

practice-1

1. Topic: network traffic analysis via packets capturing and detection of anomalies / possible attacks
 - capturing and analyzing network traffic
 - identifying normal communication patterns
 - detecting possible anomalies or malicious activities
 - using professional networking tools

Solution and practice

1. Installed and configured Wireshark in Windows (winget install Wireshark-Foundation.Wireshark)
 - source code: <https://gitlab.com/wireshark/wireshark>
 - Wireshark in CLI form: tshark, tcpdump, or alternative
 - However, Wireshark deprecated native Python binding / API for interaction (reference: <https://wiki.wireshark.org/python>)
 - it is possible to use Lua (<https://wiki.wireshark.org/lua>)
 - that is why, downloaded additional library: pyshark (<https://github.com/KimiNewt/pyshark>)
 - network adapter need to be set in promiscuous mode for proper packet capture

2. setup environment for analysis:

```
python3 -m venv venv
source .venv/bin/activate
pip install -r requirements.txt
```

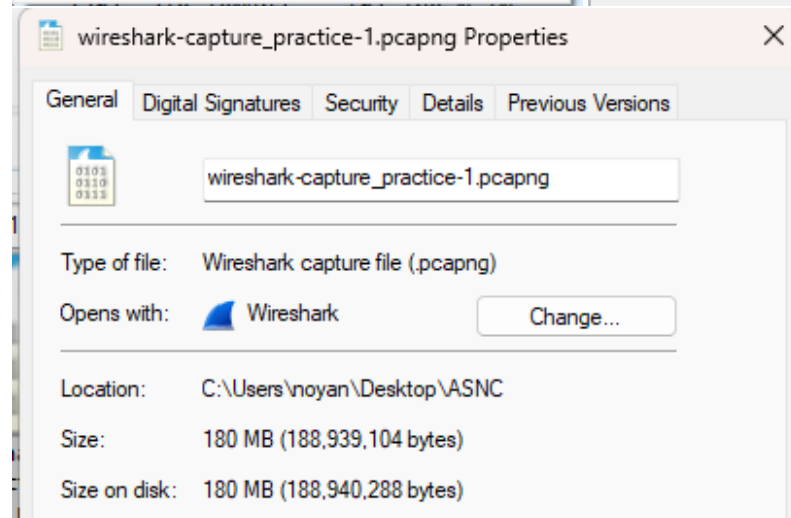
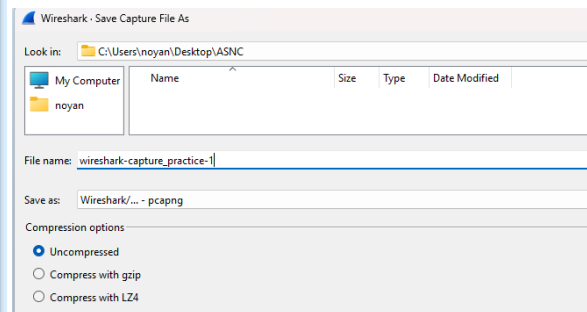
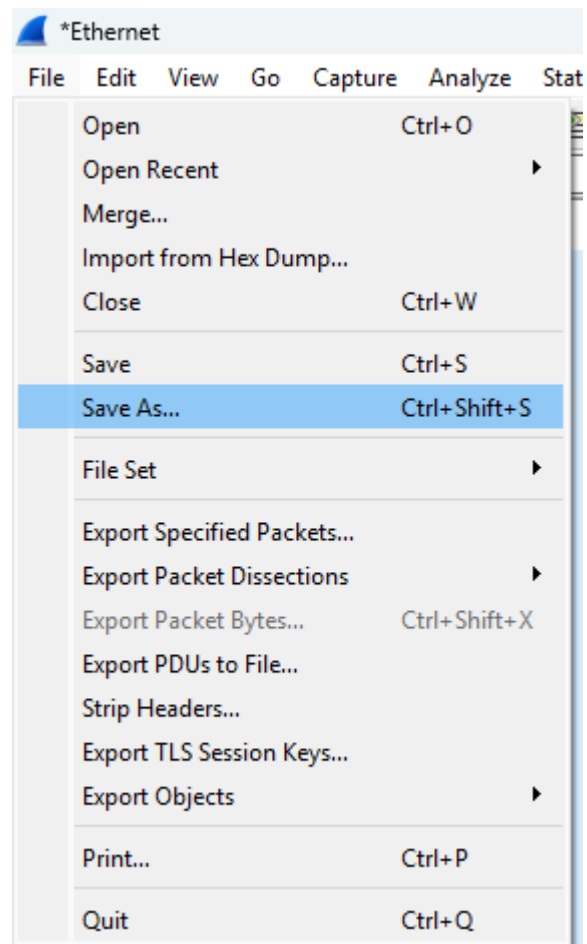
3. Captured traffic from Windows machine for 5+ minutes with web browsing, file downloads, DNS requests, VPN connection, etc.
 - or download traffic sample like <https://www.wireshark.org/resources/sample-captures>
 - saved file in .pcap format in /local_data
 - analyzed protocols: most frequent protocols in captured data in summary table (protocol / percentage of traffic)

– “wireshark” -> “statistics” -> “protocol hierarchy”

The screenshot displays the Wireshark network protocol analyzer interface. The top menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Windows, Tools, and Help. Below the menu is a toolbar with icons for various functions. The main window is divided into several panes:

- Packet List:** A table showing captured packets. The first packet is an ARP request from 192.168.0.10 to 192.168.0.1.
- Packet Details:** A hierarchical view of the selected packet's structure, showing Ethernet II, Internet Protocol Version 4, and Internet Control Message Protocol (ICMP) Echo (ping request).
- Packet Bytes:** A hex dump and ASCII representation of the packet data.

The bottom status bar indicates the current capture status: "Ready to load or capture".



The image shows the Wireshark interface with the 'Protocol Hierarchy Statistics' pane on the left and the 'Packet List' pane on the right.

Protocol Hierarchy Statistics:

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Blk/s	End Packets	End Bytes	End Blk/s	PDUs
Total	100.0	239292	100.0	15893516	3245.0	0	0	0	239292
Ethernet	100.0	239292	1.9	3357594	94.4	0	0	0	239292
Internet Protocol Version 4	100.0	239292	2.6	4764768	1274.0	0	0	0	239292
User Datagram Protocol	38.9	111473	0.6	1017784	28.6	0	0	0	111473
Data	48.8	119225	37.9	6953320	1723.4	119225	6953320	1723.4	119225
OSPFv2 LSP	4.8	11497	5.8	1043441	282.4	11497	1036640	269.4	11497
Domain Name System	0.3	607	0.0	42338	1069	607	42338	1069	607
Multicast Domain Name System	0.0	60	0.0	14599	367	60	14599	367	60
Teredo IPv6 over UDP tunneling	0.0	23	0.0	2080	52	0	0	0	23
Internet Protocol Version 6	0.0	23	0.0	1000	25	1	40	1	23
Internet Control Message Protocol v6	0.0	24	0.0	672	16	24	672	16	24
NatBios Name Service	0.0	21	0.0	1050	26	21	1050	26	21
Simple Service Discovery Protocol	0.0	16	0.0	4996	124	16	4996	124	16
Network Time Protocol	0.0	14	0.0	672	16	14	672	16	14
Sham in Home Streaming Discovery Protocol	0.0	6	0.0	711	17	6	711	17	6
NatBios Datagram Service	0.0	1	0.0	62	2	0	0	0	1
SMB (Server Message Block Protocol)	0.0	1	0.0	119	2	0	0	0	1
SMB Hostid Protocol	0.0	1	0.0	25	0	0	0	0	1
Microsoft Windows Browser Protocol	0.0	1	0.0	33	0	1	33	0	1
Malformed Packet	0.0	1	0.0	0	0	1	0	0	1
Transmission Control Protocol	0.0	1	1.5	279576	69.4	1494	208050	52.4	107755
Transport Layer Security	0.0	1	36.1	6330863	2094.4	3241	5143860	1304.4	34974
Data	0.2	464	0.4	63236	174	464	63236	174	464
Hypertext Transfer Protocol	0.0	36	0.0	76121	1913	33	9876	250	36
Media Type	0.0	22	0.0	6120	159	22	6120	159	22
Online Certificate Status Protocol	0.0	1	0.0	314	7	1	314	7	1
Internet Control Message Protocol	0.0	8	0.0	320	8	8	320	8	8
Internet Group Management Protocol	0.0	2	0.0	32	0	2	32	0	2
Internet Protocol Version 6	0.0	53	0.0	2098	52	0	0	0	53
User Datagram Protocol	0.0	50	0.0	400	10	0	0	0	50
Multicast Domain Name System	0.0	50	0.0	12373	311	50	12373	311	50
Internet Control Message Protocol v6	0.0	2	0.0	96	1	2	96	1	2
Link Layer Discovery Protocol	0.0	1	0.0	53	1	1	53	1	1
Address Resolution Protocol	0.0	1	0.0	28	0	1	28	0	1

Packet List:

No.	Time	Source	Destination	Protocol	Length	Info
72517	1.89429711	192.168.0.1	192.168.0.10	DNS	80	Standard query response 0x0f0 Server: failure HTTPS wakefiletutorial.com
96579	240.153190	192.168.0.1	192.168.0.10	DNS	141	Standard query response 0x7580 No such name A wpad.apsetup.link SOA dns17.hichina.com
96579	240.742176	192.168.0.1	192.168.0.10	DNS	141	Standard query response 0x20ff No such name A wpad.apsetup.link SOA dns17.hichina.com
101369	245.136528	192.168.0.1	192.168.0.10	DNS	141	Standard query response 0x20ff No such name A wpad.apsetup.link SOA dns17.hichina.com
156419	353.143679	192.168.0.1	192.168.0.10	DNS	141	Standard query response 0xfaf6 No such name A wpad.apsetup.link SOA dns17.hichina.com
237849	290.559807	192.168.0.1	192.168.0.10	DNS	141	Standard query response 0xc150 No such name A wpad.apsetup.link SOA dns17.hichina.com
237931	305.568262	192.168.0.1	192.168.0.10	DNS	141	Standard query response 0x80b3 No such name A wpad.apsetup.link SOA dns17.hichina.com

Packet Details:

Frame 72517: 80 bytes on wire (640 bits), 80 bytes captured (640 bits) on interface 0

Ethernet II, Src: YingdaKangTe_1b12b139 (28:90:d8:1b:2b:39), Dst: MicroStarINT_6e:04:

Internet Protocol Version 4, Src: 192.168.0.1, Dst: 192.168.0.10

User: Datagram Protocol, Src Port: 53, Dst Port: 53463

Domain Name System (response)

0000 04 7c 16 6e d4 9d 20 98 08 1b 2b 39 08 00 45 00 | n + P : E

0010 00 42 ae 44 40 00 40 11 00 00 c0 a0 00 01 c0 a0 | B 0 0 #

0020 00 00 00 35 00 07 00 2e ef 90 0a f0 81 82 00 01 | . B

0030 00 00 00 00 00 10 6d 61 69 65 66 69 6c 65 74 | m a k e f i l e t

0040 75 74 67 72 69 61 6c 03 63 67 6d 00 41 00 01 | u t o r i a l . c o m - A -

- #### 4. Anomaly and attack detection
- detecting possible anomalies in the traffic (e.g., repeated failed DNS requests, ARP spoofing attempts, unusual port scanning activity)
 - suspicious flows with malicious activity
 - provided practical graph representation in main.py to identify connections

