `PC2`, `PC3`), one switch, and 2 routers (`R1`, `R2`).
- Rename routers from `R1` and `R2` to `R2` and `R3` to not get confused while following the lab docs.
- Connect 2 computers into a switch, switch into a router, router into another router, and this final router into the third computer using `Link` menu.
- Turn everything on.
- Open console for `R2` and configure its `f0/0` interface to use ip `10.1.1.1/24` by typing the following commands

  ```
  config t
  interface f0/0
  ```

```
ip addr 10.1.1.1 255.255.255.0
no shut
exit
exit
```

- Check that it worked by typing `show ip interface brief`
- The output should look like this:

```
        R2> show ip interface brief

        Interface                  IP-Address      OK? Method Status
        FastEthernet0/0            10.1.1.1        YES manual up
        FastEthernet0/1            unassigned      YES unset  down
                ```
- Open console for `PC1` and assign its one and only interface to ip `10.1.1.2/24` to `R2` by typing
        ```Shell
        ip 10.1.1.2/24
        ip 10.1.1.2/24 10.1.1.1
        ```
- Run `show ip` to check if it worked. The output should look like this
        ```Shell
        PC1> show ip

        NAME        : PC1[1]
        IP/MASK     : 10.1.1.2/24
        GATEWAY     : 10.1.1.1
        DNS         :
        MAC         : 00:50:79:66:68:00
        LPORT       : 10018
        RHOST:PORT  : 127.0.0.1:10019
        MTU:        : 1500
        ```
- Do the same for `PC2`, but change ip to `10.1.1.3/24`
- Run `show ip` to check if it worked. The output should look like this
        ```Shell
        PC2> show ip

        NAME        : PC2[1]
        IP/MASK     : 10.1.1.3/24
        GATEWAY     : 10.1.1.1
        DNS         :
        MAC         : 00:50:79:66:68:01
        LPORT       : 10020
```

```
        RHOST:PORT  : 127.0.0.1:10021
        MTU:        : 1500
        ```
- Map interface `f0/1` on `R2` to `192.1.1.1 255.255.255.0`
  ```Shell
        config t
        interface f0/1
        ip addr 192.1.1.1 255.255.255.0
        no shut
        exit
        exit
```

- Run `show ip interface brief` to check if it worked. Output should be:

```
R2> show ip interface brief
Interface                  IP-Address      OK? Method Status
FastEthernet0/0            10.1.1.1        YES manual up
FastEthernet0/1            192.1.1.1       YES manual up
```

- Assign interface `f0/0` in `R3` to `192.1.1.2 255.255.255.0`, and interface `f0/1` to `10.2.1.1 255.255.255.0` using the same commands. Run `shou ip interface brief` to check if it worked. Output should be:

```
R3> show ip interface brief
Interface                  IP-Address      OK? Method Status
FastEthernet0/0            192.1.1.2       YES manual up
FastEthernet0/1            10.2.1.1        YES manual up
```

- Open console for `PC3` and assign its one and only interface to ip `10.2.1.2/24` to `R2` by typing:

```
ip 10.2.1.2/24
ip 10.2.1.2/24 10.2.1.1
```

- Run `show ip` to check if it worked. The output should look like this

```
PC3> show ip

NAME        : PC3[1]
IP/MASK     : 10.2.1.2/24
GATEWAY     : 10.2.1.1
DNS         :
```

```
MAC          : 00:50:79:66:68:02
LPORT        : 10022
RHOST:PORT   : 127.0.0.1:10023
MTU:         : 1500
```

- Now you can use `PC1` console and try to ping ip addresses of `PC2` and `PC3`. You should be able to ping `PC2` because it's in the same local network. However, you won'd be able to ping `PC3`. You can also try to ping `R2` and `R3` and see that `PC1` is only able to ping up to `R2`.

Step 2

- To enable communication between `PC1` and `PC3`/`R3` we need to set up routing on `R2` to forward traffic to `R3`. We can do that by typing

```
config t
ip route 0.0.0.0 0.0.0.0 192.1.1.2
```

- `R3` in its turn also should set up routing with

```
config t
ip route 0.0.0.0 0.0.0.0 192.1.1.1
```

- Now that the routing is established `PC1` and `PC3` can communicate

Step 3

- Indeed they are! `PC3` can now ping both `PC1` and `PC2`, and both `PC1` and `PC2` can ping `PC3`

Step 4

- Now for some reason configure interface `f0/0` on `R2` as a NAT by typing:

```
config t
interface f0/0
ip nat inside
interface f0/1
ip nat outside
```

Step 5

- Create range of addresses that NAT will transmit over to `f0/1` with:

```
exit
access-list 10 permit 10.1.1.0 0.0.0.255
ip nat inside source list 10 interface f0/1 overload
```

Step 6

- Check the cool NAT table on `R2` by enabling debugging with `debug ip nat` command.
- Start pinging different devices from `PC1`, `PC2`, or `PC3`. The NAT table on `R2` will look something like this:

```
R2>debug ip nat
IP NAT debugging is on
*Mar  1 00:23:50.051: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [46460]
*Mar  1 00:23:52.047: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [46461]
*Mar  1 00:23:53.127: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [46460]
*Mar  1 00:23:53.135: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [46461]
*Mar  1 00:23:54.075: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [46462]
*Mar  1 00:23:54.111: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [46462]
*Mar  1 00:23:55.159: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [46463]
*Mar  1 00:23:55.203: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [46463]
*Mar  1 00:23:56.247: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [46464]
*Mar  1 00:23:56.291: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [46464]
*Mar  1 00:24:06.107: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.1 [46476]
*Mar  1 00:24:06.139: NAT*: s=10.2.1.1, d=192.1.1.1->10.1.1.2 [46476]
*Mar  1 00:24:07.175: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.1 [46477]
*Mar  1 00:24:07.207: NAT*: s=10.2.1.1, d=192.1.1.1->10.1.1.2 [46477]
*Mar  1 00:24:08.251: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.1 [46478]
*Mar  1 00:24:08.279: NAT*: s=10.2.1.1, d=192.1.1.1->10.1.1.2 [46478]
*Mar  1 00:24:09.315: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.1 [46479]
*Mar  1 00:24:09.343: NAT*: s=10.2.1.1, d=192.1.1.1->10.1.1.2 [46479]
*Mar  1 00:24:10.379: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.1 [46480]
*Mar  1 00:24:10.415: NAT*: s=10.2.1.1, d=192.1.1.1->10.1.1.2 [46480]
*Mar  1 00:24:13.411: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [46483]
*Mar  1 00:24:13.455: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [46483]
*Mar  1 00:24:14.499: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [46484]
*Mar  1 00:24:14.547: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [46484]
*Mar  1 00:24:15.583: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [46485]
```

```
*Mar  1 00:24:15.631: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [46485]
*Mar  1 00:24:16.675: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [46486]
*Mar  1 00:24:16.723: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [46486]
*Mar  1 00:24:17.763: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [46487]
*Mar  1 00:24:17.811: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [46487]
```

- The final configurations (results of `show ip` and `show ip interface brief`) should look like this:

```
PC1> show ip
NAME        : PC1[1]
IP/MASK     : 10.1.1.2/24
GATEWAY     : 10.1.1.1
DNS         :
MAC         : 00:50:79:66:68:00
LPORT       : 10018
RHOST:PORT  : 127.0.0.1:10019
MTU:        : 1500
        ```
```Shell
PC2> show ip
NAME        : PC2[1]
IP/MASK     : 10.1.1.3/24
GATEWAY     : 10.1.1.1
DNS         :
MAC         : 00:50:79:66:68:01
LPORT       : 10020
RHOST:PORT  : 127.0.0.1:10021
MTU:        : 1500
```

```
PC3> show ip
NAME        : PC3[1]
IP/MASK     : 10.2.1.2/24
GATEWAY     : 10.2.1.1
DNS         :
MAC         : 00:50:79:66:68:02
LPORT       : 10022
RHOST:PORT  : 127.0.0.1:10023
MTU:        : 1500
```

```
R2> show ip interface brief
Interface                  IP-Address      OK? Method Status
FastEthernet0/0            10.1.1.1        YES manual up
```

```
FastEthernet0/1            192.1.1.1       YES manual up
NVI0                       unassigned      NO  unset  up
```

```
R3> show ip interface brief
Interface                  IP-Address      OK? Method Status
FastEthernet0/0            192.1.1.2       YES manual up
FastEthernet0/1            10.2.1.1        YES manual up
```

Step 7

- Demo your work to the TA, write up a step-by-step guide on how to do what you just did, and submit it to camino.