

COMP3010HK Security Operations and Incident Management 2025/2026

Coursework 1: Set Exercises - PCAP Analysis



Student ID: 10953435

Github repo: <https://github.com/aka331/COMP3010HK/tree/main/CW1>

Word count: 988

Section 1 – Introduction

This report presents a network traffic investigation based on the provided PCAP file and associated quiz questions. The objective is to identify the infected system, understand how the infection occurred, and determine the nature of the malware or attack involved.

The structure of this report is as follows:

Section No.	Content
Section 1	highlights the report objective;
Section 2	describes the tools and methodology used to analyse the PCAP and extract key indicators of compromise (IOCs);
Section 3	presents the main findings, drawing on the quiz answers and supporting evidence from the traffic analysis;
Section 4	concludes with prevention techniques, open challenges, and relevant references.

Table 1. Report Structure

Table of Content

Section 1 – Introduction.....	2
Section 2 – Methodology.....	3
2.1 Wireshark.....	3
2.2.1 Identifying the Infected Host.....	4
2.2.2 How the system got infected.....	6
2.2.3 Infection Indicators.....	6
2.2 ZUI.....	7
2.3 Virustotal.....	9
Section 3 – Results.....	10
Section 4 – Conclusion and References.....	19
4.1 Conclusion.....	19
4.2 References.....	20

Section 2 – Methodology

2.1 Wireshark

Firstly, Wireshark was used to investigate the PCAP. I filtered HTTP GET traffic to find the first malicious download and extracted the timestamp, domain, file name, and server headers. I exported the downloaded archive via Export Objects to confirm the embedded file. I then used DNS/TLS evidence and Conversations statistics to identify C2 IPs and map them to domains/Host headers. Lastly, I followed the SMTP TCP stream and decoded Base64 authentication data to obtain the required email and password details. Below are the expressions used for analyzing the traffic in the pcap file.

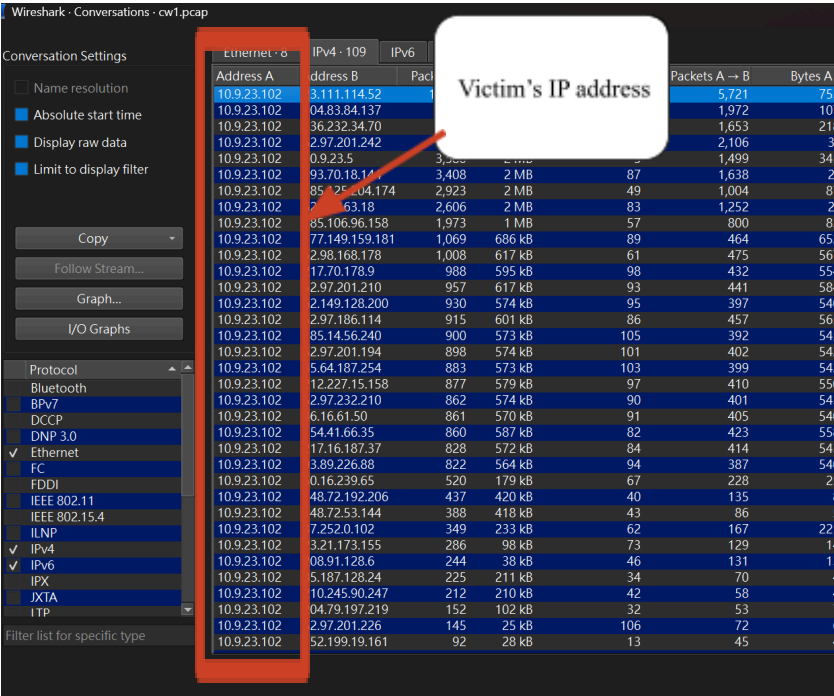
```
## Filtering only HTTP request traffic
http.request
## Filtering only HTTP GET requests
http.request.method == "GET"
## Filtering only the source IP from the victim and the HTTP POST
traffic
ip.addr==10.9.23.102 && http.request.method=="POST"
## Filtering only the source IP from 10.9.23.102 and the DNS
protocol traffic
ip.src==10.9.23.102 && dns
## Filtering only the IP from 185.106.96.158 (C2 server) and the
HTTP traffic (port 80)
ip.addr == 185.106.96.158 && http
## Filtering only the IP from 185.125.204.174 (C2 server) and the
port 8080 protocol traffic
ip.addr == 185.125.204.174 && tcp.port == 8080
## Filtering only the source IP from the victim, the DNS protocol
traffic, and containing "api"
ip.addr==10.9.23.102 && dns && frame contains "api"
```

```
## Filtering only the source IP from the victim, the SMTP protocol
traffic, and containing "FROM"
ip.addr==10.9.23.102 && smtp && frame contains "FROM"
```

Expression used in Wireshark

2.1.1 Identifying the Infected Host

The figure below shows that Address A(**10.9.23.102**) has a lot of conversations in descending order with other IP addresses (internal or external), which means the victim host is “**10.9.23.102**”



Wireshark - Conversations - cw1.pcap

Conversation Settings

- ☐ Name resolution
- ☒ Absolute start time
- ☒ Display raw data
- ☒ Limit to display filter

Copy Follow Stream... Graph... I/O Graphs

Protocol

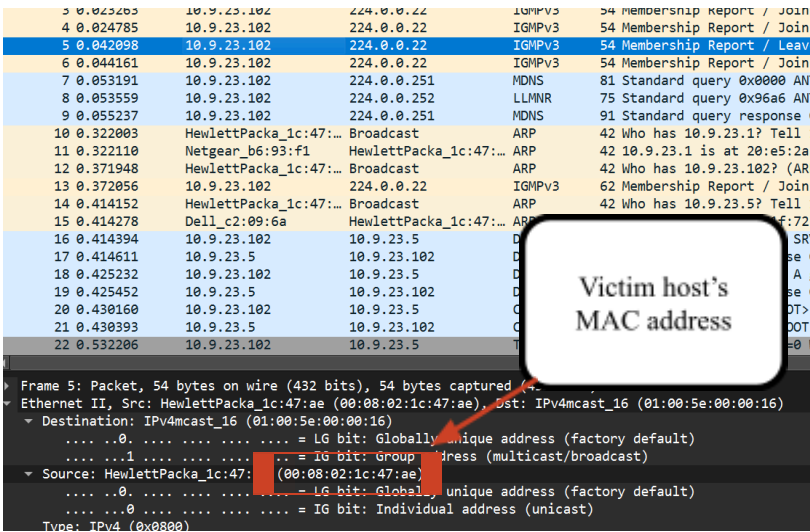
- ☐ Bluetooth
- ☐ BPv7
- ☐ DCCP
- ☐ DNP 3.0
- ☒ Ethernet
- ☐ FC
- ☐ FDDI
- ☐ IEEE 802.11
- ☐ IEEE 802.15.4
- ☐ ILNP
- ☒ IPv4
- ☒ IPv6
- ☐ IPX
- ☐ JXTA
- ☐ LTP

Filter list for specific type

Address A	Address B	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A
10.9.23.102	3.111.114.52	5,721	753		
10.9.23.102	04.83.84.137	1,972	107		
10.9.23.102	36.232.34.70	1,653	218		
10.9.23.102	2.97.201.242	2,106	31		
10.9.23.102	0.9.23.5	1,499	345		
10.9.23.102	93.70.18.14	3,408	2 MB	87	1,638
10.9.23.102	85.25.104.174	2,923	2 MB	49	1,004
10.9.23.102	2.16.63.18	2,606	2 MB	83	1,252
10.9.23.102	85.106.96.158	1,973	1 MB	57	800
10.9.23.102	77.149.159.181	1,069	686 kB	89	464
10.9.23.102	2.98.168.178	1,008	617 kB	61	475
10.9.23.102	17.70.178.9	988	595 kB	98	432
10.9.23.102	2.97.201.210	957	617 kB	93	441
10.9.23.102	2.149.128.200	930	574 kB	95	397
10.9.23.102	2.97.186.114	915	601 kB	86	457
10.9.23.102	85.14.56.240	900	573 kB	105	392
10.9.23.102	2.97.201.194	888	574 kB	101	402
10.9.23.102	5.64.187.254	883	573 kB	103	399
10.9.23.102	12.227.15.158	877	579 kB	97	410
10.9.23.102	2.97.232.210	862	574 kB	90	401
10.9.23.102	6.16.61.50	861	570 kB	91	405
10.9.23.102	54.41.66.35	860	587 kB	82	423
10.9.23.102	17.16.187.37	828	572 kB	84	414
10.9.23.102	3.89.226.88	822	564 kB	94	387
10.9.23.102	0.16.239.65	520	179 kB	67	228
10.9.23.102	48.72.192.206	437	420 kB	40	135
10.9.23.102	48.72.53.144	388	418 kB	43	86
10.9.23.102	7.25.20.102	349	233 kB	62	167
10.9.23.102	3.21.173.155	286	98 kB	73	129
10.9.23.102	08.91.128.6	244	38 kB	46	131
10.9.23.102	5.187.128.24	225	211 kB	34	70
10.9.23.102	10.245.90.247	212	210 kB	42	58
10.9.23.102	04.79.197.219	152	102 kB	32	53
10.9.23.102	2.97.201.226	145	25 kB	106	72
10.9.23.102	52.199.19.161	92	28 kB	13	45

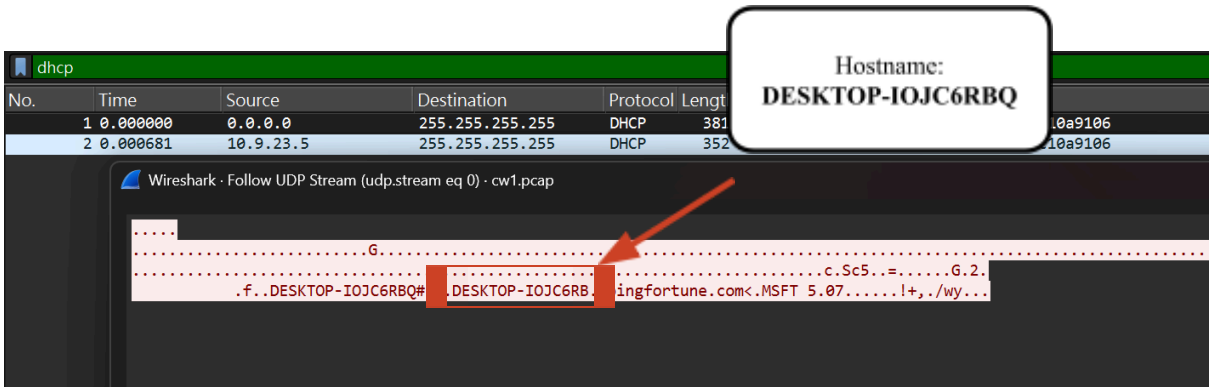
Conversation table in Wireshark

Select frame 5, where the source IP address(10.9.23.102), and review the MAC address(00:08:02:1c:47:ae) of the victim host.



Victim host MAC address

For hostname, run expression “dhcp”, then right click “Follow > UDP Stream”, the UDP stream shows the hostname is DESKTOP-IOJC6RBQ



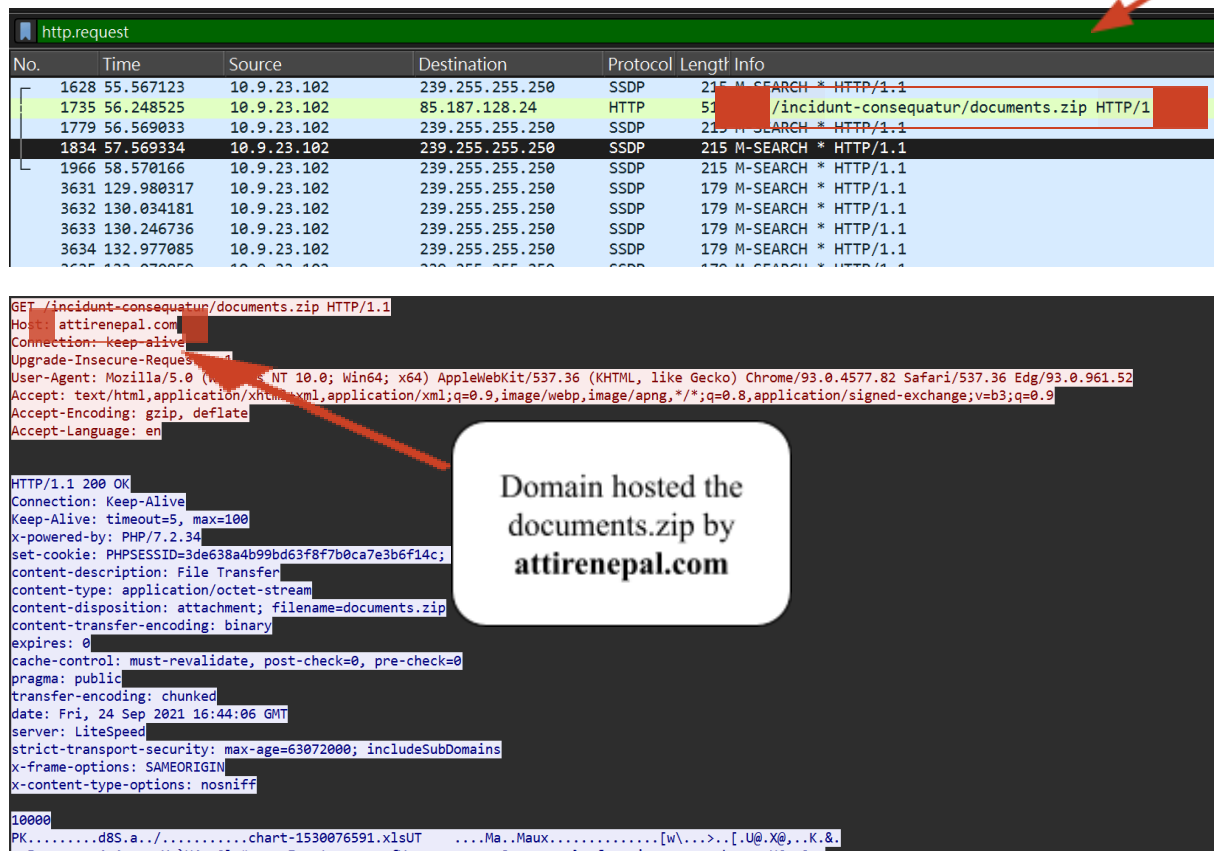
Victim hostname

The infected system information

Item	Information
IP address	10.9.23.102
MAC address	00:08:02:1c:47:ae
Hostname	DESKTOP-IOJC6RBQ

2.1.2 How the system got infected

Apply the HTTP request filter, which shows the victim, and download the **documents.zip**, which activates the malicious activity.



The image shows a Wireshark packet capture. The top packet list shows an HTTP request (No. 1735) from 10.9.23.102 to 85.187.128.24. The packet details pane shows the request for `/incident-consequatur/documents.zip` from `attirenepal.com`. The request headers include `Host: attirenepal.com`, `Connection: keep-alive`, `Upgrade-Insecure-Request`, `User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.82 Safari/537.36 Edg/93.0.961.52`, `Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9`, and `Accept-Encoding: gzip, deflate`. The response status is `HTTP/1.1 200 OK`. The response headers include `Connection: Keep-Alive`, `Keep-Alive: timeout=5, max=100`, `x-powered-by: PHP/7.2.34`, `set-cookie: PHPSESSID=3de638a4b99bd63f8f7b0ca7e3b6f14c;`, `content-description: File Transfer`, `content-type: application/octet-stream`, `content-disposition: attachment; filename=documents.zip`, `content-transfer-encoding: binary`, `expires: 0`, `cache-control: must-revalidate, post-check=0, pre-check=0`, `pragma: public`, `transfer-encoding: chunked`, `date: Fri, 24 Sep 2021 16:44:06 GMT`, `server: LiteSpeed`, `strict-transport-security: max-age=63072000; includeSubDomains`, `x-frame-options: SAMEORIGIN`, and `x-content-type-options: nosniff`. A text box overlay states: "Domain hosted the documents.zip by attirenepal.com".

Malicious file from **attirenepal.com**

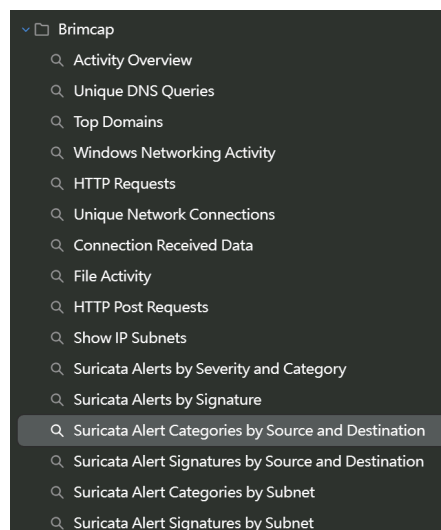
2.2 ZUI

Second, ZUI is a network forensic tool that provides a GUI interface to analyze and visualize large amounts of network traffic. It was utilized to analyze the CW1.pcap to investigate the network traffic that easily identifies the alert inside the PCAP file. It is similar to the SOC dashboard environment that allows us to filter out the event types of interest.



ZUI Screenshot 1. Activity overview

The preloaded [query.json](#) from brimcap on GitHub was used in the analysis of the pcap file. There is some information that can be viewed, such as activity overview, file activity, HTTP post requests, show IP subnets, Suricata Alerts by Severity and Category, Suricata Alerts by Signature, etc. Those preloaded queries are useful for analyzing malicious events.



ZUI Screenshot 2. Preloaded Query

For alert triage, key attributes are those that can uniquely identify a rule, describe the relevant network session, and allow association with other logs. There are key attributes that can be used to analyze alert events, such as timestamp, src_ip, src_port, dest_ip, dest_port, proto, app_proto, alert.signature, alert.signature_id, alert.category, and alert.severity.

```

{
  event_type: alert (3),
  ts: 2021-09-24T17:04:53.048515Z,
  src_ip: 10.9.23.102,
  src_port: 63757 (port=(uint16)),
  dest_ip: 52.97.232.194,
  dest_port: 25 (port=(uint16)),
  vlan: null ([uint16]),
  proto: "TCP",
  app_proto: "failed",
  alert: > {severity: 3 (uint16), signature: "SURICATA Applayer No TLS after STARTTLS", category: "Generic Protocol Command Decode", action: "allo
  flow_id: 721582510190689 (uint64),
  pcap_cnt: null,
  tx_id: null,
  icmp_code: null,
  icmp_type: null,
  tunnel: null ({src_ip:ip,src_port:port=(uint16),dest_ip:ip,dest_port:port=(uint16),proto:string,depth:uint64}),
  community_id: "1:Q$79pxV3Y5oz8EruLd/10F09op4="
}

```

ZUI Screenshot 3. One of the JSON files in the alert events

From ZUI, we can identify the possible or potential attack that the victim is dealing with. The screenshot below shows that the attack could be a network trojan or malware activity.

Query Page
Suricata Alerts by Severity and Category

← → Suricata Alerts by Severity and Category⁴

from cw1.pcap

1 event_type=="alert" | count() by alert.severity,alert.category | sort count

No data.

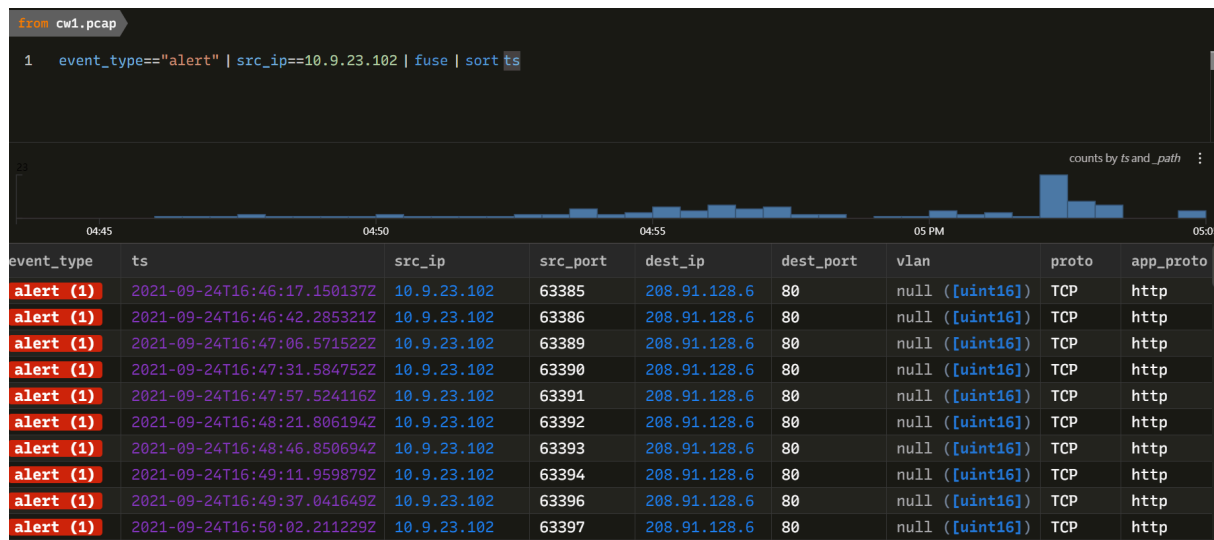
alert	count
> {severity: 1, category: A Network Trojan was detected}	26
> {severity: 3, category: Misc activity}	40
> {severity: 3, category: Unknown Traffic}	45
> {severity: 1, category: Malware Command and Control Activity Detected}	52
> {severity: 3, category: Generic Protocol Command Decode}	54

ZUI Screenshot 4. Suricata Alerts by Severity and Category

2.2.1 Infection Indicators

Firstly, ZUI was used as a primary analysis tool to identify infection indicators within the captured PCAP file. The malicious connection was started on

2021-09-24T16:46:17.150137Z



```
{
  event_type: "alert",
  ts: 2021-09-24T16:46:17.150137Z,
  src_ip: 10.9.23.102,
  src_port: 63385 (port=(uint16)),
  dest_ip: 208.91.128.6,
  dest_port: 80 (port=(uint16)),
  vlan: null ([uint16]),
  proto: "TCP",
  app_proto: "http",
  alert: {
    severity: 1 (uint16),
    signature: "ET MALWARE SQUIRRELWAFFLE Loader Acti",
    category: "A Network Trojan was detected",
    action: "allowed",
    signature_id: 2033939 (uint64),
    gid: 1 (uint64),
    rev: 6 (uint64),
    metadata: {
      signature_severity: [
        0: "Major"
      ],
      former_category: null ([string]),
      attack_target: null ([string]),
    }
  }
}
```

Second, there has been a malicious file named “**documents.zip**” downloaded on **2021-09-24T16:44:40.408441Z**; the documents.zip classifier has been marked as a malicious file based on Virustotal analysis.

(<https://www.virustotal.com/gui/file/77229c744a0b1470afc7989a774cfe821386c11c0165e7e3fb5e9897a789a8cb>)

from cw1.pcap

```
1 filename!=null | cut _path, ts, tx_hosts, rx_hosts, conn_uids, mime_type, filename, md5, sha1 | sort ts
```

_path	ts	tx_hosts	rx_hosts	conn_uids	mime_type	filename
files	2021-09-24T16:44:40.408441Z	> error(missing)	> error(missing)	> error(missing)	application/zip	documents.zip
files	2021-09-24T17:03:33.277422Z	> error(missing)	> error(missing)	> error(missing)	application/zip	=?UTF-8?B?Q2xhaW0tOTA4M
files	2021-09-24T17:04:01.943684Z	> error(missing)	> error(missing)	> error(missing)	application/zip	=?UTF-8?B?Q2xhaW0tODQ2M
files	2021-09-24T17:04:47.940714Z	> error(missing)	> error(missing)	> error(missing)	application/zip	=?UTF-8?B?Q2xhaW0tNDg2N

The file server is hosted at **85.187.128.24**, which is a malicious IP address. According to Virustotal analysis, this classifier has been flagged as a malicious IP address.

```
{
  orig_h: 10.9.23.102,
  orig_p: 62245,
  resp_h: 85.187.128.24,
  resp_p: 80
}
```

VIRUSTOTAL

2 / 93 Community Score

2/93 security vendors flagged this IP address as malicious

85.187.128.24 (85.187.128.0/19) SG Last Analysis Date 17 hours ago

AS 55293 (A2HOSTING)

DETECTION DETAILS RELATIONS COMMUNITY

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

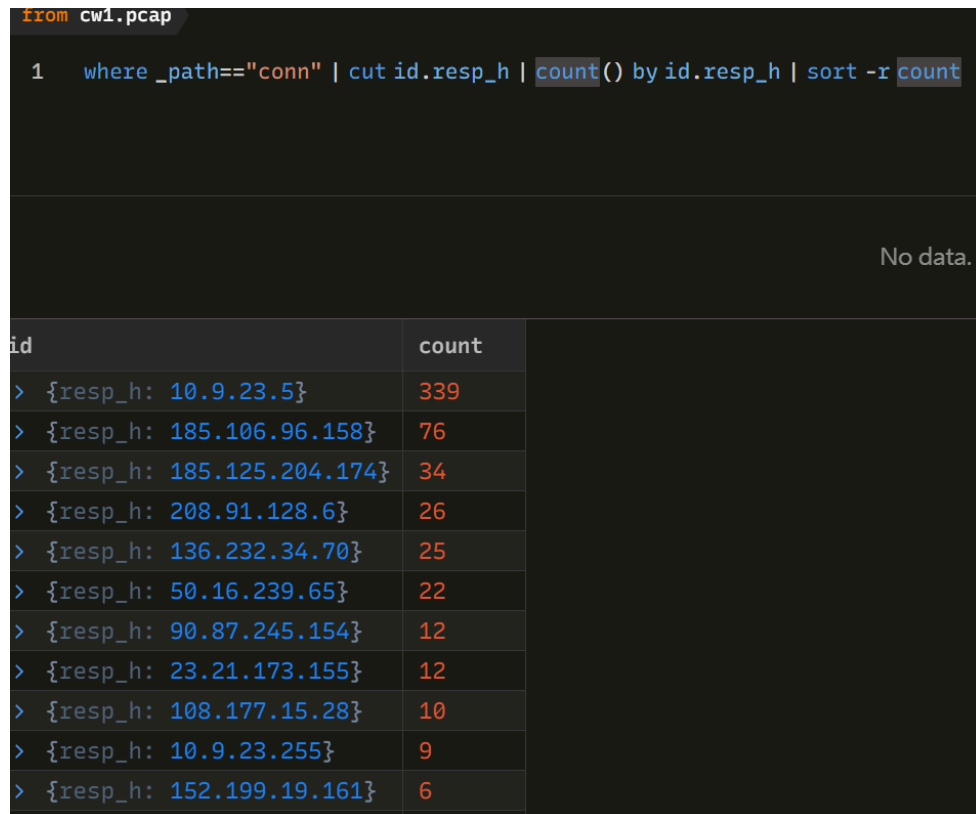
Security vendors' analysis

Criminal IP Malicious

Kaspersky Malware

AbuseIP Clean

Third, to filter out what destination IP address communicates the most between the victim, run the zed language query “**where _path=="conn" | cut id.resp_h | count() by id.resp_h | sort -r count**”, which is the result as below:



```
from cw1.pcap

1  where _path=="conn" | cut id.resp_h | count() by id.resp_h | sort -r count
```

No data.

id	count
> {resp_h: 10.9.23.5}	339
> {resp_h: 185.106.96.158}	76
> {resp_h: 185.125.204.174}	34
> {resp_h: 208.91.128.6}	26
> {resp_h: 136.232.34.70}	25
> {resp_h: 50.16.239.65}	22
> {resp_h: 90.87.245.154}	12
> {resp_h: 23.21.173.155}	12
> {resp_h: 108.177.15.28}	10
> {resp_h: 10.9.23.255}	9
> {resp_h: 152.199.19.161}	6

3 IP addresses request the most, which are 10.9.23.5, 185.106.96.158, and 185.125.204.174. Then submit those addresses to Virustotal. It found that **185.106.96.158** and **185.125.204.174** are C2 servers that allow attackers to remotely control infected systems, receive data from them, and issue malicious commands. (Virustotal Screenshot 1&2)

Fourth, when the malicious activity occurs, it has to analyze what DNS (Domain Name Server) is communicated after the victim opens the malicious excel file. Run the zed language query “**where _path=="dns" | where ts >= 2021-09-24T16:45:11Z | where ts <= 2021-09-24T17:45:35Z | cut ts, query | sort ts**” based on the hint.

ts	query
2021-09-24T16:45:11.44058Z	finejewels.com.au
2021-09-24T16:45:16.91361Z	self.events.data.microso
2021-09-24T16:45:20.26264Z	_ldap._tcp.default-first
2021-09-24T16:45:20.487123Z	thietbiagt.com
2021-09-24T16:45:25.457503Z	new.americold.com

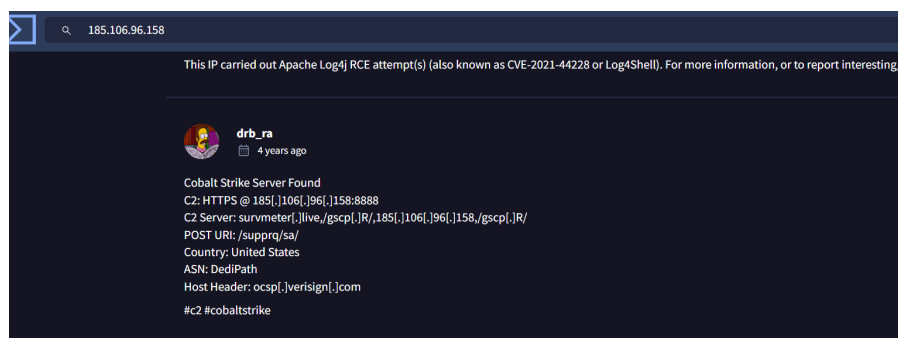
After applying the Virustotal checking, **finejewels.com.au**, **thietbiagt.com**, and **new.americold.com** were malicious.

2.3 Virustotal

Virustotal is used to determine whether an external IP address, domain name, URL, suspicious file, or MD5 hash value is malicious. These evaluations may have come from the security community.



Virustotal Screenshot 1. Virustotal review (1)

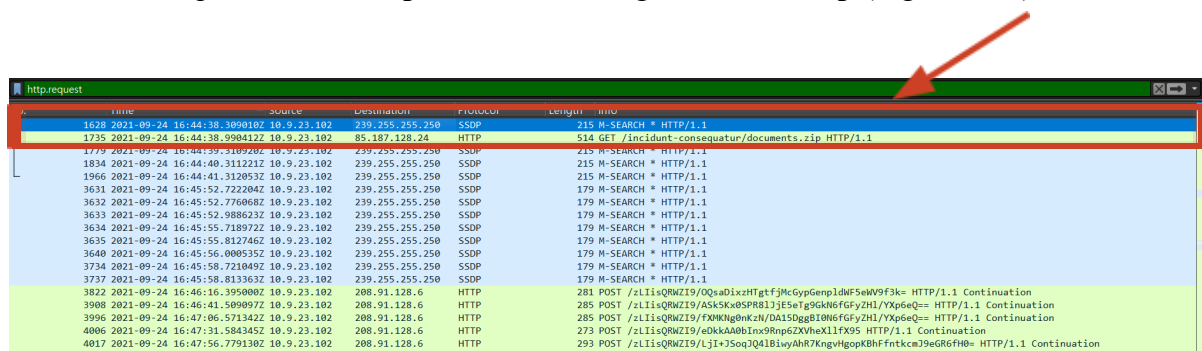


Virustotal Screenshot 2. Virustotal review (2)

Section 3 – Results

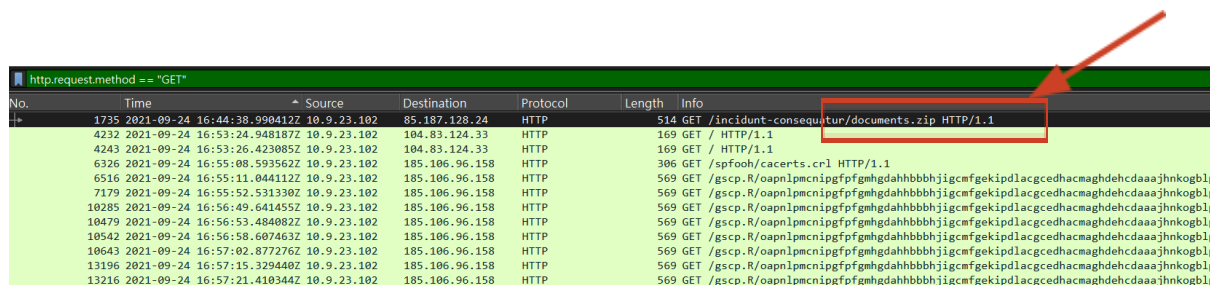
After the investigation, we found that the victim's host was infected after downloading a ZIP file from a suspicious website. That ZIP contained a malicious Excel file. Shortly after, the host began beaconing to attacker-controlled infrastructure (C2) and later started sending large volumes of phishing/spam emails directly to many SMTP servers (port 25), including messages with a malicious ZIP attachment.

Initial infection & file transfer: At **2021-09-24 16:44:38** (Figure 1), the victim made an HTTP GET request to **attirenepal.com**, retrieving **documents.zip** (Figure 2&3).



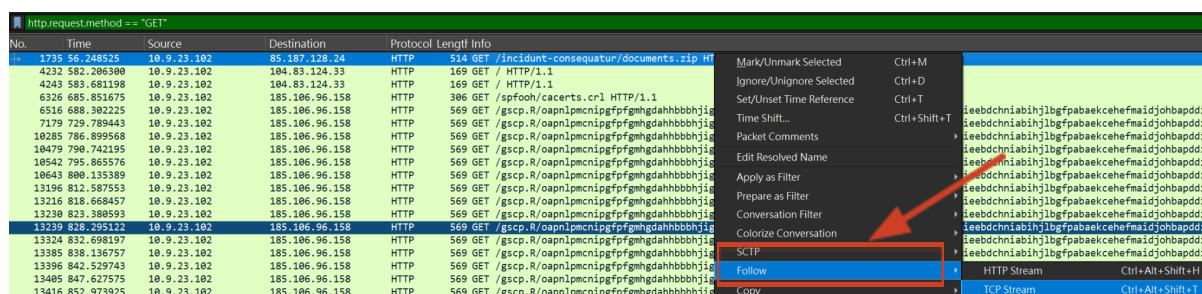
No.	Time	Source	Destination	Protocol	Length	Info
1735	2021-09-24 16:44:38.9904122	10.9.23.102	85.187.128.24	HTTP	514	GET /incident-consequatur/documents.zip HTTP/1.1
1779	2021-09-24 16:44:39.3102602	10.9.23.102	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1
1834	2021-09-24 16:44:40.3112212	10.9.23.102	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1
1966	2021-09-24 16:44:41.3120532	10.9.23.102	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1
3631	2021-09-24 16:45:52.7222042	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3632	2021-09-24 16:45:52.7768682	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3633	2021-09-24 16:45:52.9886232	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3634	2021-09-24 16:45:55.7189722	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3635	2021-09-24 16:45:55.8127462	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3640	2021-09-24 16:45:56.0005352	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3734	2021-09-24 16:45:58.7210492	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3737	2021-09-24 16:45:58.8133632	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3822	2021-09-24 16:46:16.3950002	10.9.23.102	208.91.128.6	HTTP	281	POST /zLIIsQRWZ19/OQaDixzHTgtfJMcGypGepnldwF5eW9f3k= HTTP/1.1 Continuation
3908	2021-09-24 16:46:41.5090972	10.9.23.102	208.91.128.6	HTTP	285	POST /zLIIsQRWZ19/ASKSkx0SPR81jE5eTg9Gkh6FGyZHL/YxpeQ= HTTP/1.1 Continuation
3996	2021-09-24 16:47:06.5713422	10.9.23.102	208.91.128.6	HTTP	285	POST /zLIIsQRWZ19/FXWKhGbnkN/DA15dg8I0M6FGyZHL/YxpeQ= HTTP/1.1 Continuation
4006	2021-09-24 16:47:31.5843452	10.9.23.102	208.91.128.6	HTTP	273	POST /zLIIsQRWZ19/eKNA00bJm0RnpG2XVhX1LFX9S HTTP/1.1 Continuation
4017	2021-09-24 16:47:56.7791302	10.9.23.102	208.91.128.6	HTTP	293	POST /zLIIsQRWZ19/Lj1+35oq3Q4lBisyAhR7KngvlgppKBHfvtckm39eGR6FH0= HTTP/1.1 Continuation
4037	2021-09-24 16:48:31.0867372	10.9.23.102	208.91.128.6	HTTP	180	POST /zLIIsQRWZ19/0RNM0C-A4-8DvLcEMg/TTTCE76ZmV1V1E7M1/0= HTTP/1.1 Continuation

Figure 1. HTTP request



No.	Time	Source	Destination	Protocol	Length	Info
1735	2021-09-24 16:44:38.9904122	10.9.23.102	85.187.128.24	HTTP	514	GET /incident-consequatur/documents.zip HTTP/1.1
4232	2021-09-24 16:53:24.9481872	10.9.23.102	104.83.124.33	HTTP	169	GET / HTTP/1.1
4243	2021-09-24 16:53:26.4230852	10.9.23.102	104.83.124.33	HTTP	169	GET / HTTP/1.1
6326	2021-09-24 16:55:08.5935622	10.9.23.102	185.106.96.158	HTTP	306	GET /spfooh/cacerts.cr1 HTTP/1.1
6516	2021-09-24 16:55:11.0441122	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
7179	2021-09-24 16:55:52.5313902	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
10285	2021-09-24 16:56:49.6414552	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
10479	2021-09-24 16:56:53.4840932	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
10542	2021-09-24 16:56:58.6074632	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
10643	2021-09-24 16:57:02.8772762	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
13196	2021-09-24 16:57:15.3294402	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
13216	2021-09-24 16:57:21.4193442	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp

Figure 2. HTTP GET request that discovers document.zip



No.	Time	Source	Destination	Protocol	Length	Info
1735	2021-09-24 16:44:38.9904122	10.9.23.102	85.187.128.24	HTTP	514	GET /incident-consequatur/documents.zip HTTP/1.1
4232	2021-09-24 16:53:24.9481872	10.9.23.102	104.83.124.33	HTTP	169	GET / HTTP/1.1
4243	2021-09-24 16:53:26.4230852	10.9.23.102	104.83.124.33	HTTP	169	GET / HTTP/1.1
6326	2021-09-24 16:55:08.5935622	10.9.23.102	185.106.96.158	HTTP	306	GET /spfooh/cacerts.cr1 HTTP/1.1
6516	2021-09-24 16:55:11.0441122	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
7179	2021-09-24 16:55:52.5313902	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
10285	2021-09-24 16:56:49.6414552	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
10479	2021-09-24 16:56:53.4840932	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
10542	2021-09-24 16:56:58.6074632	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
10643	2021-09-24 16:57:02.8772762	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
13196	2021-09-24 16:57:15.3294402	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp
13216	2021-09-24 16:57:21.4193442	10.9.23.102	185.106.96.158	HTTP	569	GET /gscp.R/oapnlpmcnigpfpfmgdahlbbbhjgicmfgekipdlacgedhacmaghdhcdadaajhknogblp

Figure 3. Select TCP Stream to view the dedicated traffic from GET /incident-consequatur/documents.zip HTTP/1.1

Exporting HTTP objects confirmed the archive contained **chart-1530076591.xls** (Figures 4 & 5). The malicious web server responded with Server: **LiteSpeed** with **PHP/7.2.34** (Figure 6), indicating the attacker-controlled infrastructure delivering the initial payload.

```
GET /incidunt-consequatur/documents.zip HTTP/1.1
Host: attirenepal.com
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.82 Safari/537.36 Edg/93.0.961.52
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: en

HTTP/1.1 200 OK
Connection: Keep-Alive
Keep-Alive: timeout=5, max=100
X-powered-by: PHP/7.2.34
Set-Cookie: PHPSESSID=3de639a4b99bd63f8f7b0ca7e3b6f14c; path=/
Content-Description: File Transfer
Content-Type: application/octet-stream
Content-Disposition: attachment; filename=documents.zip
Content-Transfer-Encoding: binary
Expires: 0
Cache-Control: must-revalidate, post-check=0, pre-check=0
Pragma: public
Transfer-Encoding: chunked
Date: Fri, 24 Sep 2021 16:44:06 GMT
Server: LiteSpeed
Strict-Transport-Security: max-age=3072000; includeSubDomains
X-Frame-Options: SAMEORIGIN
X-Content-Type-Options: nosniff

10000
PK.....d8S.a./.....chart-1530076591.xlsUT.....Ma..Maux.....[w...>..{.U8.X8...K.&
..S...c.4.g.X`H4..Ql.#...n.I...^.....sfY.....8.s.y.<..}.sfgv..j...
...8.U...[...
...K...h.OB...>...x.?..a...;..p...40.tn.gsc?...'^..
Q>.../X..B.h...MX.B.+.....x7&...g..!..Hkjkj..h7ox..1....~..w;..].8r..s....kp...$
..Q>...f.B.N...Gc...n...@p...../N..[.....[.1..#.....C.....
..$Ff...t...f.bxt.....Zo...;..f.g...=..s...N..".....k1.....na`.....p
..D..l.../...n...$.S...Pb..O;..C.wk.....(.w..w...N.Gv...v.....J...$.>..6..~T...
..b...b...S...].j...~...~...~...#R.....d/..).Q.AK..{G..+.....h
..f..I..Yp...n...^*.....t.3...;.....$. HR... $b..d.@..L..@0W&.Z...S
..B@t..*9...U.C.@49A...p...".1Q.D. b.....: ...1S...;...pg.B=...V...;..
..j...f.....@.....
..[5]F...u...7..$.S.....*j.MT ...F.Q...Ja..B6.(dc..H..6../.....{.....:
...;..eV...sq...zRA.....g..W.t...e.....0...V\...#G..$@a.&... @6..X
...D..V..K...Y...x/Xk...^...6.../...b...{...
...8..U...0K\...
OX9e...c...k...>...8
```

Figure 4. TCP Stream for /incidunt-consequatur/documents.zip

```
Strict-Transport-Security: max-age=3072000; includeSubDomains
X-Frame-Options: SAMEORIGIN
X-Content-Type-Options: nosniff

10000
PK.....d8S.a./.....chart-1530076591.xlsUT.....Ma..Maux.....[w...>..{.U8.X8...K.&
..S...c.4.g.X`H4..Ql.#...n.I...^.....sfY.....8.s.y.<..}.sfgv..j...
...8.U...[...
...K...h.OB...>...x.?..a...;..p...40.tn.gsc?...'^..
Q>.../X..B.h...MX.B.+.....x7&...g..!..Hkjkj..h7ox..1....~..w;..].8r..s....kp
..G<...n...@p...../N..[.....[.1..#.....C.....
..t...f.bxt.....Zo...;..f.g...=..s...N..".....k1.....na`.....p
..n...$.S...Pb..O;..C.wk.....(.w..w...N.Gv...v.....J...$.>..6..~T...
..b...b...S...].j...~...~...~...#R.....d/..).Q.AK..{G..+.....h
..f..I..Yp...n...^*.....t.3...;.....$. HR... $b..d.@..L..@0W&.Z...S
..J.C.@49A...p...".1Q.D. b.....: ...1S...;...pg.B=...V...;..
..j...f.....@.....
..[5]F...u...7..$.S.....*j.MT ...F.Q...Ja..B6.(dc..H..6../.....{.....:
...;..eV...sq...zRA.....g..W.t...e.....0...V\...#G..$@a.&... @6..X
...D..V..K...Y...x/Xk...^...6.../...b...{...
...8..U...0K\...
OX9e...c...k...>...8
```

Figure 5. chart-1530076591.xls inside the stream

```

HTTP/1.1 200 OK
Connection: Keep-Alive
Keep-Alive: timeout=5, max=100
x-powered-by: PHP/7.2.34
set-cookie: PHPSESSID=5de638a4b99bd63f8f7b0ca7e3b6f14c; path=/
content-description: File Transfer
content-type: application/octet-stream
content-disposition: attachment; filename=documents.zip
content-transfer-encoding: binary
expires: 0
cache-control: must-revalidate, post-check=0, pre-check=0
pragma: public
transfer-encoding: chunked
date: Fri, 24 Sep 2021 16:44:06 GMT
server: LiteSpeed
strict-transport-security: max-age=63072000; includeSubDomains
x-frame-options: SAMEORIGIN
x-content-type-options: nosniff

```

Figure 6. Web server and version

Additional download infrastructure: Within 16:45:11–16:45:30 UTC, additional domains contacted by the victim were **finejewels.com.au**, **thietbiagt.com**, and **new.americold.com**, as shown via DNS/TLS evidence (Figure 7). For the first domain, the TLS certificate issuer (CA) was **GoDaddy** (Figure 8).

2427	2021-09-24 16:45:11.840716Z	10.9.23.102	148.72.192.206	TLSv1.2	247 Client Hello	(SNI=finejewels.com.au)
2646	2021-09-24 16:45:17.228469Z	10.9.23.102	13.52.109.131	TLSv1.2	242 Client Hello	(SNI=events.data.microsoft.com)
2909	2021-09-24 16:45:20.389994Z	10.9.23.102	20.54.36.229	TLSv1.2	238 Client Hello	(SNI=mail-us-east-1.amazonaws.com)
3009	2021-09-24 16:45:21.314812Z	10.9.23.102	210.245.90.247	TLSv1.2	244 Client Hello	(SNI=thietbiagt.com)
3229	2021-09-24 16:45:25.731116Z	10.9.23.102	148.72.53.144	TLSv1.2	247 Client Hello	(SNI=new.americold.com)

Figure 7. DNS traffic

```

Wireshark - Follow TCP Stream (tcp.stream eq 90) - cw1.pcap
.....aN....Q..l. N^...v^..i/.6..[EE(UI...&.,+.0./.$.#.(.'
.....=<.5./
...i.....finejewels.com.au.
.....#.....h2.http/1.1.....
...N...J...s.%.....w.....ts.Zb.*:K2...0...#.....h2.....0...0.....
*.H..
....0..1.0..U...US1.0..U...Arizona1.0..U...
icottsdale1.0..U...
GoDaddy.com, Inc.1-0+..U...$http://certs.godaddy.com/repository/1301..U...Go Daddy Secure Certificate Authority - G2
00410090438Z.
00410090438Z0?1!0...U...Domain Control Validated1.0..U...finejewels.com.au0..0
*.H..
.....0..
....&.G.H.lj*x.%..[w-pB].%.w.G.W.M.IYd5...'.{...f.u...'.%w.cx...1|...5....Q...w.....16.....x.?+...-...G.....*.vS.v.
...{Q...s...V.2.K1...S5..P...R...h...R...1...n...w...Qx...Q..F.T..I.liTb_i...
.q...[l].l<m...v...S...H0..D0...U...0.0...U...%...0...+...08..U...10/0-+...).ht
http://certs.godaddy.com/repository/1301..U...$http://certs.godaddy.com/repository/1301..U...

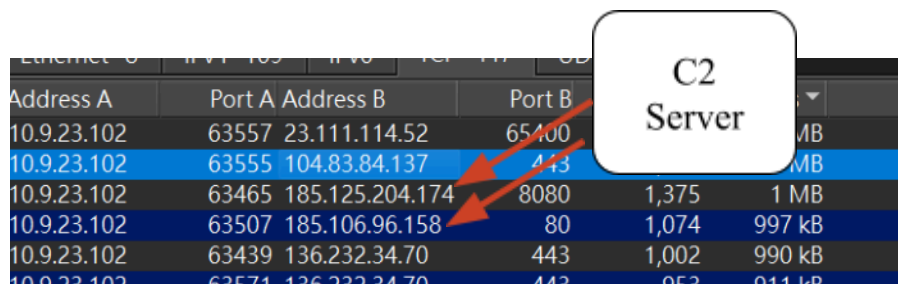
```

Figure 8. CA issuer

Suspected C2 infrastructure consistent: Using Statistics → Conversations, two C2 candidate servers were identified: **185.125.204.174** (Port 8080) and **185.106.96.158** (Port 80) (Figures 9 & 10 & 11).

DNS analysis linked **185.106.96.158** (Port 80) to **survmeter.live** (Figure 12), and the Host/SNI value observed for this infrastructure was **ocsp.verisign.com** (Figure 13).

For **185.125.204.174**, HTTP POST traffic revealed an association with **securitybusinpuff.com** (Figure 14).



Address A	Port A	Address B	Port B		
10.9.23.102	63557	23.111.114.52	65400		
10.9.23.102	63555	104.83.84.137	443		
10.9.23.102	63465	185.125.204.174	8080	1,375	1 MB
10.9.23.102	63507	185.106.96.158	80	1,074	997 kB
10.9.23.102	63439	136.232.34.70	443	1,002	990 kB
10.9.23.102	63571	136.232.34.70	443	953	911 kB

Figure 9. TCP stream statistics in order of packets



Figure 10. Virustotal review (1)

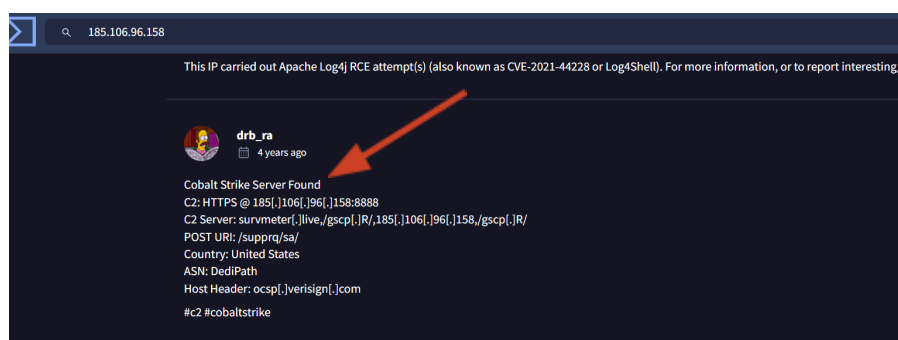


Figure 11. Virustotal review (2)

ip.addr == 185.106.96.158 && http

No.	Time	Source	Destination
6326	2021-09-24 16:55:08.593562Z	DESKTOP-IOJC6RB.goingfortune.com	survnmeter.live
6505	2021-09-24 16:55:10.600344Z	survnmeter.live	DESKTOP-IOJC6RB.goingfortune.com
6516	2021-09-24 16:55:11.044112Z	DESKTOP-IOJC6RB.goingfortune.com	survnmeter.live
6524	2021-09-24 16:55:11.290931Z	survnmeter.live	DESKTOP-IOJC6RB.goingfortune.com
7179	2021-09-24 16:55:52.531330Z	DESKTOP-IOJC6RB.goingfortune.com	survnmeter.live
7181	2021-09-24 16:55:52.831411Z	survnmeter.live	DESKTOP-IOJC6RB.goingfortune.com

Figure 12. Domain linked to 185.106.96.158

ip.addr == 185.106.96.158 && http

No.	Time	Source	Destination
6326	2021-09-24 16:55:08.593562Z	10.9.23.102	185.106.96.158
6505	2021-09-24 16:55:10.600344Z	185.106.96.158	10.9.23.102
6516	2021-09-24 16:55:11.044112Z	10.9.23.102	185.106.96.158
6524	2021-09-24 16:55:11.290931Z	185.106.96.158	10.9.23.102
7179	2021-09-24 16:55:52.531330Z	10.9.23.102	185.106.96.158
7181	2021-09-24 16:55:52.831411Z	185.106.96.158	10.9.23.102
10285	2021-09-24 16:56:49.641455Z	10.9.23.102	185.106.96.158
10291	2021-09-24 16:56:49.892169Z	185.106.96.158	10.9.23.102
10479	2021-09-24 16:56:53.484082Z	10.9.23.102	185.106.96.158
10487	2021-09-24 16:56:53.726722Z	185.106.96.158	10.9.23.102
10542	2021-09-24 16:56:58.607463Z	10.9.23.102	185.106.96.158
10550	2021-09-24 16:56:58.864872Z	185.106.96.158	10.9.23.102
10643	2021-09-24 16:57:02.877276Z	10.9.23.102	185.106.96.158
12145	2021-09-24 16:57:09.344753Z	185.106.96.158	10.9.23.102
12436	2021-09-24 16:57:10.251419Z	10.9.23.102	185.106.96.158
12954	2021-09-24 16:57:10.555165Z	185.106.96.158	10.9.23.102
13196	2021-09-24 16:57:15.329440Z	10.9.23.102	185.106.96.158
13198	2021-09-24 16:57:15.574022Z	185.106.96.158	10.9.23.102
13205	2021-09-24 16:57:15.825481Z	10.9.23.102	185.106.96.158
13207	2021-09-24 16:57:16.188305Z	185.106.96.158	10.9.23.102
13216	2021-09-24 16:57:21.410344Z	10.9.23.102	185.106.96.158
13218	2021-09-24 16:57:21.740566Z	185.106.96.158	10.9.23.102

Frame 6326: Packet, 306 bytes on wire (2448 bits), 306 bytes captured (2448 bits) on interface 0
 Ethernet II, Src: Hewlett-Packard_1c:47:ae (08:00:02:1c:47:ae), Dst: Netgear_b6:93:02 (08:00:27:08:00:27)
 Internet Protocol Version 4, Src: 10.9.23.102, Dst: 185.106.96.158
 Transmission Control Protocol, Src Port: 63447, Dst Port: 80, Seq: 1, Ack: 1, Len: 0
 Hypertext Transfer Protocol
 GET /spfooh/cacerts.crl HTTP/1.1\r\n
 Host: ocsf.verisign.com\r\n
 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.164 Safari/537.36\r\n
 Connection: Close\r\n
 Cache-Control: no-cache\r\n
 \r\n
 [Response in frame: 6505]
 [Full request URI: http://ocsp.verisign.com/spfooh/cacerts.crl]

Figure 13. Request URL for 185.106.96.158

ip.addr == 185.125.204.174 && tcp.port == 8080

No.	Time	Source	Destination	Protocol
4216	2021-09-24 16:53:24.084722Z	DESKTOP-IOJC6RB.goingfortune.com	securitybusinpu...com	TCP
4217	2021-09-24 16:53:24.246485Z	securitybusinpu...com	DESKTOP-IOJC6RB.goingfort...	TCP
4218	2021-09-24 16:53:24.246766Z	DESKTOP-IOJC6RB.goingfortune.com	securitybusinpu...com	TCP
4219	2021-09-24 16:53:24.251794Z	DESKTOP-IOJC6RB.goingfortune.com	securitybusinpu...com	TCP
4220	2021-09-24 16:53:24.251887Z	securitybusinpu...com	DESKTOP-IOJC6RB.goingfort...	TCP
4221	2021-09-24 16:53:24.425150Z	securitybusinpu...com	DESKTOP-IOJC6RB.goingfort...	TCP
4222	2021-09-24 16:53:24.425395Z	DESKTOP-IOJC6RB.goingfortune.com	securitybusinpu...com	TCP

Figure 14. Domain linked to 185.125.204.174

Post-infection beaconing traffic was primarily directed to **maldivehost.net** (Figure 15), with the victim sending data beginning with **zLIisQRWZI9** (Figure 16). The first packet from the victim to the C2 server had a frame length of **281** bytes (Figure 16). Server-side response headers from **maldivehost.net** indicated Server: **Apache/2.4.49 (cPanel) OpenSSL/1.1.1l mod_bwlimited/1.4** (Figure 17), which further supports that this domain was attacker-controlled and actively responding to client beacons.

ip.addr==10.9.23.102 && http.request.method=="POST"

No.	Time	Source	Destination
3822	2021-09-24 16:46:16.395000Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net
3908	2021-09-24 16:46:41.509097Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net
3996	2021-09-24 16:47:06.571342Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net
4006	2021-09-24 16:47:31.584345Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net
4017	2021-09-24 16:47:56.779130Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net
4027	2021-09-24 16:48:21.805873Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net
4037	2021-09-24 16:48:46.850457Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net
4046	2021-09-24 16:49:11.959706Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net
4090	2021-09-24 16:49:37.041462Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net
4099	2021-09-24 16:50:02.211046Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net
4109	2021-09-24 16:50:27.298936Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net
4118	2021-09-24 16:50:52.306135Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net

Figure 15. Post-infection beaconing traffic

No.	Time	Source	Destination	Protocol	Length	Info
3822	2021-09-24 16:46:16.395000Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net	HTTP	281	POST /zLIisQRWZI9/OQsaDixzHTgtfj
3908	2021-09-24 16:46:41.509097Z	DESKTOP-IOJC6RB.goingfortune.com	maldivehost.net	HTTP	285	POST /zLIisQRWZI9/ASK5Kx0SPR8lJj

Figure 16. Post-infection beaconing traffic

Wireshark · Follow TCP Stream (tcp.stream eq 111) · cw1.pcap

```
POST /zLIisQRWZI9/LjI+JS0qJQ4lBiwyAhR7KngvHgopKBhFfntkcmJ9eGR6fH0= HTTP/1.1
Host: maldivehost.net
Content-Length: 112
```

Server
Header

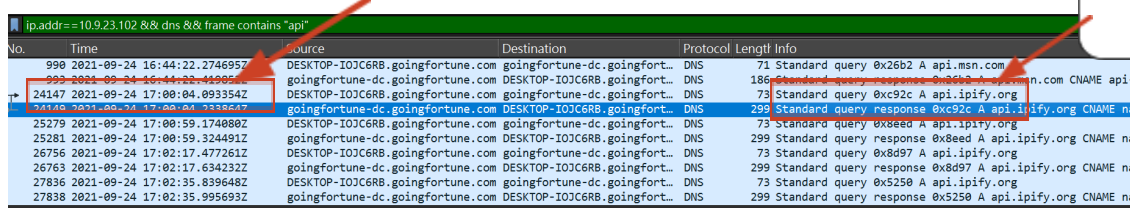
```
Dw8YBxsEGmYFAAEJfR4NQkMmLTyqZDk5KyQmOyRGQglxEBo4Lzk/EyYrMi1hOT8vIyM7IhcNPas0gAgIIDQUZGB0FD0Jf
```

```
HTTP/1.1 200 OK
Date: Fri, 24 Sep 2021 16:47:55 GMT
Server: Apache/2.4.49 (cPanel) OpenSSL/1.1.1l mod_bwlimited/1.4
X-Powered-By: PHP/5.6.40
Content-Length: 302
Strict-Transport-Security: ...max-age=15552000...
Connection: close
Content-Type: text/html; charset=UTF-8
```

```
eXp7QUVCQ0FBfn15eX1/eXp8e0JBQ0JGQnpzeWJ+eXtleH11f3xBRUJDQUELDhKAGAAbZWIDBQh8GQ5GQicqNS51OD4oICc6I0VGCHAXGTWu0DgQIiozKmI9Pi4KIDeQDBQsJBBgFGQE0Q0JB RUJDQUEBBAQ0QUVCQ0FB AQQEDkFFQkNBQ0EEBA5BRUJDQUFCQUNCRkJGQ0EJFRUZHQUVGQudGRkZCQ0EzBRUJDQUFCQUNCRkJGQ0EJFRUZHQUVG
```

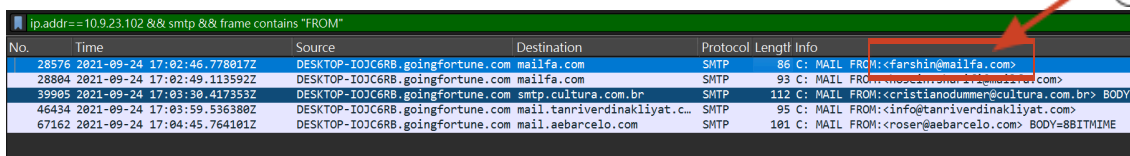
Figure 17. Server header

Final exfiltration/check-in: The malware performed an external IP check via DNS at **2021-09-24 17:00:04**, querying **api.ipify.org** (Figure 18). SMTP traffic showed the first MAIL FROM address as **farshin@mailfa.com** (Figure 19). Following the same TCP stream, the password was transmitted using AUTH LOGIN (Base64), and the password associated with ho3ein.sharifi's password is **13691369** (Figures 20 & 21).



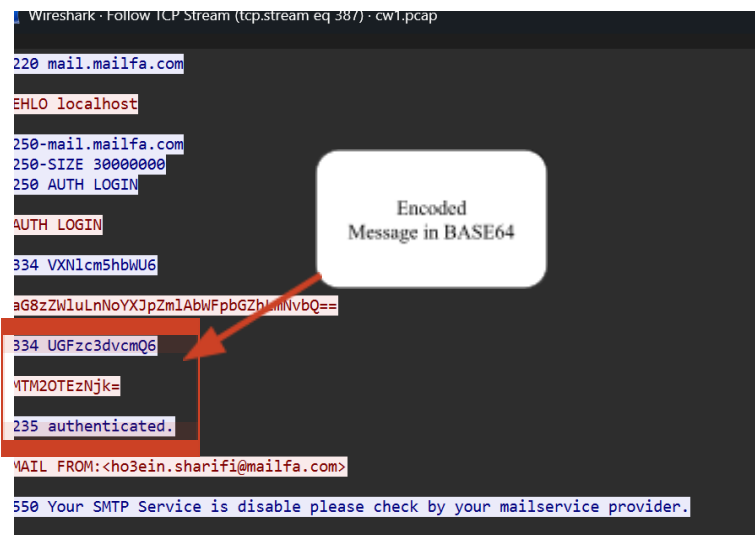
No.	Time	Source	Destination	Protocol	Length	Info
990	2021-09-24 16:44:22.2746957	DESKTOP-IOJ6R8B.goingfortune.com	goingfortune-dc.goingfort...	DNS	71	Standard query 0x26b2 A api.msn.com
993	2021-09-24 16:44:23.4490581	goingfortune-dc.goingfortune.com	DESKTOP-IOJ6R8B.goingfort...	DNS	186	Standard query response 0x26b2 A api.msn.com CNAME api...
24147	2021-09-24 17:00:04.093354Z	DESKTOP-IOJ6R8B.goingfortune.com	goingfortune-dc.goingfort...	DNS	73	Standard query 0xc92c A api.ipify.org
24148	2021-09-24 17:00:04.2338647	goingfortune-dc.goingfortune.com	DESKTOP-IOJ6R8B.goingfort...	DNS	299	Standard query response 0xc92c A api.ipify.org CNAME n...
25279	2021-09-24 17:00:59.174080Z	DESKTOP-IOJ6R8B.goingfortune.com	goingfortune-dc.goingfort...	DNS	73	Standard query 0xc92c A api.ipify.org
25281	2021-09-24 17:00:59.324491Z	goingfortune-dc.goingfortune.com	DESKTOP-IOJ6R8B.goingfort...	DNS	299	Standard query response 0xc92c A api.ipify.org CNAME n...
26756	2021-09-24 17:02:17.477261Z	DESKTOP-IOJ6R8B.goingfortune.com	goingfortune-dc.goingfort...	DNS	73	Standard query 0xc92c A api.ipify.org
26763	2021-09-24 17:02:17.634232Z	goingfortune-dc.goingfortune.com	DESKTOP-IOJ6R8B.goingfort...	DNS	299	Standard query response 0xc92c A api.ipify.org CNAME n...
27836	2021-09-24 17:02:35.839648Z	DESKTOP-IOJ6R8B.goingfortune.com	goingfortune-dc.goingfort...	DNS	73	Standard query 0xc92c A api.ipify.org
27838	2021-09-24 17:02:35.995693Z	goingfortune-dc.goingfortune.com	DESKTOP-IOJ6R8B.goingfort...	DNS	299	Standard query response 0xc92c A api.ipify.org CNAME n...

Figure 18. A DNS query occurred for the domain used by the malware



No.	Time	Source	Destination	Protocol	Length	Info
28576	2021-09-24 17:02:46.778017Z	DESKTOP-IOJ6R8B.goingfortune.com	mailfa.com	SMTP	86	C: MAIL FROM:cfarshin@mailfa.com>
28804	2021-09-24 17:02:49.113592Z	DESKTOP-IOJ6R8B.goingfortune.com	mailfa.com	SMTP	93	C: MAIL FROM:cfarshin@mailfa.com>
39985	2021-09-24 17:03:30.417353Z	DESKTOP-IOJ6R8B.goingfortune.com	smtp.cultuna.com.br	SMTP	112	C: MAIL FROM:ccristianodummer@cultuna.com.br> BODY=
46434	2021-09-24 17:03:59.536380Z	DESKTOP-IOJ6R8B.goingfortune.com	mail.tanriverdinakliyat.c...	SMTP	95	C: MAIL FROM:cinfo@tanriverdinakliyat.c...
67162	2021-09-24 17:04:45.764101Z	DESKTOP-IOJ6R8B.goingfortune.com	mail.aebacelo.com	SMTP	101	C: MAIL FROM:croser@aebacelo.com> BODY=8BITIME

Figure 19. SMTP traffic



```

Wireshark · Follow TCP Stream (tcp.stream eq 387) · cw1.pcap

220 mail.mailfa.com
EHLO localhost
250-mail.mailfa.com
250-SIZE 30000000
250 AUTH LOGIN
AUTH LOGIN
334 VXNlcmShbWU6
aG8zZWluLnNoYXJpZmIAbWpBbGZlbnNvbQ==
334 UGFzc3dvcmQ6
MTM2OTEzNjk=
235 authenticated.
MAIL FROM:<ho3ein.sharifi@mailfa.com>
550 Your SMTP Service is disable please check by your mailservice provider.
  
```

Figure 20. Packet content

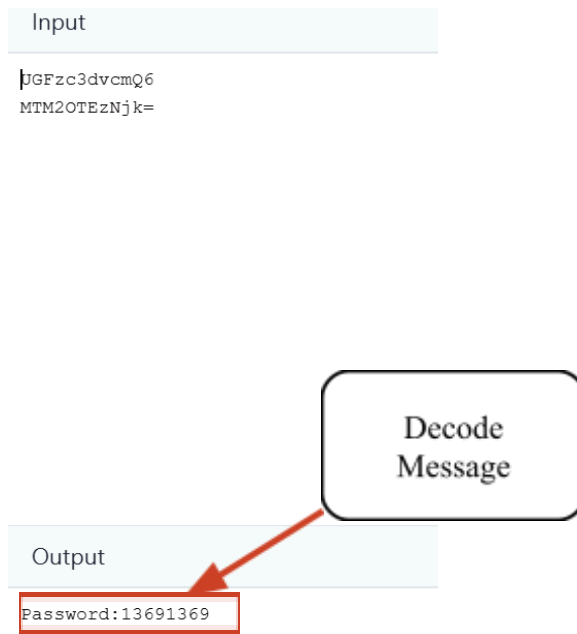


Figure 21. BASE64 decryption

Section 4 – Conclusion and References

4.1 Conclusion

This investigation identified the compromised host (10.9.23.102) and reconstructed the infection lifecycle observed in the PCAP. The incident began with the download of a malicious compressed archive containing an Excel file, which likely acted as the initial execution vector. The host subsequently established persistent command-and-control communication through HTTP POST beaconing and encrypted TLS connections, performed external IP discovery, and later initiated outbound SMTP authentication attempts, indicating that the system was repurposed for malspam or phishing activity.

To prevent the possibility of such an attack by stage:

Attack Stage	Evidence Observed	Targeted Prevention
Initial Access	HTTP download of ZIP	Web proxy filtering, file-type blocking, phishing protection, firewall rules, and antivirus scanning
Execution	XLS macro payload	Disable Office macros by default, user privilege restriction
C2	Obfuscated POST beacons	Egress filtering, TLS inspection, IDS/IPS monitoring
Recon	api.ipify.org lookup	DNS anomaly detection, threat intelligence feeds
Impact	SMTP spam	Outbound SMTP restrictions, email gateway monitoring

Table 2. Mapping of observed attack stages to preventative controls

Despite the implementation of these preventive measures, several open issues and challenges remain. A key challenge is the increasing prevalence of encrypted network traffic, which reduces payload visibility and complicates the detection of infection indicators. Another challenge lies in distinguishing legitimate but unusual user behaviour from malicious activity,

which may lead to false positives if not carefully contextualised. Additionally, modern malware may intentionally evade detection by mimicking normal traffic patterns or avoiding explicit disclosure of host identifiers.

In conclusion, while network-based intrusion analysis remains an effective approach for identifying compromised systems and reconstructing attack lifecycles, it must be complemented by layered defensive controls, secure configuration practices, and continuous monitoring to effectively prevent and respond to future incidents.

4.2 References

1. Palo Alto Networks Unit 42:
Unit 42. (n.d.). Using Wireshark display filter expressions. Palo Alto Networks.
<https://unit42.paloaltonetworks.com/using-wireshark-display-filter-expressions/>
2. Wireshark User's Guide:
Wireshark Foundation. (2019). Wireshark user's guide (Version 4.7.0).
https://www.wireshark.org/docs/wsug_html_chunked/
3. Brim Data. (2026). ZUI GitHub. <https://github.com/brimdata/zui>

Appendix

Part 1: Initial Infection & File Transfer

1. When did the initial malicious HTTP connection occur? (Provide the date and time in yyyy-mm-dd hh:mm:ss format).

Ans:

No.	Time	Source	Destination	Protocol	Length	Info
1628	2021-09-24 16:44:38.309010Z	10.9.23.102	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1
1735	2021-09-24 16:44:38.990412Z	10.9.23.102	85.187.128.24	HTTP	514	GET /incidunt-consequatur/documents.zip HTTP/1.1
1779	2021-09-24 16:44:39.310970Z	10.9.23.102	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1
1834	2021-09-24 16:44:40.311221Z	10.9.23.102	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1
1966	2021-09-24 16:44:41.312053Z	10.9.23.102	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1
3631	2021-09-24 16:45:52.722204Z	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3632	2021-09-24 16:45:52.776068Z	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3633	2021-09-24 16:45:52.988623Z	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3634	2021-09-24 16:45:55.798972Z	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3635	2021-09-24 16:45:55.812746Z	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3640	2021-09-24 16:45:56.000535Z	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3734	2021-09-24 16:45:58.721049Z	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3737	2021-09-24 16:45:58.813363Z	10.9.23.102	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3822	2021-09-24 16:46:16.395008Z	10.9.23.102	208.91.128.6	HTTP	281	POST /z/1iisqRwMZ/0Qqad1xztGtf9KcGyngp1dWf5Ww9f3k= HTTP/1.1 Continuation
3908	2021-09-24 16:46:41.509097Z	10.9.23.102	208.91.128.6	HTTP	285	POST /z/1iisqRwMZ/ASk5v00SPR81jZ5eTg6M6fGfyZHLfYXp6eQ= HTTP/1.1 Continuation
3996	2021-09-24 16:47:06.517342Z	10.9.23.102	208.91.128.6	HTTP	285	POST /z/1iisqRwMZ/FYwKlNgkNz/DAl5Dg8qB10M6fGfyZHLfYXp6eQ= HTTP/1.1 Continuation
4006	2021-09-24 16:47:31.584345Z	10.9.23.102	208.91.128.6	HTTP	237	POST /z/1iisqRwMZ/e0kAA0b1zn9n6pG2XvHk1fV95 HTTP/1.1 Continuation
4021	2021-09-24 16:47:56.779130Z	10.9.23.102	208.91.128.6	HTTP	293	POST /z/1iisqRwMZ/Lj1r75oqJQ4L8iayAHR7XgXvHk1fV95 HTTP/1.1 Continuation

```
1735  2021-09-24 16:44:38.990412Z      10.9.23.102  85.187.128.24 HTTP 514
GET /incidunt-consequatur/documents.zip HTTP/1.1
```

UTC Arrival Time: Sep 24, 2021 16:44:38.309010000 UTC

2021-09-24 16:44:38

2. What is the name of the compressed file that the victim downloaded?

Ans:

No.	Time	Source	Destination	Protocol	Length	Info
1735	2021-09-24 16:44:38.9984127	10.9.2.3.192	85.187.128.24	HTTP	514	GET /incident-consecurator/documents.zip HTTP/1.1
4232	2021-09-24 16:53:24.9481877	10.9.2.3.192	104.83.124.3	HTTP	169	GET / HTTP/1.1
4243	2021-09-24 16:53:26.4230855	10.9.2.3.192	104.83.124.3	HTTP	169	GET / HTTP/1.1
6326	2021-09-24 16:55:08.5935622	10.9.2.3.192	185.106.96.158	HTTP	306	/spfooh/cacerts.crl HTTP/1.1
6516	2021-09-24 16:55:11.0441122	10.9.2.3.192	185.106.96.158	HTTP	569	GET /gscp./R/oaaplmcnigpffgmghdabhhbbjigcnfgekpidlagcedhacmaghdcdadajhknqoblp
7179	2021-09-24 16:55:52.5313987	10.9.2.3.192	185.106.96.158	HTTP	569	GET /gscp./R/oaaplmcnigpffgmghdabhhbbjigcnfgekpidlagcedhacmaghdcdadajhknqoblp
10285	2021-09-24 16:56:49.6414552	10.9.2.3.192	185.106.96.158	HTTP	569	GET /gscp./R/oaaplmcnigpffgmghdabhhbbjigcnfgekpidlagcedhacmaghdcdadajhknqoblp
10479	2021-09-24 16:56:53.4894827	10.9.2.3.192	185.106.96.158	HTTP	569	GET /gscp./R/oaaplmcnigpffgmghdabhhbbjigcnfgekpidlagcedhacmaghdcdadajhknqoblp
10542	2021-09-24 16:56:59.5074637	10.9.2.3.192	185.106.96.158	HTTP	569	GET /gscp./R/oaaplmcnigpffgmghdabhhbbjigcnfgekpidlagcedhacmaghdcdadajhknqoblp
10643	2021-09-24 16:57:02.8772762	10.9.2.3.192	185.106.96.158	HTTP	569	GET /gscp./R/oaaplmcnigpffgmghdabhhbbjigcnfgekpidlagcedhacmaghdcdadajhknqoblp
13196	2021-09-24 16:57:15.3294402	10.9.2.3.192	185.106.96.158	HTTP	569	GET /gscp./R/oaaplmcnigpffgmghdabhhbbjigcnfgekpidlagcedhacmaghdcdadajhknqoblp
13216	2021-09-24 16:57:21.4103447	10.9.2.3.192	185.106.96.158	HTTP	569	GET /gscp./R/oaaplmcnigpffgmghdabhhbbjigcnfgekpidlagcedhacmaghdcdadajhknqoblp

[Full request URI: <http://attirenepal.com/incidunt-consequatur/documents.zip>]

documents.zip

3. Which domain hosted the malicious compressed file?

Ans:

attirenepal.com

4. What is the name of the file located inside the compressed archive?

Ans:

```
pragma: public
transfer-encoding: chunked
date: Fri, 24 Sep 2021 16:44:06 GMT
server: LiteSpeed
strict-transport-security: max-age=63072000; includeSubDomains
-x-frame-options: SAMEORIGIN
-content-type-options: nosniff

0000
K.....d8S.a../.....chart-1530076591.xlsUT.....Ma..Maux.....[w\...>..[
..5... ..c.4.4.g...X.`H4..Ql.#...n.I...^.....sfY.....8.s.y...<..}.sfgv..j.....+.h.
...8.U~.....[.....
...K...h.OB....>...x.?a.;..p....40.tn.gsc?...'^..
<.....^/X..@.h.<..MX.B.+.....x7&....g..!.Hkjkj..h7ox..1...~.w;..].8r..s....kp...ft..
Q`>..f.8.N...G<....n=.....@p...../N..[.....[.1..#.....C.....
...$tF.....t...f.bxt.....Zo..;..f.g...=.s....N.".....kl.....na`.....p.[.....
.D.l1../...n...$.S...Pb..O;..C.wk.....(.w.w...N.Gv..v.....J...$.>.6...~T....K..p..$
.X.....b..b..z.....,S.]...j...~...~...#R.....d/.)..Q.AK..{G.:+..
...."....8,y7R....8...v/m.....f..I..Yp...n.....^*.....t.3.;.....$. HR
...fO;z.A.....v.O;h...NO;....^O;w.!...6...0..@t.*9..<U.C.@49A....p....".1Q.D. b.
...V...7=...H...9...8...A...H...f'...0...H...f'...0...
```

chart-1530076591.xls

5. Identify the specific web server software (Server header) running on the malicious IP address that served the compressed file.

Ans:

```
transfer-encoding: chunked
date: Fri, 24 Sep 2021 16:44:06 GMT
server: LiteSpeed
strict-transport-security: max-age=63072000; includeSubDomains
-x-frame-options: SAMEORIGIN
-x-content-type-options: nosniff

10000
PK.....d8S.a../.....chart-1530076591.xlsUT
..5... ..c.4.4.g...X.`H4..Ql.#...n.I...^.....sfY.....
...8.U~.....[.....
...K...h.OB....>...x.?a.;..p....40.tn.gsc?...'^..
Q<.....^/X..@.h.<..MX.B.+.....x7&....g..!.Hkjkj..
.Q`>..f.8.N...G<....n=.....@p...../N..[.....[.1..
...$tF.....t...f.bxt.....Zo..;..f.g...=.s....N.
.D.l1../...n...$.S...Pb..O;..C.wk.....(.w.w...
:X.....b..b..z.....,S.]...j...~...~...
...."....8,y7R....8...v/m.....f..I..Yp...n.....
client pkt(s), 148 server pkt(s), 1 turn(s).
Entire conversation (199 kB) Show
ind: server
```

LiteSpeed

6. What is the version number of the web server identified in the previous question?

Ans:

PHP/7.2.34

7. Identify the three additional domains that were involved in downloading malicious files to the victim host.

Hint: Inspect HTTPS traffic and focus on the time window between 16:45:11 and 16:45:30. Note this range is in UTC, not BST.

Ans:

2427	2021-09-24	16:45:11.840716Z	10.9.23.102	148.72.192.206	TLSv1.2	247 Client Hello (SNI=finejewels.com.au)
2646	2021-09-24	16:45:17.228469Z	10.9.23.102	13.69.109.131	TLSv1.2	242 Client Hello (SNI=self.events.data.microsoft.com)
2909	2021-09-24	16:45:20.389994Z	10.9.23.102	20.54.36.229	TLSv1.2	238 Client Hello (SNI=client.wns.windows.com)
3009	2021-09-24	16:45:21.314012Z	10.9.23.102	210.245.90.247	TLSv1.2	244 Client Hello (SNI=thietbiagt.com)
3229	2021-09-24	16:45:25.731116Z	10.9.23.102	148.72.53.144	TLSv1.2	247 Client Hello (SNI=new.americold.com)

finejewels.com.au

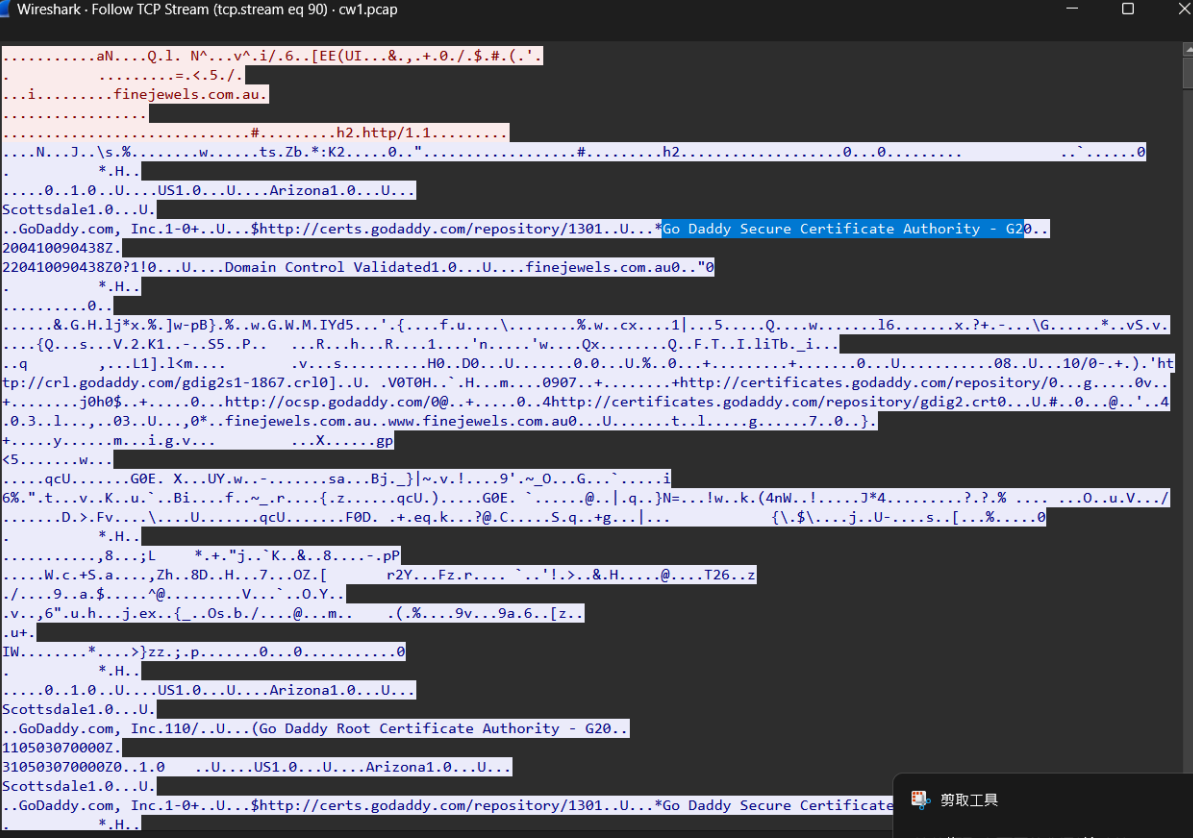
thietbiagt.com

new.americold.com

Part 2: Command and Control (C2) Activity

8. Which Certificate Authority (CA) issued the SSL certificate for the first domain identified in question 7?

Ans:



```
.....aN...Q.l. N^...v^i/.6..[EE(UI...&.,+.0./.$.#.(.'
.....=.<.5./
.....finejewels.com.au.
.....#.....h2.http/1.1.....
...N...J...s.%.....w.....ts.Zb.*:K2.....0.....#.....h2.....0...0.....
*H..
.....0..1.0..U...US1.0..U...Arizona1.0..U...
Scottsdale1.0..U...
..GoDaddy.com, Inc.1-0+..U...$http://certs.godaddy.com/repository/1301..U...*Go Daddy Secure Certificate Authority - G20..
200410090438Z.
220410090438Z0?110...U...Domain Control Validated1.0..U...finejewels.com.au0..0
*H..
.....0..
.....&G.H.lj*x.%]w-pB}.%.w.G.W.M.IYd5...'.{...f.u...\......%w..cx....1|...5....Q...w.....16.....x.?+...G.....*..vS.v.
...{Q...s...V.2.K1...S5..P...R...h...R...1...'n....'w...Qx.....Q..F.T.I.liTb._i...
..q...l1].l<m...v...s...H0..D0..U...0..0..U...0...+.....0..U.....08..U...10/0-+.).'ht
tp://crl.godaddy.com/gdig2s1-1867.crl0].U..V0T0H..H...m...0907..+.....+http://certificates.godaddy.com/repository/0...g....0v...
+.....j0h0$.+.....0...http://ocsp.godaddy.com/0@..+.....0..4http://certificates.godaddy.com/repository/gdig2.crt0...U.#..0...@...'.4
.0.3..l...03..U...0*..finejewels.com.au..www.finejewels.com.au..U.....t..l....g.....7..0..}.
+.....y.....m..i.g.v...X.....gp
<5.....w...
.....qcU.....G0E. X...UY.w...sa...Bj_)|~.v.!...9'.~_O...G...^.....i
6%."t...v..K..u..Bi...f...r...{.z.....qcU.).....G0E. ^.....@..|.q..}N=...!w..k.(4nW..!.....J*4.....?..%.....0..u.V.../
.....D..>.Fv...U.....qcU.....F0D..+eq.k...?@.C....S.q..+g...|. {\. $ \...j..U-...s...[...%.....0
*H..
.....8...;L...+."j..`K..&..8....-pP
.....W.c.+S..a....Zh..8D..H...7...OZ.[r2Y...Fz.r...`'!.>..&H....@....T26..z
/.....9..a.$..^@.....V...`O.Y...
.v...6".u.h...j.ex..{..Os.b./...@...m...(.%...9v...9a.6..[z..
.u+
IW.....*.....>zz;j.p.....0...0.....0
*H..
.....0..1.0..U...US1.0..U...Arizona1.0..U...
Scottsdale1.0..U...
..GoDaddy.com, Inc.110/..U...(Go Daddy Root Certificate Authority - G20..
110503070000Z.
310503070000Z0..1.0..U...US1.0..U...Arizona1.0..U...
Scottsdale1.0..U...
..GoDaddy.com, Inc.1-0+..U...$http://certs.godaddy.com/repository/1301..U...*Go Daddy Secure Certificate
*H..
Packet 2433. 5 client pkt(s), 294 server pkt(s), 5 turn(s). Click to select.
Entire conversation (395 kB) Show as ASCII No delta times
```

GoDaddy

9. What are the two IP addresses of the Cobalt Strike servers? (Provide them in sequential order).

Hint: Inspect the Conversations menu option


Ans:

Address A	Port A	Address B	Port B	Packets	Bytes
10.9.23.102	63557	23.111.114.52	65400	18,002	16 MB
10.9.23.102	63555	104.83.84.137	443	9,074	10 MB
10.9.23.102	63465	185.125.204.174	8080	1,375	1 MB
10.9.23.102	63507	185.106.96.158	80	1,074	997 kB
10.9.23.102	63439	136.232.34.70	443	1,002	990 kB
10.9.23.102	63571	136.232.34.70	443	953	911 kB

Virustotal

185.106.96.158

This IP carried out Apache Log4j RCE attempt(s) (also known as CVE-2021-44228 or Log4Shell). For more information, or to report interesting/in

 **drb_ra**
4 years ago

Cobalt Strike Server Found
C2: HTTPS @ 185[.]106[.]96[.]158:8888
C2 Server: survmeter[.]live[.]gscp[.]R/185[.]106[.]96[.]158[.]gscp[.]R/
POST URI: /supprq/sa/
Country: United States
ASN: DediPath
Host Header: ocsp[.]verisign[.]com
#c2 #cobaltstrike

185.125.204.174

This IP carried out Apache Log4j RCE attempt(s) (also known as CVE-2021-44228 or Log4Shell). For more information, or to report interesting/in

 **drb_ra**
4 years ago

Cobalt Strike Server Found
C2: HTTPS @ 185[.]125[.]204[.]174:4444
C2 Server: securitybusinbuff[.]com/jquery-3[.]3[.]1[.]min[.]js,185[.]125[.]204[.]174/jquery-3[.]3[.]1[.]min[.]js
POST URI: /jquery-3[.]3[.]2[.]min[.]js
Country: N/A
ASN: Hydra Communications Ltd
#c2 #cobaltstrike

185.106.96.158

185.125.204.174

10. What is the value of the Host header for the first Cobalt Strike IP address?

Hint: Apply a filter to isolate DNS queries.

Ans:

ocsp.verisign.com

11. What is the domain name associated with the first Cobalt Strike IP address?

Hint: Take a closer look at HTTPS (443)

Ans:

survmeter.live

12. What is the domain name associated with the second Cobalt Strike IP address?

Hint: Apply a filter to capture HTTP POST requests.

Ans:

securitybusinpuff.com

13. What is the domain name used for the post-infection traffic?

Ans:

maldivehost.net

14. What are the first eleven characters of the data the victim host sends to the malicious domain identified in the previous question?

Ans:

zLIisQRWZI9

15. What was the length of the first packet the victim sent to the C2 server?

Ans:

281

16. What was the Server header value for the malicious domain from question 13?

Ans:

Apache/2.4.49 (cPanel) OpenSSL/1.1.1l mod_bwlimited/1.4

Part 3: Final Exfiltration/Check-in

17. What was the date and time (in yyyy-mm-dd hh:mm:ss UTC format) when the DNS query occurred for the domain used by the malware to check the victim's external IP address?

Ans:

2021-09-24 17:00:04

18. What was the domain name in the DNS query from the previous question?

Ans:

api.ipify.org

19. What was the first email address observed in the SMTP traffic in the pcap file (the MAIL FROM address)?

Ans:

28576 2021-09-24 17:02:46.778017Z 10.9.23.102 185.4.29.135 SMTP 86 C:
MAIL FROM:<farshin@mailfa.com>

farshin@mailfa.com

20. Follow the stream from Q19. What is ho3ein.sharifi's password?

Ans:

220 mail.mailfa.com

EHLO localhost

250-mail.mailfa.com

250-SIZE 30000000

250 AUTH LOGIN

AUTH LOGIN

334 VXNlcm5hbWU6

ZmFyc2hpbkBtYWlsZmEuY29t

334 UGFzc3dvcmQ6

ZGluYW1pdA==

235 authenticated.

MAIL FROM:<farshin@mailfa.com>

550 Your SMTP Service is disabled please check by your mailservice provider.

Due to the communication using BASE64, which is decryptable, the password was found in the decrypt output.

Password:**13691369**

Run the query below to get the clear details of the alert event:

```
event_type=="alert" | cut
ts,src_ip,dest_ip,dest_port,flow_id,alert.signature,alert.severity
,alert.category | sort -r ts
```

The result can be found on the GitHub repository

(https://github.com/aka331/COMP3010HK/blob/main/CW1/event_alert.csv)

Timeline (key events with exact times)

All times are from the PCAP timestamps.

1. Initial download (infection vector)

- **16:44:38.990** — Victim downloads a ZIP over HTTP:
GET /incidunt-consequatur/documents.zip from **attirenepal.com** (IP: **85.187.128.24**)

Extracted that file from the PCAP, and it contains:

- chart-1530076591.xls (inside documents.zip)

This is a classic pattern: **ZIP** → **Excel lure** → **macro/dropper execution**.

2. First C2 / malware check-in

- **16:46:16.395** — Victim starts POST beacons to **maldivehost.net** (IP: **208.91.128.6**)
with highly obfuscated URI paths like:
POST /zLIisQRWZI9/<base64-ish blob>

This looks like **malware C2 polling/tasking**.

3. Second C2 channel over TLS

- **16:53:28.188** — Victim initiates TLS sessions to **185.125.204.174:8080** with SNI:
securitybusinpuff.com
Certificate chain shows **Let's Encrypt (R3)**, which is common for attacker infrastructure too.

4. External IP discovery

- **17:00:04.093** — DNS query for **api.ipify.org**

This is often used by malware to learn the victim's **public IP**.

5. **Botnet spamming/phishing begins**

- **17:02:43.150** — Clear SMTP commands observed from victim to external mail servers (port **25**), e.g., EHLO localhost
- The victim then performs many SMTP deliveries to different domains/servers (direct-to-MX behavior).

The victim host (10.9.23.102) became infected and began communicating with a known malware loader infrastructure. ZUI/Suricata generated repeated “ET MALWARE SQUIRRELWAFFLE Loader Activity (POST)” alerts for HTTP POST beacons to 208.91.128.6 starting at 16:46Z, followed by “SQUIRRELWAFFLE Server Response” alerts indicating successful C2 responses. Shortly after, the host initiated multiple TLS sessions that matched an “ET JA3 Hash – Possible Dridex” fingerprint (including traffic to 185.125.204.174:8080 and 136.232.34.70:443), suggesting a second-stage bot/trojan component. The host then performed external IP discovery via api.ipify.org. Finally, SMTP anomalies and email-delivery behavior indicate the compromised machine was used to send malspam/phishing emails (often via direct SMTP to external servers), consistent with botnet/spambot activity.

Student Declaration of AI Tool use in this Assessment

Please indicate your level of usage of generative AI for this assessment - please tick the appropriate category(s).

If the “Assisted Work” or “Partnered Work” category is selected, please expand on the usage and in which elements of the assignment the usage refers to.

Solo Work	S1 - Generative AI tools have not been used for this assessment.	<input type="checkbox"/>
Assisted Work	A1 – Idea Generation and Problem Exploration Used to generate project ideas, explore different approaches to solving a problem, or suggest features for software or systems. Students must critically assess AI-generated suggestions and ensure their own intellectual contributions are central.	<input type="checkbox"/>
	A2 - Planning & Structuring Projects AI may help outline the structure of reports, documentation and projects. The final structure and implementation must be the student’s own work.	<input type="checkbox"/>
	A3 – Code Architecture AI tools maybe used to help outline code architecture (e.g. suggesting class hierarchies or module breakdowns). The final code structure must be the student’s own work.	<input type="checkbox"/>
	A4 – Research Assistance Used to locate and summarise relevant articles, academic papers, technical documentation, or online resources (e.g. Stack Overflow, GitHub discussions. The interpretation and	<input type="checkbox"/>

	integration of research into the assignment remain the student's responsibility.	
	A5 - Language Refinement Used to check grammar, refine language, improve sentence structure in documentation not code. AI should be used only to provide suggestions for improvement. Students must ensure that the documentation accurately reflects the code and is technically correct.	<input type="checkbox"/>
	A6 – Code Review AI tools can be used to check comments within the code and to suggest improvements to code readability, structure or syntax. AI should be used only to provide suggestions for improvement. Students must ensure that the code accurately reflects their knowledge and is technically correct.	<input type="checkbox"/>
	A7 - Code Generation for Learning Purposes Used to generate example code snippets to understand syntax, explore alternative implementations, or learn new programming paradigms. Students must not submit AI-generated code as their own and must be able to explain how it works.	<input type="checkbox"/>
	A8 - Technical Guidance & Debugging Support AI tools can be used to explain algorithms, programming concepts, or debugging strategies. Students may also help interpret error messages or suggest possible fixes. However, students must write, test, and debug their own code independently and understand all solutions submitted.	<input type="checkbox"/>
	A9 - Testing and Validation Support AI may assist in generating test cases, validating outputs, or suggesting edge cases for software testing. Students are	<input type="checkbox"/>

	responsible for designing comprehensive test plans and interpreting test results.	
	A10 - Data Analysis and Visualization Guidance AI tools can help suggest ways to analyse datasets or visualize results (e.g. recommending chart types or statistical methods). Students must perform the analysis themselves and understand the implications of the results.	<input type="checkbox"/>
	A11 - Other uses not listed above Please specify:	<input type="checkbox"/>
Partnered Work	P1 - Generative AI tool usage has been used integrally for this assessment Students can adopt approaches that are compliant with instructions in the assessment brief. Please Specify: <ul style="list-style-type: none"> - The analysis of the pcap file to verify the answer of founding - README.md - report drafting and improvement - generate suggested expression in Wireshark 	<input checked="" type="checkbox"/>

Please provide details of AI usage and which elements of the coursework this relates to:

improve the report, pcap analysis

I understand that the ownership and responsibility for the academic integrity of this submitted assessment falls with me, the student.	<input checked="" type="checkbox"/>
I confirm that all details provide above are an accurate description of how AI was used for this assessment.	<input checked="" type="checkbox"/>