COMP 3010 CW1

1) Introduction

The report presents a comprehensive intrusion analysis of a social engineering attack observed in a captured PCAP file, in which a victim downloaded a malicious ZIP file disguised as a legitimate document. As a security analyst, the aim is to analyze the breach by interpreting intrusion detection alerts, identifying the threat, and explaining how the attacker compromised the system. The report is structured as follows: Section 2 explains the investigation methodology, Section 3 presents the findings with supporting packet evidence, and Section 4 gives recommendations for prevention.

2) Methodology

The investigation was carried out in Wireshark with the following steps taken to identify the infected system information, an indicator of the infection, and how the system got infected:

2a) Infected System Information

IP Address and MAC Address:

- The analysis began by isolating HTTP traffic in Wireshark to find suspicious downloads and identify the source IP.

http

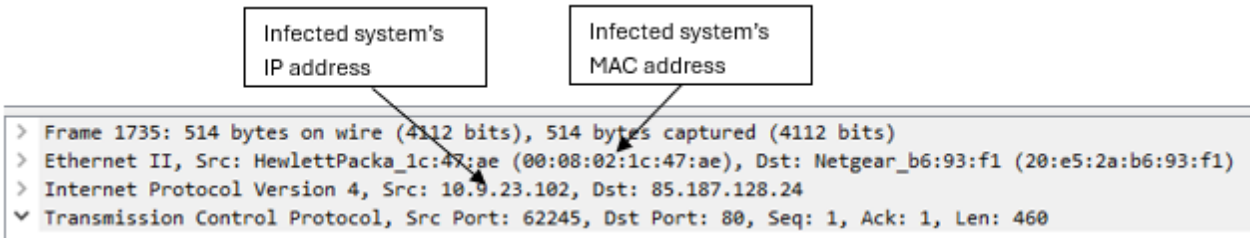
No.	Time	Source	Destination	Protocol	Lengtl	Info
1735	56.248525	10.9.23.102	85.187.128.24	HTTP	514	GET /incidunt-consequatur/documents.zip HTTP/1.1
2173	59.234150	85.187.128.24	10.9.23.102	HTTP	580	HTTP/1.1 200 OK
3822	153.653113	10.9.23.102	208.91.128.6	HTTP	281	POST /zLIisQRWZI9/OQsaDixzHTgtfjMcGypGenpldWF5eWV9f3k= HTTP/1.1 Continuation
3851	154.401688	208.91.128.6	10.9.23.102	HTTP	634	HTTP/1.1 200 OK (text/html)
3908	178.767210	10.9.23.102	208.91.128.6	HTTP	285	POST /zLIisQRWZI9/ASk5Kx0SPR8lJjE5eTg9GkN6fGFyZHl/YXp6eQ== HTTP/1.1 Continuation
3912	179.543303	208.91.128.6	10.9.23.102	HTTP	634	HTTP/1.1 200 OK (text/html)
3996	203.829455	10.9.23.102	208.91.128.6	HTTP	285	POST /zLIisQRWZI9/fXMKNg0nKzN/DA15DggBI0N6fGFyZHl/YXp6eQ== HTTP/1.1 Continuation
4000	204.546015	208.91.128.6	10.9.23.102	HTTP	634	HTTP/1.1 200 OK (text/html)
4006	228.842458	10.9.23.102	208.91.128.6	HTTP	273	POST /zLIisQRWZI9/eDkkAA0bInx9Rnp6ZXVheXllfX95 HTTP/1.1 Continuation
4010	229.568579	208.91.128.6	10.9.23.102	HTTP	634	HTTP/1.1 200 OK (text/html)
4017	254.037243	10.9.23.102	208.91.128.6	HTTP	293	POST /zLIisQRWZI9/LjI+JSoqJQ4lBiwyAhR7KngvHgopKBhFfntkcmJ9eGR6fH0= HTTP/1.1 Continuation
4021	254.776306	208.91.128.6	10.9.23.102	HTTP	634	HTTP/1.1 200 OK (text/html)
4027	279.063986	10.9.23.102	208.91.128.6	HTTP	289	POST /zLIisQRWZI9/HDN9NScAAw8PKwEFMi0/JTI5PEZ6emV1YXl5ZX1/eQ== HTTP/1.1 Continuation

- The first malicious HTTP GET request was isolated and inspected at packet level.

http

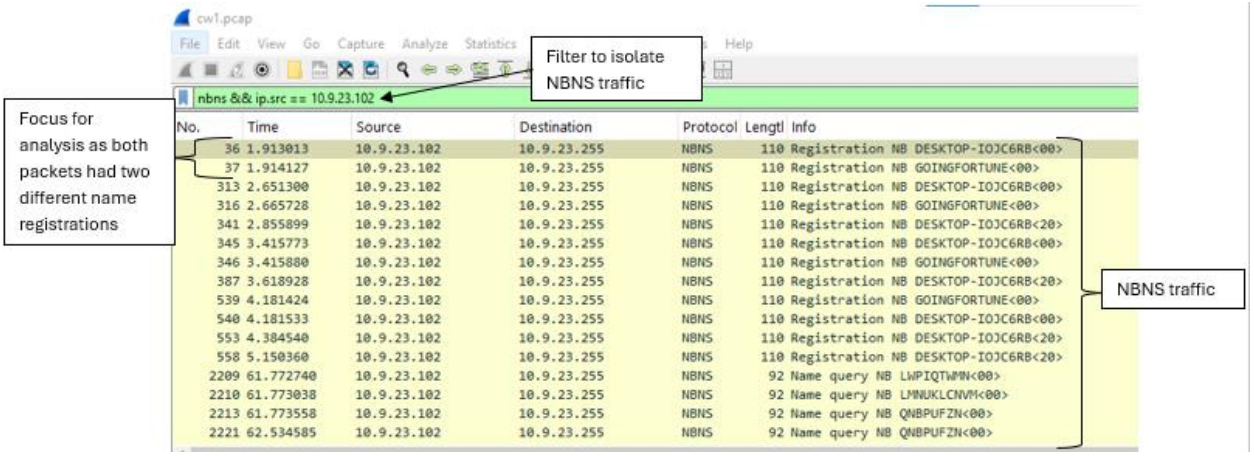
No.	Time	Source	Destination	Protocol	Lengtl	Info
1735	56.248525	10.9.23.102	85.187.128.24	HTTP	514	GET /incidunt-consequatur/documents.zip HTTP/1.1

- Expansion of packet 1735's Ethernet II and IPv4 headers revealed the internal system's IP and MAC address. This establishes which host initiated the download and will be used to tie all subsequent activity back to that same machine.

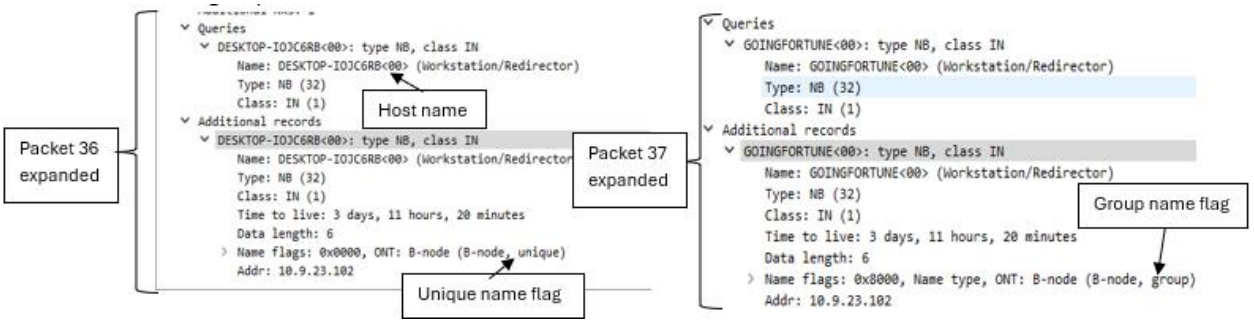


Host Name:

- Next, NBNS traffic was filtered as NetBIOS Name Service packets contain name registration data that exposes a Windows system's hostname.

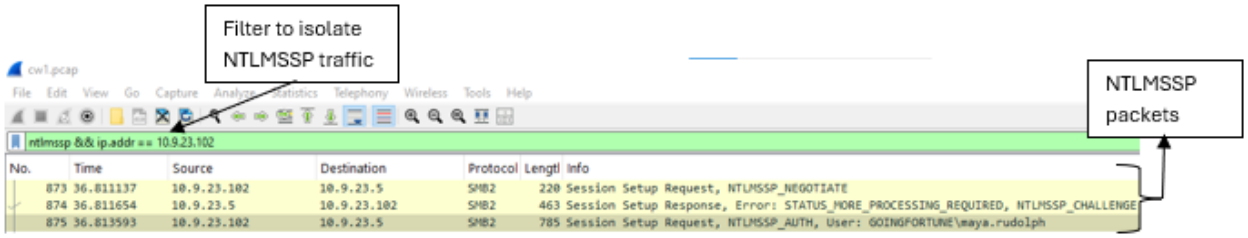


- By reviewing those packets, the workstation name of the host was recovered and distinguished from any broadcast group/domain names.

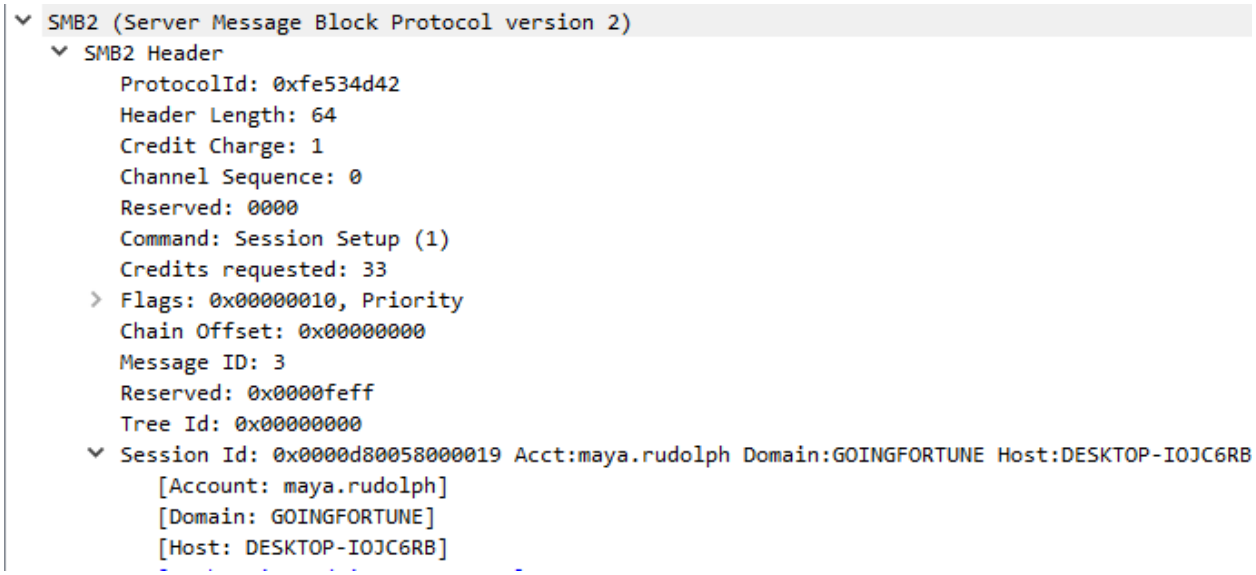


User Account Name:

- To identify the user account name, NTLMSSP traffic was inspected because it displays packets that often carry user authentication data such as usernames and domains (Blin n.d.).

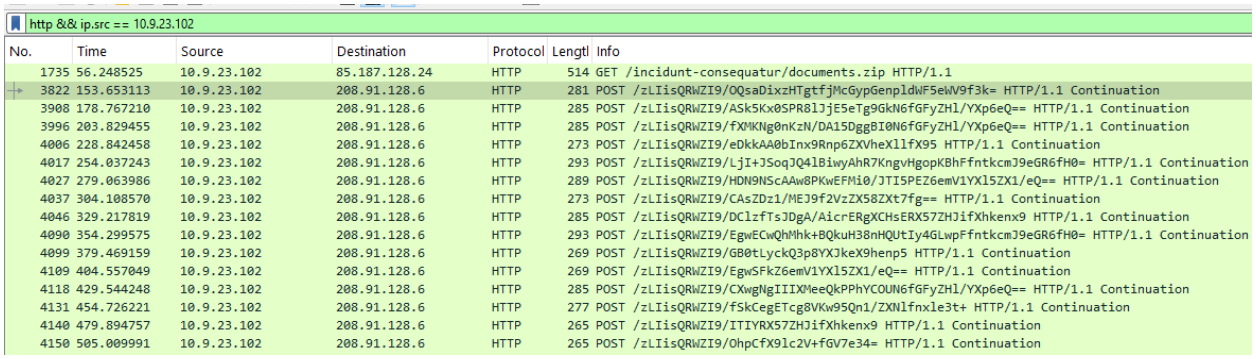


- By examining the authentication messages, the user account name of the infected machine was obtained (Click packet 875 > SMB2 > SMB2 Header > Session Id). This confirms that the activity was tied to an interactive user account



2b) Indicator of Infection

- To find an IoI, http traffic from the victim's IP was analyzed. This revealed multiple HTTP POST requests to an external IP address containing randomized data.



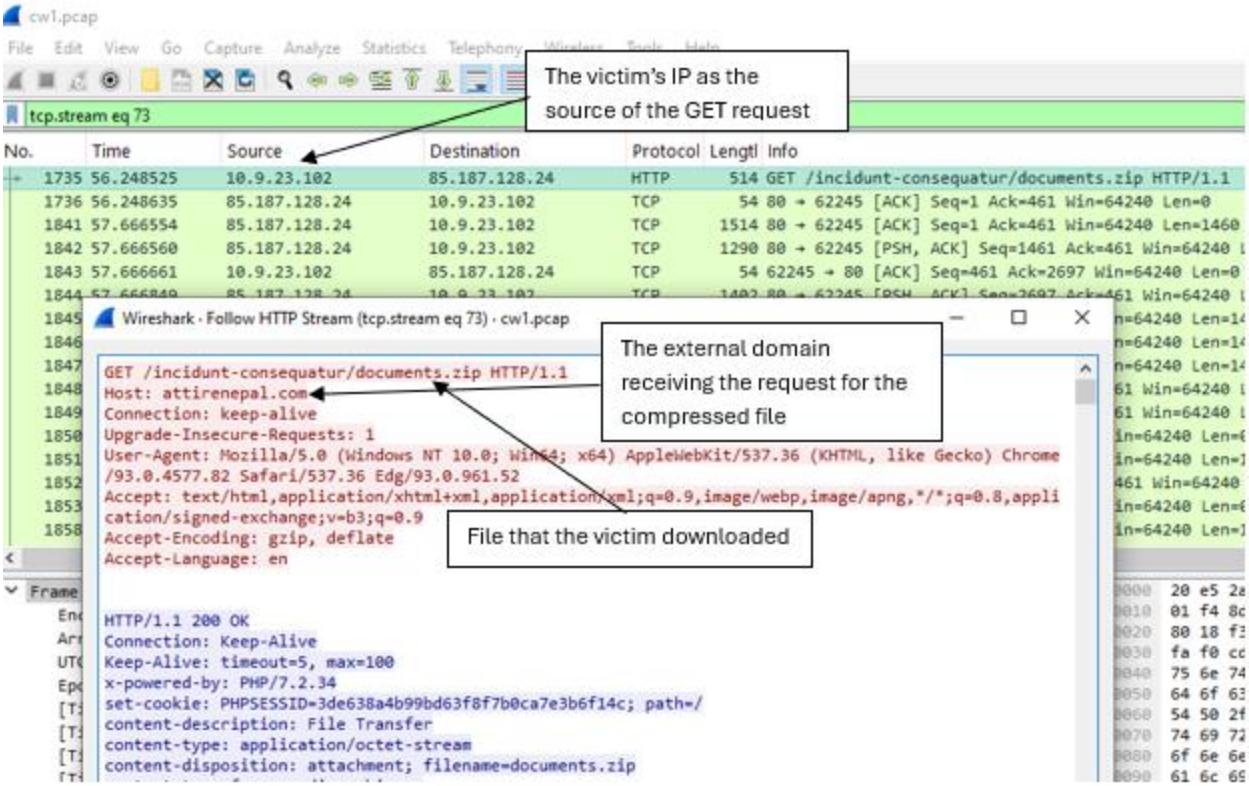
- These POST requests were reviewed and I discovered they all occurred repeatedly in short intervals as seen in the arrival times of the first 3 packets with POST requests below.

```
Frame 3822: 281 bytes on wire (2248 bits), 281 bytes captured (2248 bits)
    Encapsulation type: Ethernet (1)
    Arrival Time: Sep 24, 2021 17:46:16.395000000 W. Central Africa Standard Time
    UTC Arrival Time: Sep 24, 2021 16:46:16.395000000 UTC
Frame 3908: 285 bytes on wire (2280 bits), 285 bytes captured (2280 bits)
    Encapsulation type: Ethernet (1)
    Arrival Time: Sep 24, 2021 17:46:41.509097000 W. Central Africa Standard Time
    UTC Arrival Time: Sep 24, 2021 16:46:41.509097000 UTC
Frame 3996: 285 bytes on wire (2280 bits), 285 bytes captured (2280 bits)
    Encapsulation type: Ethernet (1)
    Arrival Time: Sep 24, 2021 17:47:06.571342000 W. Central Africa Standard Time
    UTC Arrival Time: Sep 24, 2021 16:47:06.571342000 UTC
```

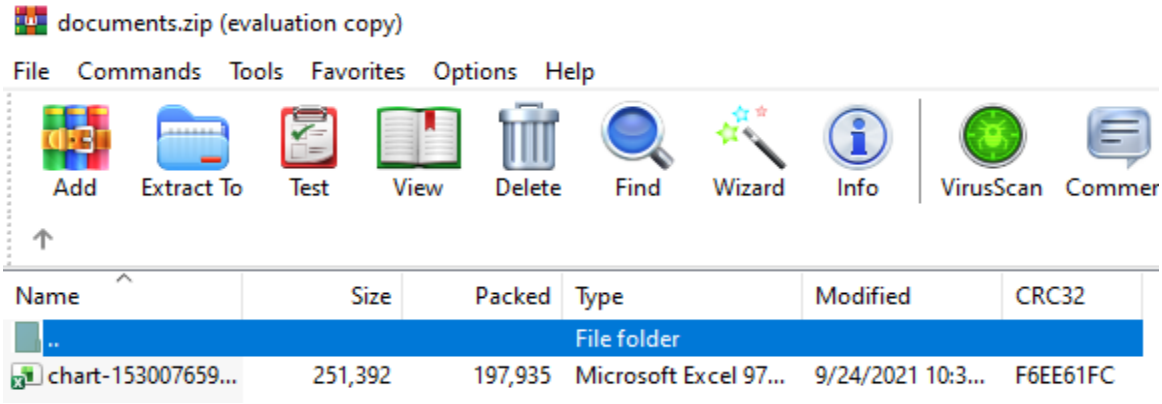
- This encoded communication and repetition pattern indicates malware command-and-control activity, confirming the system is infected.

2c) How the system got infected

- Following the HTTP stream of packet 1735, where the initial malicious HTTP connection occurred, the HTTP response headers confirmed that a ZIP archive was downloaded.



- The archive contained a macro-enabled Excel file. Opening this file triggered malicious code on the victim host and led to outbound command-and-control traffic to attacker infrastructure.

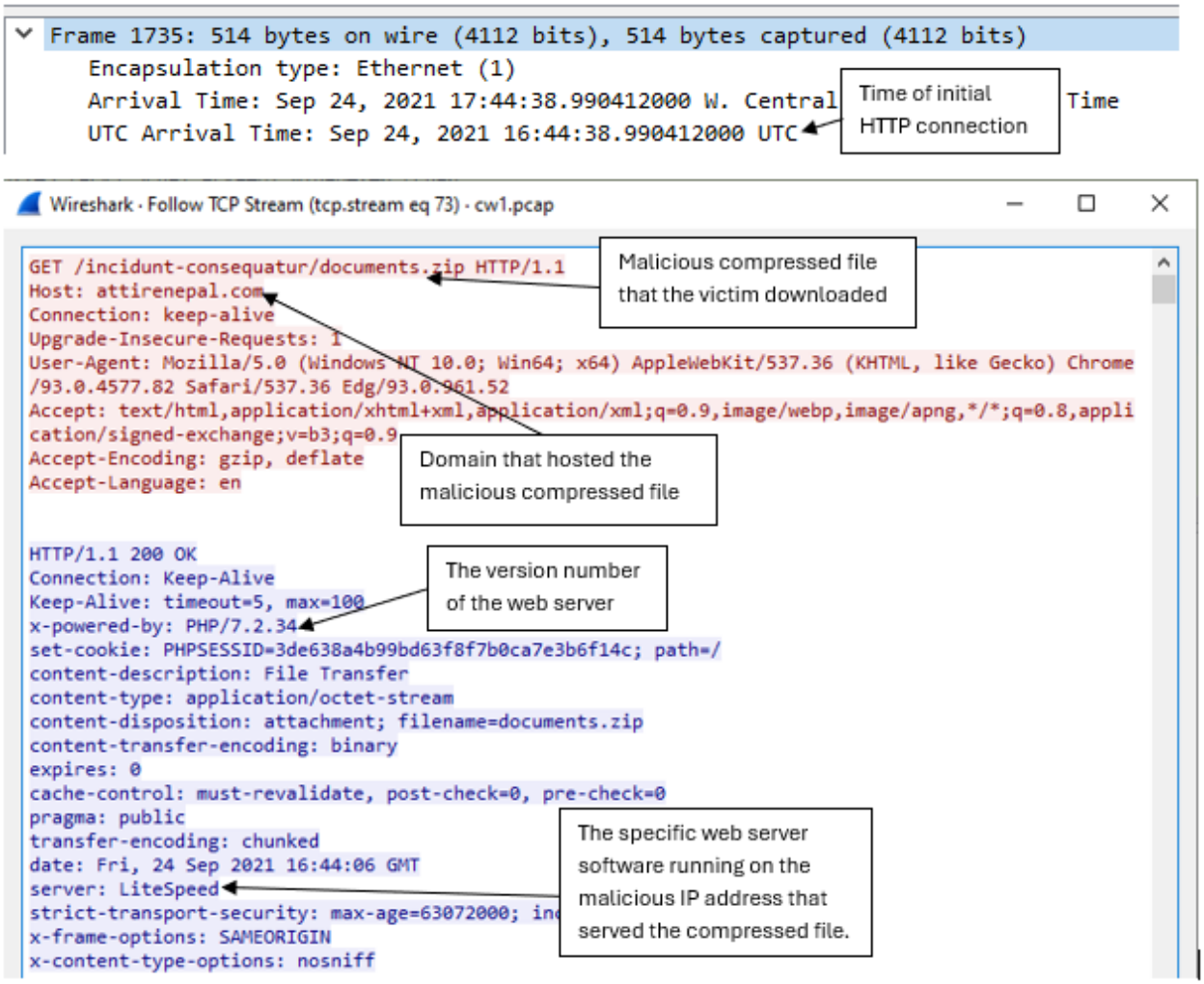


3) Results

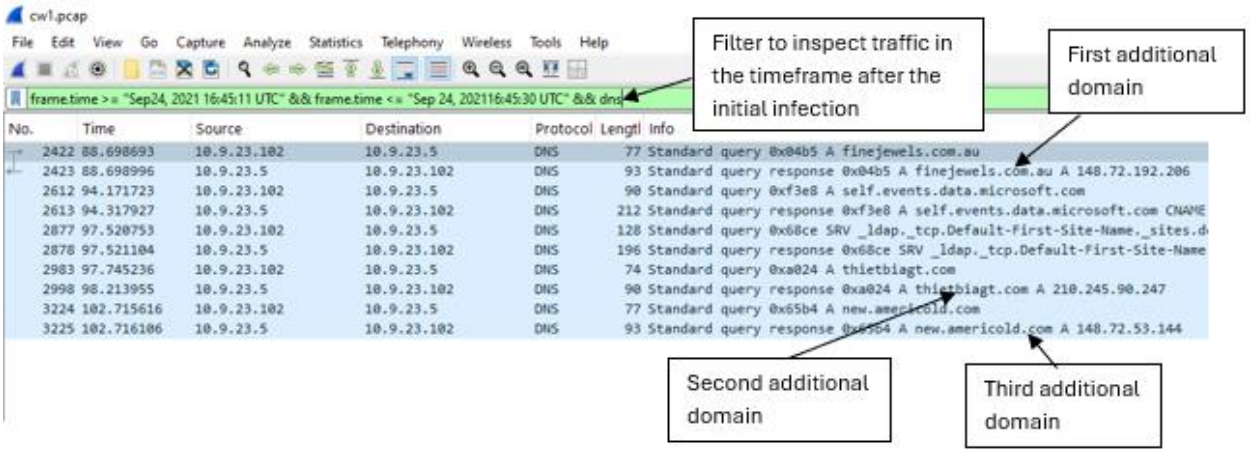
This section presents findings from the analysis and supporting evidence that explain the quiz answers:

3a) Initial Infection & File Transfer

The victim host made an HTTP GET request for documents.zip from attirenepal.com at 2021-09-24 16:44:38 UTC. A ZIP file containing chart-1530076591.xls was sent with the response. LiteSpeed was recognized with PHP/7.2.34 in the Server header, suggesting that a PHP-based shared server was probably compromised. The downloaded Excel file was opened by the user, and its macro started post-exploitation actions and became more persistent by establishing external connections.

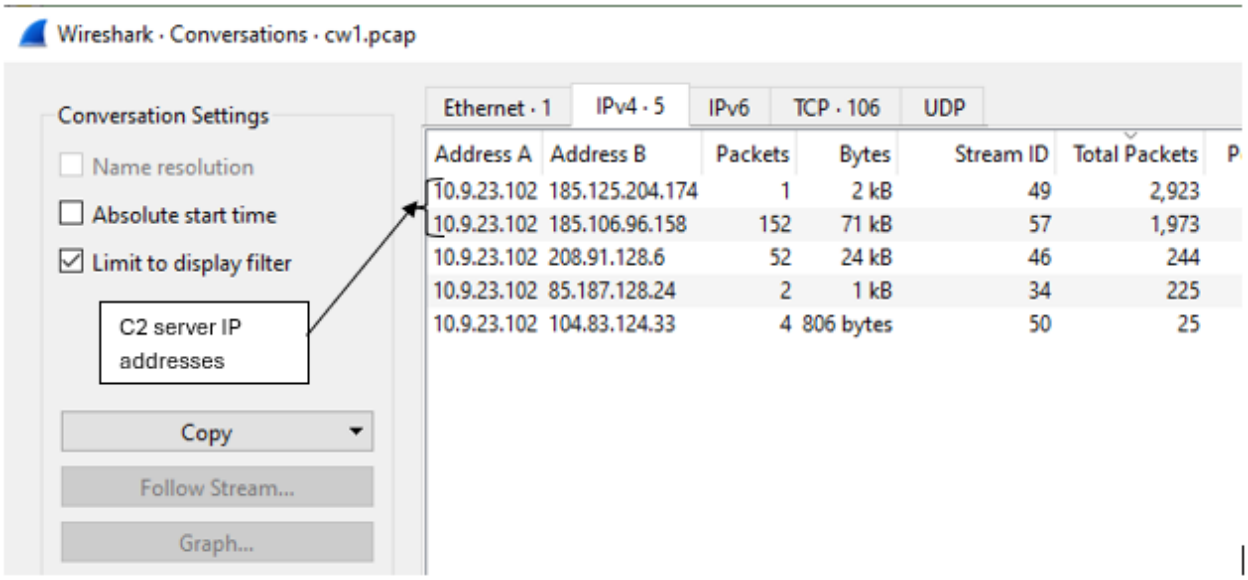


After initial infection, inspection of subsequent DNS traffic within the 16:45:11 – 16:45:30 UTC range revealed three additional domains were involved in downloading malicious files to the victim host.

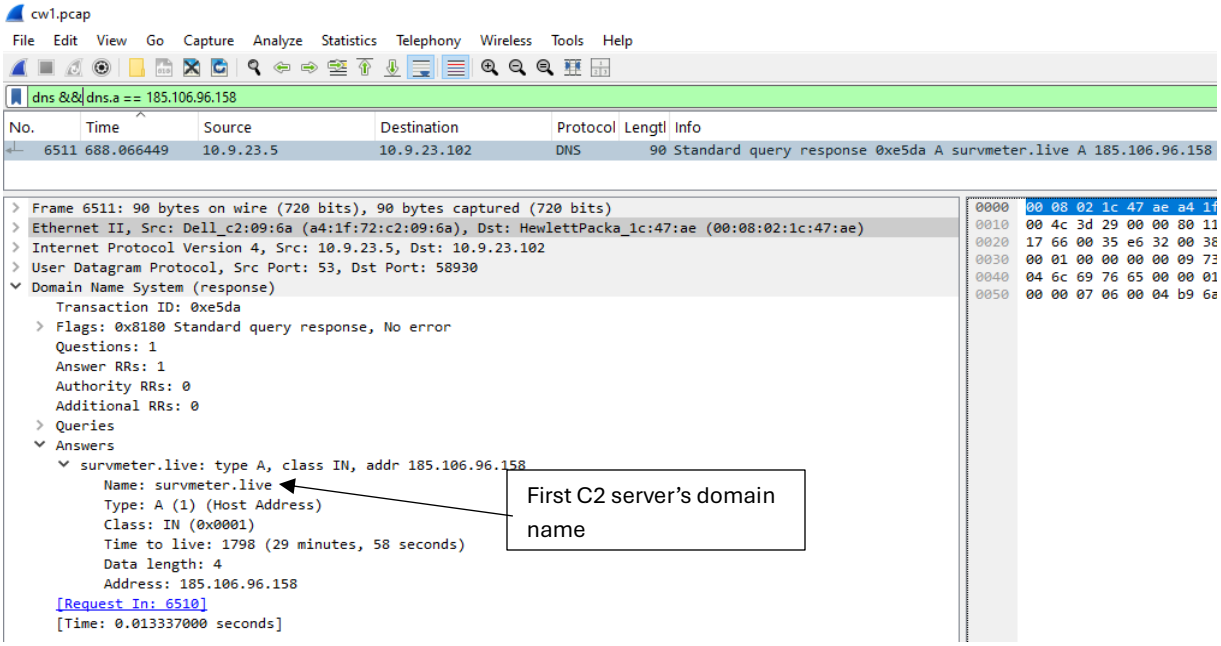


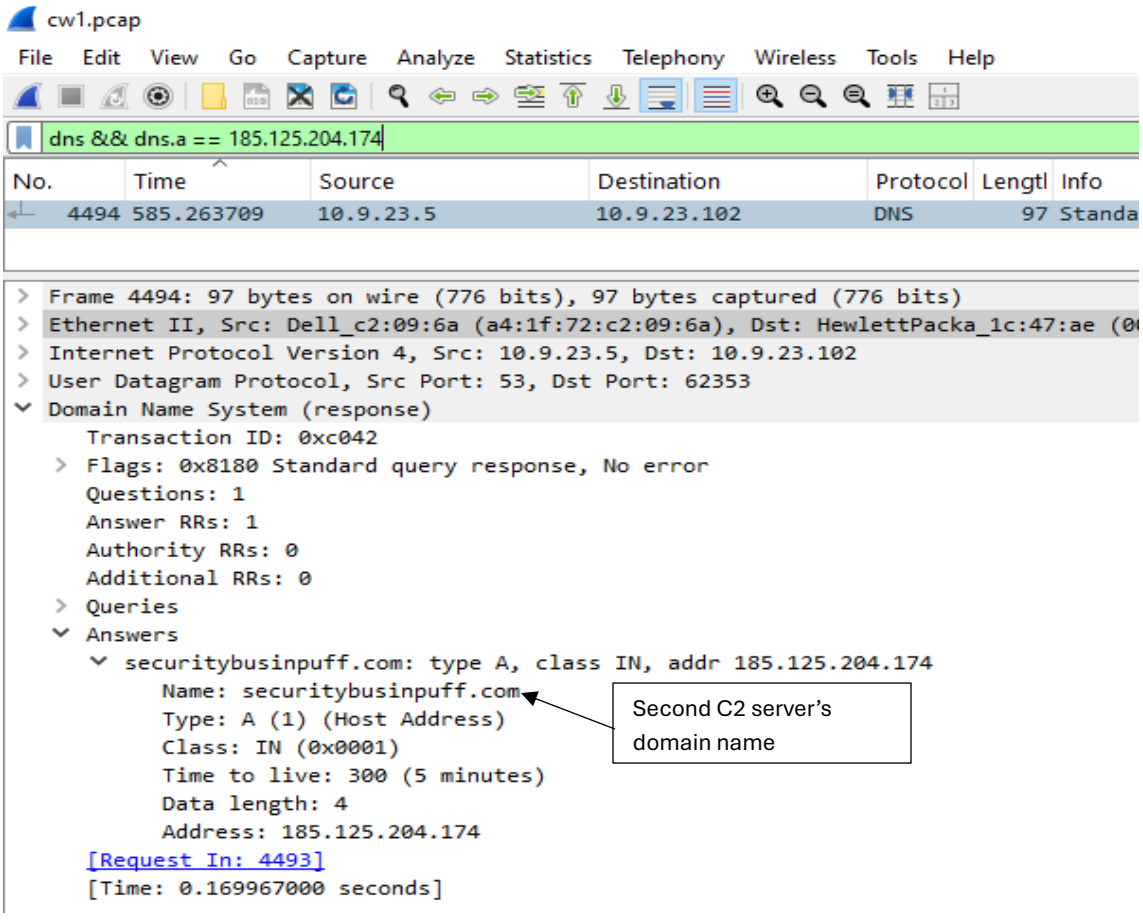
3b) Command & Control Activity

Post-execution, the host initiated encrypted sessions to 185.106.96.158 and 185.125.204.174, conversations analysis flagged these as top candidates by packet count/bytes. TLS certificate inspection of the first additional domain using its IP showed issuer GoDaddy as the CA.



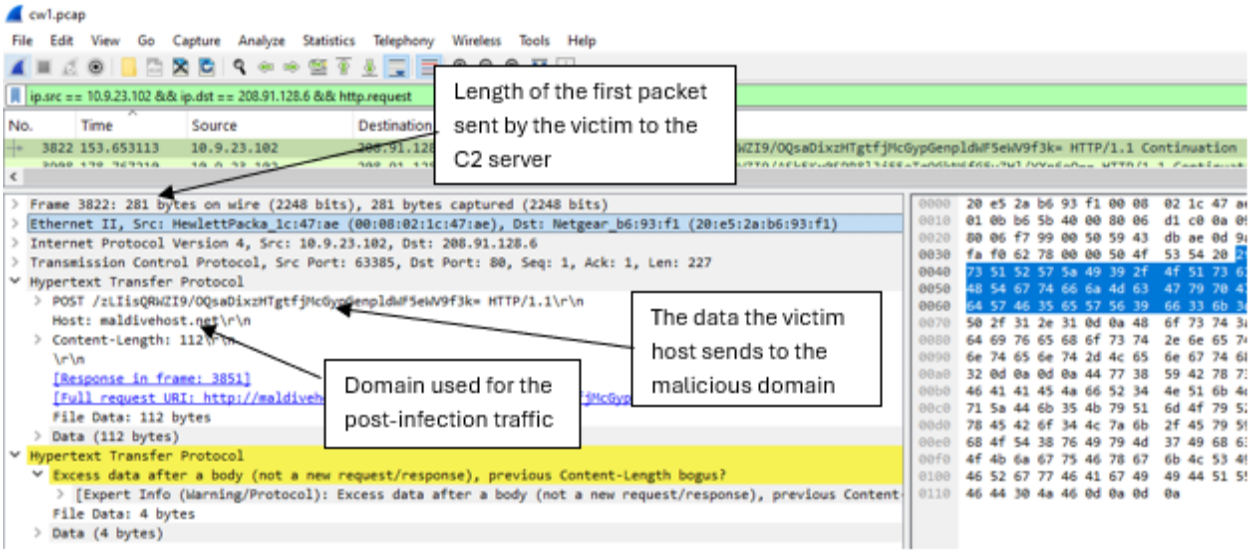
Host header/SNI evidence included ocsp.verisign.com, and DNS answers associated the C2 IPs with these domains: survmeter.live (185.106.96.158) and securitybusinpuff.com (185.125.204.174).





HTTP POST requests exhibited short, periodic payloads with total packet length around 281 bytes, consistent with Cobalt Strike beaconing (Rahman 2021). Lack of standard HTML

body supports the interpretation of automated beacon traffic rather than normal user browsing.

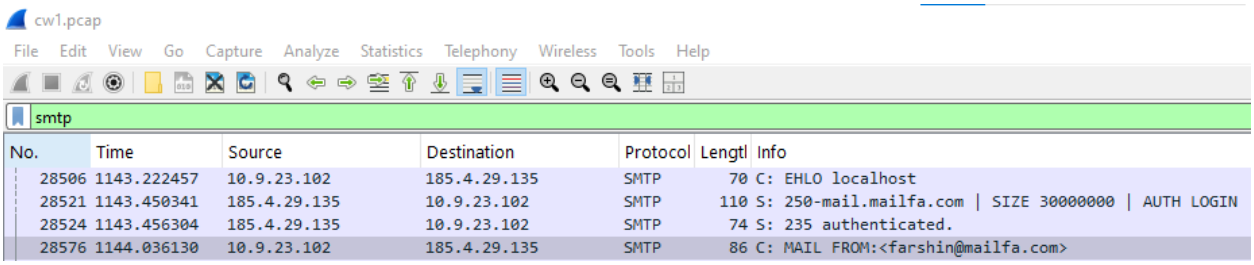


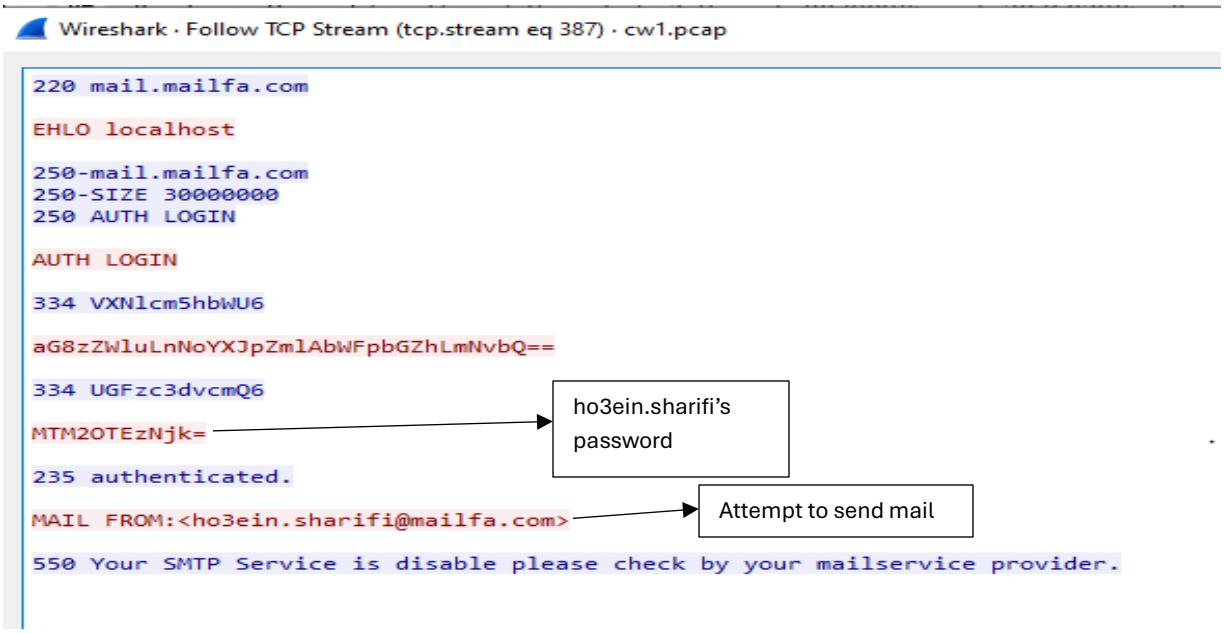
3c) Final Exfiltration

At 2021-09-24 17:00:04 UTC, the infected host queried api.ipify.org. This service is commonly used by malware to discover the victim's external/public IP address before it reports back to its operator (Ipify.org, 2014).

After this IP check, the host initiated SMTP traffic. The first SMTP transaction in the pcap shows MAIL FROM:<farshin@mailfa.com>. In a later SMTP session from the same host, the client performed AUTH LOGIN and supplied Base64-encoded credentials. Decoding those values reveals the password 13691369 for the user ho3ein.sharifi@mailfa.com. The server responded with 235 authenticated., and the client then attempted to send mail using MAIL FROM:<ho3ein.sharifi@mailfa.com>.

This demonstrates that the host is attempting to authenticate to an external mail service using exposed credentials and then send outbound email.





4) Conclusion

The captured traffic shows a successful compromise of an internal Windows host, followed by command-and-control communication and credentialed outbound activity. The attacker didn't just breach the victim's system but achieved post-exploitation control and attempted to move data out of the environment.

To reduce the impact of similar incidents, the organization should:

- Use network-based intrusion detection systems (NIDS) to detect odd patterns such as anomalous outbound traffic, executable macro-enabled attachments, and enable email gateway filtering in order to stop similar instances.
- Train its users on the dangers of opening unexpected attachments.
- Use Intrusion prevention systems (IPS) to actively stop malicious sessions or block traffic before it reaches important assets.
- Use EDR, XDR, and SIEM solutions to enhance visibility across hosts, networks, and cloud environments, supporting the integration of threat intelligence and automated response capabilities which is useful for the early detection of C2 activity.

Open challenges remain because most of the post infection traffic is encrypted or encoded and blends in with normal HTTPS and SMTP. Attackers rely on this more to evade traditional defense systems. Future work should integrate real-time threat intelligence feeds and sandboxing environments to improve early detection and response to similar threats.

5) References

- Blin, K. (n.d.) Implementing the NTLM Secure Service Provider for Wine.
- Rahman, A. (2021) 'Defining Cobalt Strike Components So You Can BEA-CONfident in Your Analysis.' *Mandiant*.
- Ipify.org. (2014). *ipify - A Simple Public IP Address API*. [online] Available at: https://www.ipify.org/.

6) Github

https://github.com/ukashley/COMP-3010-Security-Operations-Incident-Management-

7) Appendix

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.**	☐
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: I used AI for critical feedback and grading simulation.	☐

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: I used generative AI to assist with a small part of the investigative analysis, report structuring and summarization, research assistance and technical language improvement. It helped me critically assess my methodology, clarify grammar, and simulate grading feedback to improve the quality of my submission. All documentation, explanations, analysis of results and screenshots were performed independently as per assignment instructions.	☑

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

Generative AI was used to support the development of this coursework. It assisted with structuring the report, improving technical phrasing, and clarifying grammar. Specific elements where AI was used include the Methodology and Results sections, where it helped articulate packet analysis findings and streamline explanations.

I understand that the ownership and responsibility for the academic integrity of this submitted assessment falls with me, the student.	☑
I confirm that all details provide above are an accurate description of how AI was used for this assessment.	☑