در این پروژه قصد داریم مجموعه‌ای از ابزارهایی بسازیم تا در نهایت به هدف شناخت کامل پروتکل‌ها، نحوه استخراج تمام اطلاعات یک بسته، ساخت انواع بسته ها با رعایت قوانین پروتکل مربوطه و ایجاد ابزارهای اسکنر برسیم. این پروژه از چندین قسمت تشکیل شده است که در هر قسمت شما باید ویدئویی کامل از نحوه قسمت های کد خود به همراه نحوه اجرا و همچنین نحوه خروجی در آن را بیان کنید. شاید در نگاه اول این پروژه کمی دشوار بنظر برسد اما در نهایت دانش شما را در حوزه شبکه بسیار ارتقا میبخشد. همچنین پیشنهاد میشود که از سیستم عامل لینوکس و توزیع اوبونتو و همچنین زبان پایتون جهت پیاده سازی این پروژه استفاده کنید. یقینن استفاده از سیستم عامل ویندوز یا زبان های دیگر سختی های مخصوص به خود را دارد. زمان تقریبی برای این پروژه حدودا ٤ روز میباشد، پس در نتیجه پروژه را به روز های آخر موکول نکنید.

قسمت اول) (۸ نمره) در این قسمت پروژه قصد داریم تا یک برنامه تحت کنسول شبیه به وایرشارک به صورت کاملا ابتدایی پیاده سازی کنیم. در شکل زیر نمونه ای از یک بسته را میتوانید ببینید :

```
Ethernet Frame:
        - Destination: 00:0C:29:C4:63:BC, Source: 08:10:79:E5:15:C8, Protocol: 8
        - IPv4 Packet:
                - version: 4,Header Length: 20,Type of Service: 0,Total Length: 1100,Identification: 43230,IP Flags: 2,Fragment Offset: 0,Ti
me To Live: 49,Protocole: 6,Header Checksum: 0xd072,Source Address: 151.101.122.217,Destination Addresss: 172.20.14.8
        - TCP Segment:
                - Source Port: 443, Destination Port: 37542
                - Sequence: 456922973, Acknowledgment: 2487348688
                - Offset: 32, Reserved: 0
                - Flags:
                        - NS: 0, CWR: 0, ECE: 0
                        - URG: 0, ACK: 1, PSH:0
                        - RST: 0, SYN: 0, FIN:0
                - Windows Size: 256, Checksum: 0xeb13, Urgent Pointer: 0
                - TCP Data:
                        b'D%\x99\xe8K\xd4B\x130\x0c\xb5\xfbo\xb2n\xe6^\xed\xd5\xbc\xac\xf7\xccb(WB
                        \x90\xc7\xdb\x13\x01\x9c\xd1\xa9rk\x862\x8a|\x02G\xbf\xa0\x8e\x8e\xf4\xa8\
                        x06\x80\x1dP\x91\xf3\xa8\x99\x96\x8c\xe5\x00\xca\xc3uW\x9a?;D\x935\xe1\x8f
                        0F\x1b\x1bK\xb4\xfai\x17\x01\x15^\x07\xa9\x06\x8e: \xee\xc5\xf5}\xf2\x98\x
                        91\xa9D\xb5v\x00\x91>\xec\x9a\xea8\xa2\xda\x94\xb3Y\x1e-tVD\xa7\xd5\x11\xa
                        1\x8a\x83\x96\x03\xc6\x84\xcc\xd6\x9b\x7f\xd8+qBh\x9a\xab\xe0_S\xa0\xee\xf
                        1R}\xa8\xfc\xfd\xf2\x0e\x15\x1b\xbc\xc7U%p\xe0\xb4\x19\xd9u\x15\x9crln\xfb
                        \x86=\x85A\xae\x0c\xe6/#\|\x8d\xb8,.}\xe9\x02\xcf\tY\xfc\xcc\x8f\x9a\xe7\x
                        80\xc9P\xcf:\xa5\xa3\x8b\xf8\xa15\x85\xdf\x97y\xe0\xc1\xd7\xb1\xaf\x0f\x00
                        \xd3\x03e\xd1\x94\x7f\xed\xe9\x1f\xf8\xa0\x91\xe2]\xa4\xe3\x1d\x8d\xe2\xb0
                        \xbcwd\xb0\xb1\x98\xfb\xd2\x04`\xd5\xac\xb1\xd5\x13\xf1\xde\xdcL\x8d\xaa\x
                        89\xcc\xaa\x12\xb7Q\xa4\xe72\xcb\xe6\xb4"\x8a\x0eAx\xe5j\|.\xe7H\x1aR\x83\
                        x94\xf0\xc3R\x88\x12&\x03\xf9\x12\xb1\x1f(\x0e\x01\xf3\x90!\x02\xc4\x06[v\
                        xb3x\x96$\xfe\x9e\x9e\x06\xb2\xaa\xc8\x94\x02\xa6\x89\x8a\xf8F\x88UY\x10R\
                        xd2\xb0\x05F0\xb6\xcf%`\xe3\x17@\x17lzx\x17\x05\xa5`I=2L\xd4\xa5\x93#\x1bn
                        \xaa\xb4\xa8\xfa\x9bX7\xe5p\xad99u\x15%\xd4hk\xc9\x1c@(@\x9f\x83{\x0b\x80\
                        x81\xb9Qg\xb2\xa8\xfd+N\x1d\xa8\xe8\xf3\x85,\x84\xfb\xde\xa5\xf1\x13\xc8[h
                        qL\x9d\xd6\xbd\x80\x84\xc0\t\xcb\x0e\xfe\x0fNK\xa0VS\xaa\xafs\x91\x85@\xfe
                        \xbcV\xfa\xb0\xd1f\xe5g\xcd\x8c\x97{4\x05\x04\xf9b\xf5\x89\x81\x8e\xb8\x83
                        \x8f\xbcY\x01\xa2\x89{\xd95\xd2JS\xb8\x10[\xe6\x83)\xec\x93s\x07\xc6\x04b^
                        \xd1\xc4\xfb4\x86\xca\xb5\xa9\xba\x98\x0b\xb4\xf8B\xdc\xc0\xd0\xc7\x8cP{\x
                        0b\xd5\x1d\x80N\xae\xf6\xf2yB\x88\xe1\xcb7\xda\x9c\xc2j\x1b\x8e\xa7\xd2\xa
```

برای این کار شما ابتدا باید کارت شبکه خود را در حالت promiscuous قرار دهید تا بتوانید تمام نوع بسته (حتی آن دسته از بسته‌هایی که آدرس MAC مقصد، مربوط به کارت شبکه شما نباشد) را ببینید. این کار را میتوانید با یکی از دو دستور زیر در سیستم عامل لینوکس به صورت زیر انجام دهید :

```
amirmnoohi@linux:~$ sudo ifconfig ens33 promisc #Enabling Promiscuous on ens33
amirmnoohi@linux:~$ sudo ifconfig ens33 -promisc #Disabling Promiscuous on ens33
amirmnoohi@linux:~$ sudo ip link set ens33 promisc on #Enabling promiscuous on ens33
amirmnoohi@linux:~$ sudo ip link set ens33 promisc off #Disabling promiscuous on ens33
```

پس از آن نیازمند این هستید تا با روشی تمام بسته‌هایی که برروی کارت شبکه دریافت میکنید را در کد خود دسترسی پیدا کنید و سپس اقدام به استخراج بیت به بیت آن کنید. برای این کار روش های زیادی وجود دارد. یکی از روش ها استفاده از توابع سوکت موجود در خود پایتون میباشد. در زیر این روش به تفصیل توضیح داده شده است :

۱- ابتدا یک سوکت ایجاد کنید :

```
conn = socket.socket(socket.AF_PACKET, socket.SOCK_RAW, socket.ntohs(3))
```

۲- سپس هر بسته ای که برروی کارت شبکه مشخص شده می آید را توسط این سوکت دریافت کنید :

```
raw_data, addr = conn.recvfrom(65535)
```

۳- سپس هدر های هر بسته را به صورت لایه به لایه با کمک تابع unpack جداسازی میکنید. ترتیب لایه ها به صورت زیر میباشد :

a. Ether: از بایت [۰ تا ۱۴)

```
ether_header = ether(raw_data)
def ether(data):
    dest_mac, src_mac, proto = unpack('! 6s 6s H', data[:14])
    return [get_mac_addr(dest_mac), get_mac_addr(src_mac), socket.htons(proto), data[14:]]
```

b. IP: از بایت ۱۴ تا ۲۰

```
ip_header = ip(ether_header[3])
def ip(data):
    maindata = data
    data = unpack('!BBHHHBBH4s4s', data[:20])
    return [(data[0] >> 4), (data[0] & 0xF) * 4, data[1], data[2],
        data[3], data[4] >> 13, data[4] & 0x1FFF, data[5],
        data[6], hex(data[7]), socket.inet_ntoa(data[8]),
        socket.inet_ntoa(data[9]), maindata[((data[0] & 0xF) * 4):]]
```

c. سپس با توجه به نوع پروتکل ارتباطی در قسمت IP باید تشخیص دهید چند بایت برای لایه انتقال جدا کنید که در زیر اگر پروتکل موجود در لایه IP برابر ۱ بود به معنی ICMP بودن بسته میباشد که باید به صورت زیر هدر آن استخراج پیدا کند :

```
if ip_header[8] == 1:
    icmp_header = icmp(ip_header[-1])
def icmp(data):
    type, code, checksum = unpack('!BBH', data[:4])
    return [type, code, hex(checksum), repr(data[4:])]
```

کد شما به محض اجرا باید اقدام به ذخیره سازی بسته های دریافتی به فرمت pcap کند. لیست پروتکل‌هایی که باید به صورت کامل استخراج کنید در زیر آمده است :

1- Ethernet 2- ARP 3- ICMP 4- DNS 5- TCP 6- UDP 7- HTTP

قسمت دوم) (۷ نمره) در این قسمت هدف نوشتن یک ابزار جهت بررسی پورت‌های باز یک IP یا FQDN میباشد. نرم افزار مشابه این ابزار nmap میباشد. که شما باید خروجی شبیه به nmap داشته باشید. نرم افزار شما جهت اجرا نیازمند ۴ ورودی میباشد:

۱- آدرس آیپی یا FQDN مقصد
۲- بازه ی پورت‌ها جهت اسکن
۳- مدل اسکن
۴- تاخیر در بررسی بسته های دریافت شده

برای مثال در زیر نمونه دستور برای اجرای برنامه را میتوانید ببینید :

$$\$\ python3\ main.py\ -t\ iut.ac.ir\ -p\ 0-100\ -s\ CS\ -d\ 3$$

مدل اسکن شما جمعا دارای 5 مدل میباشد :

- Connect Scan
- Ack Scan
- Syn Scan
- Fin Scan
- Windows Scan

منطق هر یک از روش های بالا را میتوانید مشابه با nmap که در این لینک میباشد، پیاده سازی کنید.

منطق کلی شما به این صورت هست که برای Connect Scan میتوانید از تابع connect موجود در خود socket استفاده کنید اما برای دیگر اسکن ها باید به صورت دستی یک بسته ی TCP ساخته و برای مقصد ارسال کرده و وضعیت آن پورت را از نتیجه مشخص کنید. همچنین در این فایل لیستی از سرویس برروی پورت ها را میتوانید مشاهده و جهت نمایش صحیح استفاده کنید.

قسمت سوم) (تا ۵ نمره) (اختیاری جهت تکمیل ۳۰ درصد تکالیف وایرشارک و برنامه‌نویسی) در این قسمت قصد داریم تا کدی بنویسیم تا به تمام بسته های Arp، DNS، ICMP به صورت اتوماتیک به محض دریافت پاسخ دهد. دقت شود محتوای این بسته‌های پاسخ داده شده را میتوانید به دلخواه پر کنید. فقط دقت کنید که تمام checksum ها را به درستی پر شوند. خروجی Wireshark خود را برای هر مدل در فایل نهایی ضمیمه کنید.

نکات مهم :

- فایلی که در سامانه lms آپلود میکنید باید به فرم name-stdnumber.zip باشد.
- این پروژه را میتوانید به صورت انفرادی یا ۲ نفره تحویل دهید.
- ممکن است تحویل بعضی از افراد از طریق اسکایپ انجام شود که متعاقبا بعد از تحویل مشخص میشود.
- پیشنهاد میشود از گیت جهت مدیریت کد های خود استفاده کنید.
- درصورتی که سوالی داشتید میتوانید از طریق ایمیل highlimner@gmail.com در ارتباط باشید.

موفق باشید