ESOF 422:

Advanced Software Engineering: Cyber Practices

Network Forensics, Course Conclusion

Reese Pearsall

Spring 2025

Announcements

HW6 is fully done. No part 3 (sorry).

- Friday will be an optional workday for HW6 (no new materials)

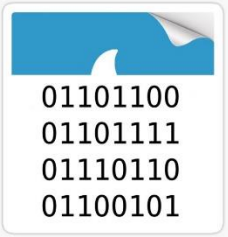
Final Exam: Wednesday May 7th 2:00 PM – 3:50 PM

Fill out the course evaluation

Network Forensics

Network Forensics- digital evidence is coming from traffic of a computer network

Able to see data leaving and coming to machine
May be helpful for timelining



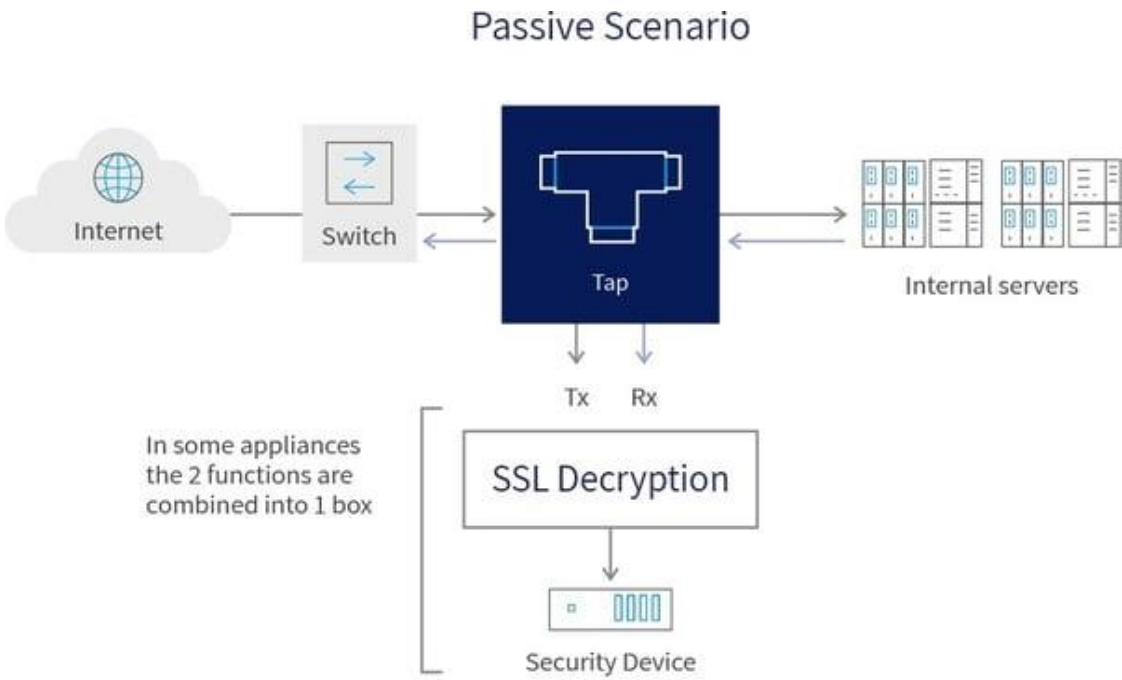
You will usually be working with PCAP files

Challenges with Network Forensics

- Capturing network traffic for analysis is becoming less and less feasible due to data transmission and storage limitations
- Takes a long time to sift through thousands of packets

- 100 MBPS x 7 days per week = 7.56 TB
- 10 GBPS x 7 days per week = 756 TB

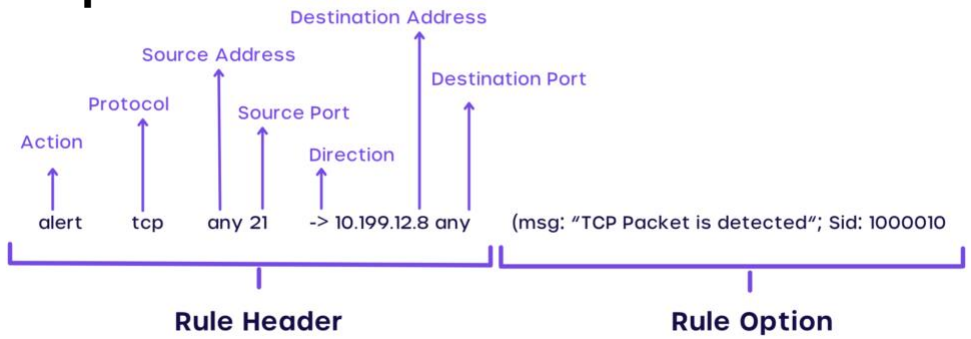
Think about keeping track of this for all machines at your organization…



- May have to decrypt in-line/MITM traffic
- Difficult to capture in a cloud network

SNORT

A set of rules to detect suspicious/malicious traffic



If we satisfy a rule, an alert can be generated or that packet can be logged

Alerts for malicious IP addresses, typical malware behavior, log unusual ports

This rule would generate an alert for web traffic that is not using the secure version of HTTP

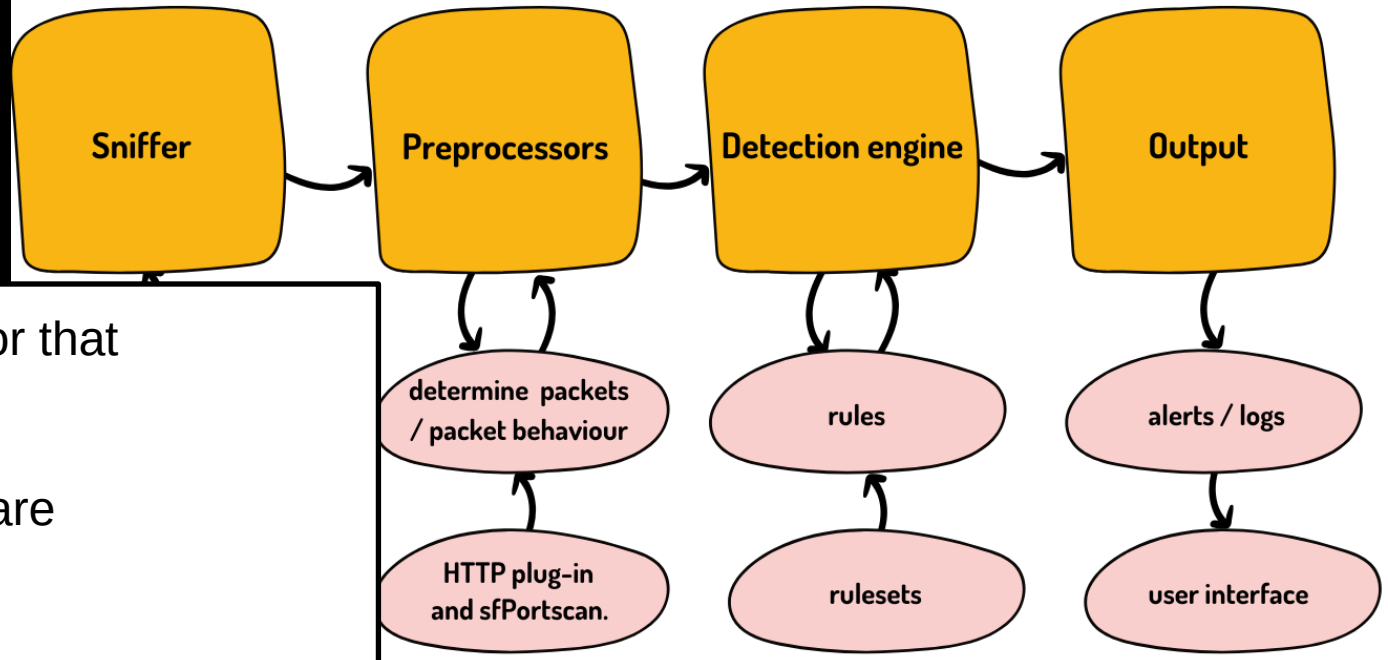
```
alert tcp any any → any 80 (msg:"Sus web traffic"; flags: S; sid: 100;)
```

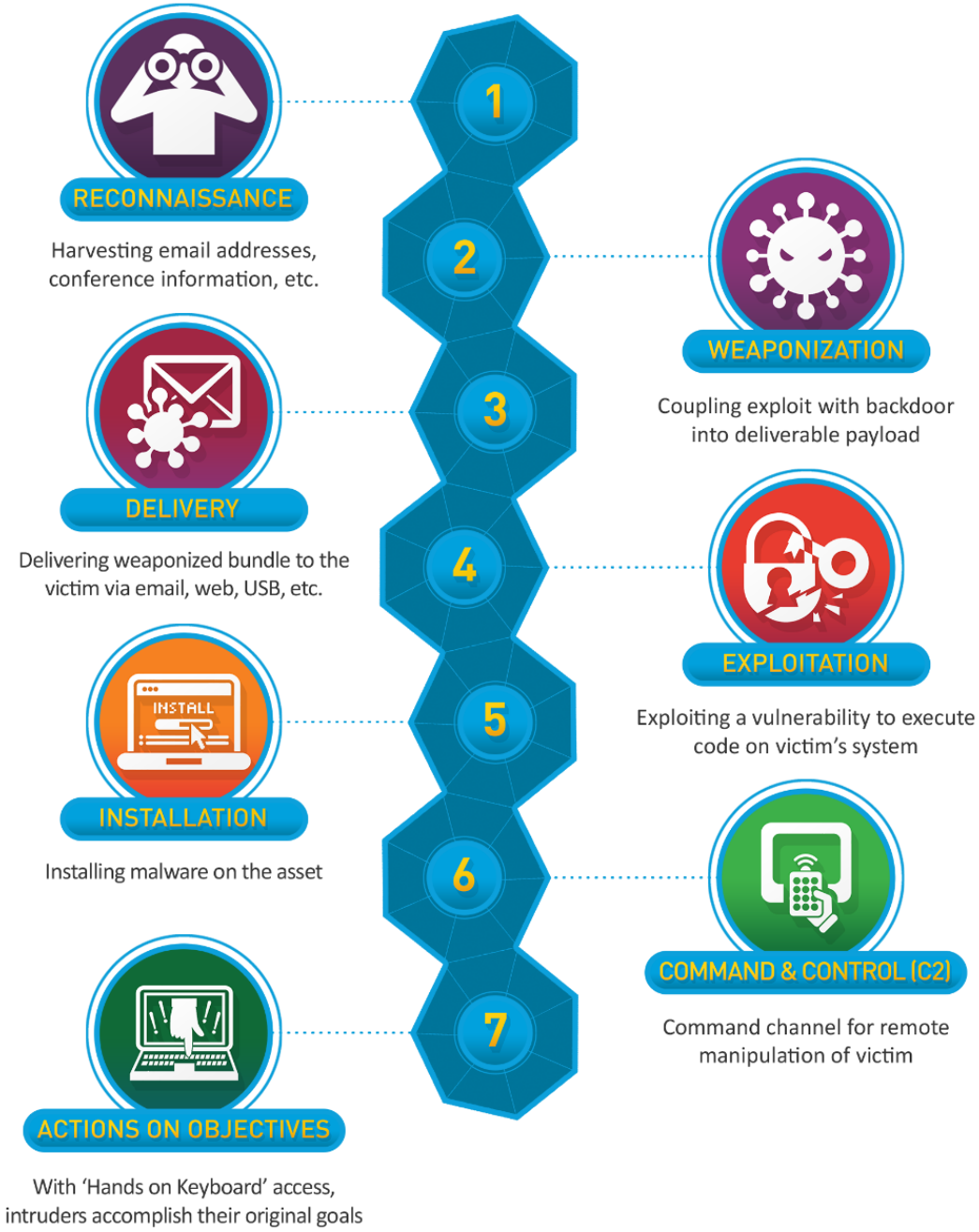
The core component that collects and identifies packet structures from network traffic.

These analyze and modify packets to determine their type or behavior before passing them to the detection engine.

This compares packet data against a predefined ruleset to identify potential threats. Packets that match the rules are forwarded to the output.

Logs and triggers alerts based on detected threats. Logs can be saved in various formats and locations, and user interfaces like Snorby or ACID help manage and view this data.



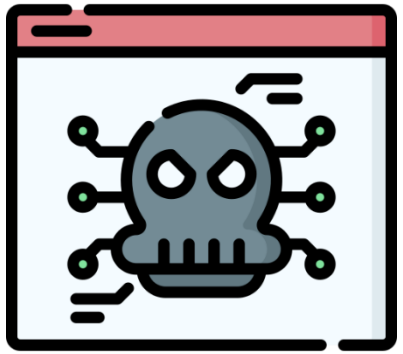


The **Cyber Kill Chain** describes the typical steps a malicious actor carries out to conduct a cyber attack

In network forensic investigations, we can see evidence of these steps occurring!

Common Malware Behavior (Network Traffic)

- Downloading malware/malware dropper through HTTP
- Communicate with a C2 server
  - Cobalt Strike
  - Metasploit
  - Covenant
- Unusual Outbound Traffic
- Malware Beaconing
- Suspicious DNS Queries to attacker domains
- Spike in connections to other devices (lateral movement)
- Failed login attempts/authentication



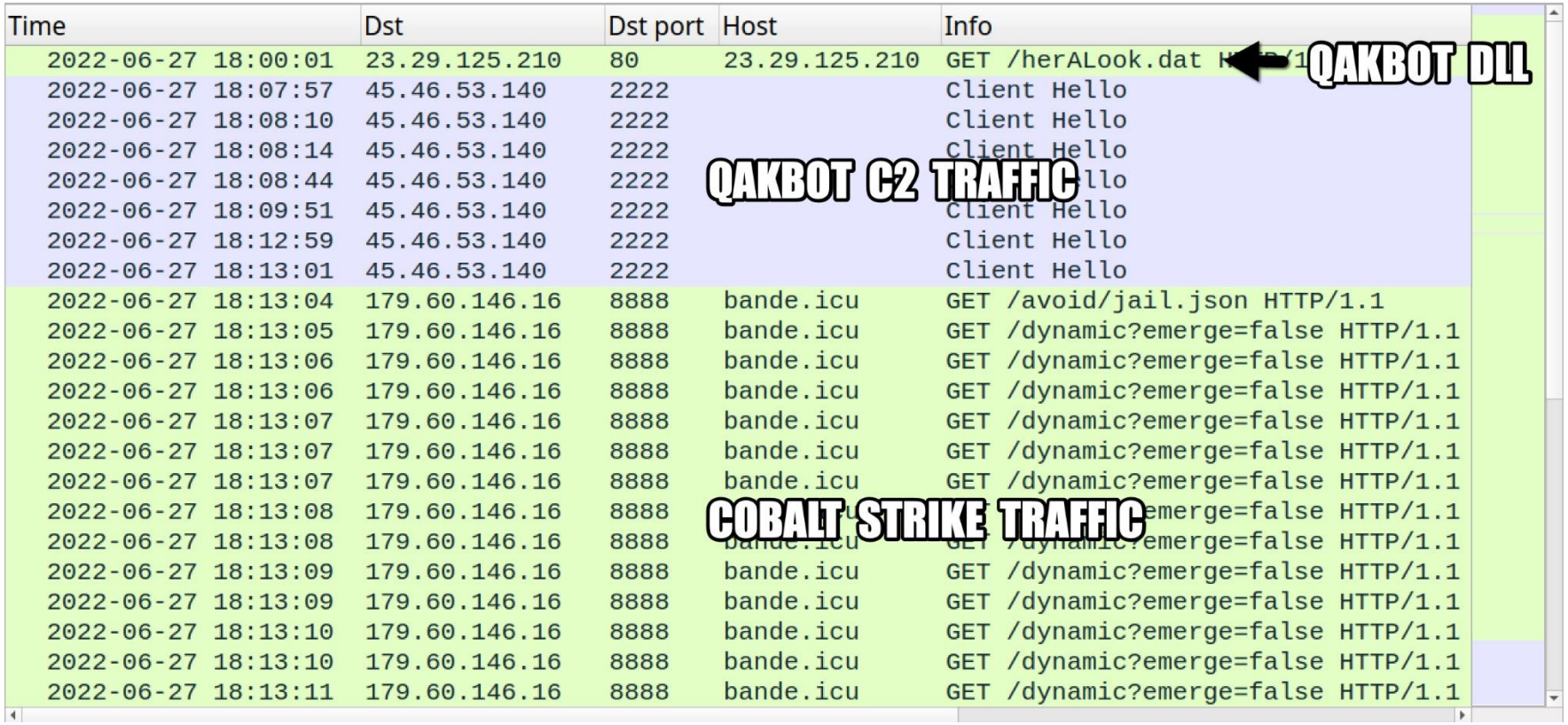
Cobalt Strike C2 Server Threat Hunting

- Use of default SSL certificates
- Increased usage of port 443, 80, 8443 with foreign IP address
- Check default responses for HTTP 404 and DNS Queries

Malware File Types to Check for

- `.exe` files - Windows executable files
- `.dll` files - Dynamic Linked Libraries
- `.msi` files – Windows installers
- `.bat` files – Windows command line script
- `.vbs` scripts – Visual Basic Scripts
- `.js` scripts – Javascript file
- `.docx` files – Microsoft Word Document (can contain macros)
- `.xlsx` files – Microsoft Excel Spreadsheet (can contain macros)
- `.zip` files – Compressed Archive Files (may have scripts when unzipped)
- `.pdf` files – Can contain suspicious links, or a PDF reader vulnerability

Wireshark

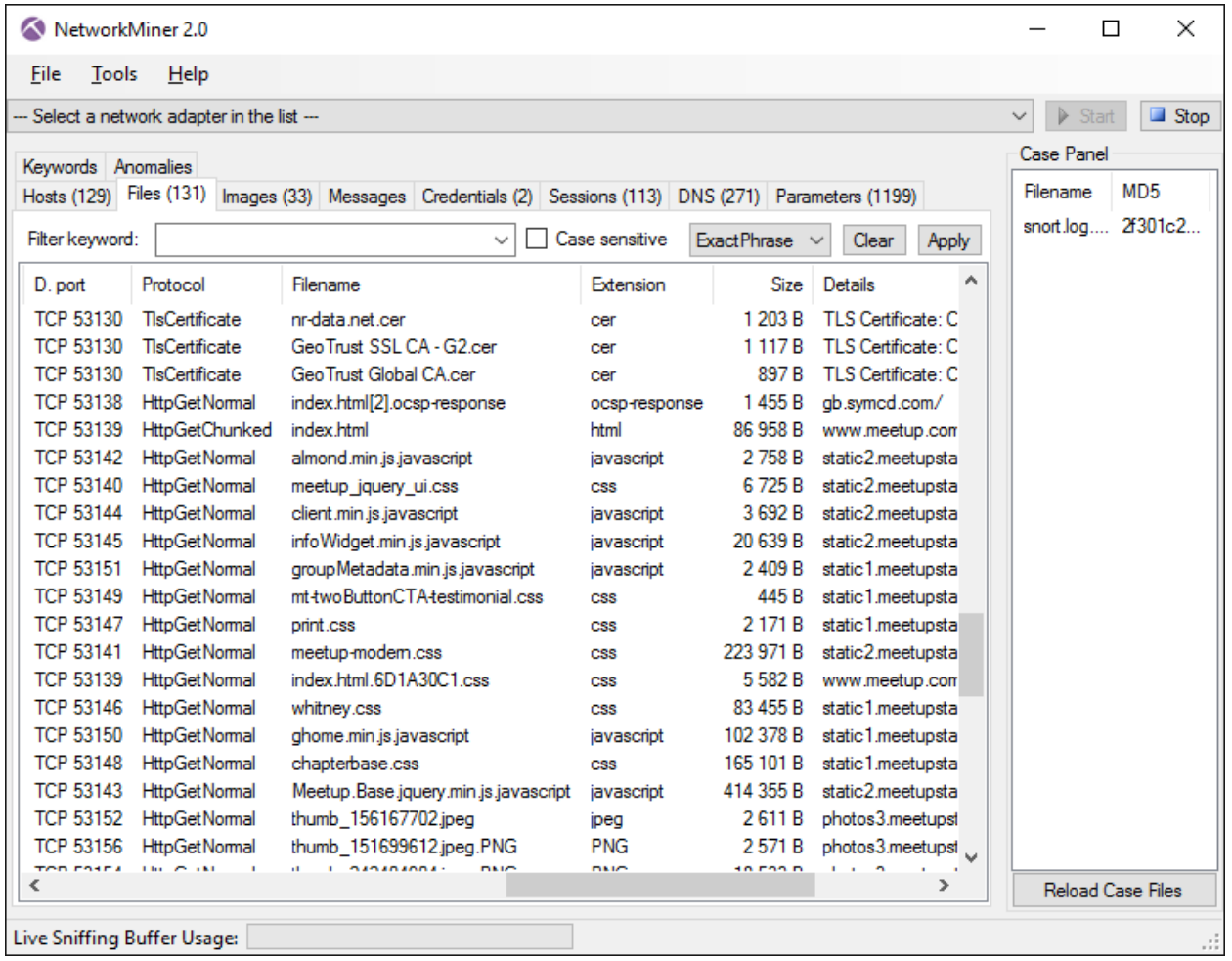


Time	Dst	Dst port	Host	Info
2022-06-27 18:00:01	23.29.125.210	80	23.29.125.210	GET /herALook.dat H[illegible]/1
2022-06-27 18:07:57	45.46.53.140	2222		Client Hello
2022-06-27 18:08:10	45.46.53.140	2222		Client Hello
2022-06-27 18:08:14	45.46.53.140	2222		Client Hello
2022-06-27 18:08:44	45.46.53.140	2222		Client Hello
2022-06-27 18:09:51	45.46.53.140	2222		Client Hello
2022-06-27 18:12:59	45.46.53.140	2222		Client Hello
2022-06-27 18:13:01	45.46.53.140	2222		Client Hello
2022-06-27 18:13:04	179.60.146.16	8888	bande.icu	GET /avoid/jail.json HTTP/1.1
2022-06-27 18:13:05	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1
2022-06-27 18:13:06	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1
2022-06-27 18:13:06	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1
2022-06-27 18:13:07	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1
2022-06-27 18:13:07	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1
2022-06-27 18:13:07	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1
2022-06-27 18:13:08	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1
2022-06-27 18:13:08	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1
2022-06-27 18:13:09	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1
2022-06-27 18:13:09	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1
2022-06-27 18:13:10	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1
2022-06-27 18:13:10	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1
2022-06-27 18:13:11	179.60.146.16	8888	bande.icu	GET /dynamic?emerge=false HTTP/1.1

We can use Wireshark to identify **specific** malicious packets, and find the exact moment where the infection started

C2 server information, Victim information, other relevant evidence

NetworkMiner

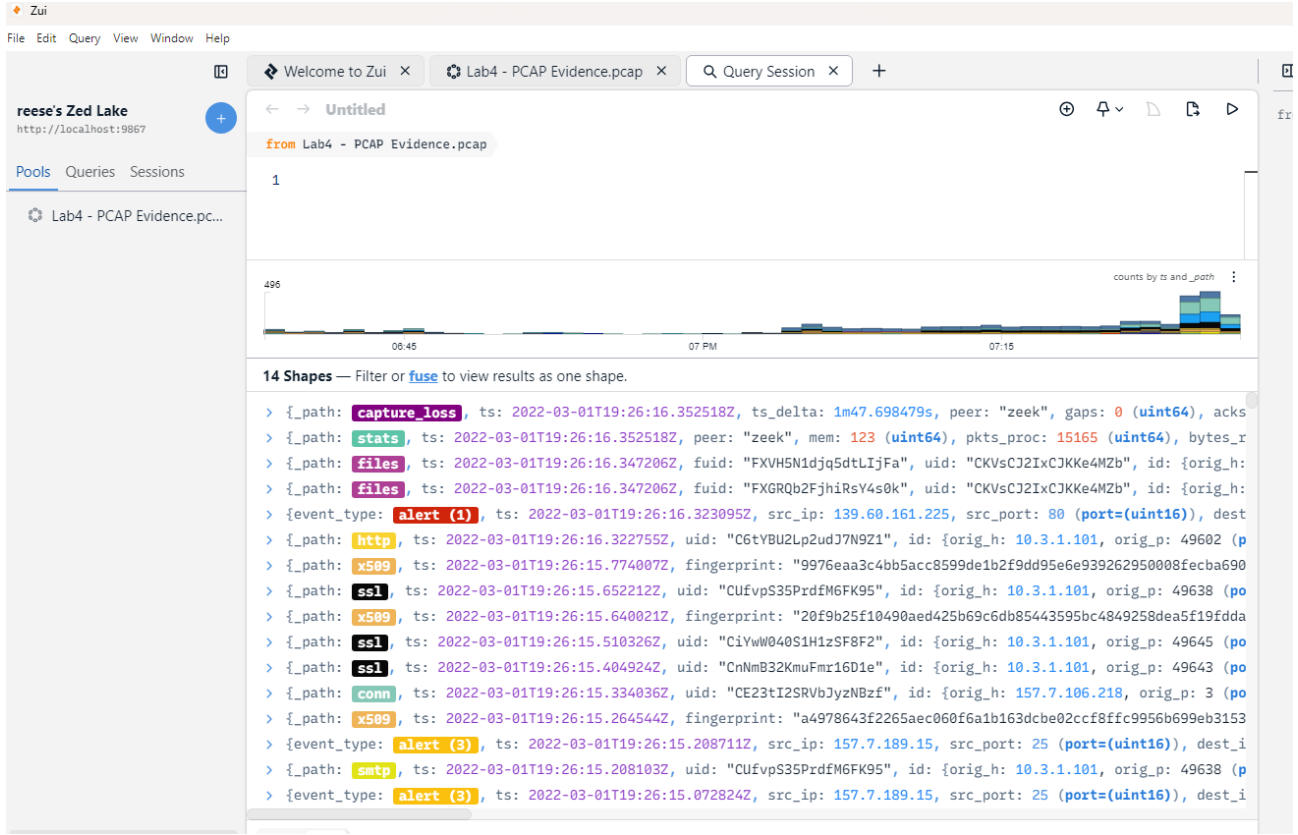


NetworkMiner is a automated flow analysis tool that will identify all hosts, downloaded files, emails, logins from a pcap file and attempt to reassemble them for analysis

Very powerful tool that can provide many helpful insights during an investigation

NetworkMiner should always be run in a sandbox environment (VM) that is disconnected from the network

Zui



Zui (formerly known as **brim**) is a automated flow analysis tool that will identify any suspicious packets, emails, certificates, files from a pcap file and create a *timeline*

Won't assemble the files, so it will be safer to use

Can help identify potential IOCs

First, let's try to find the piece of malware that was installed

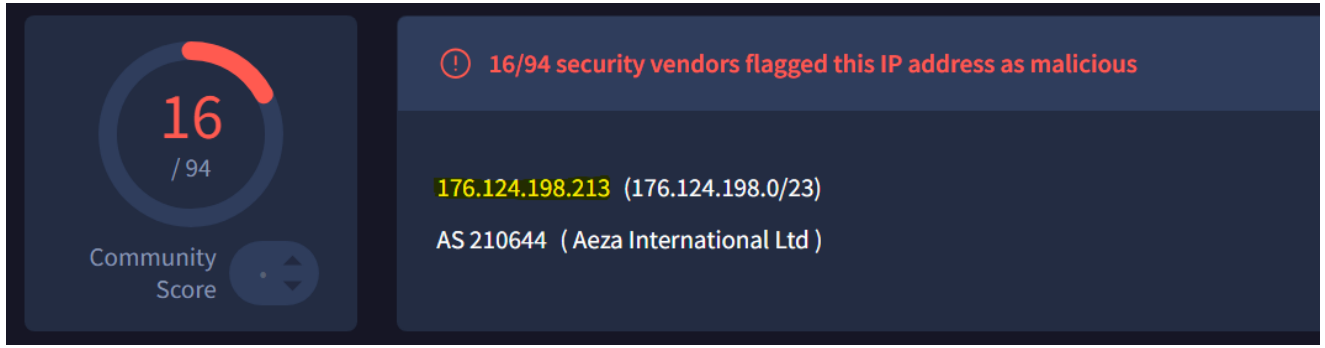
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

http

Time	Destination	Protocol	Info
12.674904	209.197.3.8	HTTP	GET /filestreamingservice/files/9f86601f-2d70-42ad-
12.729162	10.13.13.99	HTTP	HTTP/1.1 200 OK (application/x-chrome-extension)
39.446064	176.124.198.213	HTTP	GET /Fs8Py/eKTYt3dRbEXw HTTP/1.1
41.067893	10.13.13.99	HTTP	HTTP/1.1 200 OK (image/gif)

This HTTP Get request seems suspicious

- Strange URL in the header



VirusTotal confirms that this IP address is malicious

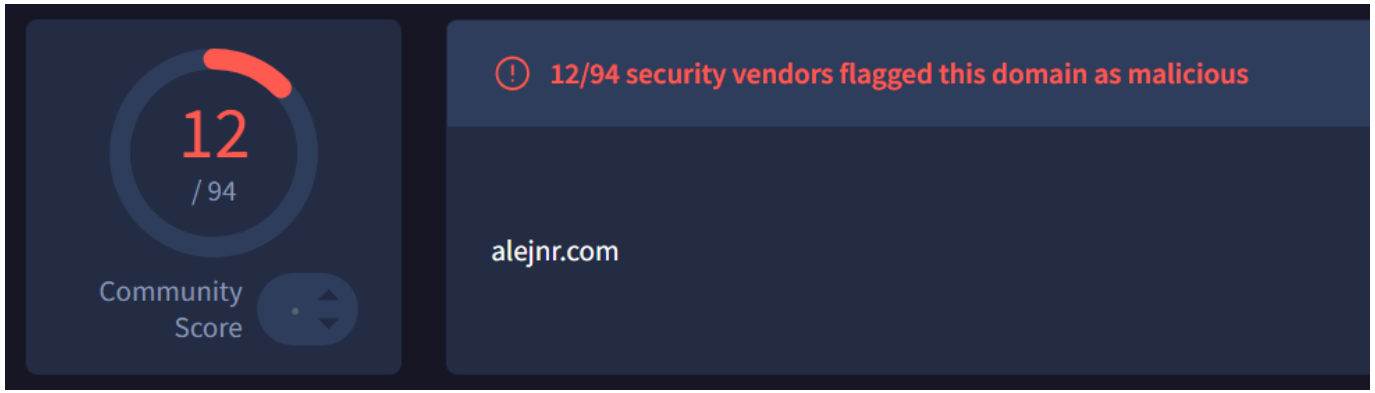
Investigating DNS

Many of the DNS requests were made to “normal” website. However, this domain seemed a bit strange and this request was made shortly before the malicious HTTP request

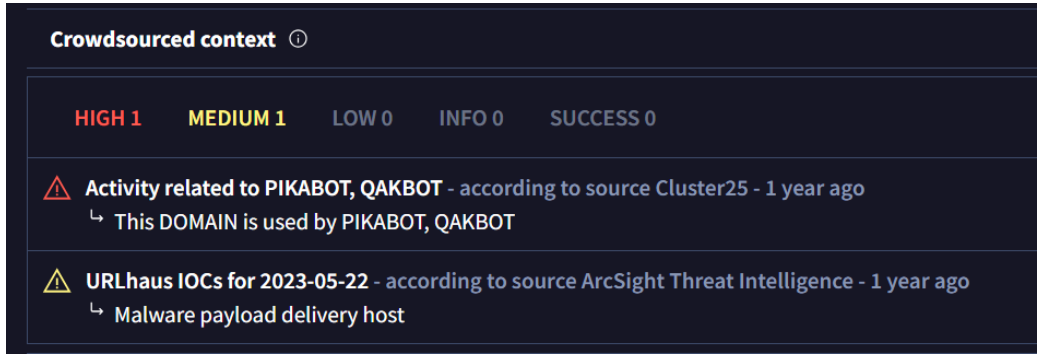
dns

Time	Destination	Protocol	Info
0.000000	10.13.13.3	DNS	Standard query 0x222a A alejnr.com
0.000186	10.13.13.3	DNS	Standard query 0x40a4 HTTPS alejnr.com
0.056796	10.13.13.99	DNS	Standard query response 0x40a4 HTTPS alejnr.com SOA pdns1.registrar-servers.com
0.056891	10.13.13.99	DNS	Standard query response 0x222a A alejnr.com A 162.213.255.36

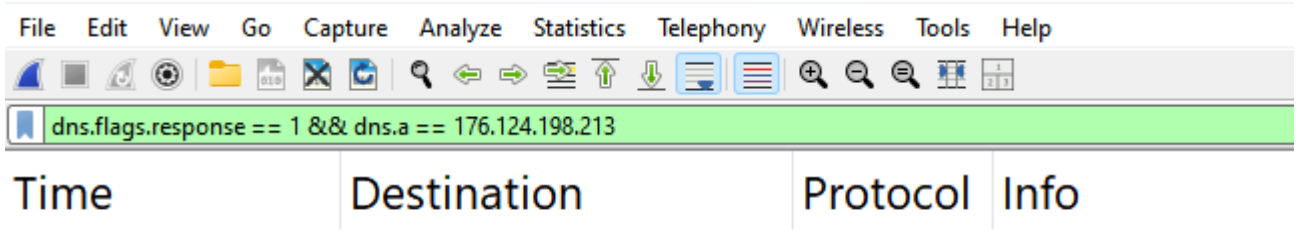
This domain had been flagged as malicious in the past



VirusTotal also provides context as to why it was flagged as malicious



Where did this IP address come from?



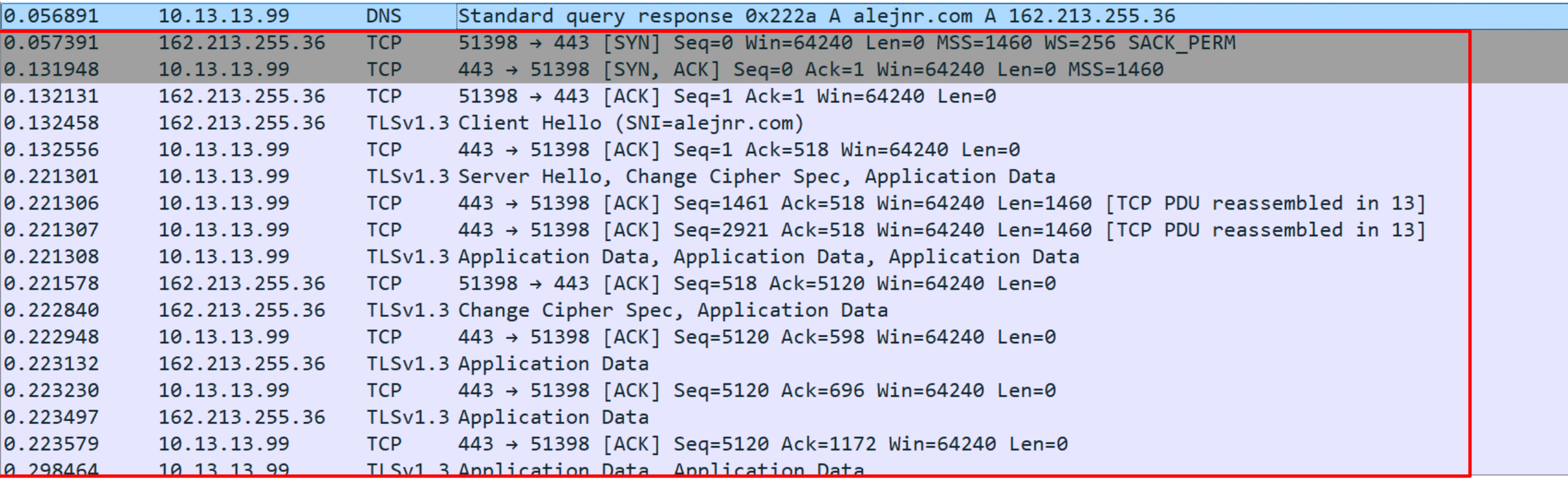
This malicious IP did not come from a DNS response.

This could be an indicator that another piece of malware contacted that IP addressed and issued the HTTP request (malware dropper)

Many of the DNS requests were made to “normal” website. However, this domain seemed a bit strange and this request was made shortly before the malicious HTTP request

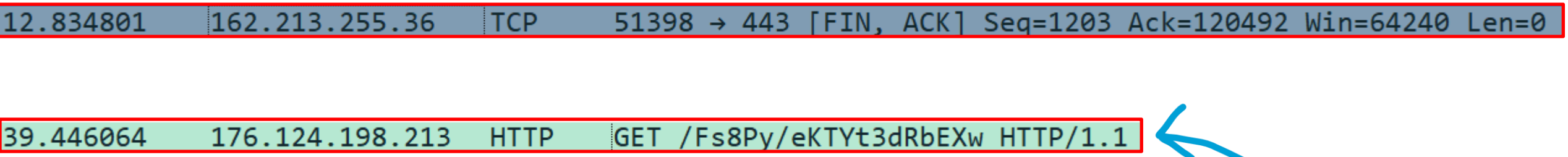
dns

Time	Destination	Protocol	Info
0.000000	10.13.13.3	DNS	Standard query 0x222a A alejnr.com
0.000186	10.13.13.3	DNS	Standard query 0x40a4 HTTPS alejnr.com
0.056796	10.13.13.99	DNS	Standard query response 0x40a4 HTTPS alejnr.com SOA pdns1.registrar-servers.com
0.056891	10.13.13.99	DNS	Standard query response 0x222a A alejnr.com A 162.213.255.36



0.056891	10.13.13.99	DNS	Standard query response 0x222a A alejnr.com A 162.213.255.36
0.057391	162.213.255.36	TCP	51398 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
0.131948	10.13.13.99	TCP	443 → 51398 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
0.132131	162.213.255.36	TCP	51398 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
0.132458	162.213.255.36	TLSv1.3	Client Hello (SNI=alejnr.com)
0.132556	10.13.13.99	TCP	443 → 51398 [ACK] Seq=1 Ack=518 Win=64240 Len=0
0.221301	10.13.13.99	TLSv1.3	Server Hello, Change Cipher Spec, Application Data
0.221306	10.13.13.99	TCP	443 → 51398 [ACK] Seq=1461 Ack=518 Win=64240 Len=1460 [TCP PDU reassembled in 13]
0.221307	10.13.13.99	TCP	443 → 51398 [ACK] Seq=2921 Ack=518 Win=64240 Len=1460 [TCP PDU reassembled in 13]
0.221308	10.13.13.99	TLSv1.3	Application Data, Application Data, Application Data
0.221578	162.213.255.36	TCP	51398 → 443 [ACK] Seq=518 Ack=5120 Win=64240 Len=0
0.222840	162.213.255.36	TLSv1.3	Change Cipher Spec, Application Data
0.222948	10.13.13.99	TCP	443 → 51398 [ACK] Seq=5120 Ack=598 Win=64240 Len=0
0.223132	162.213.255.36	TLSv1.3	Application Data
0.223230	10.13.13.99	TCP	443 → 51398 [ACK] Seq=5120 Ack=696 Win=64240 Len=0
0.223497	162.213.255.36	TLSv1.3	Application Data
0.223579	10.13.13.99	TCP	443 → 51398 [ACK] Seq=5120 Ack=1172 Win=64240 Len=0
0.298464	10.13.13.99	TLSv1.3	Application Data, Application Data

Immediately after we resolve the host name of alejnr.com, we start a TCP/TLS connection with that



We stop talking to that malicious domain, and then shortly after the malicious HTTP GET request is done

Theory:

We reached out to this malicious domain alejnr.com and downloaded a malicious file which issued this HTTP request

What is in this malicious HTTP request?

```
39.446064    176.124.198.213   HTTP    GET /Fs8Py/eKTYt3dRbEXw HTTP/1.1
```

Request

```
41.067893    10.13.13.99       HTTP    HTTP/1.1 200 OK  (image/gif)
```

Response

```
41.067893    10.13.13.99       HTTP    HTTP/1.1 200 OK  (image/gif)
› Transmission Control Protocol, Src Port: 80, Dst Port: 51399, Seq: 542069, Ack: 178, Len: 1195
› [415 Reassembled TCP Segments (543263 bytes): #260(1288), #261(1460), #262(1460), #263(1460), #264(146
⌄ Hypertext Transfer Protocol
  › HTTP/1.1 200 OK\r\n
    Date: Mon, 22 May 2023 17:01:04 GMT\r\n
    Server: Apache/2.4.41 (Ubuntu)\r\n
    Accept-Ranges: bytes\r\n
  › Content-Length: 543048\r\n
    Keep-Alive: timeout=5, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: image/gif\r\n
    \r\n
```

Wireshark thinks this file is GIF

Wireshark notices that there is something weird going on…

```
› [Expert Info (Note/Malformed): HTTP body subdissector failed, trying heuristic subdissector]
```

What is in this malicious HTTP request?

```
41.067893    10.13.13.99       HTTP    HTTP/1.1 200 OK  (image/gif)
```

Right click → Follow TCP Stream

```
GET /Fs8Py/eKTYt3dRbEXw HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT; Windows NT 10.0; en-US) WindowsPowerShell/5.1.22621.963
Host: 176.124.198.213
Connection: Keep-Alive


HTTP/1.1 200 OK
Date: Mon, 22 May 2023 17:01:04 GMT
Server: Apache/2.4.41 (Ubuntu)
Accept-Ranges: bytes
Content-Length: 543048
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: image/gif

MZ.....@.....                .!..L.!This program cannot be run in DOS mode.

$.....u.S.....]U.....]a.....]`.....X.....]W.....]d.....]P.....]Q.....]V.....Rich.....PE..L....._P.....!..
.....k.....W.....@.....K.....\C.....f..... f.@).....
..@.....x.....text..... ..`.rdata.....@..@.data...p.c..P.....B.....@....rsrc....
D....f..P...N.....@..@.reloc..V....`k.....@..B.....
.....
.....\....%8.....U..V....\.....8....E..t
V.....^].....h....%8.....A.....U..V....h.....8....E..t
V.....^].....U..VW.}.W....,.....h....G..F._..^].....t.P....f.Y....|....H..    3..H.;.t..A..H..P....H..A...t....P..P.....U.....3.....
.w8r....w13... ....w&r....w.P..h.f.....u..
W...M..u.....]...$..4...h./...D$.P.D$.h....D$.....{.....U.....f...W.N...t...\.f..D....u
h.....U..P....f....F.....t
P....f....W.~... .f.....F..F Pj.S.U.....N._].....U..j.h(...d.....PQV..P..3.P.E.d.....u..E.....F...t
P....f.....M..F...|.....t..V..P..F..V...t..P....F....F..H..E..t
V..p.f.....M.d.
....Y^..].....U..j.h....d.....P...SV..P..3.P.E.d.....t
.\.f..D....u
h.....T...1..t.....9F.tC.v...u..A.Q.....u.W.E.....f.....t/.~..t).F .M.d.
....Y^[..]..F..
..f.;.t.W.5.....3..M.d.
....Y^[..].....t.....i.....U..O...f.V.095|.f.t.h..f..Rr.....u....f....5|.f..M.i          O....f.....t.P..R...E.....tC....f..295p.f.t.h..f..B`.....u....f....5p.f...P....f.
```

This is the “raw” contents of the GIF file

There is something *very* wrong here…

What is in this malicious HTTP request?

```
41.067893    10.13.13.99       HTTP    HTTP/1.1 200 OK  (image/gif)
```

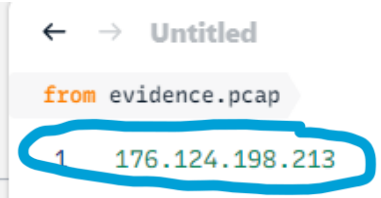
Right click → Follow TCP Stream

```
GET /Fs8Py/eKTYt3dRbEXw HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT; Windows NT 10.0; en-US) WindowsPowerShell/5.1.22621.963
Host: 176.124.198.213
Connection: Keep-Alive


HTTP/1.1 200 OK
Date: Mon, 22 May 2023 17:01:04 GMT
Server: Apache/2.4.41 (Ubuntu)
Accept-Ranges: bytes
Content-Length: 543048
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: image/gif

MZ.....@.....  .!..L.!This program cannot be run in DOS mode.

$.....u.S.....]U.....]a.....]`.....X.....]W.....]d.....]P.....]Q.....]V.....Rich.....PE..L....._P.....!..
.....k.....W.....@.....K.....\C.....f..... f.@).....
..@.....x.....text..... ..`.rdata.....@..@.data...p.c..P.....B.....@....rsrc....
D....f..P...N.....@..@.reloc..V....`k.....@..B.....
.....
.....\....%8.....U..V....\.....8....E..t
V.....^].....h....%8.....A.....U..V....h.....8....E..t
V.....^].....U..VW.}.W....,.....h....G..F._..^].....t.P....f.Y....|....H..    3..H.;.t..A..H..P....H..A...t....P..P.....U.....3.....
.w8r....w13... ....w&r....w.P..h.f.....u..
W...M..u.....]...$..4...h./...D$.P.D$.h....D$.....{.....U.....f...W.N...t...\.f..D....u
h.....U..P....f....F.....t
P....f....W.~... .f.....F..F Pj.S.U.....N._].....U..j.h(...d.....PQV..P..3.P.E.d.....u..E.....F...t
P....f.....M..F...|.....t..V..P..F..V...t..P....F....F..H..E..t
V..p.f.....M.d.
....Y^..].....U..j.h....d.....P...SV..P..3.P.E.d.....t
.\.f..D....u
h.....T...1..t.....9F.tC.v...u..A.Q.....u.W.E.....f.....t/.~..t).F .M.d.
....Y^[..]..F..
..f.;.t.W.5.....3..M.d.
....Y^[..].....t.....i.....U..O...f.V.095|.f.t.h..f..Rr.....u....f....5|.f..M.i    O....f.....t.P..R...E.....tC....f..295p.f.t.h..f..B`.....u....f....5p.f...P....f.
```

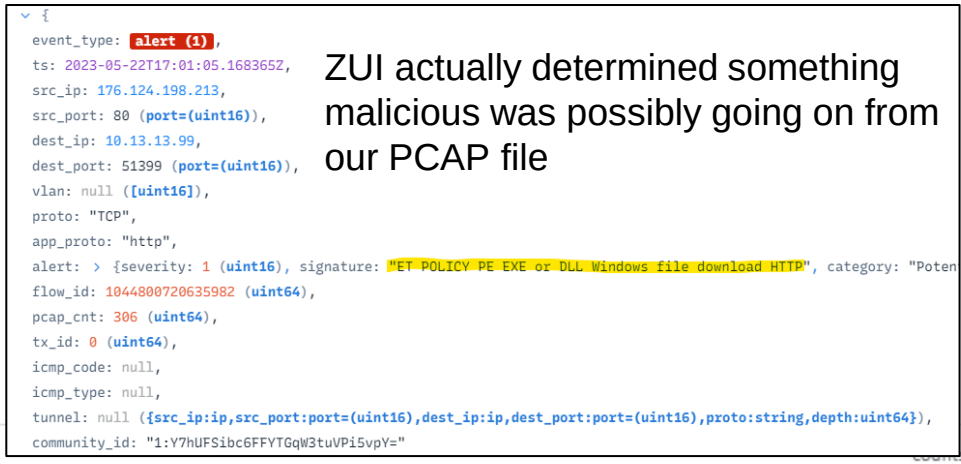
"This program cannot be run in DOS mode" is a very special string that appears in almost every Windows executable (.exe) or DLL file

It doesn't make sense that this is part of a GIF file

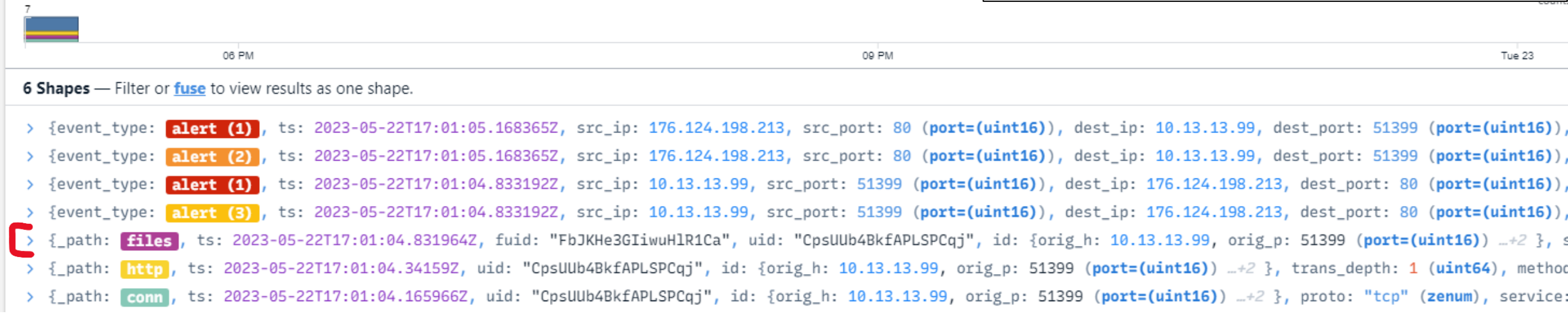
What is this malicious DLL file?



We can plug our malicious IP from the GET request into ZUI



ZUI actually determined something malicious was possibly going on from our PCAP file



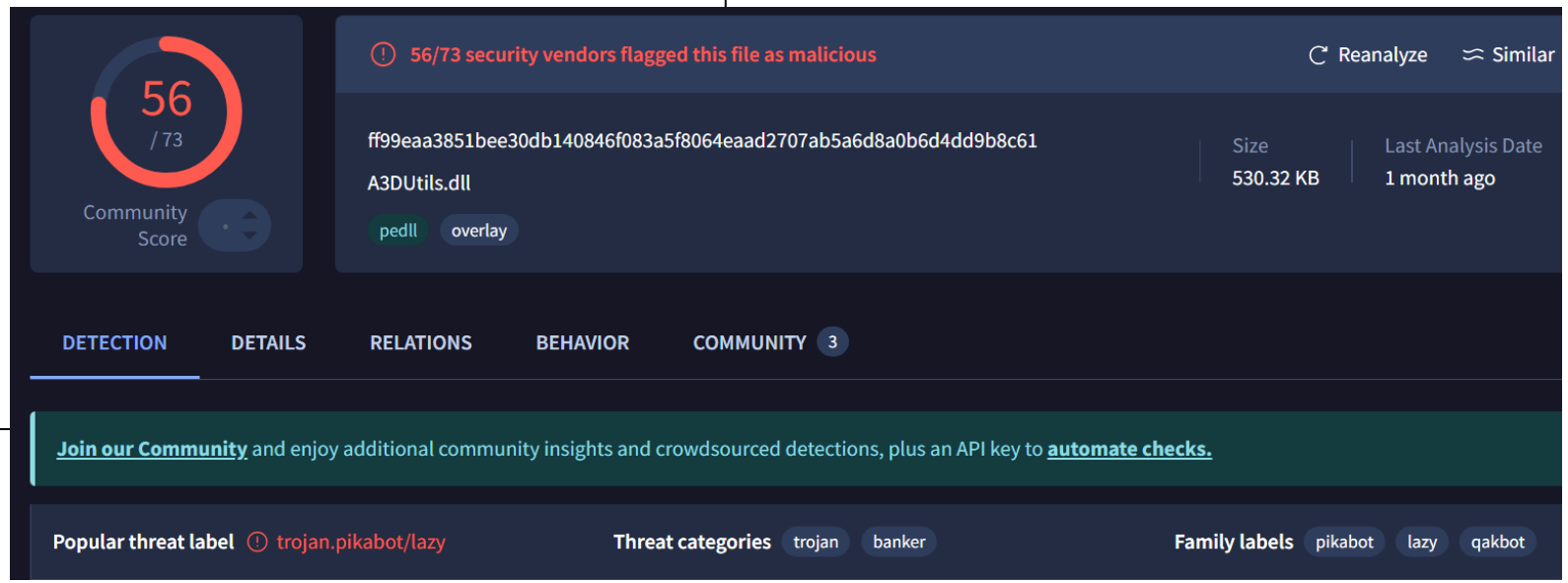
We can expand the “files” info to see information about this file that was downloaded

What is this malicious DLL file?

```
_path: files,
ts: 2023-05-22T17:01:04.831964Z,
fuid: "FbJKHe3GIiwuHlR1Ca",
uid: "CpsUUb4BkfAPLSPCqj",
id: > {orig_h: 10.13.13.99, orig_p: 51399 (port=(uint16)), resp_h: 176.124.198.213, resp_p: 80 (port=(uint16))},
source: "HTTP",
depth: 0 (uint64),
analyzers: > |["PE", "MD5", "SHA1"]|,
mime_type: "application/x-dosexec",
filename: null,
duration: 1.131455s,
local_orig: false,
is_orig: false,
seen_bytes: 543048 (uint64),
total_bytes: 543048 (uint64),
missing_bytes: 0 (uint64),
overflow_bytes: 0 (uint64),
timedout: false,
parent_fuid: null,
md5: "684f5e808312f7d7bcc7f4405ea706ad",
sha1: "6f2401d618fd2463461c5346ff64216c334842f8",
sha256: null,
extracted: null,
```

ZUI will compute the file hash for us !

By plugging it into VirusTotal, we can see that this DLL file-pretending-to-be-a-gif is a known malicious file!

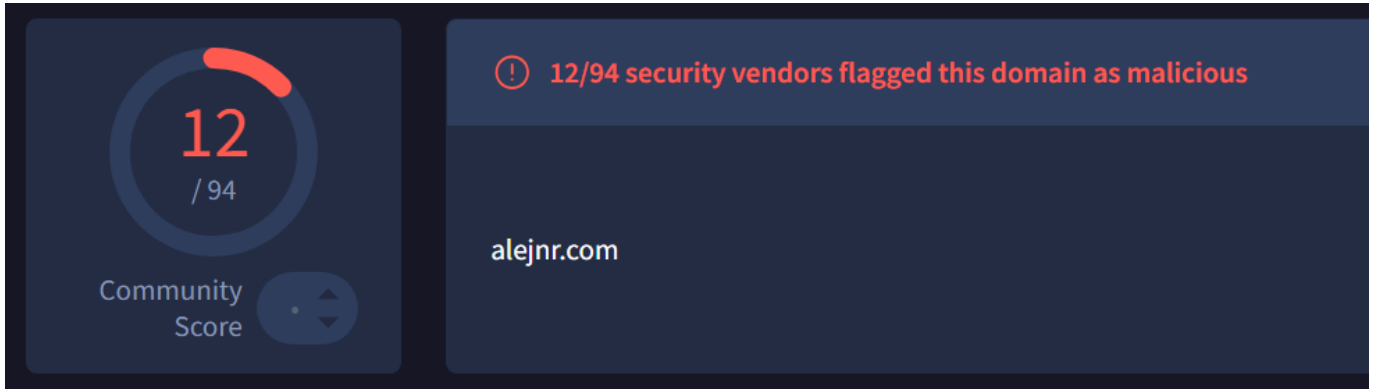


The name of this malware is "PIKABOT"

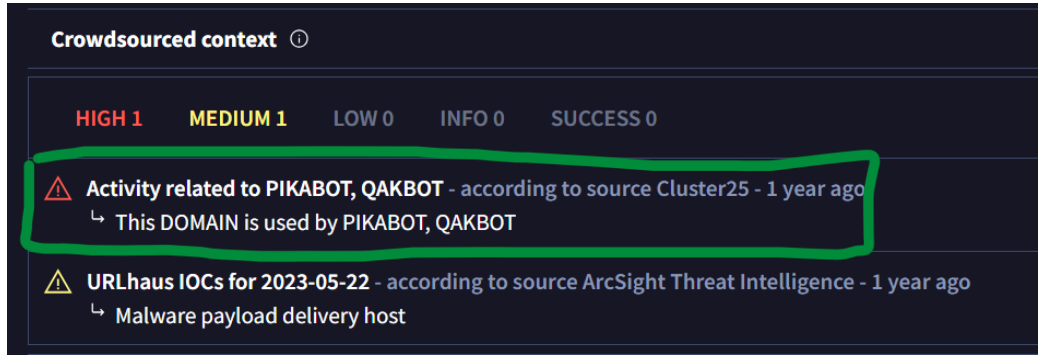
VT seemed to redirect to the SHA256 hash value

What is this malicious DLL file?

This domain had been flagged as malicious in the past

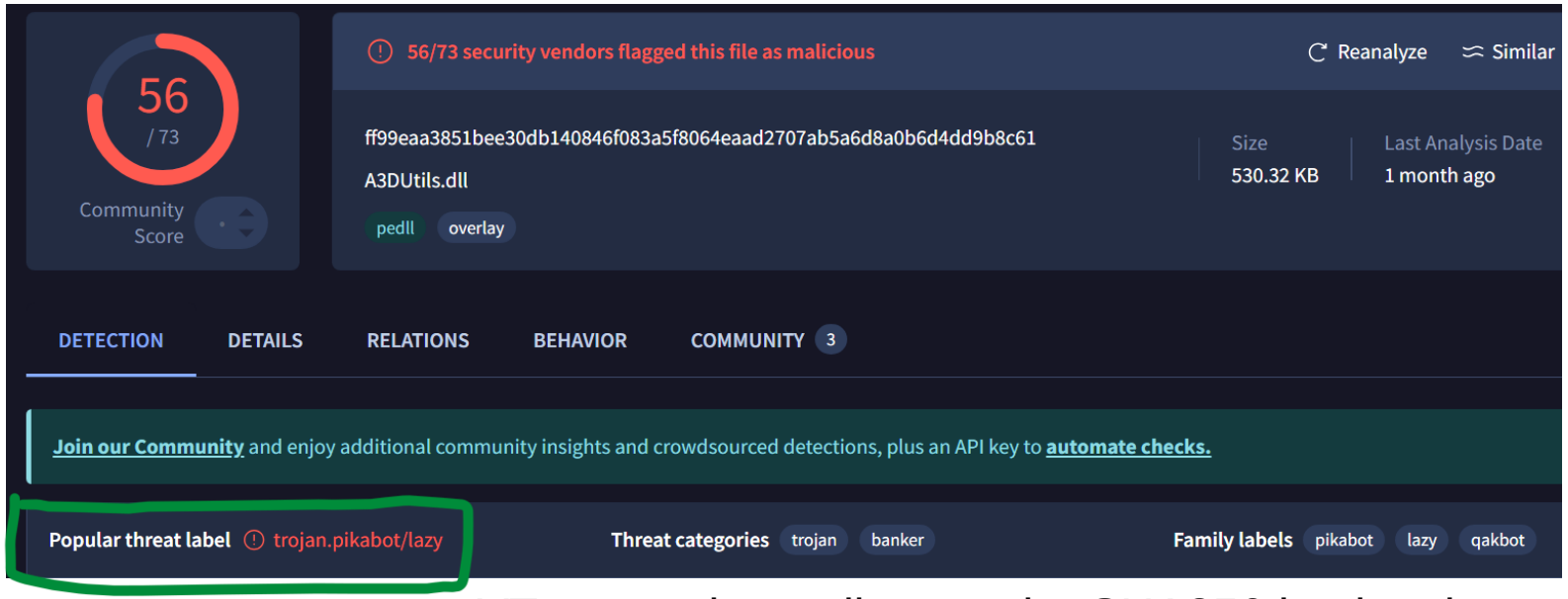


VirusTotal also provides context as to why it was flagged as malicious



We saw “PIKABOT” earlier when we were investigating **alejnr.com**

The name of this malware is “PIKABOT”



VT seemed to redirect to the SHA256 hash value

What happens after the malware was downloaded?

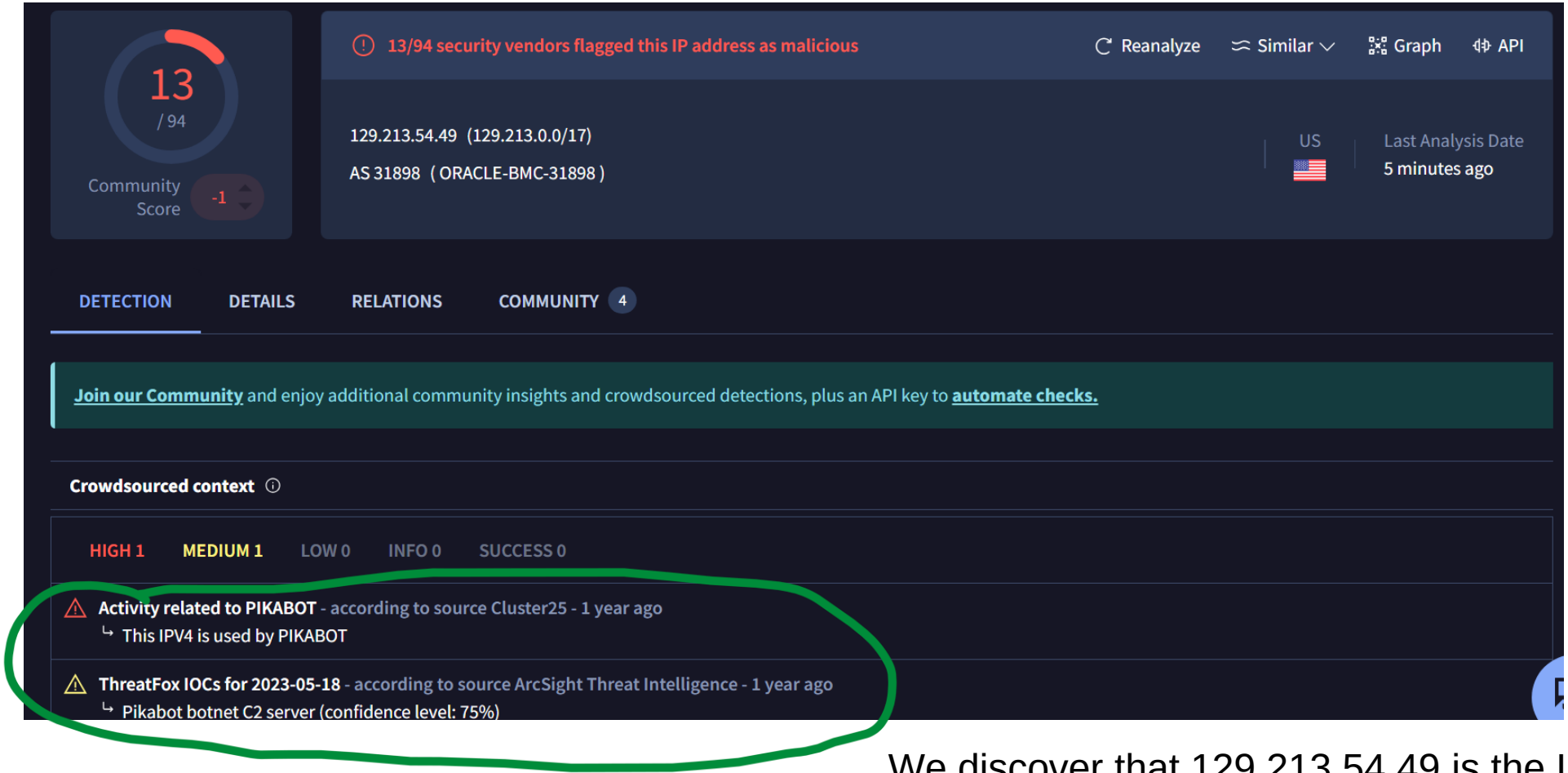
Let's see if we can find evidence of **command and control** from this cyber attack

We can start plugging in IP addresses into VirusTotal to see if we get a hit

No.	Time	Destination	Protocol	Info
1645	401.318608	129.213.54.49	TCP	51422 → 2078 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
1646	401.388236	10.13.13.99	TCP	2078 → 51422 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
1647	401.388558	129.213.54.49	TCP	51422 → 2078 [ACK] Seq=1 Ack=1 Win=65535 Len=0
1648	401.389095	129.213.54.49	TLSv1.2	Client Hello
1649	401.389336	10.13.13.99	TCP	2078 → 51422 [ACK] Seq=1 Ack=148 Win=64240 Len=0
1650	401.588331	10.13.13.99	TLSv1.2	Server Hello
1651	401.588472	10.13.13.99	TLSv1.2	Certificate, Server Key Exchange, Server Hello Done
1652	401.588675	129.213.54.49	TCP	51422 → 2078 [ACK] Seq=148 Ack=1377 Win=65535 Len=0
1653	401.588774	129.213.54.49	TCP	51422 → 2078 [ACK] Seq=148 Ack=2170 Win=65535 Len=0
1654	401.596481	129.213.54.49	TLSv1.2	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message

What happens after the malware was downloaded?

Let's see if we can find evidence of **command and control** from this cyber attack



We discover that 129.213.54.49 is the IP address for the PIKABOT C2 server

What happens after the malware was downloaded?

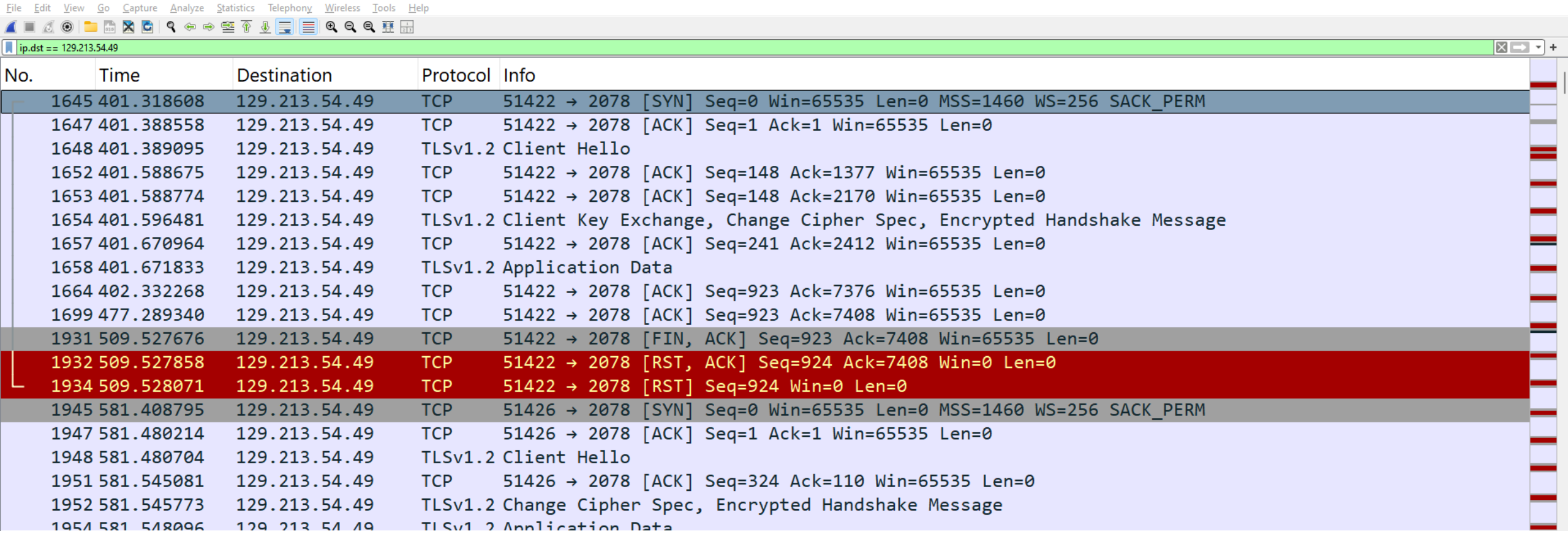
Let's see if we can find evidence of **command and control** from this cyber attack


```
{
  _path: notice,
  ts: 2023-05-22T23:47:01.833695Z,
  uid: "Cb45fX1d1qHMAfZA58",
  id: > {orig_h: 10.13.13.99, orig_p: 51976 (port=(uint16)), resp_h: 129.213.54.49, resp_p: 2078 (port=(uint16))},
  fuid: "FuX96l2lqECdlDLRZi",
  file_mime_type: null,
  file_desc: null,
  proto: "tcp" (zenum),
  note: "SSL::Invalid_Server_Cert" (zenum),
  msg: "SSL certificate validation failed with (self signed certificate)",
  sub: "CN=nonveracitygalvanometry.band,L=Wrinkles Fireless,OU=Prediet,O=Awarrant,ST=EN,C=MX",
  src: 10.13.13.99,
  dst: 129.213.54.49,
  p: 2078 (port=(uint16)),
  n: null,
  peer_descr: null,
  actions: > |["Notice::ACTION_LOG" (zenum), "Notice::ACTION_ADD_GEODATA" (zenum)]|,
  email_dest: > |[]|
```


ZUI also generated a notice that this could possibly be a C2 server

What damage was done?

A good amount of Data was exchanged with the C2 server

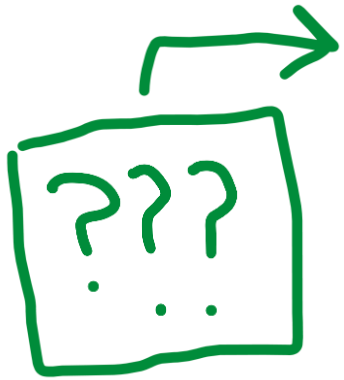


No.	Time	Destination	Protocol	Info
1645	401.318608	129.213.54.49	TCP	51422 → 2078 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
1647	401.388558	129.213.54.49	TCP	51422 → 2078 [ACK] Seq=1 Ack=1 Win=65535 Len=0
1648	401.389095	129.213.54.49	TLSv1.2	Client Hello
1652	401.588675	129.213.54.49	TCP	51422 → 2078 [ACK] Seq=148 Ack=1377 Win=65535 Len=0
1653	401.588774	129.213.54.49	TCP	51422 → 2078 [ACK] Seq=148 Ack=2170 Win=65535 Len=0
1654	401.596481	129.213.54.49	TLSv1.2	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
1657	401.670964	129.213.54.49	TCP	51422 → 2078 [ACK] Seq=241 Ack=2412 Win=65535 Len=0
1658	401.671833	129.213.54.49	TLSv1.2	Application Data
1664	402.332268	129.213.54.49	TCP	51422 → 2078 [ACK] Seq=923 Ack=7376 Win=65535 Len=0
1699	477.289340	129.213.54.49	TCP	51422 → 2078 [ACK] Seq=923 Ack=7408 Win=65535 Len=0
1931	509.527676	129.213.54.49	TCP	51422 → 2078 [FIN, ACK] Seq=923 Ack=7408 Win=65535 Len=0
1932	509.527858	129.213.54.49	TCP	51422 → 2078 [RST, ACK] Seq=924 Ack=7408 Win=0 Len=0
1934	509.528071	129.213.54.49	TCP	51422 → 2078 [RST] Seq=924 Win=0 Len=0
1945	581.408795	129.213.54.49	TCP	51426 → 2078 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
1947	581.480214	129.213.54.49	TCP	51426 → 2078 [ACK] Seq=1 Ack=1 Win=65535 Len=0
1948	581.480704	129.213.54.49	TLSv1.2	Client Hello
1951	581.545081	129.213.54.49	TCP	51426 → 2078 [ACK] Seq=324 Ack=110 Win=65535 Len=0
1952	581.545773	129.213.54.49	TLSv1.2	Change Cipher Spec, Encrypted Handshake Message
1954	581.548096	129.213.54.49	TLSv1.2	Application Data

Since this all HTTPS encrypted traffic, we cannot see the content of these messages

However, this is evidence that it is possible that data may have been exfiltrated

Timeline



alejnr.com was contacted

2023-05-22T17:00:24

162.213.255.36

Connection with alejnr.com was made and malware dropper was installed

176.124.198.213

PikaBot DLL downloaded

2023-05-22T17:01:04

129.213.54.49

PikaBot C2 server contacted

Possible Data Exfiltration ?

How did this attack begin?

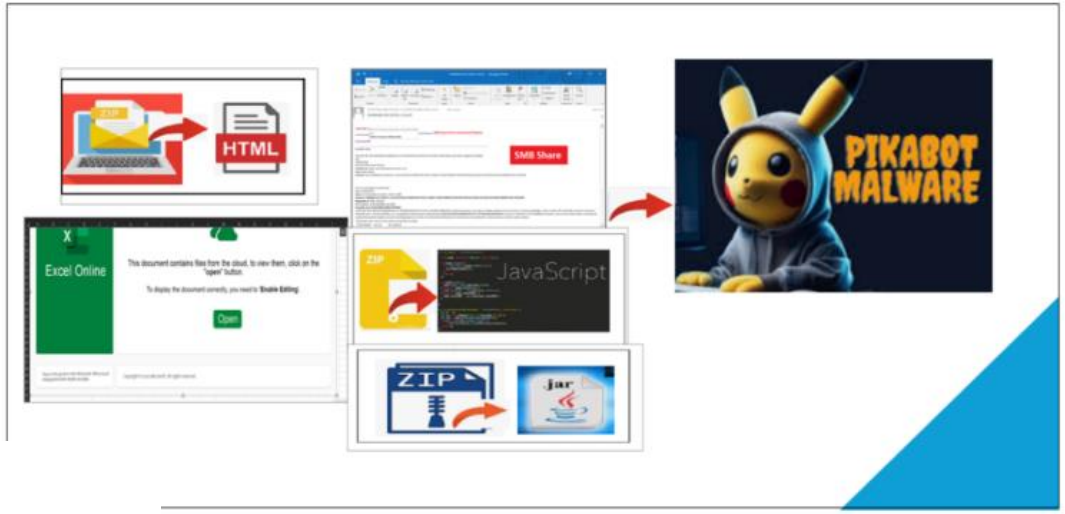
Finding Root cause

PIKABOT is a well-documented malware. We can do a little bit of investigation ourselves to figure out what happened

Distribution Methods

PikaBot, along with various other malicious loaders like QBot and DarkGate, heavily depends on email spam campaigns for distribution. Its initial access strategies are intricately crafted, utilizing geographically targeted spam emails tailored for specific countries. These emails frequently include links to external **Server Message Block (SMB)** shares hosting malicious zip files.

SMB shares refer to resources or folders on a server or computer accessible to other devices or users on a network using the SMB protocol. The threat actors frequently exploit such shares for malware distribution. In this instance, the act of downloading and opening the provided zip file leads to PikaBot infection.



How does Pikabot malware work?

Many Pikabot infections start with a malicious email, particularly using email thread hijacking; however, other cases have been distributed via malspam and malvertising [5]. Once downloaded, Pikabot runs anti-analysis techniques and checks the system's language, self-terminating if the language matches that of a Commonwealth of Independent States (CIS) country, such as Russian or Ukrainian. It will then gather key information to send to a command-and-control (C2) server, at which point additional payload downloads may be observed [2]. Early response to a Pikabot infection is important for organizations to prevent escalation to a significant compromise such as ransomware.

Full IOC: https://github.com/pan-unit42/tweets/blob/master/2023-05-22-IOCs-for-Pikabot-infection-with-Cobalt-Strike.txt

Final Exam Review

Exam Logistics

13.33% of your final grade

Wednesday May 7th 2:00 PM – 3:50PM in this classroom

Exam Length: 110 minutes (shouldn't take that full time)

This exam is **optional**
- If you don't show up, I will use the average of Exam 1 and Exam 2 for your Final Exam score

Format will be similar to your other two exams
- Short answer
- May give you a diagram, and ask you some questions about it
- Might have you look at some code

You are allowed to use any physical notes (printouts, handwritten/typed notes)
You are **not** allowed to use any digital devices

Exam will mostly cover materials from the second half of the semester, but there will be a couple questions from the first half of the semester

Stuff from first half of semester

I'd recommend reviewing the stuff from the first couple weeks

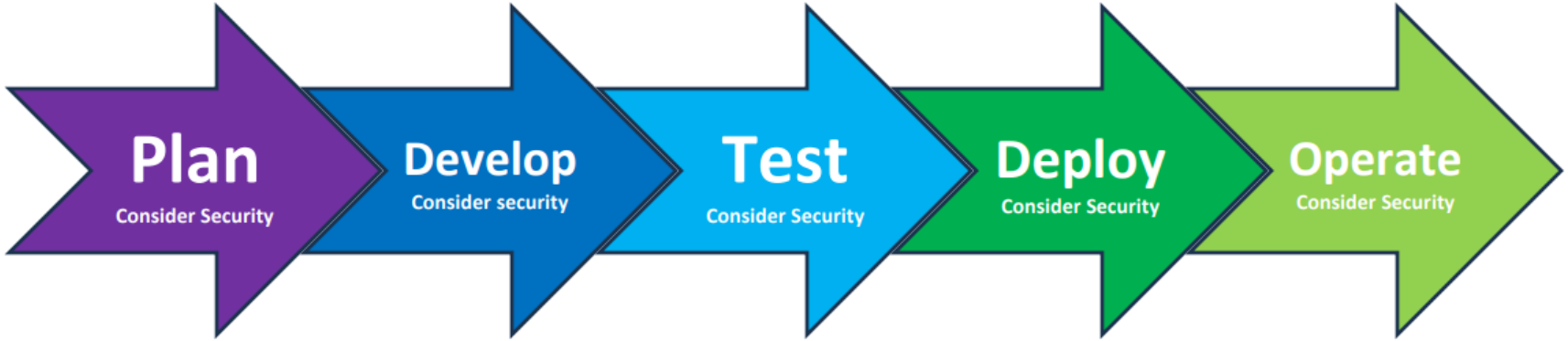
- OOP Principles and mechanisms
- Design Patterns

Secure by Design

- CIA Triad
- Domain Primitives
- Principle of Fail-Fast
- Domain Driven Design
- Object Mutability
- Input Validation Steps
- Secure Development Lifecycle
- Defense in Depth


```
public class BookOrder {

    private String title;
    private String isbn;
    private Quantity quantity;

}
```

1. **Origin** – Is the data from a legitimate sender?
2. **Size** – Is the input reasonably large?
3. **Lexical Content** – Does it contain the right characters and encoding ?
4. **Syntax**– Is the format right?
5. **Semantics**– Does it make sense?



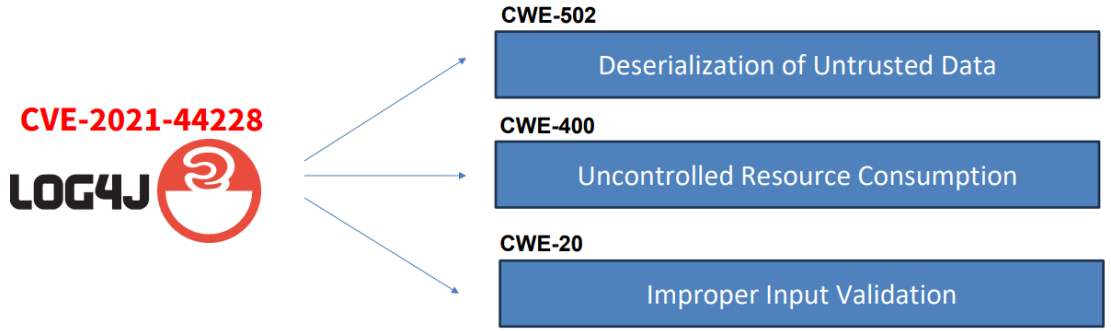
Vulnerability Analysis

- Common Vulnerabilities and Exposures (CVE)
- Common Weakness Enumeration (CWE)
- Bug Bounties
- NIST
- Attack Vectors, Attack Surface
- OWASP Top 10
- Threat Modeling
- STRIDE
- Options for dealing for threat

1. What are we building?
2. What can go wrong?
3. What are we going to do about it?
4. Did we do a good job?

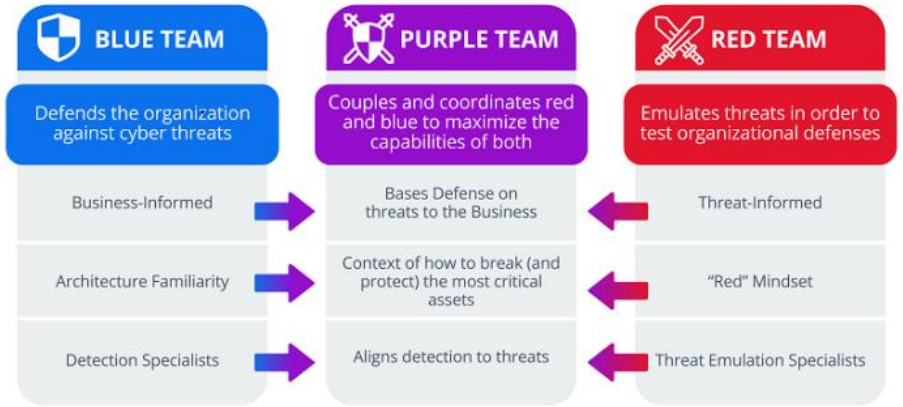
2021

- A01:2021-Broken Access Control
- A02:2021-Cryptographic Failures
- A03:2021-Injection
- A04:2021-Insecure Design
- A05:2021-Security Misconfiguration
- A06:2021-Vulnerable and Outdated Components
- A07:2021-Identification and Authentication Failures
- A08:2021-Software and Data Integrity Failures
- A09:2021-Security Logging and Monitoring Failures*
- A10:2021-Server-Side Request Forgery (SSRF)*



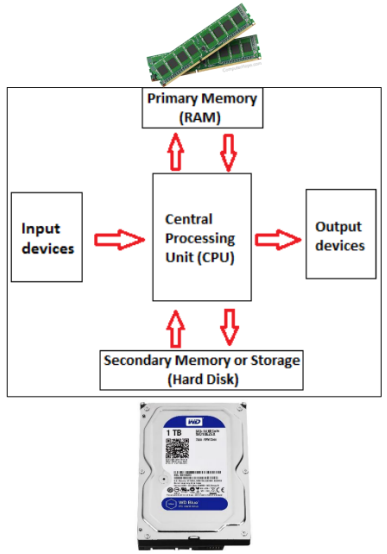
Penetration Testing

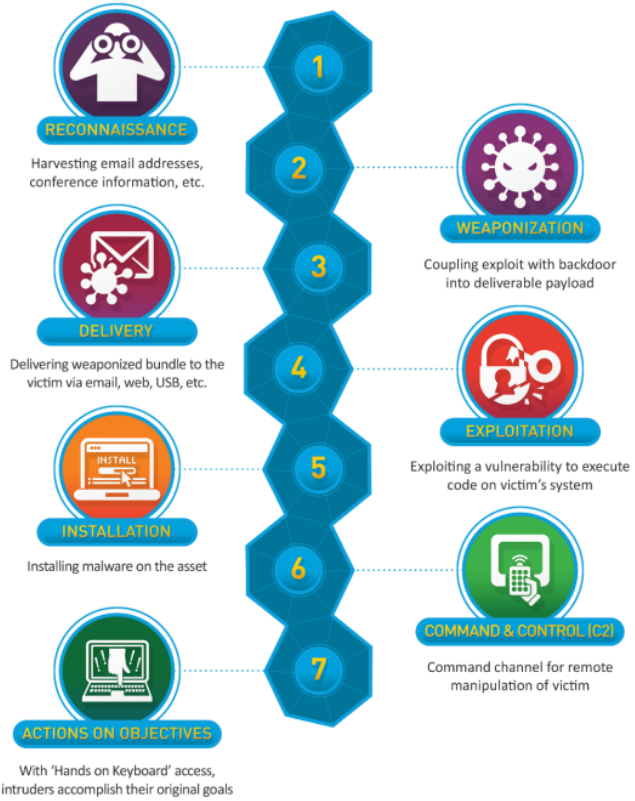
- What is penetration testing
- Types of penetration tests
- Penetration testing phases
- OSINT
- Metasploit
- Metasploit Terminology + Modules
- nmap and ports
- Post-Exploitation Tactics

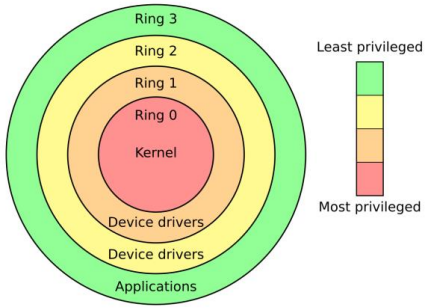


Digital Forensics

- What is digital forensics
- Goals of digital forensics
- Types of Forensics
- Digital Evidence Collection
- Principles of Digital Forensics
- Precursors, Indicators of Compromise
- Threat Detection and Analysis (Classification Errors)
- Pyramid of Pain
- Cyber Kill Chain
- Recovery, Containment, Eradication
- CISA
- OS Fundamentals
- Malicious Behavior
- Volatility
- Helpful Volatility Plugins and DF Tools
- General steps of investigation with volatility







Course Outcomes, Lessons Learned

SECURITY

Apple notifies new victims of spyware attacks across the world

Lorenzo Franceschi-Bicchierai - 1 hour ago

SECURITY

Government hackers are leading the use of attributed zero-days, Google says

Lorenzo Franceschi-Bicchierai - 1 day ago

- Deep modeling ensures a better design
- Software testing is important
- Design decisions should be made with cybersecurity in mind
- It's important to understand common vulnerabilities and weaknesses
- It's important to threat model your system
- Penetration Testing is a great way to identify flaws
- Digital Forensics investigations are vital for responding to an incident and dealing with a threat

As losses mount, CrowdStrike says bug in quality-control process led to botched update

By James Pearson

July 24, 2024 1:56 PM MDT · Updated 9 months ago

Course Objectives:

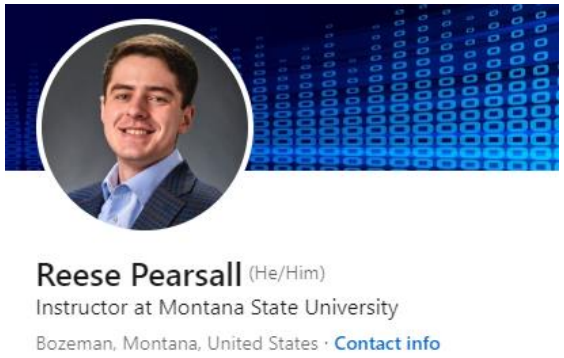
- Build expertise in modeling techniques
- Introduce software design through the use of rigorous UML
- Introduce constraint-based modeling
- Learn relevant and pragmatic topics in cybersecurity. Specifically the Security Lifecycle and analysis techniques
- Learn many relevant cybersecurity tools
- Learn advanced testing techniques also applied in cybersecurity
- Learn relevant digital forensics

Thank You!

I hope you enjoyed this class, and I hope the stuff you learned will be helpful in your career/future classes

If I can be of assistance to you for anything in the future (reference, advising, support), please let me know!

I will be teaching CSCI 232, 466, 476 next semester



Connect with me on LinkedIn! **If you find a job in cybersecurity or cyber practices, *please* keep in touch!**

Congrats to those that are graduating next weekend! I hope you find a job that you love!