ABBREVIATIONS

WAN	Wide Area Network
LAN	Local Area Network
TCP	Transmission Control Protocol
DNS	Domain Name Server
API	Application Program Interface
CLI	Command Line Interface
UDP	User Datagram Protocol
ICMP	Internet Control Message Protocol
IP	Internet Protocol
NIC	Network Interface Card

ABSTRACT

Packet sniffing is a method of tapping each packet as it flows across the network; i.e. it is a technique in which a user sniffs data belonging to other users of the network. Packet sniffers can operate as an administrative tool or for malicious purposes. It depends on the user's intent. Network administrators use them for monitoring and validating network traffic.

Packet sniffers are basically applications. They are programs used to read packets that travel across the network layer of the Transmission Control Protocol/Internet Protocol (TCP/IP) layer. (Basically, the packets are retrieved from the network layer and the data is interpreted.)

The packet sniffer listens to the data that arrives at the Network Interface Card (NIC). However, packet sniffers are not limited to Local Area Networks (LANs). Similar packet sniffers exist for Wide Area Networks (WANs). If a machine is in the path of two connected machines (A and B) on a WAN, the machine can listen to the traffic flowing from A to B.

INTRODUCTION

Packet sniffing is a technique whereby packet data flowing across the network is detected and observed. It gathers, collects and maintains logs of network packets which pass through a network.

Network administrators use packet sniffing tools to monitor and validate network traffic, while hackers may use similar tools for malicious purposes. Packets reflect the network activity in a node.

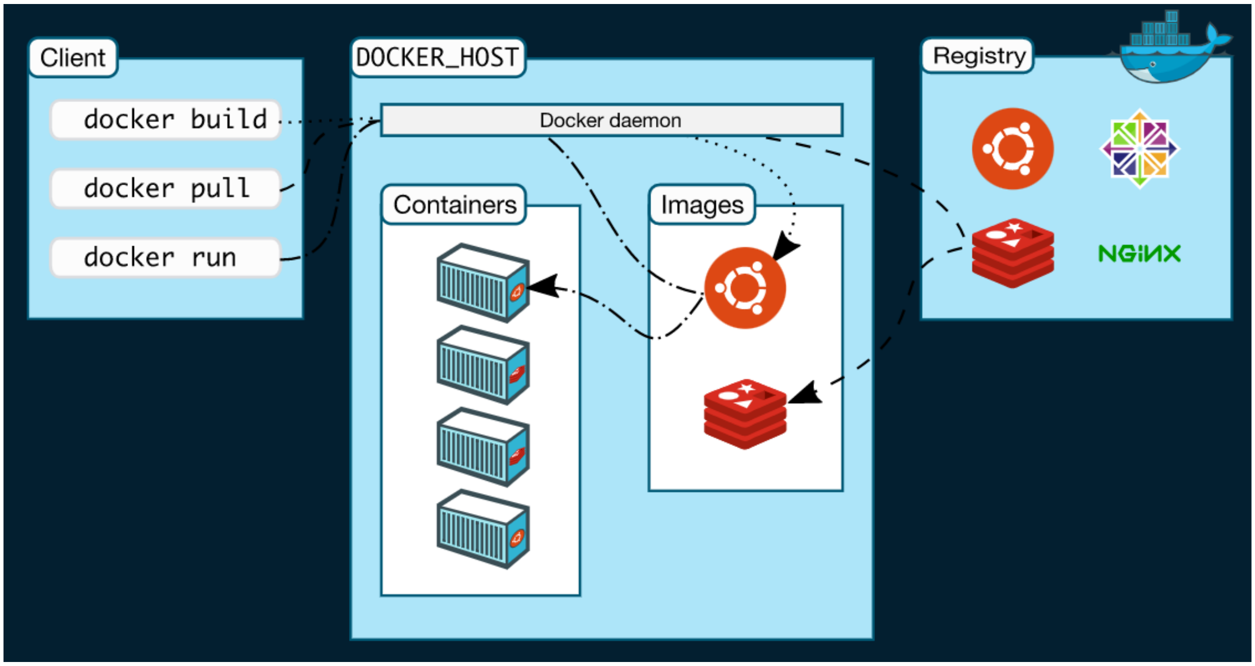
Here I will perform the role of administrator.

WHAT IS DOCKER?

- Docker is an open platform for developing, shipping and running apps.
- It is a better alternative to VirtualBox as it has less load and is an application level virtualization unlike VirtualBox which is an OS level virtualization.
- Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. (All applications have their own implementation and Docker is a virtualization of that.)
- Docker makes development efficient and significantly reduce the delay between writing code and running it in production.
- Docker containers can run on a developer’s local laptop, on physical or virtual machines in a data center, on cloud providers, or in a mixture of environments.
- Docker is portable and lightweight in nature, which makes it easy to dynamically manage workloads.
- It provides a viable, cost-effective alternative to hypervisor-based virtual machines.
- Docker Desktop is an easy-to-install application for your Mac, Windows or Linux environment that enables you to build and share containerized applications and microservices.
- Docker Desktop includes the Docker daemon (dockerd), the Docker client (docker), Docker Compose, Docker Content Trust, Kubernetes, and Credential Helper.

- A Docker registry stores Docker images. Docker Hub is a public registry that anyone can use, and Docker is configured to look for images on Docker Hub by default. You can even run your own private registry.
- When you use the docker pull or docker run commands, the required images are pulled from your configured registry. When you use the docker push command, your image is pushed to your configured registry.

DOCKER ARCHITECTURE



WHAT IS SCAPY

- Scapy is a powerful packet manipulation library supported by both Python2 and Python3. It is used for interacting with the packets on the network. It has several functionalities through which we can easily forge and manipulate the packet.
- This sniffer tool is developed in Docker with python version 3 using Scapy which has its own command line interpreter(CLI) to capture TCP, UDP and ICMP packets based on their header.
- Scapy needs to be imported. There are raw packet capture tools available like tcpdump but Scapy captures all the incoming and outgoing packets from all interfaces of the machine and classifies them into incoming and outgoing packets. With this tool we can customize the code that will sniff exactly what we need.

MODULES & SOFTWARE USED

- Docker platform is used.

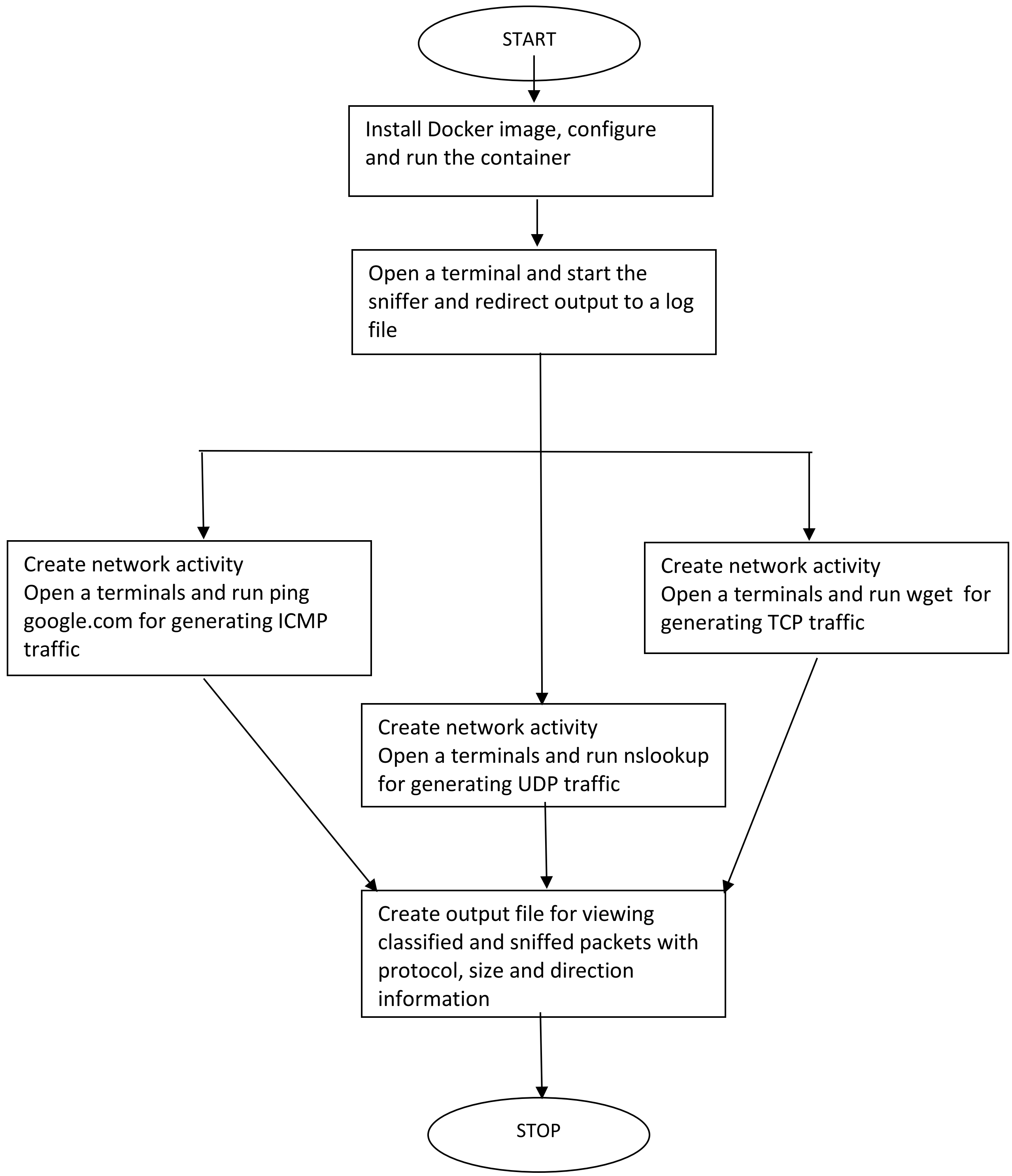
Modules and library used are:

- Scapy (a python library)
- socket
- datetime
- os
- time

PRE-REQUISITES

- Packet sniffing can be done only if the network to be monitored is in promiscuous mode, which allows a network device to intercept and read each network packet that arrives in its entirety.
- Some networks have non-promiscuous mode for security.
- In promiscuous mode, NIC is set to accept every packet that it receives, like it has no hardware filtering.
- Some other tools that use promiscuous mode - Wireshark, Tcpdump, Aircrack-ng, cain and abel(darknet, beware!), Snort, VirtualBox, etc.
- Extremely effective because of it's passive nature.
- And finally, in promisc mode the listener/sniffer has to be connected to the network.
- ifconfig eth0 promisc to enable it (if not already enabled).

FLOWCHART


START
Install Docker image, configure and run the container
Open a terminal and start the sniffer and redirect output to a log file
Create network activity
Open a terminals and run ping google.com for generating ICMP traffic
Create network activity
Open a terminals and run wget for generating TCP traffic
Create network activity
Open a terminals and run nslookup for generating UDP traffic
Create output file for viewing classified and sniffed packets with protocol, size and direction information
STOP

IMPLEMENTATION

- Download Docker Desktop for Windows. Pull Scapy Docker image.
- Run exe with administrator.
- Install WSL(Windows Linux Subsystem for Linux) for your system.
- Double click on Docker icon to start the app
- User URL to pull the Docker image for Scapy (https://hub.docker.com/r/travelping/scapy)
- Click on play button to start the docker image.
- Provide host path after this docker container will start.
- Use terminal to start.
- Create python script and run using command python packet.py >> 1.log etc..
- Using external terminal start pinging to different domains, for example ibm.com, google.com to create network activity.
- Check log file where you will get all details/packets sniffed and categorised protocol wise with direction.

SOURCE CODE

Step 1: Import required modules

Create a python file and import all the required modules.

i.e. os, socket, scapy, datetime.

```
#!/usr/bin/python
from scapy.all import *
import socket
import datetime
import os
import time
```

Step 2: Build functions

After importing all the required modules creating a function and using python built-in main function.

Also, using the prn parameter helps to sniff packets continuously. In place of prn if we use count=1 then only one packet will be sniffed.

```
def network_monitoring_for_visualization_version(pkt):
 ‘’’if __name__ == ‘__main__’:
    sniff(prn=network_monitoring_for_visualization_version)’’’
```

Step 3: Classification

Classifying packets into TCP, UDP and ICMP, then into incoming & outgoing packets.

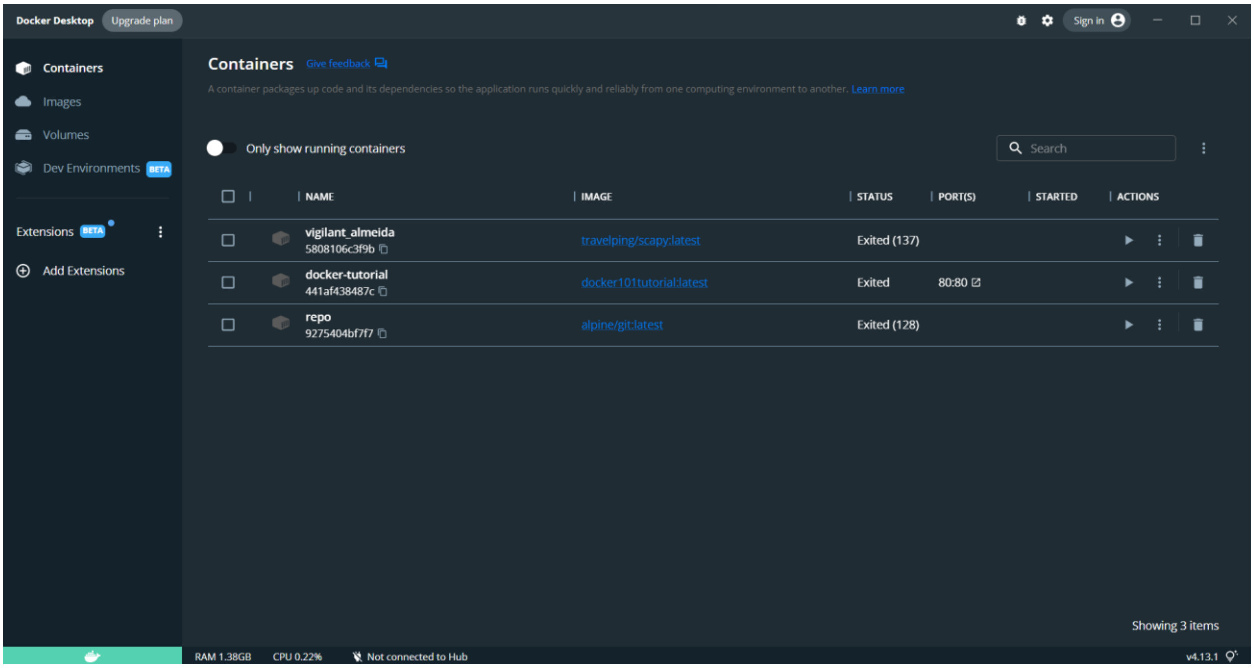
Scapy has a built-in function to check if a packet has layers of protocols. i.e. packet.haslayer(TCP), or packet.haslayer(UDP) or any protocols supported by Scapy. And finally print all the required information in the console.

```
time=datetime.now() # importing time to get the exact time when packets are sniffed.

# classifying packets into TCP

    if pkt.haslayer(TCP):
# classifying packets into TCP Incoming packets
        if socket.gethostbyname(socket.gethostname())== pkt[IP].dst:
            print(str("[")+str(time)+str("]")+"    "+"TCP-IN:{}".format(len(pkt[TCP]))+ "Bytes"+"    "+"SRC-MAC:" +str(pkt.src)+"    "+ "DST-MAC:"+str(pkt.dst)+"    "+ "SRC-PORT:"+str(pkt.sport)+"       "+"DST-PORT:"+str(pkt.dport)+"       "+"SRC-IP:"+str(pkt[IP].src  )+"    "+"DST-IP:"+str(pkt[IP].dst))

        if socket.gethostbyname(socket.gethostname())==pkt[IP].src:
            print(str("[")+str(time)+str("]")+"                "+"TCP-OUT:{}".format(len(pkt[TCP]))+ " Bytes"+"    "+"SRC-MAC:" +str(pkt.src)+"    "+ "DST-MAC:"+str(pkt.dst)+"       "+    "SRC-PORT:"+str(pkt.sport)+"       "+"DST-PORT:"+str(pkt.dport)+"            "+"SRC-IP:"+str(pkt[IP].src)+"            "+"DST-IP:"+str(pkt[IP].dst))

#classifying packets into UDP

     if pkt.haslayer(UDP):
        if socket.gethostbyname(socket.gethostname())==pkt[IP].src:
            # classifying packets into UDP Outgoing packets
```

```
            print(str("[")+str(time)+str("]")+"                                                    "+"UDP-OUT:{}".format(len(pkt[UDP]))+" Bytes "+
"    "+"SRC-MAC:" +str(pkt.src)+"    "+"DST-MAC:"+ str(pkt.dst)+"    "+"SRC-PORT:"+ str(pkt.sport) +"     "+"DST-PORT:"+ str(pkt.dport)+"     "+"SRC-IP:"+ str(pkt[IP].src)+"    "+"DST-IP:"+ str(pkt[IP].dst))

        if socket.gethostbyname(socket.gethostname())==pkt[IP].dst:
            # classifying packets into UDP Incoming packets
            print(str("[")+str(time)+str("]")+"                                                    "+"UDP-IN:{}".format(len(pkt[UDP]))+" Bytes "+
"    "+"SRC-MAC:" +str(pkt.src)+"    "+"DST-MAC:"+ str(pkt.dst)+"    "+"SRC-PORT:"+ str(pkt.sport) +"     "+"DST-PORT:"+ str(pkt.dport)+"     "+"SRC-IP:"+ str(pkt[IP].src)+"    "+"DST-IP:"+ str(pkt[IP].dst))


#classifying packets into ICMP
    if pkt.haslayer(ICMP):
        # classifying packets into ICMP Incoming packets
        if socket.gethostbyname(socket.gethostname())==pkt[IP].src:
            print(str("[")+str(time)+str("]")+"                                                  "+"ICMP-OUT:{}".format(len(pkt[ICMP]))+"         Bytes"+"                                      "+"IP-Version:"+str(pkt[IP].version) +"     "*1+" SRC-MAC:"+str(pkt.src)+"     "+"DST-MAC:"+str(pkt.dst)+"             "+"SRC-IP:   "+str(pkt[IP].src)+   "         "+"DST-IP:"+str(pkt[IP].dst))

          # classifying packets into ICMP Outgoing packets
        if socket.gethostbyname(socket.gethostname())==pkt[IP].dst:
            print(str("[")+str(time)+str("]")+"                                                  "+"ICMP-IN:{}".format(len(pkt[ICMP]))+"         Bytes"+"                                      "+"IP-
```

```
Version:"+str(pkt[IP].version)+"    "*1+"    SRC-MAC:"+str(pkt.src)+"    "+"DST-MAC:"+str(pkt.dst)+"          "+"SRC-IP:   "+str(pkt[IP].src)+   "          "+"DST-IP: "+str(pkt[IP].dst))

if __name__ == '__main__':
    sniff(prn=network_monitoring_for_visualization_version)
```

HOW THE DOCKER INTERFACE LOOKS



PROCEDURE

- Host path is where you mount the container, in this case E drive.
- To explain the sniffed outputs, I will use three use cases:

    - Case 1 where only packet.py is run after pinging google.com. The output is redirected to a file 1.out which gives the output when one terminal is trying to reach google

    - Case 2 where we ping google.com in the first terminal and ibm.com in the external terminal and in another external terminal cat 1.out is given, meaning the file is directed to 2.out instead of original output location. And ***cat*** command is used to concatenate. 2.out is the output when one terminal is trying to reach google, another is trying to reach ibm.

    - Case 3 we do an nslookup check after pinging both for sastra.ac.in and also do nslookup sastra.ac.in 8.8.8.8(this is global DNS).

STEPS

OUTPUT – wget -TCP

```
docker exec -it aaa4697a65ffec75bc0beecdec217f1845b859c13332d4f25fb0f07a9260dd5a /bin/sh
[2022-12-12 16:24:34.040127]  ICMP-IN:64 Bytes    IP-Version:4          SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.196.174   DST-IP:  172.17.0.2
[2022-12-12 16:24:35.013915]  ICMP-OUT:64 Bytes    IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-IP: 172.17.0.2   DST-IP:  142.250.196.174
[2022-12-12 16:24:35.039844]  ICMP-IN:64 Bytes    IP-Version:4          SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.196.174   DST-IP:  172.17.0.2
[2022-12-12 16:24:36.013161]  ICMP-OUT:64 Bytes    IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-IP: 172.17.0.2   DST-IP:  142.250.196.174
[2022-12-12 16:24:36.040954]  ICMP-IN:64 Bytes    IP-Version:4          SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.196.174   DST-IP:  172.17.0.2
[2022-12-12 16:24:37.013455]  ICMP-OUT:64 Bytes    IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-IP: 172.17.0.2   DST-IP:  142.250.196.174
[2022-12-12 16:24:37.045137]  ICMP-IN:64 Bytes    IP-Version:4          SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.196.174   DST-IP:  172.17.0.2
[2022-12-12 16:24:38.019980]  ICMP-OUT:64 Bytes    IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-IP: 172.17.0.2   DST-IP:  142.250.196.174
[2022-12-12 16:24:39.015107]  ICMP-OUT:64 Bytes    IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-IP: 172.17.0.2   DST-IP:  142.250.196.174
[2022-12-12 16:24:39.043468]  ICMP-IN:64 Bytes    IP-Version:4          SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.196.174   DST-IP:  172.17.0.2
[2022-12-12 16:25:13.604098]  UDP-OUT:38 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:43406   DST-PORT:53   SRC-IP:172.17.0.2   DST-IP:192.168.65.5
[2022-12-12 16:25:13.607384]  UDP-OUT:38 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:43406   DST-PORT:53   SRC-IP:172.17.0.2   DST-IP:192.168.65.5
[2022-12-12 16:25:14.757845]  UDP-IN:118 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:53   DST-PORT:43406   SRC-IP:192.168.65.5   DST-IP:172.17.0.2
[2022-12-12 16:25:14.761108]  UDP-IN:66 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:53   DST-PORT:43406   SRC-IP:192.168.65.5   DST-IP:172.17.0.2
[2022-12-12 16:25:26.677526]  UDP-OUT:39 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:37556   DST-PORT:53   SRC-IP:172.17.0.2   DST-IP:192.168.65.5
[2022-12-12 16:25:26.679965]  UDP-OUT:39 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:37556   DST-PORT:53   SRC-IP:172.17.0.2   DST-IP:192.168.65.5
[2022-12-12 16:25:27.528585]  UDP-IN:279 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:53   DST-PORT:37556   SRC-IP:192.168.65.5   DST-IP:172.17.0.2
[2022-12-12 16:25:28.801026]  UDP-IN:457 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:53   DST-PORT:37556   SRC-IP:192.168.65.5   DST-IP:172.17.0.2
[2022-12-12 16:25:47.094590]  UDP-OUT:40 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:34517   DST-PORT:53   SRC-IP:172.17.0.2   DST-IP:192.168.65.5
[2022-12-12 16:25:47.096933]  UDP-OUT:40 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:34517   DST-PORT:53   SRC-IP:172.17.0.2   DST-IP:192.168.65.5
[2022-12-12 16:25:47.099767]  UDP-IN:82 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:53   DST-PORT:34517   SRC-IP:192.168.65.5   DST-IP:172.17.0.2
[2022-12-12 16:25:47.102789]  UDP-IN:70 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:53   DST-PORT:34517   SRC-IP:192.168.65.5   DST-IP:172.17.0.2
[2022-12-12 16:25:47.105545]  TCP-OUT:40 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:50132   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.196.164
[2022-12-12 16:25:47.107531]  TCP-IN:28 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:50132   SRC-IP:142.250.196.164   DST-IP:172.17.0.2
[2022-12-12 16:25:47.109517]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:50132   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.196.164
[2022-12-12 16:25:47.110610]  TCP-OUT:97 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:50132   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.196.164
[2022-12-12 16:25:47.111483]  TCP-IN:20 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:50132   SRC-IP:142.250.196.164   DST-IP:172.17.0.2
[2022-12-12 16:25:47.227357]  TCP-IN:1406 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:50132   SRC-IP:142.250.196.164   DST-IP:172.17.0.2
[2022-12-12 16:25:47.229541]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:50132   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.196.164
/packet-sniffing #
```

OUTPUT – watch wget -TCP

```
docker exec -it aaa4697a65ffec75bc0beecdec217f1845b859c13332d4f25fb0f07a9260dd5a /bin/sh
[2022-12-12 16:34:50.911259]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60750   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:50.912577]  TCP-IN:1480 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60750   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:50.913824]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60750   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:50.915006]  TCP-IN:1480 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60750   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:50.916267]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60750   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:50.917930]  TCP-IN:1212 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60750   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:50.919313]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60750   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:50.920124]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60750   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:52.913830]  UDP-OUT:40 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:37586   DST-PORT:53   SRC-IP:172.17.0.2   DST-IP:192.168.65.5
[2022-12-12 16:34:52.916855]  UDP-OUT:40 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:37586   DST-PORT:53   SRC-IP:172.17.0.2   DST-IP:192.168.65.5
[2022-12-12 16:34:52.921583]  UDP-IN:82 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:53   DST-PORT:37586   SRC-IP:192.168.65.5   DST-IP:172.17.0.2
[2022-12-12 16:34:52.925603]  UDP-IN:70 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:53   DST-PORT:37586   SRC-IP:192.168.65.5   DST-IP:172.17.0.2
[2022-12-12 16:34:52.928422]  TCP-OUT:40 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60752   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:52.930728]  TCP-IN:28 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60752   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:52.933270]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60752   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:52.934997]  TCP-OUT:97 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60752   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:52.936137]  TCP-IN:20 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60752   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:53.052497]  TCP-IN:1420 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60752   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:53.054764]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60752   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:53.056925]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60752   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:53.058498]  TCP-IN:1480 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60752   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:53.059755]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60752   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:53.060777]  TCP-IN:1480 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60752   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:53.061755]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60752   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:53.062775]  TCP-IN:1300 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60752   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:53.064250]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60752   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:53.065111]  TCP-IN:20 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60752   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:53.066945]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60752   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:53.069312]  TCP-IN:20 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60752   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:53.070538]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60752   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:53.072552]  TCP-IN:20 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60752   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:53.074450]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60752   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
[2022-12-12 16:34:53.075302]  TCP-IN:20 Bytes    SRC-MAC:02:42:95:38:2b:9c   DST-MAC:02:42:ac:11:00:02   SRC-PORT:80   DST-PORT:60752   SRC-IP:142.250.205.228   DST-IP:172.17.0.2
[2022-12-12 16:34:53.076184]  TCP-OUT:20 Bytes    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:95:38:2b:9c   SRC-PORT:60752   DST-PORT:80   SRC-IP:172.17.0.2   DST-IP:142.250.205.228
```

OUTPUT(packet.py in case 1)

```
docker exec -it 5808106c3f9b
/ # ls
bin              home             mnt              proc             sbin             tmp
dev              lib              opt              root             srv              usr
etc              media            packet-sniffing  run              sys              var
/ # cd packet-sniffing
/packet-sniffing # ls
1.out      2.out      log.out    log1.out   packet.py
/packet-sniffing # packet.py
/bin/sh: packet.py: not found
/packet-sniffing # python3 packet.py
[2022-11-09 17:46:42.913630] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:2c:e4:79:c4   SRC-IP: 172.17.0.2   DST-IP:
 142.250.76.46
[2022-11-09 17:46:42.936044] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:2c:e4:79:c4   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.76.46
  DST-IP:  172.17.0.2
[2022-11-09 17:46:43.913754] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:2c:e4:79:c4   SRC-IP: 172.17.0.2   DST-IP:
 142.250.76.46
[2022-11-09 17:46:43.939165] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:2c:e4:79:c4   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.76.46
  DST-IP:  172.17.0.2
[2022-11-09 17:46:44.914227] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:2c:e4:79:c4   SRC-IP: 172.17.0.2   DST-IP:
 142.250.76.46
[2022-11-09 17:46:44.936797] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:2c:e4:79:c4   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.76.46
  DST-IP:  172.17.0.2
[2022-11-09 17:46:45.914647] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:2c:e4:79:c4   SRC-IP: 172.17.0.2   DST-IP:
 142.250.76.46
[2022-11-09 17:46:45.938922] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:2c:e4:79:c4   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.76.46
  DST-IP:  172.17.0.2
[2022-11-09 17:46:46.914838] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:2c:e4:79:c4   SRC-IP: 172.17.0.2   DST-IP:
 142.250.76.46
[2022-11-09 17:46:46.935318] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:2c:e4:79:c4   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.76.46
  DST-IP:  172.17.0.2
[2022-11-09 17:46:47.915669] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:2c:e4:79:c4   SRC-IP: 172.17.0.2   DST-IP:
 142.250.76.46
[2022-11-09 17:46:47.935815] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:2c:e4:79:c4   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.76.46
  DST-IP:  172.17.0.2
[2022-11-09 17:46:48.916283] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:2c:e4:79:c4   SRC-IP: 172.17.0.2   DST-IP:
 142.250.76.46
[2022-11-09 17:46:48.941621] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:2c:e4:79:c4   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.76.46
  DST-IP:  172.17.0.2
```

OUTPUT 1.out case 1

```
docker exec -it 5808106c3f9b
/ # ls
bin              home             mnt              proc             sbin             tmp
dev              lib              opt              root             srv              usr
etc              media            packet-sniffing  run              sys              var
/ # cd packet-sniffing
/packet-sniffing # ls
1.out      2.out      log.out    log1.out   packet.py
/packet-sniffing # python3 packet.py>>1.out
^C/packet-sniffing # cata 1.out
/bin/sh: cata: not found
/packet-sniffing # cat 1.out
[2022-11-08 17:52:59.798672] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:01:73:02:cb   SRC-IP: 172.17.0.2   DST-IP:
 142.250.67.46
[2022-11-08 17:52:59.822412] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:01:73:02:cb   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.67.46
  DST-IP:  172.17.0.2
[2022-11-08 17:53:00.799321] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:01:73:02:cb   SRC-IP: 172.17.0.2   DST-IP:
 142.250.67.46
[2022-11-08 17:53:00.846538] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:01:73:02:cb   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.67.46
  DST-IP:  172.17.0.2
[2022-11-08 17:53:01.799516] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:01:73:02:cb   SRC-IP: 172.17.0.2   DST-IP:
 142.250.67.46
[2022-11-08 17:53:01.825883] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:01:73:02:cb   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.67.46
  DST-IP:  172.17.0.2
[2022-11-08 17:53:02.799533] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:01:73:02:cb   SRC-IP: 172.17.0.2   DST-IP:
 142.250.67.46
[2022-11-08 17:53:02.824062] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:01:73:02:cb   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.67.46
  DST-IP:  172.17.0.2
[2022-11-08 17:53:03.800240] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:01:73:02:cb   SRC-IP: 172.17.0.2   DST-IP:
 142.250.67.46
[2022-11-08 17:53:03.832389] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:01:73:02:cb   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.67.46
  DST-IP:  172.17.0.2
[2022-11-08 17:53:04.800383] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:01:73:02:cb   SRC-IP: 172.17.0.2   DST-IP:
 142.250.67.46
[2022-11-08 17:53:04.825298] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:01:73:02:cb   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.67.46
  DST-IP:  172.17.0.2
[2022-11-08 17:53:05.800262] ICMP-OUT:64 Bytes   IP-Version:4    SRC-MAC:02:42:ac:11:00:02   DST-MAC:02:42:01:73:02:cb   SRC-IP: 172.17.0.2   DST-IP:
 142.250.67.46
[2022-11-08 17:53:05.826049] ICMP-IN:64 Bytes   IP-Version:4          SRC-MAC:02:42:01:73:02:cb   DST-MAC:02:42:ac:11:00:02   SRC-IP: 142.250.67.46
  DST-IP:  172.17.0.2
```

OUTPUT 2.out case 3

```
docker exec -it 5808106c3f9b
/ # ls
bin              home             mnt              proc             sbin             tmp
dev              lib              opt              root             srv              usr
etc              media            packet-sniffing  run              sys              var
/ # cd packet-sniffing
/packet-sniffing # ls
1.out       log.out    log1.out    packet.py
/packet-sniffing # python3 packet.py>>2.out
^C/packet-sniffing # cat 2.out
[2022-11-08 18:05:57.756283]  ICMP-OUT:64 Bytes    IP-Version:4     SRC-MAC:02:42:ac:11:00:02     DST-MAC:02:42:01:73:02:
cb    SRC-IP: 172.17.0.2     DST-IP:  142.250.196.78
[2022-11-08 18:05:57.781363]  ICMP-IN:64 Bytes    IP-Version:4           SRC-MAC:02:42:01:73:02:cb     DST-MAC:02:42:ac
:11:00:02    SRC-IP: 142.250.196.78    DST-IP:  172.17.0.2
[2022-11-08 18:05:57.980461]  ICMP-OUT:64 Bytes    IP-Version:4     SRC-MAC:02:42:ac:11:00:02     DST-MAC:02:42:01:73:02:
cb    SRC-IP: 172.17.0.2     DST-IP:  104.85.124.77
[2022-11-08 18:05:58.005754]  ICMP-IN:64 Bytes    IP-Version:4           SRC-MAC:02:42:01:73:02:cb     DST-MAC:02:42:ac
:11:00:02    SRC-IP: 104.85.124.77    DST-IP:  172.17.0.2
[2022-11-08 18:05:58.755939]  ICMP-OUT:64 Bytes    IP-Version:4     SRC-MAC:02:42:ac:11:00:02     DST-MAC:02:42:01:73:02:
cb    SRC-IP: 172.17.0.2     DST-IP:  142.250.196.78
[2022-11-08 18:05:58.781865]  ICMP-IN:64 Bytes    IP-Version:4           SRC-MAC:02:42:01:73:02:cb     DST-MAC:02:42:ac
:11:00:02    SRC-IP: 142.250.196.78    DST-IP:  172.17.0.2
[2022-11-08 18:05:58.981183]  ICMP-OUT:64 Bytes    IP-Version:4     SRC-MAC:02:42:ac:11:00:02     DST-MAC:02:42:01:73:02:
cb    SRC-IP: 172.17.0.2     DST-IP:  104.85.124.77
[2022-11-08 18:05:59.005701]  ICMP-IN:64 Bytes    IP-Version:4           SRC-MAC:02:42:01:73:02:cb     DST-MAC:02:42:ac
:11:00:02    SRC-IP: 104.85.124.77    DST-IP:  172.17.0.2
[2022-11-08 18:05:59.755786]  ICMP-OUT:64 Bytes    IP-Version:4     SRC-MAC:02:42:ac:11:00:02     DST-MAC:02:42:01:73:02:
cb    SRC-IP: 172.17.0.2     DST-IP:  142.250.196.78
[2022-11-08 18:05:59.782132]  ICMP-IN:64 Bytes    IP-Version:4           SRC-MAC:02:42:01:73:02:cb     DST-MAC:02:42:ac
:11:00:02    SRC-IP: 142.250.196.78    DST-IP:  172.17.0.2
[2022-11-08 18:05:59.981864]  ICMP-OUT:64 Bytes    IP-Version:4     SRC-MAC:02:42:ac:11:00:02     DST-MAC:02:42:01:73:02:
```

OUTPUT 1.out Case 3

```
/packet-sniffing # cat 1.out
[2022-11-08 17:52:59.798672]  ICMP-OUT:64 Bytes    IP-Version:4     SRC-MAC:02:42:ac:11:00:02     DST-MAC:02:42:01:73:02:
cb    SRC-IP: 172.17.0.2     DST-IP:  142.250.67.46
[2022-11-08 17:52:59.822412]  ICMP-IN:64 Bytes    IP-Version:4           SRC-MAC:02:42:01:73:02:cb     DST-MAC:02:42:ac
:11:00:02    SRC-IP: 142.250.67.46    DST-IP:  172.17.0.2
[2022-11-08 17:53:00.799321]  ICMP-OUT:64 Bytes    IP-Version:4     SRC-MAC:02:42:ac:11:00:02     DST-MAC:02:42:01:73:02:
cb    SRC-IP: 172.17.0.2     DST-IP:  142.250.67.46
[2022-11-08 17:53:00.846538]  ICMP-IN:64 Bytes    IP-Version:4           SRC-MAC:02:42:01:73:02:cb     DST-MAC:02:42:ac
:11:00:02    SRC-IP: 142.250.67.46    DST-IP:  172.17.0.2
[2022-11-08 17:53:01.799516]  ICMP-OUT:64 Bytes    IP-Version:4     SRC-MAC:02:42:ac:11:00:02     DST-MAC:02:42:01:73:02:
cb    SRC-IP: 172.17.0.2     DST-IP:  142.250.67.46
[2022-11-08 17:53:01.825883]  ICMP-IN:64 Bytes    IP-Version:4           SRC-MAC:02:42:01:73:02:cb     DST-MAC:02:42:ac
:11:00:02    SRC-IP: 142.250.67.46    DST-IP:  172.17.0.2
[2022-11-08 17:53:02.799533]  ICMP-OUT:64 Bytes    IP-Version:4     SRC-MAC:02:42:ac:11:00:02     DST-MAC:02:42:01:73:02:
cb    SRC-IP: 172.17.0.2     DST-IP:  142.250.67.46
[2022-11-08 17:53:02.824062]  ICMP-IN:64 Bytes    IP-Version:4           SRC-MAC:02:42:01:73:02:cb     DST-MAC:02:42:ac
:11:00:02    SRC-IP: 142.250.67.46    DST-IP:  172.17.0.2
[2022-11-08 17:53:03.800240]  ICMP-OUT:64 Bytes    IP-Version:4     SRC-MAC:02:42:ac:11:00:02     DST-MAC:02:42:01:73:02:
cb    SRC-IP: 172.17.0.2     DST-IP:  142.250.67.46
[2022-11-08 17:53:03.832389]  ICMP-IN:64 Bytes    IP-Version:4           SRC-MAC:02:42:01:73:02:cb     DST-MAC:02:42:ac
:11:00:02    SRC-IP: 142.250.67.46    DST-IP:  172.17.0.2
[2022-11-08 17:53:04.800383]  ICMP-OUT:64 Bytes    IP-Version:4     SRC-MAC:02:42:ac:11:00:02     DST-MAC:02:42:01:73:02:
cb    SRC-IP: 172.17.0.2     DST-IP:  142.250.67.46
[2022-11-08 17:53:04.825298]  ICMP-IN:64 Bytes    IP-Version:4           SRC-MAC:02:42:01:73:02:cb     DST-MAC:02:42:ac
:11:00:02    SRC-IP: 142.250.67.46    DST-IP:  172.17.0.2
```

INPUT nslookup in case 3

```
docker exec -it aaa4697a65ffec75bc0beecdec217f1845b859c13332d4f25fb0f07a9260dd5a /bin/sh
Name:   sastra.ac.in
Address: 115.240.199.108
Name:   sastra.ac.in
Address: 139.167.67.12
Name:   sastra.ac.in
Address: 14.139.181.236

Non-authoritative answer:
*** Can't find sastra.ac.in: No answer

/ # nslookup google.com
Server:         192.168.65.5
Address:        192.168.65.5:53

Non-authoritative answer:
Name:   google.com
Address: 2404:6800:4007:823::200e

Non-authoritative answer:
Name:   google.com
Address: 172.217.163.206

/ # nslookup 8.8.8.8
Server:         192.168.65.5
Address:        192.168.65.5:53

Non-authoritative answer:
*** Can't find 8.8.8.8.in-addr.arpa: No answer

/ #
```

TCP ports are analysed

TCP PORTS ARE ANALYZED

- For 443 TCP connections use below command

  wget https://www.google.com (wget means web get provided it is permitted)
- wget www.google.com -o /dev/null (o is output)
- The output will be in /dev/null which means null device, i.e. it acts as a vacuum for errors
- Use same command with http instead of https so connection will be on 80 port with TCP:

  watch wget http://www.google.com -o /dev/null
- This will download index.html by default and make it null.
- After this, check output file you can see 443 or 80 connection as per your command.

OUTPUT TCP port 443

```
Every 2.0s: wget https://www.google.com -O/dev/null                                202

Connecting to www.google.com (172.217.167.132:443)
saving to '/dev/null'
null                 100% |*******************************************************************************************************| 16235  0:00:00 ETA
'/dev/null' saved
```

- That's size of the index file, index.html
- It is downloading file but saved in /dev/null so whatever you download there it will nullify
- And you wont see anything as downloaded
- Use ctrl c to stop
- watch(a Linux command) will execute same command for every 2 seconds
- So you will continue to get same packets in the packet sniffer

ANALYSIS

- The output shows many fields such as source IP, destination IP, source MAC, destination MAC, IP version, date, time(hour, minute, second and microseconds included) & packets inbound or outbound and their protocol.
- If source address in packet is IP address of my machine then it is an outgoing packet.
- If destination address in packet is IP address of my machine then it is an incoming packet.
- If a packet is missing you will get request timed out in a ping terminal.
- We cannot know if a packet is corrupted or not because with this code we are only checking for the credentials and not the content.
- But it could be detected with checksum in network layers,
- If sender end and receiver end checksum do not match then the packet may be corrupted in transit.

- Although Scapy is all powerful, it’s takes a lot of memory when reading packets so analysing larger packet will take toll on your system memory.

FUTURE SCOPE OF WORK

- Penetration testing using Scapy
    - A penetration test (pen test) is an authorized simulated attack performed on a computer system to evaluate its security.
    - Penetration tests are used to identify the level of technical risk emanating from software and hardware vulnerabilities.
    - Scapy can be used for penetration testing
- Packet manipulation and crafting
    - Packet Editing / Crafting is the modification of created or captured packets. This involves modifying packets in manners which are difficult or impossible to do in the Packet Assembly stage, such as modifying the payload of a packet
    - You can write servers, routers, firewalls, network tracing tools, and pretty much anything in Scapy, due to its ability to sniff, send, and respond to packets. All of these properties make it very useful for network based attacks.
- Detection of hidden wireless networks; Wi-Fi
    - Scapy can as well be used to detect hidden WiFi networks