Jake Pole – 10761617

Section 1 - Introduction

This report contains a detailed analysis using Wireshark to inspect a network packet capture (PCAP) file which shows internet traffic from the 24th of September 2021. The analysis follows the traffic from its first identifiable network action to the final stage of exploitation.

Focusing on the following three key questions:

- Which system in the network has been compromised?
- What method was used to infect the system?
- What type of malware or attack caused the infection?

Section 2 - Methodology

The primary tool used for this investigation has been **Wireshark**. Wireshark is a network packet analyser[1]. This tool has allowed for deep investigation of the PCAP file, this has involved applying various display filters including Time, HTTP, DNS, and SMTP. Additionally using features such as Follow TCP Stream to reconstruct payloads. Additionally, VirusTotal[2] has been used to reference IP Addresses identified during the analysis to confirm whether the IP was maliciously used at the time the PCAP file was generated.

Section 3 – Analysis and Results

The first malicious activity which was observed in the PCAP file was a HTTP connection from **10.9.23.102 with MAC address 00:08:02:1c:47:ae** (an internal device), this was done through Wireshark by applying a filter to only show HTTP traffic. This occurred on the **24th of September at 16:44:38 UTC** which is demonstrated in figure 1. As shown in figure 1, the connection made is a HTTP GET request for a file named "**documents.zip**".

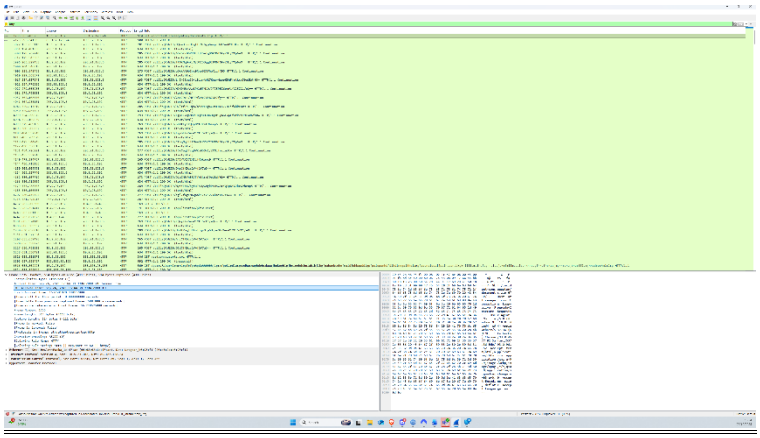

Figure 1 - Q1 & Q2

[1] Wireshark. *Chapter 1*. [Online]. Wireshark. Available at: https://www.wireshark.org/docs/wsug_html_chunked/ChapterIntroduction.html [Accessed 22 October 2025].

[2] VirusTotal. *VirusTotal*. [Online]. Available: https://www.virustotal.com/ [Accessed: 22 October 2025].

After further analysis of the HTTP GET packet, it is observed that the domain hosting the file was "**attirenepal.com**" as shown in Figure 2.

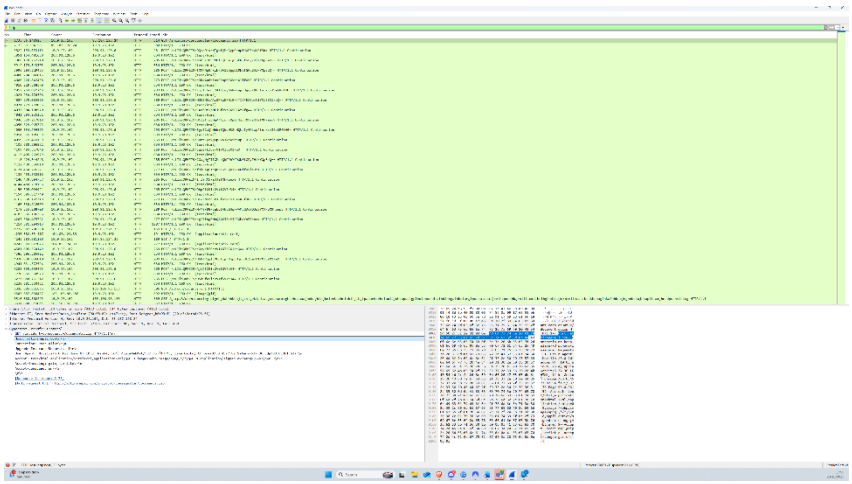

Figure 2 - Q3.

This demonstrates the method, which was used to infect the system, after analysing the HTTP stream, the "**documents.zip**" file is shown to contain a singular file called "**chart-1530076591.xls**" as seen in Figure 3 b using the "Follow TCP Stream" feature. An excel file containing excel macros which can execute scripts. After analysing the server header from "**attirenepal.com**", it is shown that the web server software is **LiteSpeed**, which is powered by **PHP/7.2.34** this can be seen in figure 4.

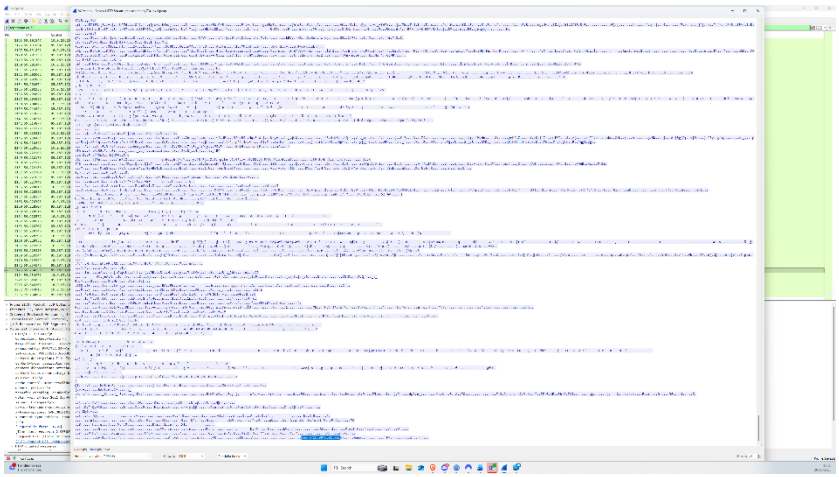

Figure 3 - Q4.

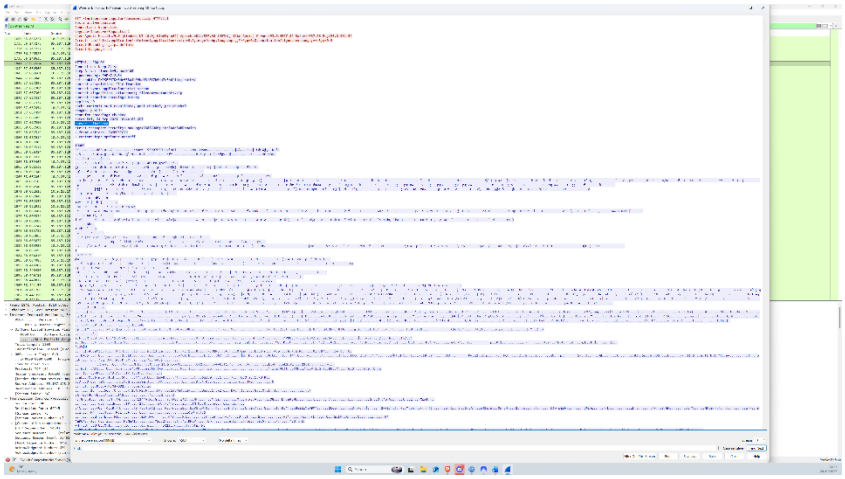

Figure 4 - Q5 and Q6

Following the initial download, phase 2 of the infection begins, by filtering the HTTPS traffic which occurred between 16:45:11 and 16:45:30, DNS queries and connections are spotted for three suspected malicious domains (**finejewels.com.au, thietbiagt.com, new.americold.com**) shown in Figures 5 & 6.

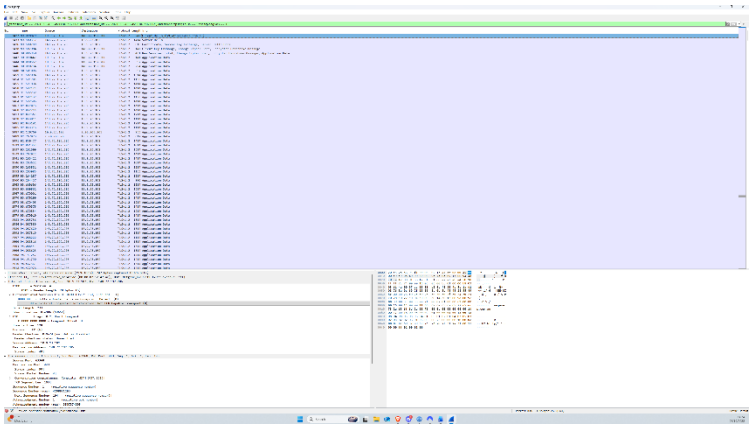

Figure 5 - Q7.

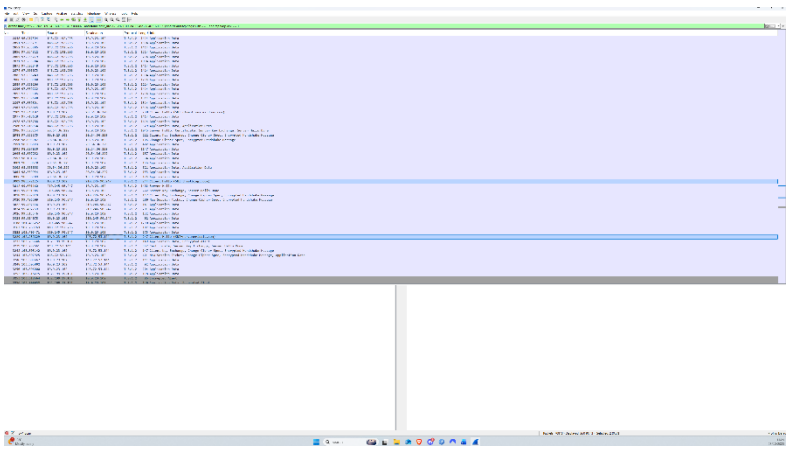

Figure 6 - Q7.

Further analysis of the TCP Stream with **finejewels.com.au** shows that the Certificate Authority who issued the SSL certificate was **GoDaddy** as shown in Figure 7.

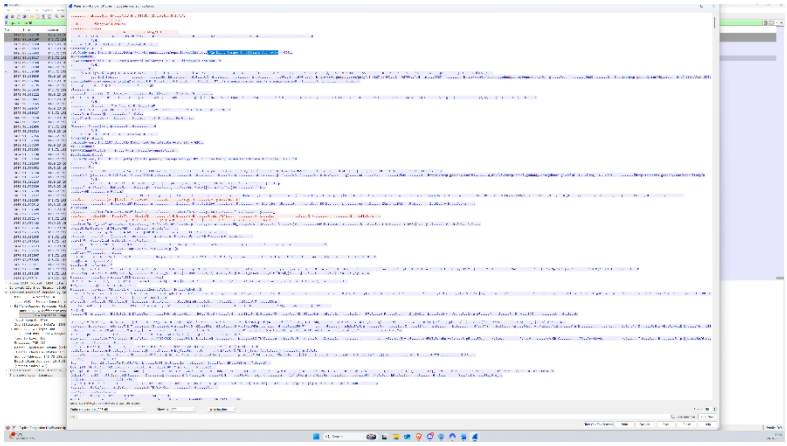

Figure 7 - Q8.

By accessing the Conversations menu within Wireshark, two IP addresses can be seen which have a high volume of packets, which can highlight the two C2 servers (**185.106.96.158 and 185.125.204.174)** in figure 8. After inspecting the first IP address and looking at the Host header, a fake host header is being used of "**ocsp.verisign.com**" (Figure 9), making the C2 traffic appear more legitimate. This is known as "domain fronting".

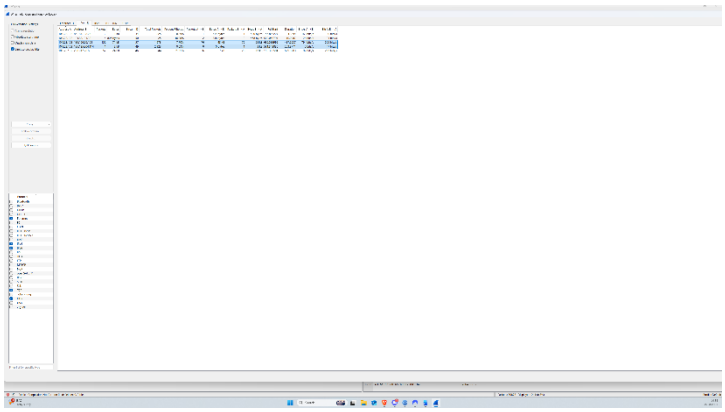

Figure 8 - Q9.

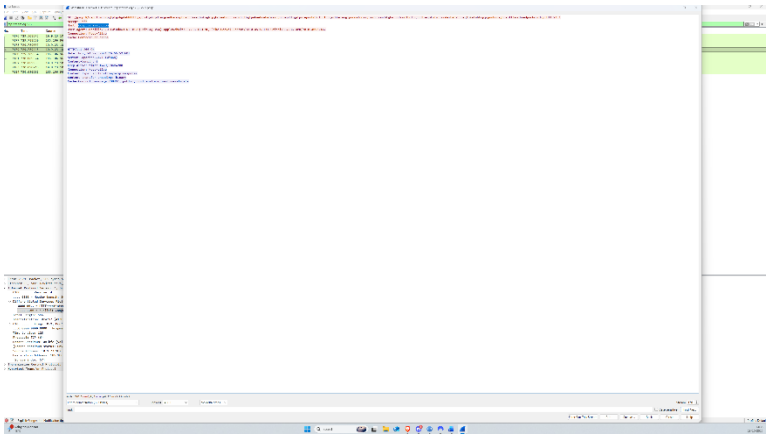

Figure 9 - Q10.

After taking the two highlights C2 servers, a Virus Total search shows that at the time of the attack, 185.106.96.158 was "**survmeter.live**" domain (see figure 10). With a similar method being done for 185.125.204.174 , we can see in virus total that this C2 server is s**ecuritybusinpuff.com** (Figure 11).

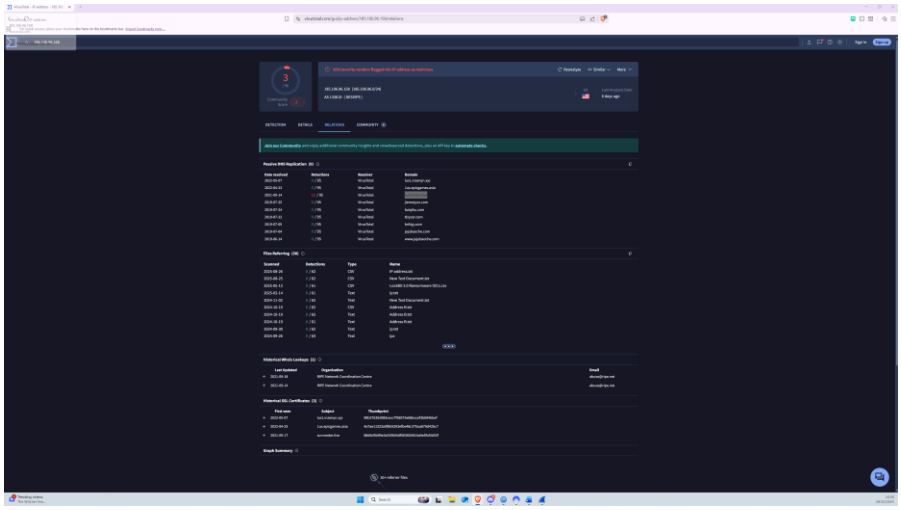

Figure 10 - Q11

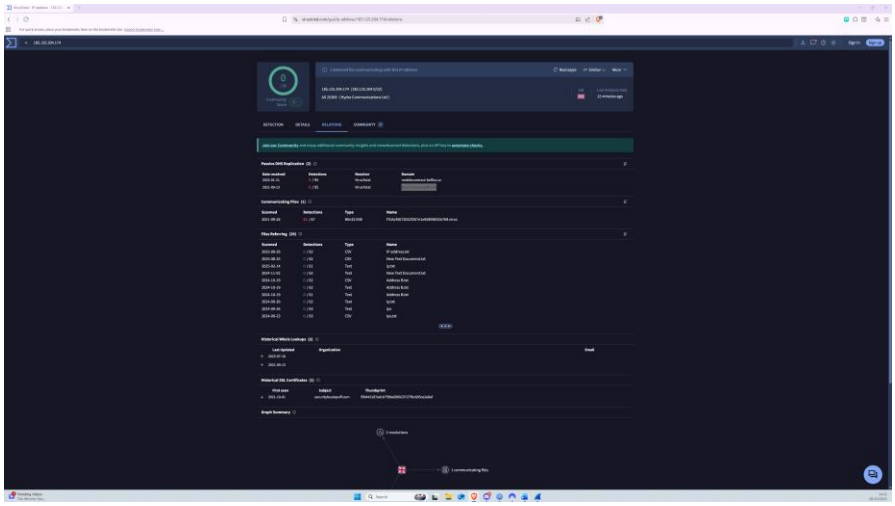

Figure 11 - Q12.

By filtering the traffic, HTTP POST requests, reveals a domain which is used for the post-infection traffic "**maldivehost.net**" (Figure 12). By following the HTTP stream for the POST traffic, the data sent by the compromised workstation begins with the following eleven characters "**LIisQRWZI9**" (see Figure 13),

analysing this shows that this is encoded C2 beacon data. The packet length for the first POST request is **281 bytes** (see Figure 14).

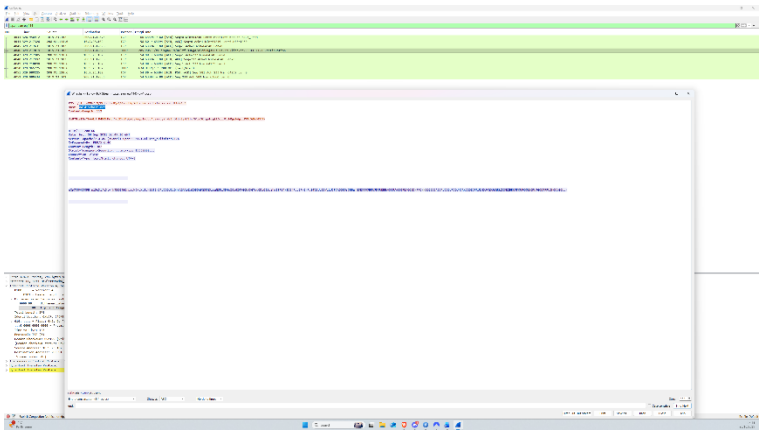

Figure 12 - Q13.

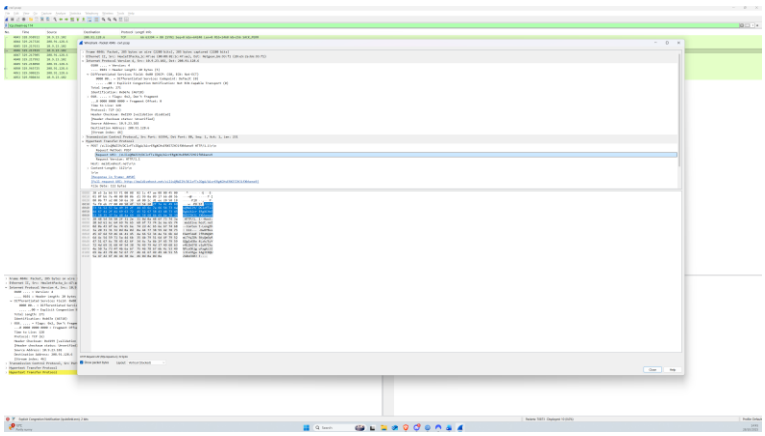

Figure 13 - Q14.

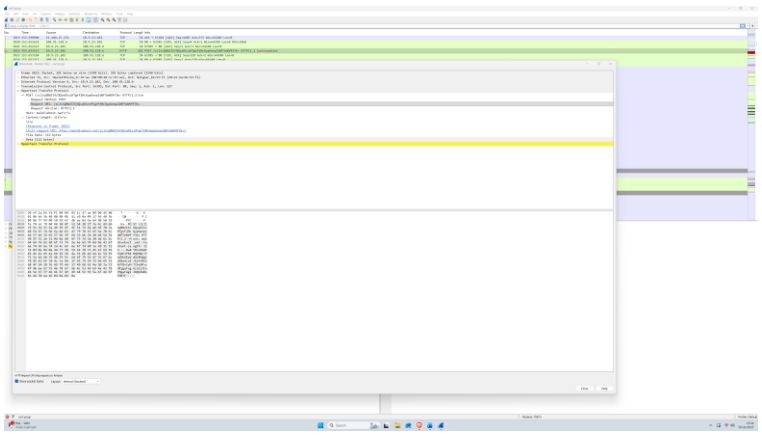

Figure 14 - Q15.

By analysing the Server header value from maldivehost.net is "**Apache/2.4.49 (cPanel) OpenSSL/1.1.11 mod_bwlimited/1.4**" as shown in Figure 15.

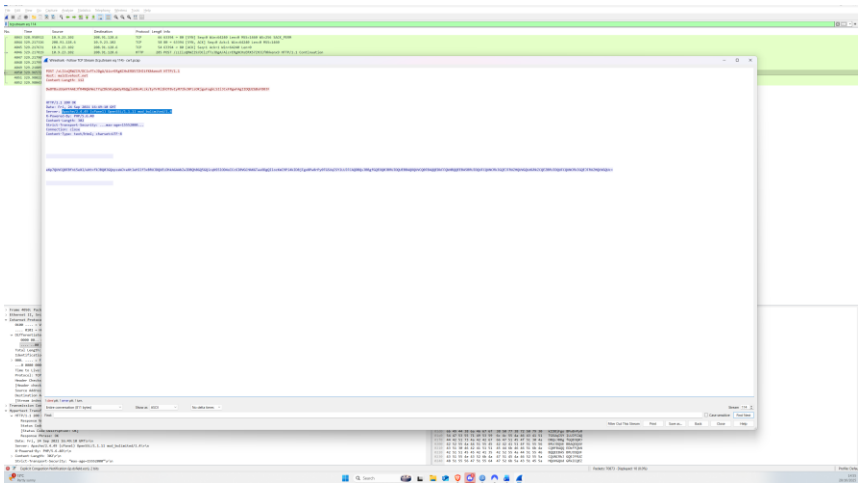

Figure 15 - Q16.

Analysing the DNS queries within the PCAP file, the malware performs a query to check the public IP, which occurred on the **24th of September 2021 at 17:00:04** (Figure 16). And the domain queried was "**api.ipify.org**" as shown in Figure 17.

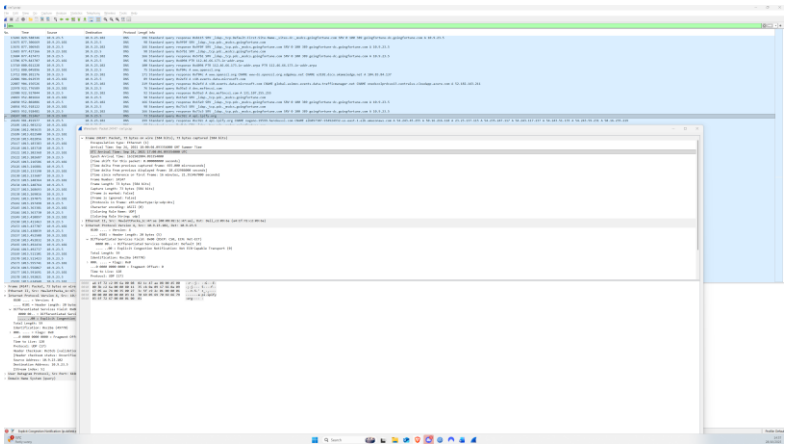

Figure 16 - Q17.

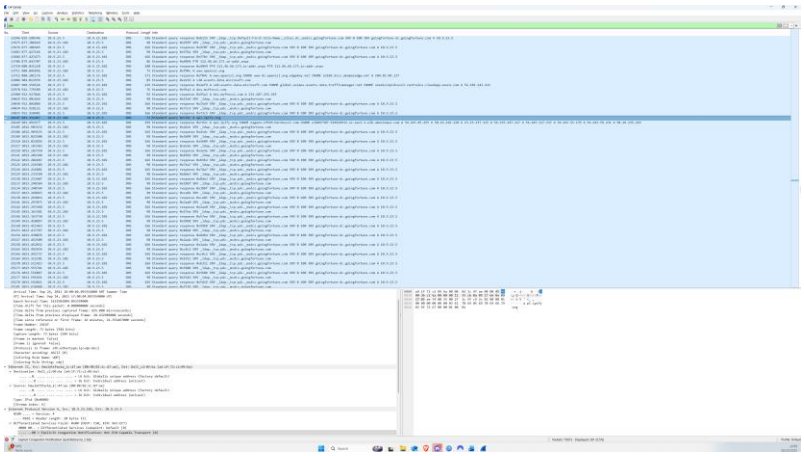

Figure 17 - Q18.

Following this, a large number of SMTP traffic is then observed, with the first MAIL FROM being "**farshin@mailfa.com**" (Figure 18). And then the SMTP stream reveals that the password used for ho3ein.sharifi@mailfa.com is MTM2OTEzNjk=. Using this we can base64 decode[3] this to reveal the plaintext password is "**13691369**" (Figure 19) revealing the credentials used for a spam campaign.

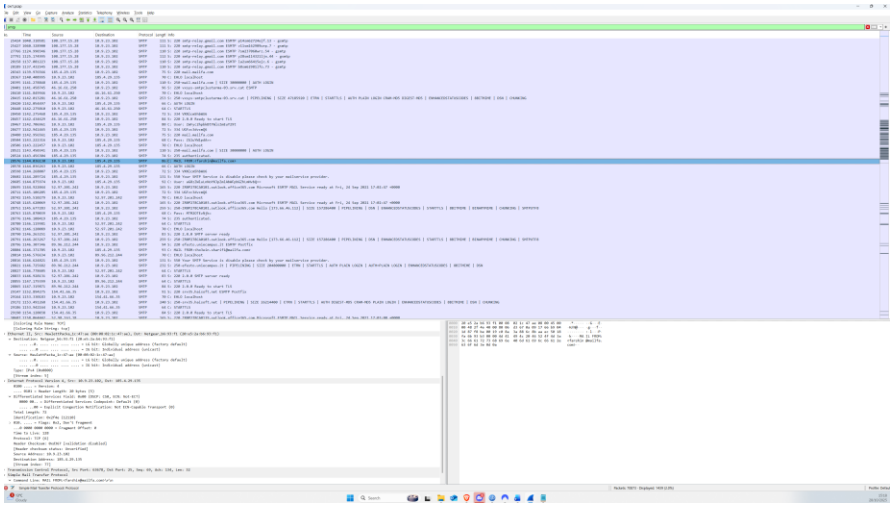

Figure 18 - Q19.

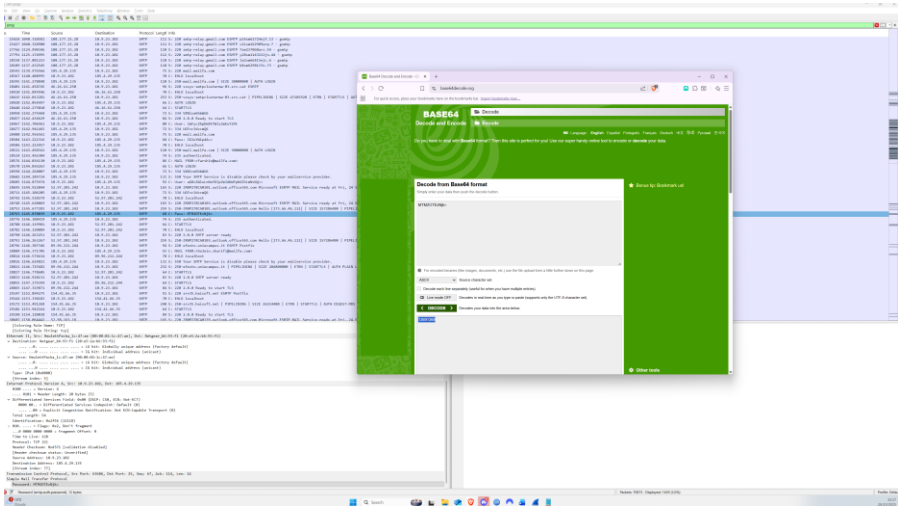

Figure 19 - Q20.

Section 4 - Conclusion

Based on this report, the compromised system is the host which uses the internal IP address "**10.9.23.102**". This is the source of all malicious downloads and the Command and Control (C2) communications. The infection was a multi-stage attack, initiating at **16:44:38** when the host downloaded **documents.zip** from the malicious domain "**attirenepal.com**". This contained a file called "**chart-1530076591.xls**" which is an excel file containing macros which furthered the attack. The attack can be defined as an advanced Cobalt Strike C2 (Command and Control). This is clarified using two different C2 server with associated domains discovered through use of Virus Total, domain fronting to

[3] Base64Decode. *Base64Decode*. [Online]. Available: https://www.base64decode.org/ [Accessed: 22 October 2025].

make the traffic appear more legitimate, HTTP POST communicating with a C2 domain, and additional post-infection reconnaissance and further malicious spam.

To prevent this incident from occurring again in the future, I would recommend implementing several processes. This would include email filtering to block emails from unknown sources and blocking attachments from unknown sources. As well as blocking Office macros by default as Office macros are commonly a method of compromising a device, and if used within an organization, a trusted member of an IT Security team can allow the user to use Office macros.

Link to Video Demonstration - https://youtu.be/O7TAI9Pr7TI

Link to GitHub Repository (Also containing all figures)- https://github.com/JakeP2400/COMP3010-Security-Operations-Incident-Management

Jake Pole – 10761617

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.**	☐
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.	☐
	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.	☐
	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.	☐
	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 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.	☐
	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.	☐
	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.	☐
	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.	☐

	A9 - Testing and Validation Support AI may assist in generating test cases, validating outputs, or suggesting edge cases for software testing. Students are 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.	☐
	A11 - Other uses not listed above Please specify:	☐
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: • Language Refinement during write-up • Markdown File Generation • Report Analysis and Improvements Suggestion	☒

Please provide details of AI usage and which elements of the coursework this relates to:
Report Analysis and Improvements, Markdown File Generation for GitHub

I understand that the ownership and responsibility for the academic integrity of this submitted assessment falls with me, the student.	☒

Jake Pole – 10761617

I confirm that all details provide above are an accurate description of how AI was used for this assessment.	☒