Name : Suhan Ashraf
Reg No : 210905014
Date : 31-08-2023

CN Lab 5
Computer Network Design using HUB in GNS3

Q1. Design network configuration shown in Figure 5.29 for all parts. Connect all four VMs to a single Ethernet segment via a single hub as shown in Figure 5.29. Configure the IP addresses for the PCs as shown in Table 6.1.

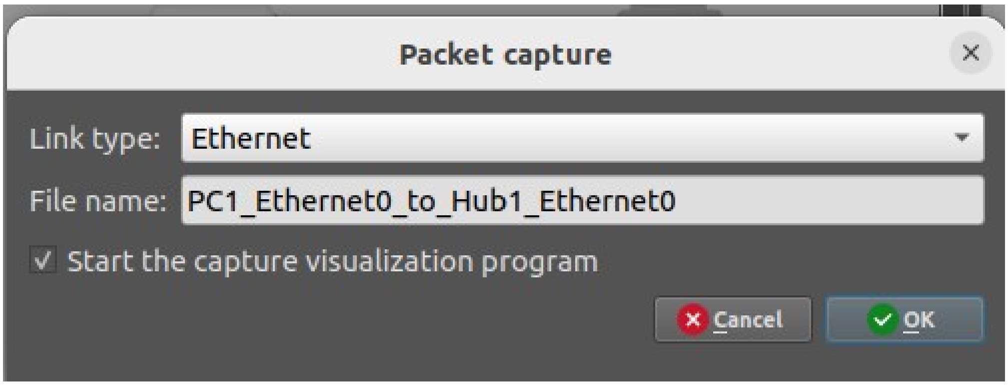
Table 5.1: IP Address of PCs

a. On PC1, view the ARP cache with show arp

```
PC1> show arp

arp table is empty
```

b. Start Wireshark on PC1-Hub1 link with a capture filter set to the IP address of PC2.



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	Private_66:68:02	Broadcast	ARP	64	Who has 10.0.1.11? Tell 10.0.1.13
2	0.000279	Private_66:68:00	Private_66:68:02	ARP	64	10.0.1.11 is at 00:50:79:66:68:00
3	0.000984	10.0.1.13	10.0.1.11	ICMP	98	Echo (ping) request id=0x0365, seq=1/256, ttl=64 (reply in 4)
4	0.001162	10.0.1.11	10.0.1.13	ICMP	98	Echo (ping) reply id=0x0365, seq=1/256, ttl=64 (request in 3)
5	1.002239	10.0.1.13	10.0.1.11	ICMP	98	Echo (ping) request id=0x0465, seq=2/512, ttl=64 (reply in 6)
6	1.002484	10.0.1.11	10.0.1.13	ICMP	98	Echo (ping) reply id=0x0465, seq=2/512, ttl=64 (request in 5)
7	2.003488	10.0.1.13	10.0.1.11	ICMP	98	Echo (ping) request id=0x0565, seq=3/768, ttl=64 (reply in 8)
8	2.003720	10.0.1.11	10.0.1.13	ICMP	98	Echo (ping) reply id=0x0565, seq=3/768, ttl=64 (request in 7)
9	3.004664	10.0.1.13	10.0.1.11	ICMP	98	Echo (ping) request id=0x0665, seq=4/1024, ttl=64 (reply in 10)
10	3.004911	10.0.1.11	10.0.1.13	ICMP	98	Echo (ping) reply id=0x0665, seq=4/1024, ttl=64 (request in 9)
11	4.005935	10.0.1.13	10.0.1.11	ICMP	98	Echo (ping) request id=0x0765, seq=5/1280, ttl=64 (reply in 12)
12	4.006088	10.0.1.11	10.0.1.13	ICMP	98	Echo (ping) reply id=0x0765, seq=5/1280, ttl=64 (request in 11)

c. Issue a ping command from PC1 to PC2:
PC1% ping 10.0.1.13 –c 3

```
PC1> ping 10.0.1.12 -c 3

84 bytes from 10.0.1.12 icmp_seq=1 ttl=64 time=0.240 ms
84 bytes from 10.0.1.12 icmp_seq=2 ttl=64 time=0.304 ms
84 bytes from 10.0.1.12 icmp_seq=3 ttl=64 time=0.464 ms
```

d. View the ARP cache again with the command arp -a. Note that ARP cache entries can get refreshed/deleted fairly quickly (~2 minutes).

show arp

```
PC1> show arp

arp table is empty
```

e. Save the results of Wireshark.

Q2. To observe the effects of having more than one host with the same (duplicate) IP address in a network.
After completing Exercise 1, the IP addresses of the Ethernet interfaces on the four PCs are as shown in Table 6.2 below. Note that PC1 and PC4 are assigned the same IP address.

a. Delete all entries in the ARP cache on all Pcs.

```
PC4> ip 10.0.1.11/24 10.0.1.5
Checking for duplicate address...
10.0.1.11 is being used by MAC 00:50:79:66:68:00
Address not changed
```

b. Run Wireshark on PC3-Hub1 link and capture the network traffic to and from the duplicate IP address 10.0.1.11.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	Private_66:68:02	Broadcast	ARP	64	Who has 10.0.1.11? Tell 10.0.1.13
2	0.000279	Private_66:68:00	Private_66:68:02	ARP	64	10.0.1.11 is at 00:50:79:66:68:00
3	0.000984	10.0.1.13	10.0.1.11	ICMP	98	Echo (ping) request id=0x0365, seq=1/256, ttl=64 (reply in 4)
4	0.001162	10.0.1.11	10.0.1.13	ICMP	98	Echo (ping) reply id=0x0365, seq=1/256, ttl=64 (request in 3)
5	1.002239	10.0.1.13	10.0.1.11	ICMP	98	Echo (ping) request id=0x0465, seq=2/512, ttl=64 (reply in 6)
6	1.002484	10.0.1.11	10.0.1.13	ICMP	98	Echo (ping) reply id=0x0465, seq=2/512, ttl=64 (request in 5)
7	2.003488	10.0.1.13	10.0.1.11	ICMP	98	Echo (ping) request id=0x0565, seq=3/768, ttl=64 (reply in 8)
8	2.003720	10.0.1.11	10.0.1.13	ICMP	98	Echo (ping) reply id=0x0565, seq=3/768, ttl=64 (request in 7)
9	3.004664	10.0.1.13	10.0.1.11	ICMP	98	Echo (ping) request id=0x0665, seq=4/1024, ttl=64 (reply in 10)
10	3.004911	10.0.1.11	10.0.1.13	ICMP	98	Echo (ping) reply id=0x0665, seq=4/1024, ttl=64 (request in 9)
11	4.005935	10.0.1.13	10.0.1.11	ICMP	98	Echo (ping) request id=0x0765, seq=5/1280, ttl=64 (reply in 12)
12	4.006088	10.0.1.11	10.0.1.13	ICMP	98	Echo (ping) reply id=0x0765, seq=5/1280, ttl=64 (request in 11)

c. From PC3, issue a ping command to the duplicate IP address, 10.0.1.11, by typing
PC3% ping 10.0.1.11 –c 5

```
PC3> ping 10.0.1.11 -c 5

84 bytes from 10.0.1.11 icmp_seq=1 ttl=64 time=0.319 ms
84 bytes from 10.0.1.11 icmp_seq=2 ttl=64 time=0.502 ms
84 bytes from 10.0.1.11 icmp_seq=3 ttl=64 time=0.274 ms
84 bytes from 10.0.1.11 icmp_seq=4 ttl=64 time=0.562 ms
84 bytes from 10.0.1.11 icmp_seq=5 ttl=64 time=0.331 ms
```

d. Stop Wireshark, save all ARP packets and screenshot the ARP cache of PC3 using the arp –a command:
PC3% arp – a

```
PC3> ping 10.0.1.11 -c 5

84 bytes from 10.0.1.11 icmp_seq=1 ttl=64 time=0.308 ms
^[[A^[[A84 bytes from 10.0.1.11 icmp_seq=2 ttl=64 time=0.655 ms
84 bytes from 10.0.1.11 icmp_seq=3 ttl=64 time=0.339 ms
84 bytes from 10.0.1.11 icmp_seq=4 ttl=64 time=0.497 ms
84 bytes from 10.0.1.11 icmp_seq=5 ttl=64 time=0.562 ms

PC3> arp -a

Invalid ID

PC3> arp -a -showall

00:50:79:66:68:00  10.0.1.11 expires in 67 seconds
```

Q3. To test the effects of changing the netmask of a network configuration.
a. Design the configuration as Exercise 1 and replace the hub with a switch, two hosts (PC2 and PC4) have been assigned different network prefixes.
Setup the interfaces of the hosts as follows:
VPCS IP Address of eth0 Network Mask
PC1 10.0.1.100/24 255.255.255.0
PC2 10.0.1.101/28 255.255.255.240
PC3 10.0.1.120/24 255.255.255.0
PC4 10.0.1.121/28 255.255.255.240

b. Run Wireshark on PC1-Hub1 link and capture the packets for the following scenarios

i. From PC1 ping PC3.

```
PC1> ping 10.0.1.120 -c 5

84 bytes from 10.0.1.120 icmp_seq=1 ttl=64 time=0.388 ms
84 bytes from 10.0.1.120 icmp_seq=2 ttl=64 time=0.522 ms
84 bytes from 10.0.1.120 icmp_seq=3 ttl=64 time=0.395 ms
84 bytes from 10.0.1.120 icmp_seq=4 ttl=64 time=0.355 ms
84 bytes from 10.0.1.120 icmp_seq=5 ttl=64 time=0.363 ms
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.0.1.100	10.0.1.120	ICMP	98	Echo (ping) request id=0x2267, seq=1/256, ttl=64 (reply in 2)
2	0.000211	10.0.1.120	10.0.1.100	ICMP	98	Echo (ping) reply id=0x2267, seq=1/256, ttl=64 (request in 1)
3	1.001150	10.0.1.100	10.0.1.120	ICMP	98	Echo (ping) request id=0x2367, seq=2/512, ttl=64 (reply in 4)
4	1.001440	10.0.1.120	10.0.1.100	ICMP	98	Echo (ping) reply id=0x2367, seq=2/512, ttl=64 (request in 3)
5	2.002382	10.0.1.100	10.0.1.120	ICMP	98	Echo (ping) request id=0x2467, seq=3/768, ttl=64 (reply in 6)
6	2.002572	10.0.1.120	10.0.1.100	ICMP	98	Echo (ping) reply id=0x2467, seq=3/768, ttl=64 (request in 5)
7	3.003639	10.0.1.100	10.0.1.120	ICMP	98	Echo (ping) request id=0x2567, seq=4/1024, ttl=64 (reply in 8)
8	3.003802	10.0.1.120	10.0.1.100	ICMP	98	Echo (ping) reply id=0x2567, seq=4/1024, ttl=64 (request in 7)
9	4.004881	10.0.1.100	10.0.1.120	ICMP	98	Echo (ping) request id=0x2667, seq=5/1280, ttl=64 (reply in 10)
10	4.005013	10.0.1.120	10.0.1.100	ICMP	98	Echo (ping) reply id=0x2667, seq=5/1280, ttl=64 (request in 9)

ii. From PC1 ping PC2.

```
PC1> ping 10.0.1.101 -c 5

84 bytes from 10.0.1.101 icmp_seq=1 ttl=64 time=0.143 ms
84 bytes from 10.0.1.101 icmp_seq=2 ttl=64 time=0.503 ms
84 bytes from 10.0.1.101 icmp_seq=3 ttl=64 time=0.396 ms
84 bytes from 10.0.1.101 icmp_seq=4 ttl=64 time=0.555 ms
84 bytes from 10.0.1.101 icmp_seq=5 ttl=64 time=0.670 ms
```

11	141.723887	Private_66:68:02	Broadcast	ARP	64	Who has 10.0.1.101? Tell 10.0.1.100
12	141.724023	Private_66:68:01	Private_66:68:02	ARP	64	10.0.1.101 is at 00:50:79:66:68:01
13	141.724942	10.0.1.100	10.0.1.101	ICMP	98	Echo (ping) request id=0xb067, seq=1/256, ttl=64 (reply in 14)
14	141.725024	10.0.1.101	10.0.1.100	ICMP	98	Echo (ping) reply id=0xb067, seq=1/256, ttl=64 (request in 13)
15	142.726262	10.0.1.100	10.0.1.101	ICMP	98	Echo (ping) request id=0xb167, seq=2/512, ttl=64 (reply in 16)
16	142.726550	10.0.1.101	10.0.1.100	ICMP	98	Echo (ping) reply id=0xb167, seq=2/512, ttl=64 (request in 15)
17	143.727600	10.0.1.100	10.0.1.101	ICMP	98	Echo (ping) request id=0xb267, seq=3/768, ttl=64 (reply in 18)
18	143.727762	10.0.1.101	10.0.1.100	ICMP	98	Echo (ping) reply id=0xb267, seq=3/768, ttl=64 (request in 17)
19	144.728846	10.0.1.100	10.0.1.101	ICMP	98	Echo (ping) request id=0xb367, seq=4/1024, ttl=64 (reply in 20)
20	144.729149	10.0.1.101	10.0.1.100	ICMP	98	Echo (ping) reply id=0xb367, seq=4/1024, ttl=64 (request in 19)
21	145.730288	10.0.1.100	10.0.1.101	ICMP	98	Echo (ping) request id=0xb467, seq=5/1280, ttl=64 (reply in 22)
22	145.730629	10.0.1.101	10.0.1.100	ICMP	98	Echo (ping) reply id=0xb467, seq=5/1280, ttl=64 (request in 21)

iii. From PC1 ping PC4.

```
PC1> ping 10.0.1.121 -c 5

10.0.1.121 icmp_seq=1 timeout
10.0.1.121 icmp_seq=2 timeout
10.0.1.121 icmp_seq=3 timeout
10.0.1.121 icmp_seq=4 timeout
10.0.1.121 icmp_seq=5 timeout
```

23	190.627990	Private_66:68:02	Broadcast	ARP	64 Who has 10.0.1.121? Tell 10.0.1.100
24	190.628310	Private_66:68:00	Private_66:68:02	ARP	64 10.0.1.121 is at 00:50:79:66:68:00
25	190.629165	10.0.1.100	10.0.1.121	ICMP	98 Echo (ping) request id=0xe167, seq=1/256, ttl=64 (reply in 30)
26	190.629492	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
27	191.630290	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
28	192.629313	10.0.1.100	10.0.1.121	ICMP	98 Echo (ping) request id=0xe367, seq=2/512, ttl=64 (reply in 36)
29	192.630465	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
30	193.630951	10.0.1.121	10.0.1.100	ICMP	98 Echo (ping) reply id=0xe167, seq=1/256, ttl=64 (request in 25)
31	193.631025	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
32	194.629932	10.0.1.100	10.0.1.121	ICMP	98 Echo (ping) request id=0xe567, seq=3/768, ttl=64 (reply in 41)
33	194.631325	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
34	195.631403	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
35	196.630436	10.0.1.100	10.0.1.121	ICMP	98 Echo (ping) request id=0xe767, seq=4/1024, ttl=64 (reply in 45)
36	196.631957	10.0.1.121	10.0.1.100	ICMP	98 Echo (ping) reply id=0xe367, seq=2/512, ttl=64 (request in 28)
37	196.632052	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
38	197.632766	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
39	198.630670	10.0.1.100	10.0.1.121	ICMP	98 Echo (ping) request id=0xe967, seq=5/1280, ttl=64 (reply in 49)
40	198.632963	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
41	199.633784	10.0.1.121	10.0.1.100	ICMP	98 Echo (ping) reply id=0xe567, seq=3/768, ttl=64 (request in 32)
42	199.633817	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
43	200.634276	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
44	201.634867	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
45	202.635663	10.0.1.121	10.0.1.100	ICMP	98 Echo (ping) reply id=0xe767, seq=4/1024, ttl=64 (request in 35)
46	202.635701	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
47	203.635796	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
48	204.636587	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
49	205.637519	10.0.1.121	10.0.1.100	ICMP	98 Echo (ping) reply id=0xe967, seq=5/1280, ttl=64 (request in 39)

iv. From PC4 ping PC1.

```
PC4> ping 10.0.1.100 -c 5

host (255.255.255.240) not reachable
```

50	273.843940	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
51	274.845137	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121
52	275.845739	Private_66:68:00	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.121

v. From PC2 ping PC4.

```
PC2> ping 10.0.1.121 -c 5

host (255.255.255.240) not reachable
```

53	410.607833	Private_66:68:01	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.101
54	411.607955	Private_66:68:01	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.101
55	412.607967	Private_66:68:01	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.101

vi. From PC2 ping PC3.

```
PC2> ping 10.0.1.120 -c 5

host (255.255.255.240) not reachable
```

56	524.079929	Private_66:68:01	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.101
57	525.080410	Private_66:68:01	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.101
58	526.081433	Private_66:68:01	Broadcast	ARP	64 Who has 255.255.255.240? Tell 10.0.1.101

c. Save the Wireshark output to a text file (using the “Packet Summary” option from “Print”) , and save the output of the ping commands. Note that not all of the above scenarios are successful. Save all the output including any error messages.

```
/tmp/wireshark_-NLVYA2.pcapng 58 total packets, 58 shown

No.     Time           Source                Destination           Protocol Length Info
      1 0.000000       10.0.1.100            10.0.1.120            ICMP     98     Echo (ping) request
id=0x2267, seq=1/256, ttl=64 (reply in 2)
Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface -, id 0
    Interface id: 0 (-)
        Interface name: -
    Encapsulation type: Ethernet (1)
    Arrival Time: Aug 31, 2023 15:40:42.726596000 IST
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1693476642.726596000 seconds
    [Time delta from previous captured frame: 0.000000000 seconds]
    [Time delta from previous displayed frame: 0.000000000 seconds]
    [Time since reference or first frame: 0.000000000 seconds]
    Frame Number: 1
    Frame Length: 98 bytes (784 bits)
    Capture Length: 98 bytes (784 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: eth:ethertype:ip:icmp:data]
    [Coloring Rule Name: ICMP]
    [Coloring Rule String: icmp || icmpv6]
Ethernet II, Src: Private_66:68:02 (00:50:79:66:68:02), Dst: Private_66:68:03 (00:50:79:66:68:03)
    Destination: Private_66:68:03 (00:50:79:66:68:03)
        Address: Private_66:68:03 (00:50:79:66:68:03)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
        .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
    Source: Private_66:68:02 (00:50:79:66:68:02)
        Address: Private_66:68:02 (00:50:79:66:68:02)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
        .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.0.1.100, Dst: 10.0.1.120
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
        0000 00.. = Differentiated Services Codepoint: Default (0)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 84
    Identification: 0x6722 (26402)
    Flags: 0x00
        0... .... = Reserved bit: Not set
        .0.. .... = Don't fragment: Not set
        ..0. .... = More fragments: Not set
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 64
    Protocol: ICMP (1)
    Header Checksum: 0xfcab [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 10.0.1.100
    Destination Address: 10.0.1.120
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0xfda3 [correct]
    [Checksum Status: Good]
    Identifier (BE): 8807 (0x2267)
    Identifier (LE): 26402 (0x6722)
    Sequence Number (BE): 1 (0x0001)
    Sequence Number (LE): 256 (0x0100)
    [Response frame: 2]
    Data (56 bytes)
0000  08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17   ................
0010  18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25 26 27   ........ !"#$%&'
0020  28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 36 37   ()*+,-./01234567
0030  38 39 3a 3b 3c 3d 3e 3f                           89:;<=>?
        Data: 08090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f202122232425262728292a2b…
        [Length: 56]
No.     Time           Source                Destination           Protocol Length Info
      2 0.000211       10.0.1.120            10.0.1.100            ICMP     98     Echo (ping) reply
id=0x2267, seq=1/256, ttl=64 (request in 1)
Frame 2: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface -, id 0
    Interface id: 0 (-)
        Interface name: -
    Encapsulation type: Ethernet (1)
    Arrival Time: Aug 31, 2023 15:40:42.726807000 IST
    [Time shift for this packet: 0.000000000 seconds]
```

Based On Lab Question 1

• What is the destination MAC address of an ARP Request packet?

Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)

• What are the different Type Field values in the Ethernet headers that you observed?

Type: ARP (0x0806)
Type: IPv4 (0x0800)

• Use the captured data to analyze the process in which ARP acquires the MAC address for IP address 10.0.1.12.

ARP resolves the IP Address to MAC Address by asking “who has the IP address 10.0.1.12” and it resolves the target MAC Address by getting a ARP reply from the specific VPC.

Based On Lab Question 2

• Explain how the ping packets were issued by the hosts with duplicate addresses.

Pings were issues only to PC1 As pc4’s ip address wasnt set as it was duplicate.

• Did the ping command result in error messages?

No errors were present.

• How can duplicate IP addresses be used to compromise the data security?

In the unlikely event of duplicate ip addresses, secure data could be leaked since the packets would be sent to both devices.

• Give an example. Use the ARP cache and the captured packets to support your explanation.

Based On Lab Question 3

• Use your output data and ping results to explain what happened in each of the ping commands.

1. PC1 to PC2 : Successful
2. PC1 to PC3: Successful
3. PC1 to PC4: Successful
4. PC2 to PC3: not reachable
5. PC2 to PC4: not reachable

• Which ping operations were successful and which were unsuccessful? Why?

Pings that were associated with pc 2 and pc 4 were not successful as they were not in the same subnet and a switch can only handle 1 network.