import masscan  
import os  
import string  
import sys  
from masscan import \*  
import pcap  
from scapy.all import rdpcap  
from scapy.all import wrpcap  
import sys  
from socket import \*  
from socket import AF\_INET, SOCK\_STREAM  
from datetime import datetime

print ("1. IP/Port scanning")  
print ("2. Network sniffer")  
print ("3. Password cracking using john")  
print ("4. Display running services")  
i=input("Enter a number to choose a service: ")  
if i==1 :  
    try:  
        mas = masscan.PortScanner()  
    except masscan.PortScannerError:  
        print("masscan binary not found", sys.exc\_info()[0])  
        sys.exit(1)  
    except:  
        print("unexppected error:", sys.exc\_info()[0])  
        sys.exit(1)  
    print ("masscan version:",mas.masscan\_version)  
    mas.scan('192.168.0.17/24', ports='21,22,80')  
    print("masscan command line:", mas.command\_line)  
    print mas.scan\_result

if i==2 :  
     
try:

s = socket.socket(socket.AF\_INET, socket.SOCK\_RAW, socket.IPPROTO\_TCP)

except socket.error , msg:

print 'Socket could not be created. Error Code : ' + str(msg[0]) + ' Message ' + msg[1]

sys.exit()

# receive a packet

while True:

packet = s.recvfrom(65565)

#packet string from tuple

packet = packet[0]

#take first 20 characters for the ip header

ip\_header = packet[0:20]

#now unpack them :)

iph = unpack('!BBHHHBBH4s4s' , ip\_header)

version\_ihl = iph[0]

version = version\_ihl &gt;&gt; 4

ihl = version\_ihl &amp; 0xF

iph\_length = ihl \* 4

ttl = iph[5]

protocol = iph[6]

s\_addr = socket.inet\_ntoa(iph[8]);

d\_addr = socket.inet\_ntoa(iph[9]);

print 'Version : ' + str(version) + ' IP Header Length : ' + str(ihl) + ' TTL : ' + str(ttl) + ' Protocol : ' + str(protocol) + ' Source Address : ' + str(s\_addr) + ' Destination Address : ' + str(d\_addr)

tcp\_header = packet[iph\_length:iph\_length+20]

#now unpack them :)

tcph = unpack('!HHLLBBHHH' , tcp\_header)

source\_port = tcph[0]

dest\_port = tcph[1]

sequence = tcph[2]

acknowledgement = tcph[3]

doff\_reserved = tcph[4]

tcph\_length = doff\_reserved &gt;&gt; 4

print 'Source Port : ' + str(source\_port) + ' Dest Port : ' + str(dest\_port) + ' Sequence Number : ' + str(sequence) + ' Acknowledgement : ' + str(acknowledgement) + ' TCP header length : ' + str(tcph\_length)

h\_size = iph\_length + tcph\_length \* 4

data\_size = len(packet) - h\_size

#get data from the packet

data = packet[h\_size:]

print 'Data : ' + data

if i==3 :  
    print ".include <john.conf>\n[List.Rules:rep]";  
    [ sys.stdout.write("s{0}\\x{1:02x}\ns{2}\\x{1:02x}\n".format(c, ord(c) - ord("a") + 1, c.upper()))for c in string.lowercase ] > rep.conf  
    for l in sys.stdin:  
        print l.rstrip("\r\n").encode("base64").replace("\n", "") < rockyou.txt > rb.txt

if i==4 :

import os

status = os.system('systemctl is-active --quiet service-name')

print(status)