Office Copy

Roll Number:

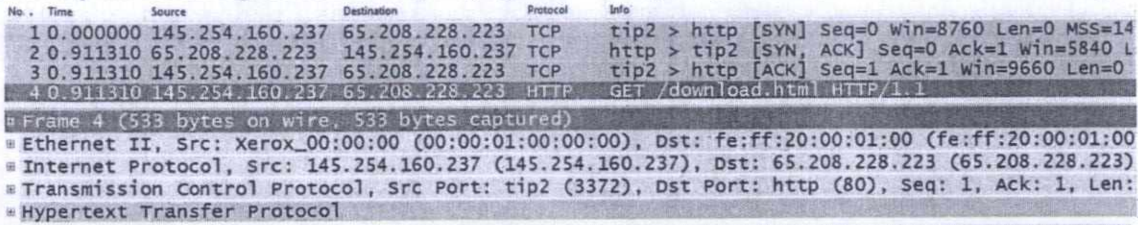
Department of Computer Science and Engineering
Thapar Institute of Engineering & Technology, Patiala

*BE (3rd year) WT*
*Elective Focus: Cyber and Information Security*
Time: 02 Hours; MM: 45

UCS534: Computer and Network Security
Dr. Maninder Singh, Dr Ravneet K, Dr. Sangita Roy
Oct 29, 2021

**ATTEMPT ANY 5 questions**

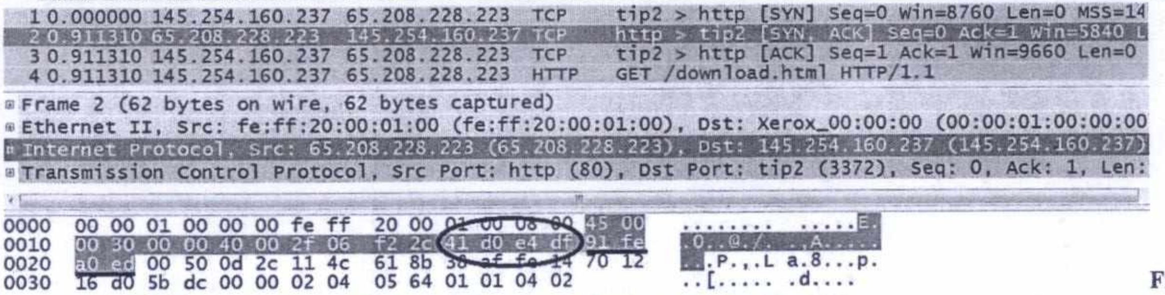
**Q1.** Study the following Wireshark outputs (**Figure1** and **Figure2**) carefully and answer the following:



```
No. . Time     Source          Destination     Protocol  Info
  1 0.000000 145.254.160.237 65.208.228.223  TCP       tip2 > http [SYN] Seq=0 Win=8760 Len=0 MSS=14
  2 0.911310 65.208.228.223  145.254.160.237 TCP       http > tip2 [SYN, ACK] Seq=0 Ack=1 Win=5840 L
  3 0.911310 145.254.160.237 65.208.228.223  TCP       tip2 > http [ACK] Seq=1 Ack=1 Win=9660 Len=0
  4 0.911310 145.254.160.237 65.208.228.223  HTTP      GET /download.html HTTP/1.1

⊞ Frame 4 (533 bytes on wire, 533 bytes captured)
⊞ Ethernet II, Src: Xerox_00:00:00 (00:00:01:00:00:00), Dst: fe:ff:20:00:01:00 (fe:ff:20:00:01:00
⊞ Internet Protocol, Src: 145.254.160.237 (145.254.160.237), Dst: 65.208.228.223 (65.208.228.223)
⊞ Transmission Control Protocol, Src Port: tip2 (3372), Dst Port: http (80), Seq: 1, Ack: 1, Len:
⊞ Hypertext Transfer Protocol
```

**Figure 1**



```
 1 0.000000 145.254.160.237 65.208.228.223  TCP    tip2 > http [SYN] Seq=0 Win=8760 Len=0 MSS=14
 2 0.911310 65.208.228.223  145.254.160.237 TCP    http > tip2 [SYN, ACK] Seq=0 Ack=1 Win=5840 L
 3 0.911310 145.254.160.237 65.208.228.223  TCP    tip2 > http [ACK] Seq=1 Ack=1 Win=9660 Len=0
 4 0.911310 145.254.160.237 65.208.228.223  HTTP   GET /download.html HTTP/1.1

⊞ Frame 2 (62 bytes on wire, 62 bytes captured)
⊞ Ethernet II, Src: fe:ff:20:00:01:00 (fe:ff:20:00:01:00), Dst: Xerox_00:00:00 (00:00:01:00:00:00
⊞ Internet Protocol, Src: 65.208.228.223 (65.208.228.223), Dst: 145.254.160.237 (145.254.160.237)
⊞ Transmission Control Protocol, Src Port: http (80), Dst Port: tip2 (3372), Seq: 0, Ack: 1, Len:

0000  00 00 01 00 00 00 fe ff  20 00 01 00 08 00 45 00   ........ ......E.
0010  00 30 00 00 40 00 2f 06  f2 2c 41 d0 e4 df 91 fe   .0..@./. .,A.....
0020  a0 ed 00 50 0d 2c 11 4c  61 8b 38 af fe 14 70 12   ...P.,.L a.8...p.
0030  16 d0 5b dc 00 00 02 04  05 64 01 01 04 02         ..[..... .d....
```

**Figure 2**

i) What is happening in the Frame 1 through Frame 3 **(ref Figure 1)**?

ii) What was user trying to access while these packets were captured **(ref Figure 1)**?

iii) In Frame 4 what is the significance of **fe:ff:20:00:01:00** address **(ref Figure 1)**?

iv) In Frame 4 what are the values associated with TCP flags **(ref Figure 1)**?

v) In the Hex Dump pane what does highlighted bytes **41 d0 e4 df** and **91 fe a0 ed** represent **(ref Figure 2).**

**9**

**Q2.** **a)** Both system() and execve() can be used to execute external programs. Why is system() unsafe while execve() is safe?

**b)** The followings are two different ways to print out environment variables. Please describe their differences:

```
$ /usr/bin/env
$ /usr/bin/strings /proc/$$/environ
```

**c)** For the Shellshock vulnerability to be exploitable, two conditions need to be satisfied. What are these two conditions, explain?

**9**

**Q3.**

**a)** A program abc invokes an external program xyz using system(), which is affected by the PATH environment variable. When we invoke abc from a shell prompt, how does the shell variable PATH in the current shell end up affecting the behavior of the system() function?

**b)** We run "nc -l 7070" on Machine 1 (IP address is 10.0.2.6), and we then type the following command on Machine 2. Describe what is going to happen?

```
$ /bin/cat < /dev/tcp/10.0.2.6/7070 >&0
```

**(4, 5)**

**Q4.**

**a)** The following function is called in a privileged program. The argument str points to a string that is entirely provided by users (the size of the string is up to 300 bytes). When this function is invoked, the address of the buffer array is 0xAABB0010, while the return address is stored in 0xAABB0050. Please write down the string that you would feed into the program, so when this string is copied to buffer and when the bof() function returns, the privileged program will run your code. In your answer, you don't need to write down the injected code, but the offsets of the key elements in your string need to be correct.

```
int bof(char *str)
{ char buffer[24];
strcpy(buffer,str);
return 1;}
```

**b)** In which memory segments are the variables in the following code located? Draw Stack & show variable locations in these segments:

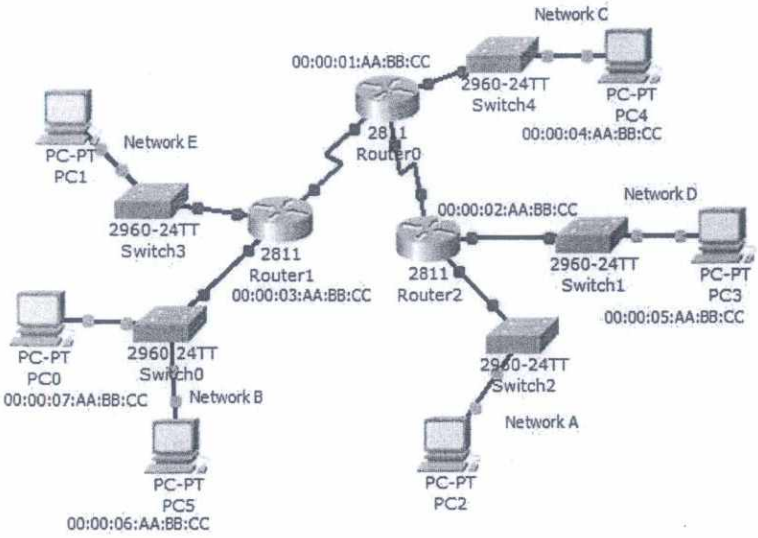
Stack:
Heap:
Data Segment:
BSS Segment:

```
int i = 0;
void func(char *str){
char *ptr = malloc(sizeof(int));
char buf[1024];
int j;
static int y;}
```

**(5, 4)**

**P.T.O**

Q5.



**a)** Give details in form of a table for the following scenarios.

Scenario 1: PC4 wishes to poison PC3
Scenario 2: PC0 wishes to poison PC5

Give initial arp -a details and list results after poisoning.

**b)** Trace layered journey starting from Application layer, where FQDN is specified (PC0 wants to access webservices hosted on PC4).
PC0 IP is: 172.31.2.6
PC4 IP is: 210.10.10.5

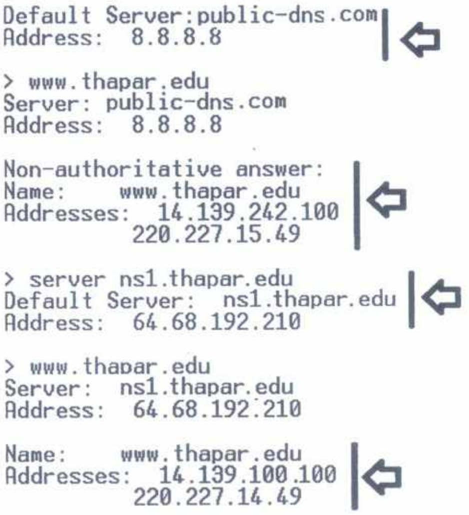
**(4, 5)**

Q6.

**a)** Study the output generated by **"nslookup"** program {given on the right-hand side} while user was connected to the Internet, Give technical comments on the highlighted parts. Emphasis should be on DNS-poisoning concept.

**b)** Following details were reported to the IT support cell at TIET: Dr. Singh was not able to communicate with Dr. Kumar, both machines were configured using DHCP server and both of these gentlemen were residing side by side in campus. Dr. Singh was connected to the network with SSID (TU) and Dr. Kumar with SSID (TU1). Dr. Kumar even complained that his machine is not able to access Internet. IP address details are given below: figure out the issues with both the machine and give detail explanation to sort out this problem.

| **Dr. Singh's Machine** | **Dr. Kumar's Machine** |
|---|---|
| Wireless LAN adapter Wi-Fi: | Wireless LAN adapter Wi-Fi: |
| Physical Address: 00-27-10-4F-FB-E8 | Physical Address: 00-27-10-4F-FB-E9 |
| IPv4 Address: 192.168.1.36 | IPv4 Address: 192.168.1.65 |
| Subnet Mask: 255.255.255.224 | Subnet Mask: 255.255.255.224 |
| Default Gateway: 192.168.1.62 | Default Gateway: 192.168.1.62 |
| DNS Servers: 8. 8. 8. 8 | DNS Servers: 8. 8. 8. 8 |


```
Default Server:public-dns.com
Address:  8.8.8.8

> www.thapar.edu
Server: public-dns.com
Address:  8.8.8.8

Non-authoritative answer:
Name:    www.thapar.edu
Addresses:  14.139.242.100
          220.227.15.49

> server ns1.thapar.edu
Default Server:  ns1.thapar.edu
Address:  64.68.192.210

> www.thapar.edu
Server:  ns1.thapar.edu
Address:  64.68.192.210

Name:    www.thapar.edu
Addresses:  14.139.100.100
          220.227.14.49
```

**(4, 5)**

Q7.

**a)** Does a SYN flooding attack cause the victim server to freeze? (Explanation is required, single line y/n will not get credit)

**b)** In the SYN flooding attack, why do we randomize the source IP address? Why cannot we just use the same IP address?

**c)** Can we launch a SYN flooding attack from a computer without using the root privilege?

**d)** What will happen if the spoofed source IP address in a SYN flooding attack does belong to a machine that is currently running?

**e)** Why do we choose to fill up the memory used for half-open connections, why cannot we directly target the memory used for holding full connections? The latter requires more memory, so the resource is much easier to exhaust.

**(1, 4x2)**