Assignment 2

IP Interfaces: Part 1

By Ziming Song

[zs2815@nyu.edu](mailto:zs2815@nyu.edu)

# **Part 0: Warm Up**

## **What is the slash notation representation of 255.255.255.0?**

/24

## **What is the dot-decimal representation of /30?**

255.255.255.252

## **What is the smallest subnet size that would accommodate 5 hosts?**

a subnet with at least 3 bits.

Need at least 7 addresses (5 hosts + 1 network + 1 broadcast). 2^N >= 7. N=3

## **Fill in the blank cells in the table below (R1 eth0 filled in by Vital system)**

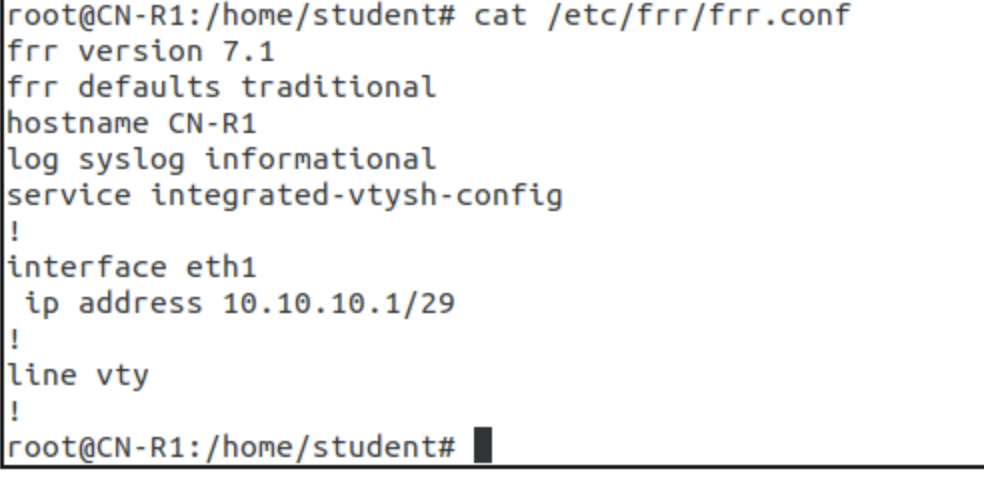
Here is one solution.

|  |  |
| --- | --- |
| VM (interface) | IP Address (CIDR Notation) |
| R1 (eth1) | 10.10.10.1/29 |
| R2 (eth0) | 10.10.10.2/29 |
| Kali (eth0) | 10.10.10.3/29 |

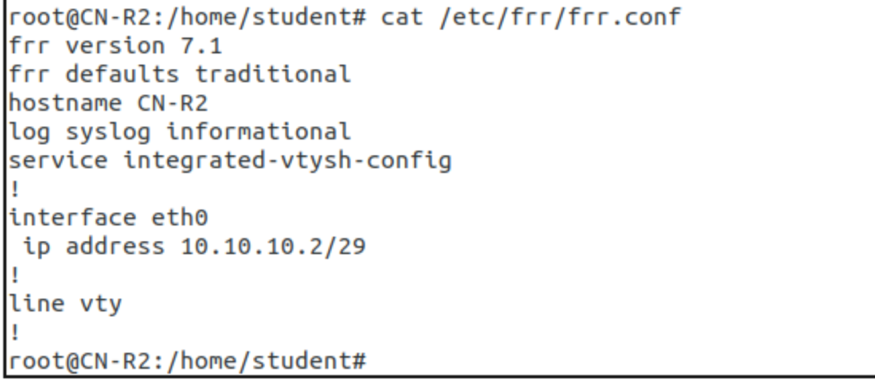
# **Part 1:** **Configuring Network Interfaces**

After configuration

R1 Screenshot of the .conf file under /etc/frr/frr.conf

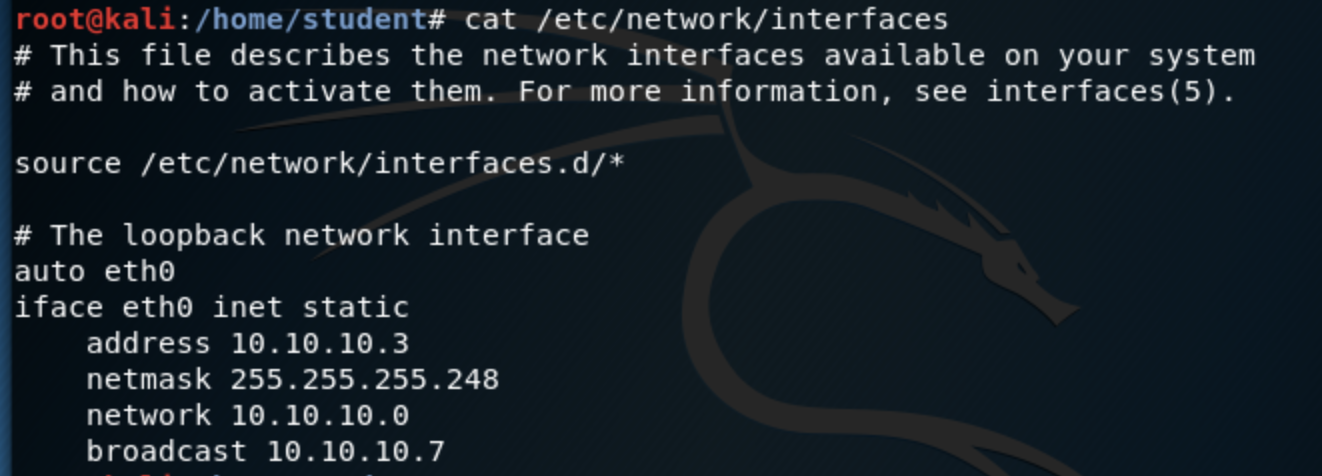


R2 Screenshot of the .conf file under /etc/frr/frr.conf



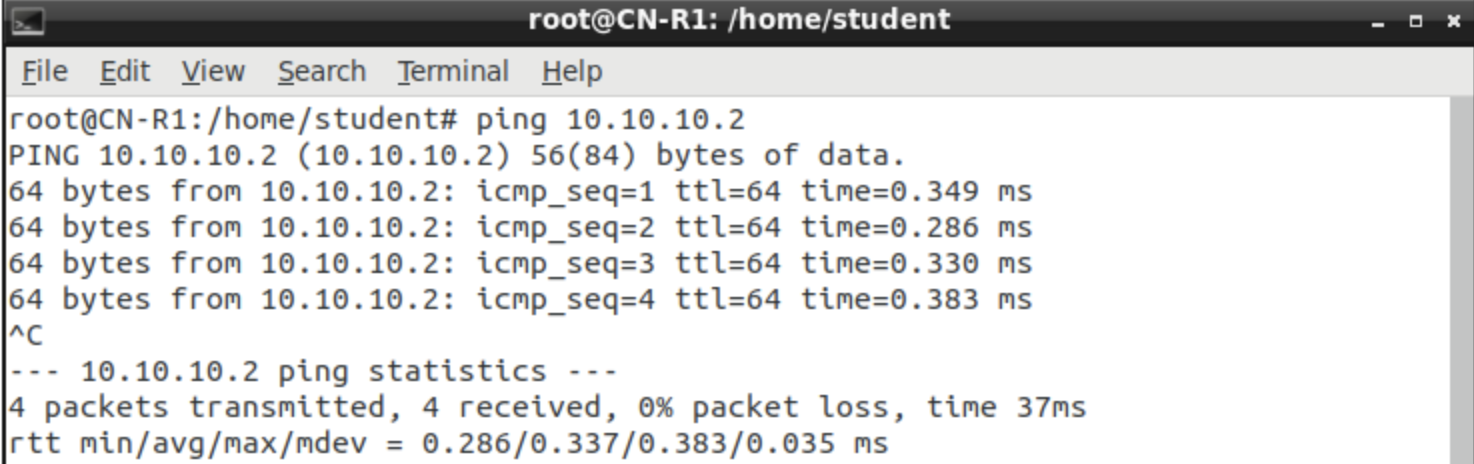
# **Part 2:** **Configuring Kali**

Screenshot of the /etc/network/interfaces file in Kali

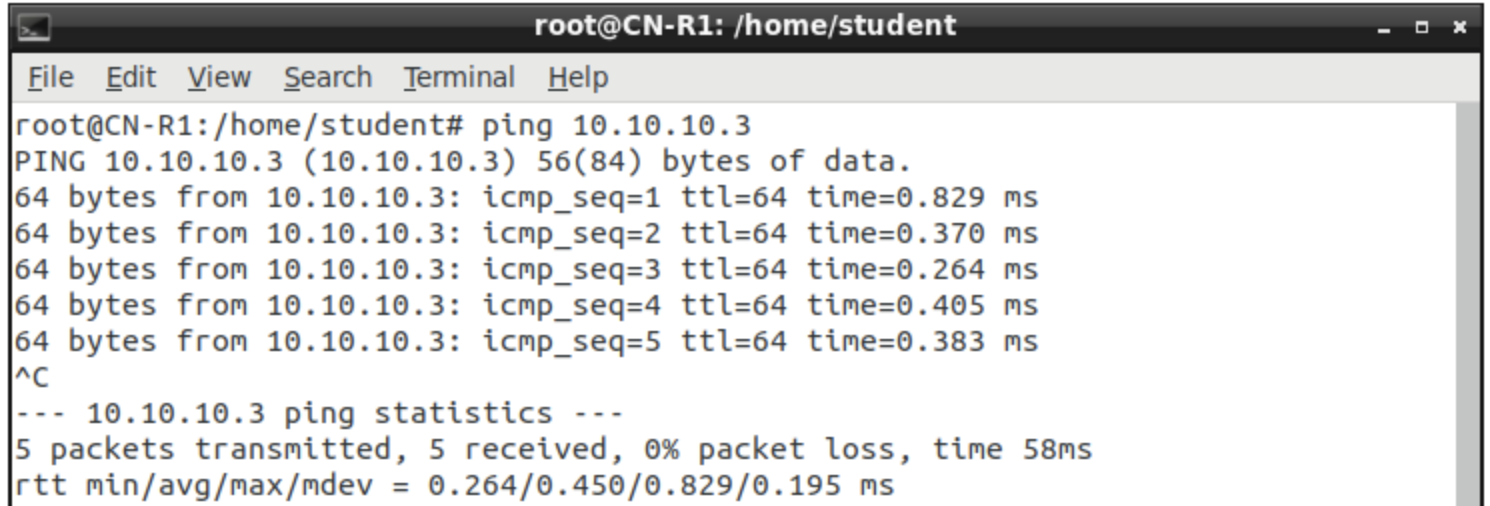


Screenshot showing that pinging works between R1, R2, and Kali

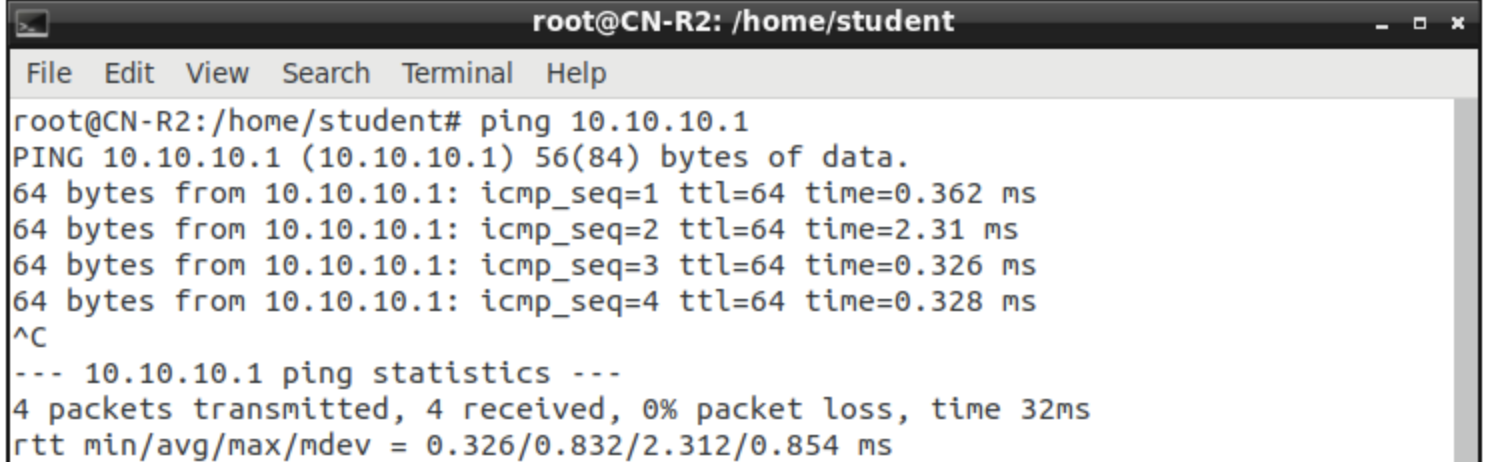
R1 ping R2



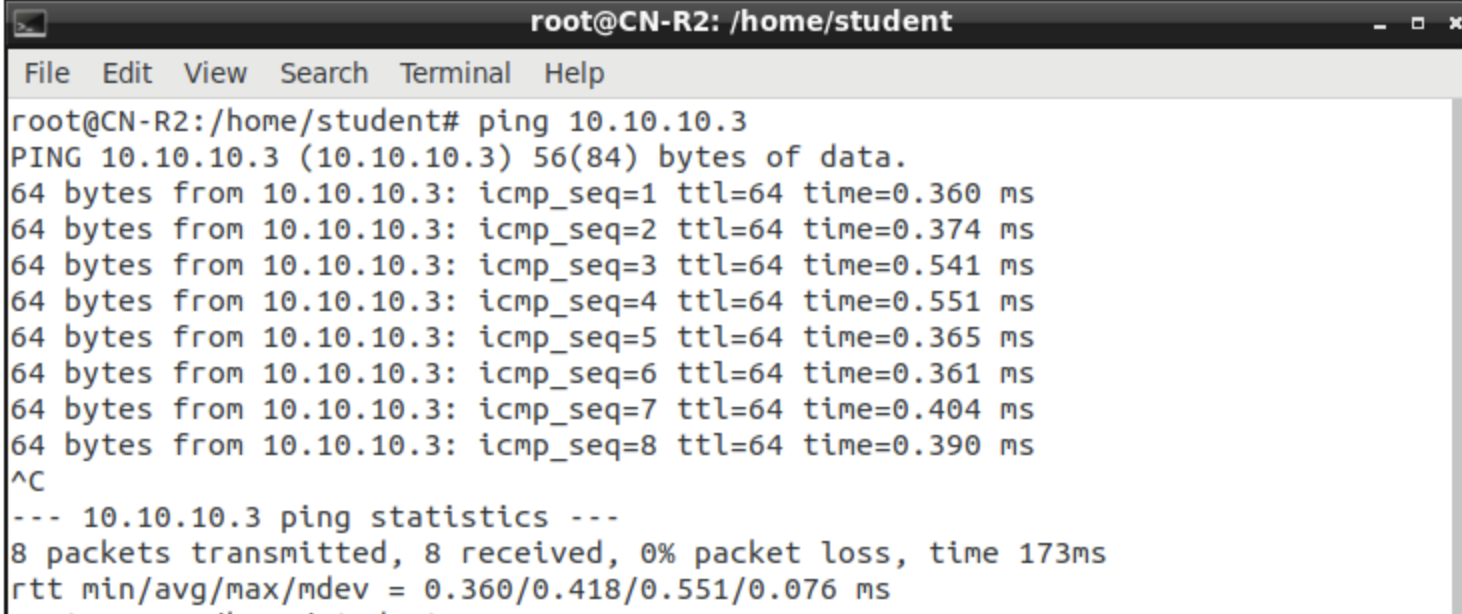
R1 ping Kali



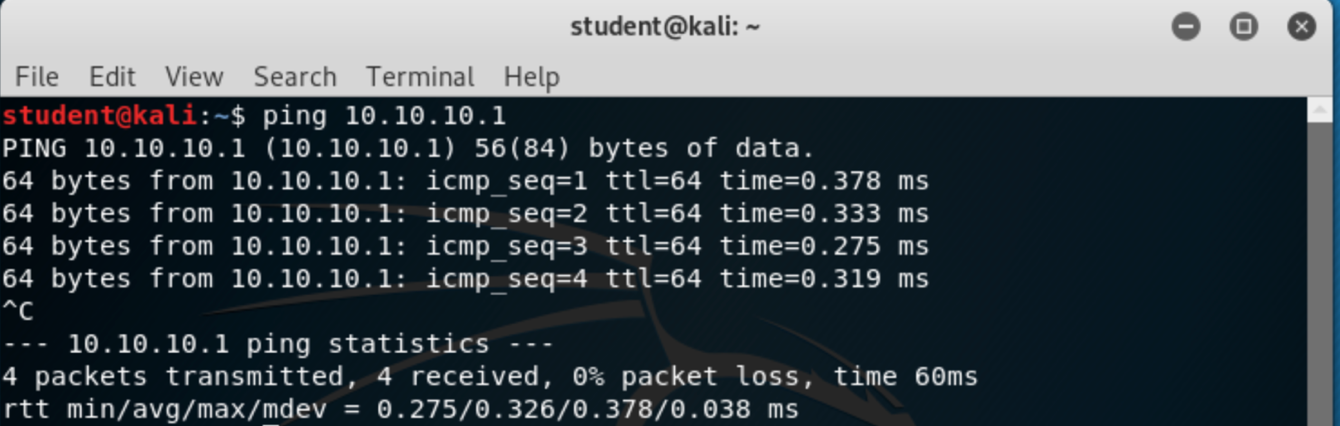
R2 ping R1

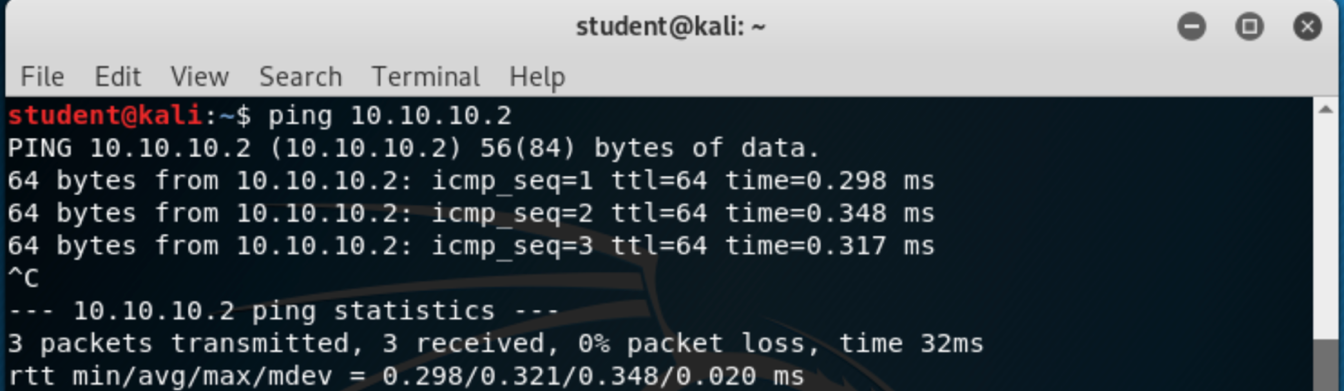
****

R2 ping Kali

****

Kali ping R1



Kali ping R2

# **Part 3: Questions**

1. **Why did we choose the /29 subnet mask for Area 0?**

Need at least 5 addresses (3 hosts + 1 network + 1 broadcast). 2^N >= 5. N=3

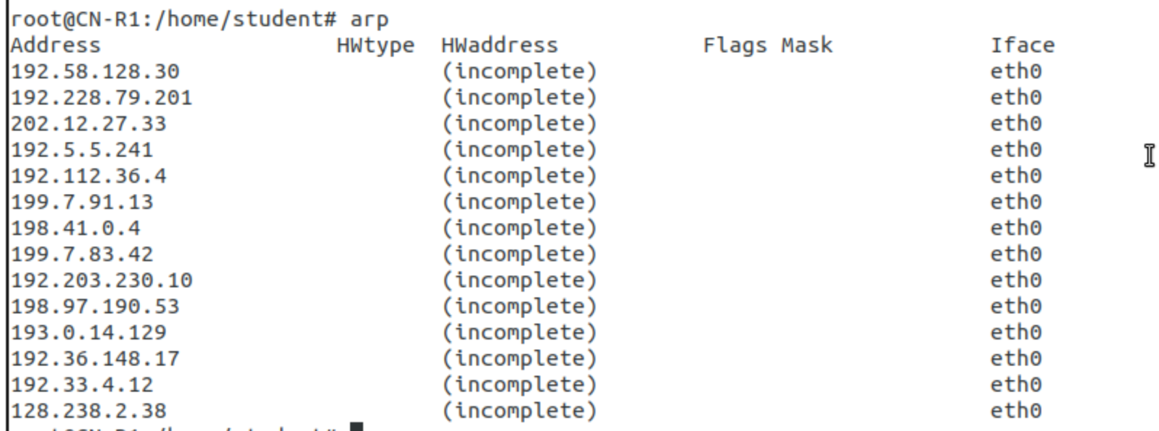
The subnet with at least 3 bits. Address length is 32 bit.

32 – 3 =29. So we choose the /29 subnet mask

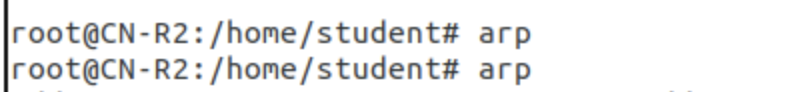
1. **The Linux arp (see man arp) command will print the current entries in the machine's address resolution protocol table. Now that you have configured Area 0, what entries are currently in R1, R2, and Kali?**

Screenshot of the ARP tables on R1, R2, and Kali, before pinging works between R1, R2, and Kali. The entries are unknow.

R1

****

R2 nothing

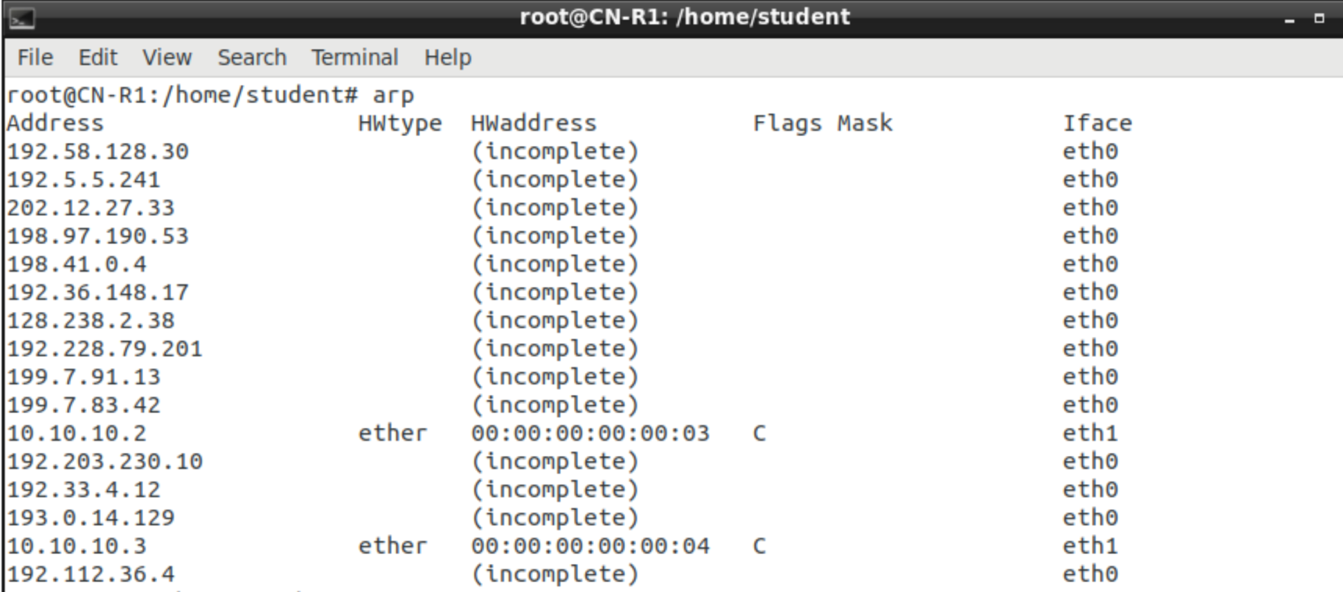


Kali nothing

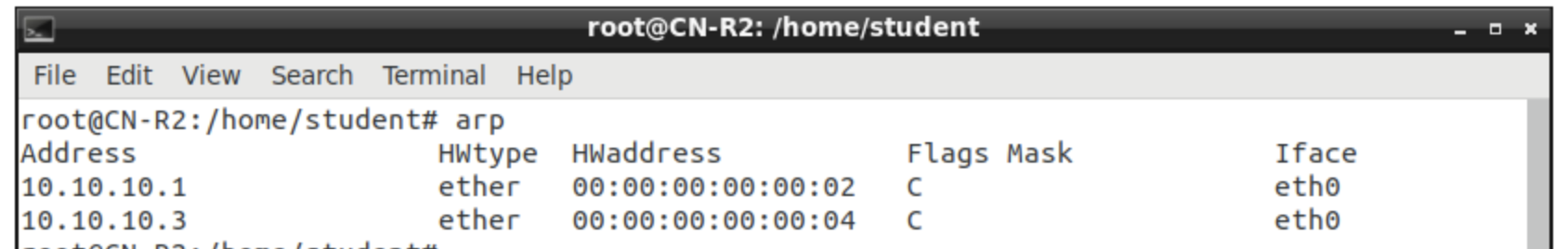


Screenshot of the ARP tables on R1, R2, and Kali, after pinging works between R1, R2, and Kali

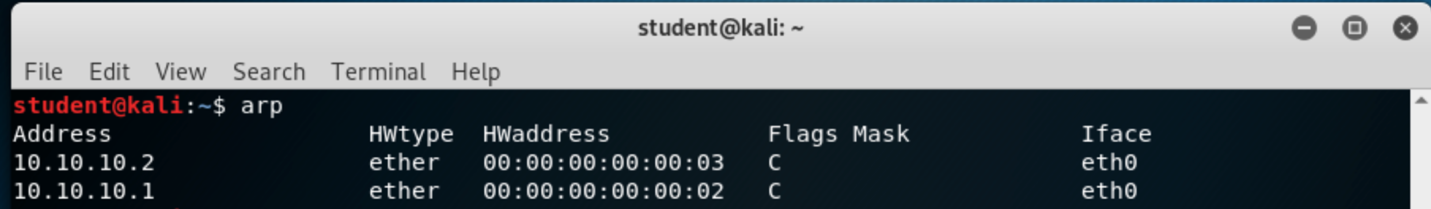
ARP table on R1:



ARP table on R2



ARP table on Kali

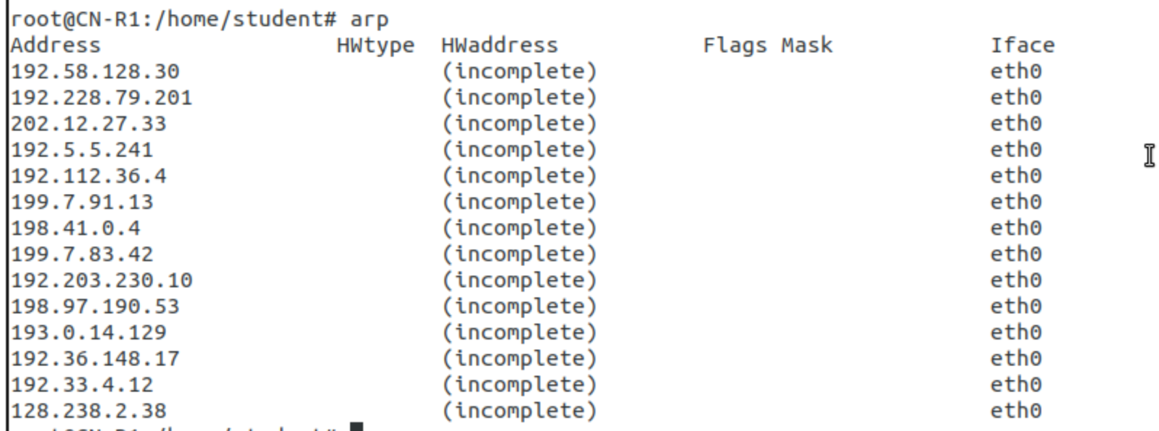


The entries are as followed.

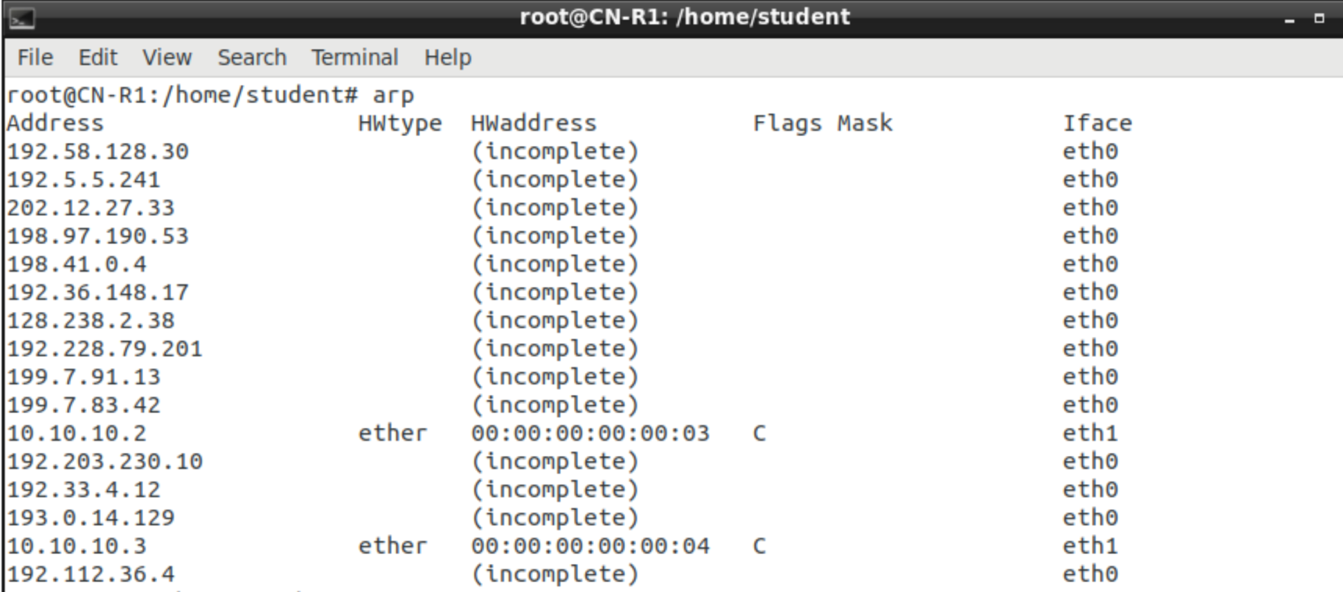
|  |  |
| --- | --- |
| R1 entry | 00:00:00:00:00:02 |
| R2 entry | 00:00:00:00:00:03 |
| Kali entry | 00:00:00:00:00:04 |

1. **Now ping both R2 and Kali from R1. Note the changes on each machine's ARP tables. At this point, R2 should be aware of R1, but why doesn't R2 have a table entry for Kali? (10 points)**

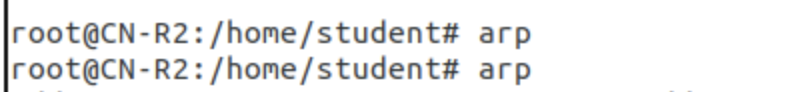
Before ping both R2 and Kali from R1, R1 ARP screenshot:

****

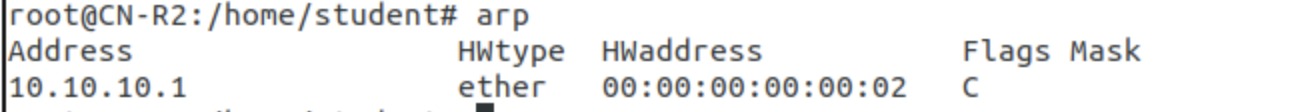
After ping both R2 and Kali from R1, R1 ARP screenshot:



Before ping both R2 and Kali from R1, R2 ARP screenshot:



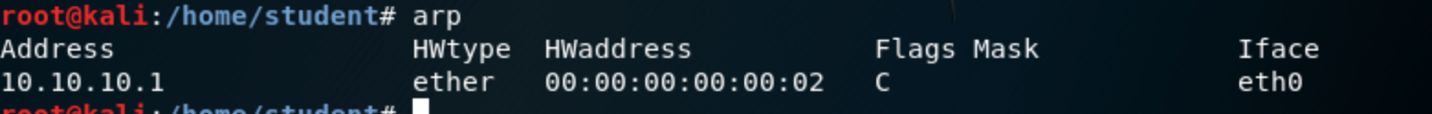
After ping both R2 and Kali from R1, R2 ARP screenshot. R2 know R1’s entry



Before ping both R2 and Kali from R1, R3 ARP screenshot:



After ping both R2 and Kali from R1, R3 ARP screenshot. Kali know R1’s entry.



Because R2 and Kali have not ping(communicate with) one to another directly.