IJCE; INTERNATIONAL JOURNAL OF CYBER ENGINEERING

E-ISSN: XXXX-XXXX
Vol. 01, No. 01, December, 2025, PP: 43-53
DOI:

ARTICLE

Network Forensic Analysis of a Hancitor–Cobalt Strike Incident: Application of the NIST Methodology

Jamaluddin [1], Ridho Surya Kusuma [2,*]

Department of Informatics, Universitas Siber Muhammadiyah, Yogyakarta, Indonesia

***Corresponding author:** ridhosuryakusuma@sibermu.ac.id, jamaluddin20220100024@sibermu.ac.id

Abstract

Cybersecurity incidents involving the Hancitor and Cobalt Strike malware have been increasing and pose a significant threat to the security of organizational information systems. This study aims to conduct a network forensic analysis of attacks that employ the combined use of Hancitor and Cobalt Strike by applying the methodology of the National Institute of Standards and Technology (NIST) SP 800-86. The research adopts a qualitative approach by implementing the four stages of the NIST framework, namely collection, examination, analysis, and reporting. Data were obtained through simulated attacks in a controlled environment using forensic tools such as Wireshark and NetworkMiner. The findings reveal an attack pattern that begins with the initial Hancitor infection delivered through a malicious Microsoft Office document, followed by command and control (C2) communication used to download and execute the Cobalt Strike payload. Network packet analysis successfully identified suspicious traffic characteristics, including Cobalt Strike beaconing and data exfiltration activities. This study concludes that the application of the NIST forensic methodology is effective in uncovering the stages of such attacks and can assist organizations in responding to similar incidents in the future. Furthermore, the research findings can serve as indicators of compromise (IoCs) to enhance early detection of attacks involving the combined use of Hancitor and Cobalt Strike.

Keywords: Network Forensics, Hancitor, Cobalt Strike, NIST SP 800-86, Malware Analysis

1. Introduction

In the current digital era, cyberattacks have become increasingly complex and sophisticated, necessitating the application of effective forensic methodologies to identify and analyze security incidents. One significant threat is the Hancitor malware, which is frequently employed as an entry point to install Cobalt Strike, a post-exploitation framework that can be leveraged for further malicious activities. Conducting network forensic analysis of such incidents is essential to understand the modus operandi of attackers and to develop effective mitigation strategies (Firdonsyah, 2022).

Hancitor is a malware family well known for its ability to propagate through phishing emails and exploit system vulnerabilities to download additional payloads such as Cobalt Strike. Although Cobalt Strike was originally designed as a threat simulation tool, it is often misused by adversaries to gain control over compromised systems. Incidents involving the combination of Hancitor and Cobalt Strike can result in significant organizational damage, including the theft of sensitive data and disruption of operations. Consequently, comprehensive network forensic analysis is required to identify traces of the attack and determine its source (Putra, 2024).

This study aims to analyze incidents involving Hancitor and Cobalt Strike through a network forensic approach by applying the methodology recommended by the National Institute of Standards and Technology (NIST). The specific objectives are: (1) to identify indicators of compromise (IoCs) associated with Hancitor and Cobalt Strike (Riadi, 2022); (2) to examine the attack flow and techniques employed by adversaries (Kurniawan, 2024); and (3) to evaluate the effectiveness of the NIST methodology in the context of network forensic investigations (Fitriana, 2020). The framework proposed by NIST provides a systematic structure for conducting network forensic analysis, encompassing four principal stages: Collection, Examination, Analysis, and Reporting (Putra, 2024; Qureshi, 2021; Riadi, 2020). This approach will be supported by the use of network forensic tools such as Wireshark for traffic analysis and Autopsy for digital data examination (Putra, 2024). The study will also draw upon contemporary literature in the fields of digital forensics and cybersecurity to ensure the relevance and accuracy of the methods employed (Kurniawan, 2024).

As part of this research, various tools and techniques applied in network forensic investigations will be discussed, along with their applicability in identifying and analyzing incidents involving Cobalt Strike. The study will further explore the challenges faced by investigators in collecting and examining data, and how the NIST-based methodology can assist in overcoming these challenges (Firdonsyah, 2022; Putra, 2024). Accordingly, this research is expected to provide a significant contribution to the understanding and practice of network forensic analysis within the context of increasingly complex cyberattacks.

2. Literature Review

Hancitor and Cobalt Strike represent serious threats within the domain of cybersecurity. Hancitor functions as a malware loader, frequently employed to distribute a variety of other malicious payloads, while Cobalt Strike is a post-exploitation framework that is often misused by adversaries to conduct harmful activities within compromised networks.

The NIST SP 800-86 methodology provides a structured framework consisting of four principal stages in the process of digital investigation. The first stage, collection, involves acquiring relevant data such as network traffic captures using Wireshark, system and firewall logs, as well as memory dumps and disk images (Riadi, 2022). The collected data then proceeds to the examination stage, where packet analysis is conducted with Wireshark, digital artifacts are extracted, and communication protocols and patterns are identified (Firdonsyah, 2022).

The subsequent analysis stage entails a deeper investigation of the examined data (Duncan, 2021). This includes reconstructing the timeline of the attack, analyzing malware communication patterns, and correlating evidence across multiple data sources (Kurniawan, 2024). Finally, the reporting stage consolidates the findings into a comprehensive report (Fitriana, 2020). This report documents the results of the investigation, outlines the reconstructed attack timeline, and provides mitigation recommendations to prevent similar incidents in the future (Duncan, 2021).

2.1. Hancitor with Cobalt Strike

Since November 5, 2020, actors deploying Hancitor have demonstrated a recurring and consistent infection pattern characterized by staged delivery and follow-on payload deployment. In this campaign structure, Hancitor typically serves as the initial loader that establishes execution on the victim host and facilitates subsequent access, while Cobalt Strike is commonly introduced in later phases to enable post-exploitation activity and operator-controlled actions within the compromised environment. To clarify this multi-stage progression, this study presents

a flow diagram that summarizes the chain of events in Figure 1, highlighting the sequential transition from initial compromise to downstream execution stages (Duncan, 2021).

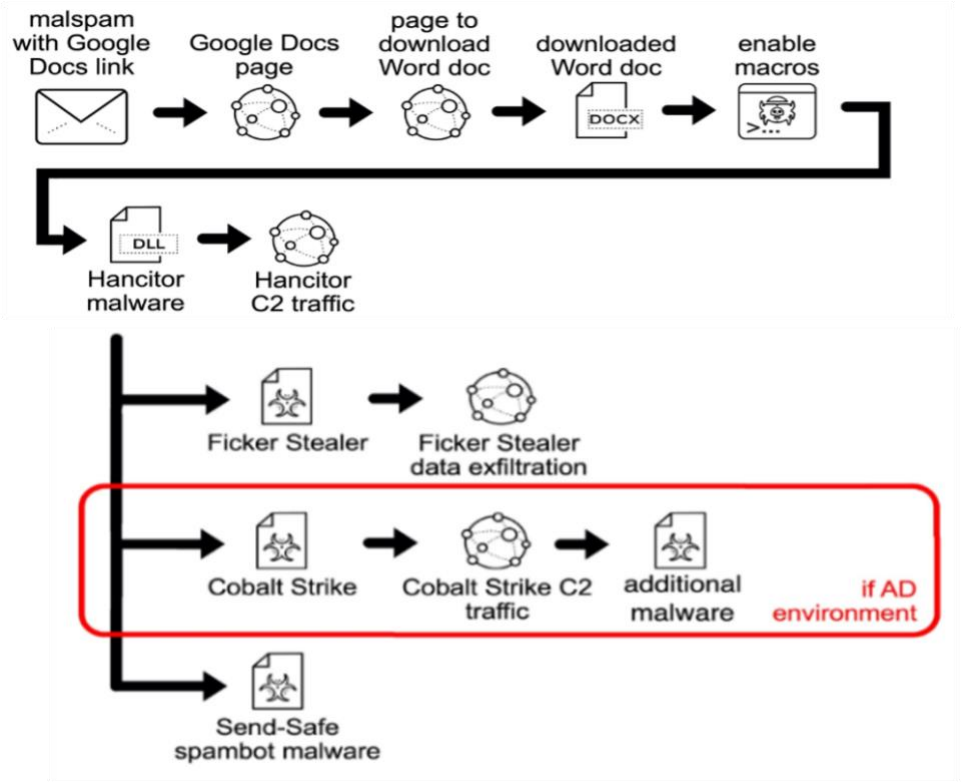


Figure 1. Hancitor chain of events.

Also known as Chanitor, Hancitor is a malware loader employed by threat actors identified as MAN1, Moskalvzapoe, or TA511. Hancitor establishes initial access on vulnerable Windows hosts and subsequently delivers additional malware. Recent Wireshark tutorials have analyzed the activity of contemporary Hancitor infections, providing guidance on how to identify Hancitor and its associated secondary payloads. These tutorials discuss examples of Hancitor infections involving Cobalt Strike, Ficker Stealer, NetSupport Manager RAT, network ping utilities, and the Send-Safe spambot (Duncan, 2021).

The Global Security Operations Center (GSOC) team at Cybereason has published a Threat Analysis Report to raise awareness of impactful threats. This report investigates the Hancitor-Cobalt Strike threat landscape and provides practical recommendations for defense (Cybereason Global SOC Team, 2022). Within the report, GSOC details three recent attack scenarios in which adversaries rapidly deployed malware loaders such as IcedID, QBot, and Emotet to distribute the Cobalt Strike framework across compromised systems (Cybereason Global SOC Team, 2022).

2.2. Tools and Technologies

2.2.1. Wireshark

Wireshark is a network analysis software that supports real-time capture and detailed inspection of network traffic, enabling analysts to observe communication behavior at the packet level. In practice, it is frequently used for troubleshooting connectivity and performance issues, identifying disruptions, and supporting security incident handling by providing direct visibility into what is transmitted across the network (P. S. Informatika, 2023). In this study, Wireshark (including Tshark for PCAP-based analysis) is utilized to structure traffic investigation through four main capabilities.

First, display filters are applied as expressive rules to selectively view packets of interest, allowing suspicious traffic to be isolated efficiently from large PCAP traces (Sharma, 2024). Second, protocol analysis is conducted to interpret how packets are constructed and exchanged across network interfaces, where timestamps, protocol interactions, and error indicators help explain observable behaviors and potential anomalies relevant to incident response (Johnson, 2024). Third, flow analysis is used to monitor traffic streams and summarize communication patterns, enabling the identification of recurring behaviors, volume characteristics, and trends that may indicate

abnormal activity (Sharma, 2024). Finally, Wireshark's statistical analysis features are employed to enrich interpretation at scale; for example, protocol hierarchies, conversations, and endpoints provide aggregated perspectives that assist in validating traffic prominence and contextualizing suspicious sessions within the overall trace (Sharma, 2024).

2.2.2. Kali Linux

Kali Linux is a Linux-based operating system recognized for its integrated toolkit dedicated to cybersecurity testing and penetration assessment. Its role in this study is to provide a controlled environment for conducting security-oriented tasks using pre-integrated utilities that are commonly adopted in practical digital security workflows (Kali Linux, 2023).

2.2.3. NetworkMiner

NetworkMiner is a network forensic tool designed to capture and analyze network traffic either by loading PCAP traces or by observing traffic directly through available interfaces on the operating system. In investigative workflows, it is particularly useful for supporting forensic reconstruction of communication events and extracting network-derived artifacts that complement packet-level observations (Alshalah, 2022).

2.2.4. VirusTotal

VirusTotal is a widely used threat-intelligence and malware-scanning service that supports multiple submission mechanisms, including a public web interface, desktop uploaders, browser extensions, and programmable APIs. In security analysis, it is commonly employed to evaluate files and URLs against a broad collection of antivirus engines and security tools, providing a practical baseline for initial triage and reputation-based checking (VTDOC, 2024).

3. Research Methodology

The methodology of this study adopts a network forensic framework based on the National Institute of Standards and Technology (NIST) SP 800-86, which provides a systematic and structured approach for handling cybersecurity incidents. The NIST framework consists of four principal stages: Collection, Examination, Analysis, and Reporting (Putra, 2024; Fitriana, 2020; Riadi, 2020). Each stage is defined by clear objectives and procedures, enabling investigators to effectively identify, acquire, and analyze digital evidence. Within the context of this research, each stage is elaborated in detail to demonstrate how the application of this methodology supports the forensic investigation of incidents involving advanced tools such as Cobalt Strike, as illustrated in Figure 2.

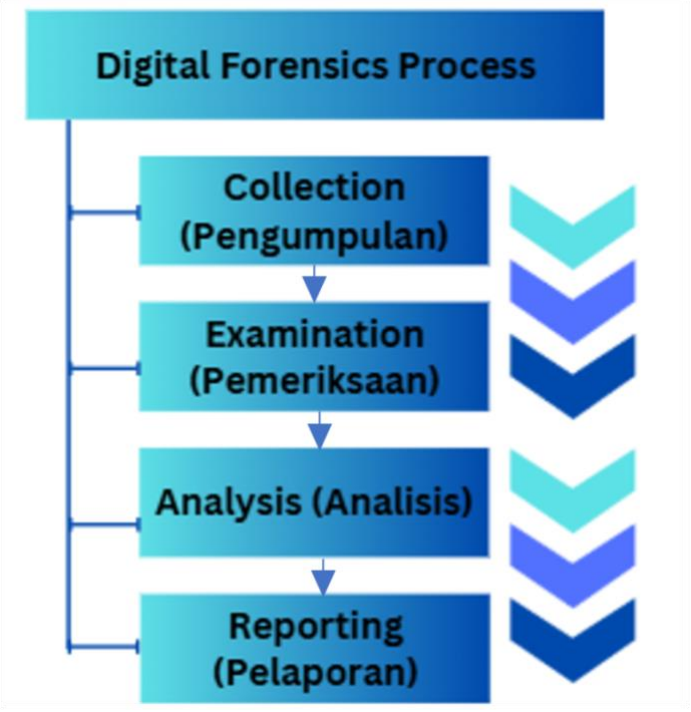


Figure 2. Digital Forensics

The stages illustrated in Figure 2 are explained as follows:

3.1. Collection

In this stage, investigators gather all relevant evidence from the network involved in the incident. This includes data acquisition from hardware, software, and other network resources. Often referred to as the preparation phase, collection involves securing evidence and tools required for digital data acquisition, while ensuring the integrity of the data throughout the process (Julian, 2023). A systematic and organized approach is essential to guarantee that all relevant evidence remains accessible for subsequent analysis.

3.2. Examination

Once data has been collected, the examination phase begins. Investigators conduct an initial evaluation of the acquired data to identify patterns or anomalies that may indicate suspicious activity (Alshalah, 2022). This process may involve log analysis, packet inspection, and the use of forensic tools to extract information from the collected data (VTDOC, 2024). During this stage, it is critical to document all findings and procedures, as these records serve as references for the subsequent analysis phase.

3.3. Analysis

The analysis stage represents the core of the forensic process, where investigators perform an in-depth evaluation of the examined data. This includes identifying the techniques employed by attackers, reconstructing the attack sequence, and assessing the impact on the network (Huda, 2020). In cases involving tools such as Cobalt Strike, analysis may encompass the identification of payloads, infiltration methods, and persistence mechanisms used by adversaries (Julian, 2023). The outcomes of this stage provide valuable insights for developing mitigation strategies and preventive measures against future incidents.

3.4. Reporting

The final stage focuses on producing a comprehensive report that consolidates the investigation's outputs, including the key findings, the methodologies applied, and recommendations for follow-up actions (P. S. Informatika, 2023). To be operationally useful, the report must present conclusions in a clear and concise manner so that it remains accessible to diverse stakeholders, such as management, technical security teams, and—when escalation is required—legal authorities (Duncan, 2021). In addition to narrative conclusions, the report should attach supporting evidence (e.g., logs, packet captures, extracted artifacts, and validation outputs) to substantiate each claim and to preserve the option of legal admissibility where applicable.

Across all phases, the integrity and security of collected data remain a primary requirement. Accordingly, the investigation must prioritize non-intrusive acquisition approaches, maintain accurate documentation of each procedural step, and ensure that evidence handling remains consistent and traceable so that results are both reliable and legally defensible (Johnson, 2024). By adhering to the NIST methodology, network forensic investigations can be conducted in a structured and dependable manner, strengthening the overall effectiveness of incident response. In this study, the NIST SP 800-86 framework is applied to guide a network forensic analysis of an incident involving Hancitor and Cobalt Strike. The next section details the tools and resources utilized in the research, as summarized in Table 1.

Table 1. Hardware and Software for Digital Forensics

No	Tool/Material	Function	Description
1	Laptop	Forensic Hardware	Intel(R) Core (TM) i5-7200U CPU @ 2.50GHz, 2.70 GHz, used as the primary forensic workstation.
2	Wireshark	Network Traffic Analysis	An open-source application for real-time network analysis, capable of capturing packets and examining communication protocols.
3	VirusTotal	Malware Scanning Service	A free online service used to analyze files and URLs against multiple antivirus engines and security tools.

No	Tool/Material	Function	Description
4	Kali Linux Operating System	Digital Forensics Environment	A Linux-based operating system equipped with a comprehensive suite of cybersecurity and forensic tools.
5	Incident Documentation	Evidence Recording	Documentation created during the initial investigation process, including incident timestamps, suspicious activities, and user involvement.

The table 1 show that this research uses a laptop workstation as the primary forensic platform (Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz, 2.70 GHz). Wireshark is employed for network traffic analysis to capture packets and inspect protocol-level communication. VirusTotal is used as a malware scanning service to evaluate files and URLs using multiple antivirus engines and security tools. Kali Linux serves as the digital forensics environment due to its comprehensive suite of cybersecurity and forensic utilities. In addition, incident documentation is maintained as a formal evidence record, capturing incident timestamps, suspicious activities, and relevant user involvement to support traceability throughout the investigation.

4. Results and Discussion

The increasing frequency and complexity of cyberattacks, particularly those involving the Hancitor malware, highlight the need for in-depth research to better understand these threats, specifically the network traffic associated with Hancitor activity. This study adopts the NIST methodology as a structured framework for conducting digital forensic analysis. The following subsections present the stages of network forensics and the findings obtained in accordance with the NIST approach.

4.1. Collection

The data collection stage in this research follows the NIST methodology. Based on an analysis of the website malware-traffic-analysis.net, particularly the post-dated September 29, 2021, several digital artifacts were successfully gathered, as illustrated in Figure 3 (Duncan, 2021).

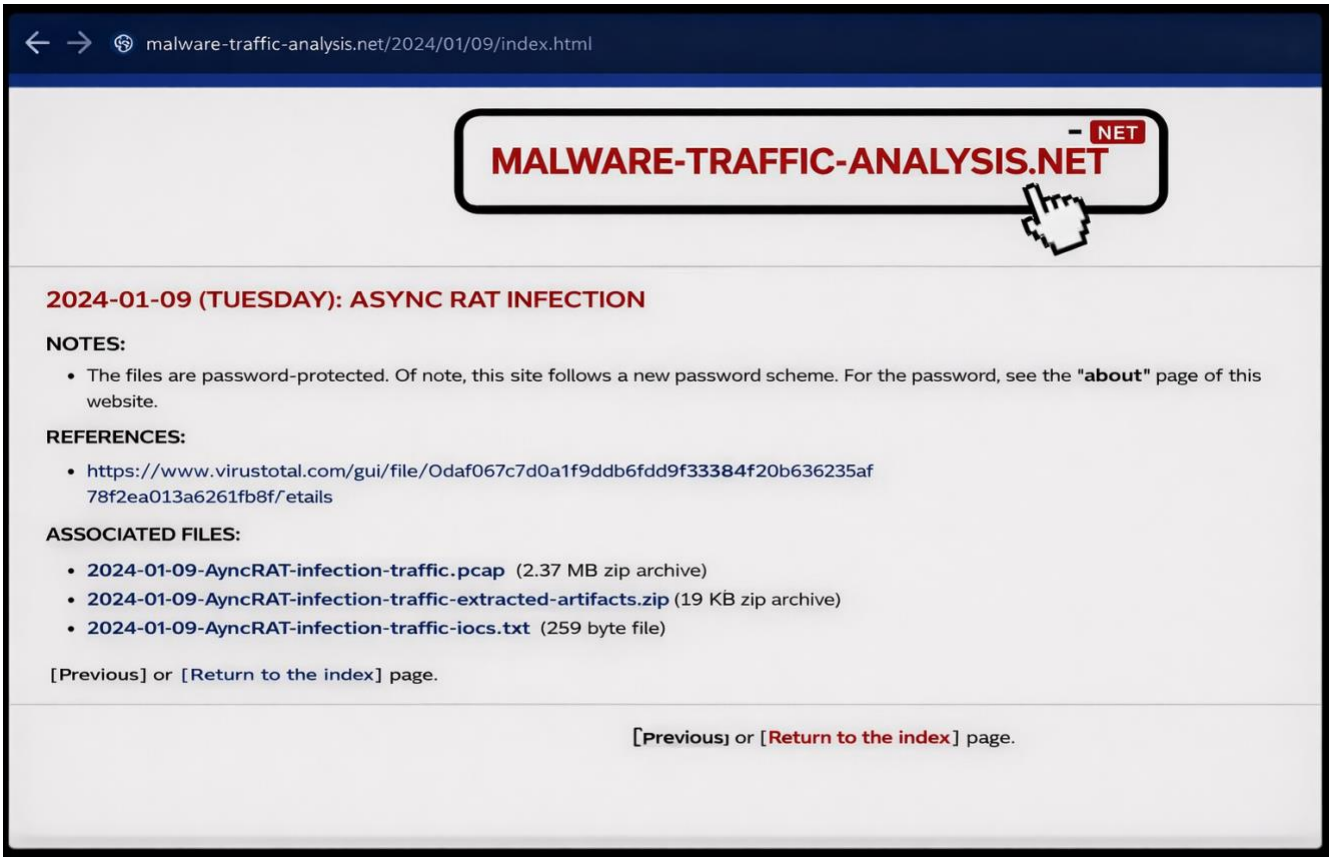


Figure 3. Data Sources

These artifacts form the evidentiary baseline for the subsequent forensic examination, ensuring that all analysis activities are grounded in authentic, case-relevant material and can be reproduced through consistent reference to the same data sources. In this study, the collected artifacts comprise four primary categories that collectively capture narrative context, delivery vectors, network-level behavior, and executable payload evidence.

4.1.1. Text Document

The archive "2021-09-29-Hancitor-with-Cobalt-Strike-notes.txt.zip" contains a supporting notes document associated with the Hancitor campaign leveraging Cobalt Strike. This text resource provides contextual anchoring for the investigation by documenting case details that can guide the identification and validation of indicators of compromise (IoCs), observed techniques, and affected assets during later stages of analysis.

4.1.2. Email Samples

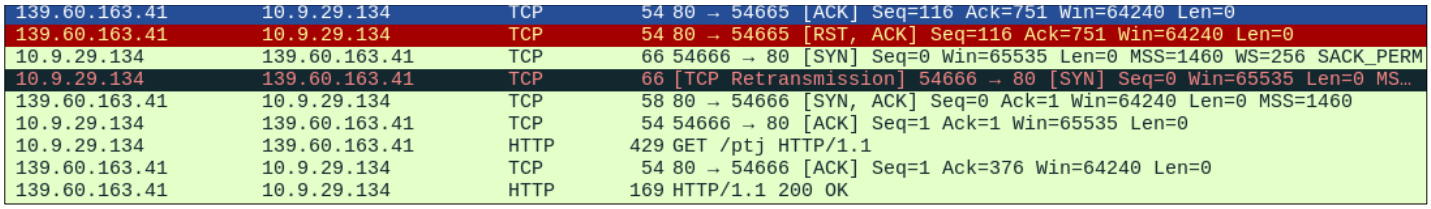
The archive "2021-09-29-Hancitor-malspam-57-examples.zip" comprises 57 malspam email specimens used as part of the distribution chain. Examining these samples enables the study to characterize the social engineering strategy and delivery patterns employed by the threat actor, including recurring lures, message structures, and other consistent traits that support clustering and comparison across phishing attempts.

4.1.3. Network Traffic Data

The archive "2021-09-29-Hancitor-with-Cobalt-Strike-traffic.pcap.zip" contains packet-capture (PCAP) evidence associated with the incident. This dataset serves as the primary basis for network-behavior reconstruction, enabling the identification of suspicious sessions, potential command-and-control (C&C) communications, and other anomalous traffic patterns that link host activity to external infrastructure.

4.1.4. Malware Samples

The archive "2021-09-29-Hancitor-malware-samples.zip" contains malware specimens involved in the attack chain. These samples enable deeper inspection of operational behavior, including execution logic, persistence-related actions, and evasion characteristics, thereby supporting attribution of observed network events to specific payload activity and clarifying potential impact on targeted systems. In addition to these core artifacts, supplementary collection also captured evidence related to ransom demands, as illustrated in Figure 4.



139.60.163.41	10.9.29.134	TCP	54	80 → 54665 [ACK] Seq=116 Ack=751 Win=64240 Len=0
139.60.163.41	10.9.29.134	TCP	54	80 → 54665 [RST, ACK] Seq=116 Ack=751 Win=64240 Len=0
10.9.29.134	139.60.163.41	TCP	66	54666 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
10.9.29.134	139.60.163.41	TCP	66	[TCP Retransmission] 54666 → 80 [SYN] Seq=0 Win=65535 Len=0 MS…
139.60.163.41	10.9.29.134	TCP	58	80 → 54666 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
10.9.29.134	139.60.163.41	TCP	54	54666 → 80 [ACK] Seq=1 Ack=1 Win=65535 Len=0
10.9.29.134	139.60.163.41	HTTP	429	GET /ptj HTTP/1.1
139.60.163.41	10.9.29.134	TCP	54	80 → 54666 [ACK] Seq=1 Ack=376 Win=64240 Len=0
139.60.163.41	10.9.29.134	HTTP	169	HTTP/1.1 200 OK

Figure 4. Network Traffic

Figure 4 illustrates the packet capture (PCAP) results obtained through Wireshark, showing a TCP conversation between two devices. This communication indicates an attempted TCP connection, most likely associated with an HTTP request. The captured traffic log reveals several notable characteristics:

a. Black Rows: These packets represent problematic transmissions, such as retransmissions or other errors. Their presence suggests that earlier packet delivery failed, necessitating retransmission.

b. Red Rows: These packets indicate terminated or reset (RST) communications. Such events occur when one party, in this case the sender, abruptly ends the connection. This behavior may reflect connection instability or potentially suspicious activity.

c. Green Rows: These packets represent successful communications, typically associated with HTTP or other protocols operating normally. Despite disruptions in certain packets, the overall communication flow remains functional.

4.2. Examination

At this stage, the captured logs are examined to identify suspicious patterns within the recorded network traffic. The examination process involves searching through the packet data to detect anomalies, irregular communication flows, or indicators of compromise that may signify malicious activity. This step provides the foundation for deeper forensic analysis in subsequent phases.

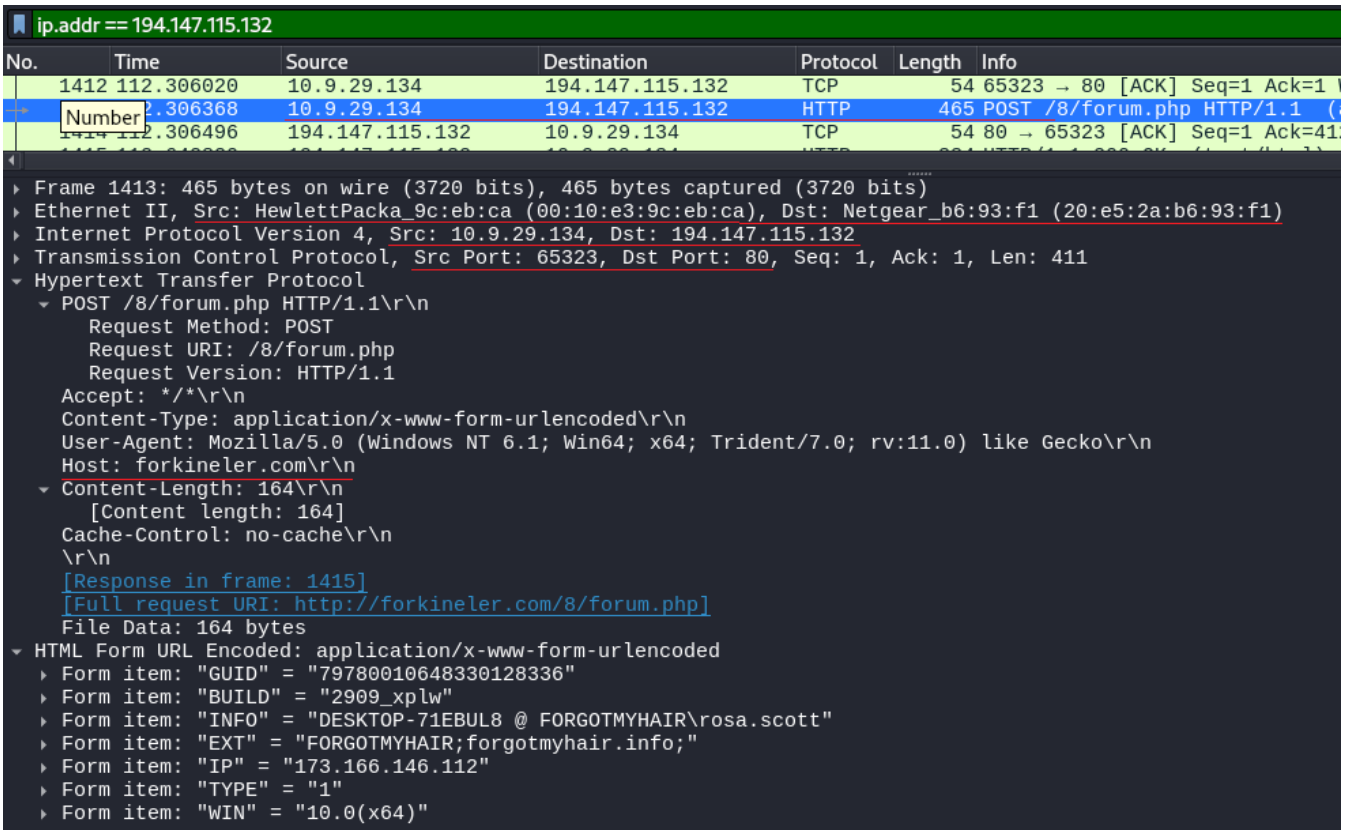


Figure 5. Network Traffic Log Tracking

Figure 5 illustrates the transmission of form data from device 10.9.29.134 to a server with IP address 194.147.115.132 via the HTTP protocol. The data was sent using the POST method, containing user information and device preferences, and directed to a specific form page.

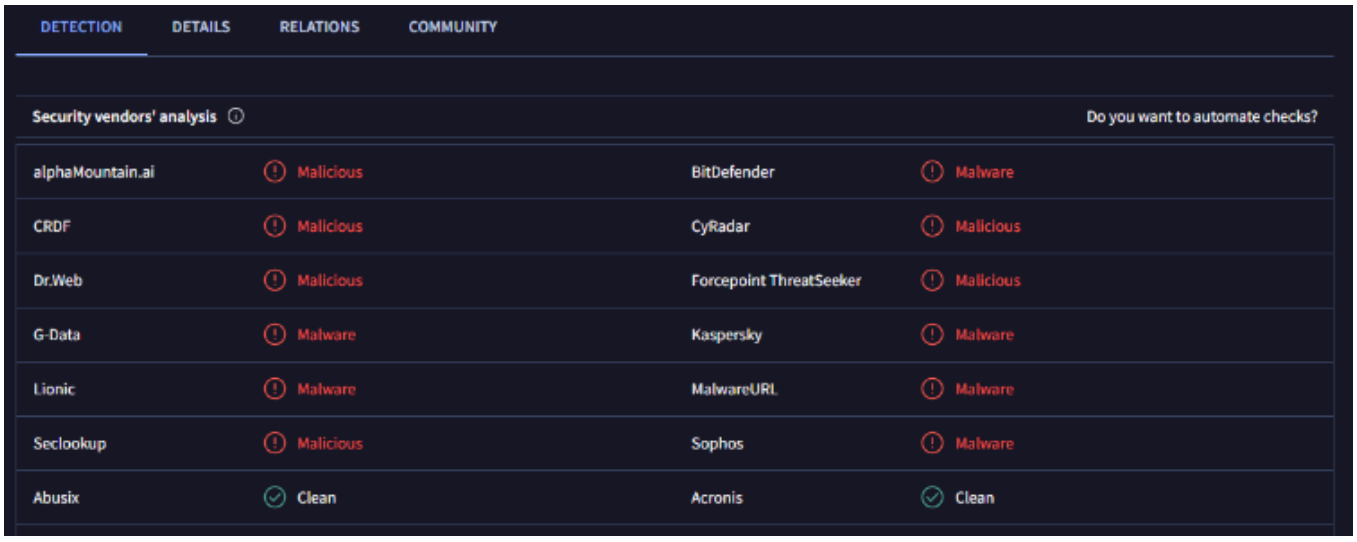


Figure 6. File Test Result

Figure 6 presents the results of file testing through VirusTotal, indicating that ten security services (alphaMountain.ai, CRDF, Dr.Web, G-Data, Lionic, Seclookup, BitDefender, CyRadar, Forcepoint ThreatSeeker, Kaspersky, MalwareURL, and Sophos) detected the file as a threat, labeling it either “Malicious” or “Malware.”

4.3. Analysis

The third stage of the forensic process, Analysis, involves correlating log data to confirm the presence of malicious activity. The logs reveal multiple detections of threats marked as “Malicious” or “Malware,” as shown in Figure 7.

Source	Destination	Protocol	Length	Source Port	Destination Port	Info
10.9.29.134	194.147.115.132	TCP	66	65331	80	65331 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS
194.147.115.132	10.9.29.134	TCP	58	80	65331	80 → 65331 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0
10.9.29.134	194.147.115.132	TCP	54	65331	80	65331 → 80 [ACK] Seq=1 Ack=1 Win=65535 Len=0
10.9.29.134	194.147.115.132	HTTP	465	65331	80	POST /8/forum.php HTTP/1.1 (application/x-www-for
194.147.115.132	10.9.29.134	TCP	54	80	65331	80 → 65331 [ACK] Seq=1 Ack=412 Win=64240 Len=0
194.147.115.132	10.9.29.134	HTTP	257	80	65331	HTTP/1.1 200 OK (text/html)
10.9.29.134	194.147.115.132	TCP	54	65331	80	65331 → 80 [ACK] Seq=412 Ack=204 Win=65535 Len=0
194.147.115.132	10.9.29.134	TCP	54	80	65331	80 → 65331 [FIN, PSH, ACK] Seq=204 Ack=412 Win=642
10.9.29.134	194.147.115.132	TCP	54	65331	80	65331 → 80 [ACK] Seq=412 Ack=205 Win=65535 Len=0
10.9.29.134	194.147.115.132	TCP	54	65331	80	65331 → 80 [FIN, ACK] Seq=412 Ack=205 Win=65535 Le
194.147.115.132	10.9.29.134	TCP	54	80	65331	80 → 65331 [ACK] Seq=205 Ack=413 Win=64239 Len=0

Figure 7. Network Traffic Log

Figure 7 displays communication between IP address 10.9.29.134 and a website hosted at 194.147.115.132 over port 80, the standard port for HTTP. The process begins with a TCP handshake, where the client sends a SYN packet, the server responds with SYN-ACK, and the client confirms with ACK. Once the connection is established, the client issues an HTTP POST request to the endpoint /8/forum.php. The request data is transmitted in the format application/x-www-form-urlencoded, commonly used for HTML form submissions. The server responds with a status code 200 OK, indicating successful processing, and returns data in text/html format. The client then acknowledges the response with an ACK packet. The communication concludes with connection termination through the exchange of FIN, ACK, and PSH packets. This analysis demonstrates a simple HTTP transaction in which the client submits data via POST. However, the communication is unencrypted, relying on HTTP rather than HTTPS, thereby increasing the risk of data exposure. Below is the advanced and suspicious network traffic log as illustrated in Figure 8.

Source	Destination	Protocol	Length	Source Port	Destination Port	Info	Host
10.9.29.134	20.189.173.5	TLSv1.2	242	65321	443	Client Hello (SNI=self.events.data.microsoft.com)	
10.9.29.134	50.17.226.156	HTTP	218	65322	80	GET / HTTP/1.1	api.ipify.org
10.9.29.134	194.147.115.132	HTTP	465	65323	80	POST /8/forum.php HTTP/1.1 (application/x-www-form-urlen…	forkineler.com
10.9.29.134	8.209.76.110	HTTP	224	65324	80	GET /41s.bin HTTP/1.1	4maurpont.ru
10.9.29.134	8.209.76.110	HTTP	223	65324	80	GET /41.bin HTTP/1.1	4maurpont.ru
10.9.29.134	139.60.163.41	HTTP	242	65325	443	GET /NfSU HTTP/1.1	139.60.163.41:443
10.9.29.134	139.60.163.41	HTTP	251	65326	80	GET /3Gvl HTTP/1.1	139.60.163.41
10.9.29.134	139.60.163.41	HTTP	441	65327	443	GET /updates.rss HTTP/1.1	139.60.163.41:443
10.9.29.134	139.60.163.41	HTTP	429	65328	80	GET /ptj HTTP/1.1	139.60.163.41
10.9.29.134	139.60.163.41	HTTP	441	65329	443	GET /updates.rss HTTP/1.1	139.60.163.41:443
10.9.29.134	139.60.163.41	HTTP	429	65330	80	GET /ptj HTTP/1.1	139.60.163.41
10.9.29.134	194.147.115.132	HTTP	465	65331	80	POST /8/forum.php HTTP/1.1 (application/x-www-form-urlen…	forkineler.com
10.9.29.134	139.60.163.41	HTTP	441	65332	443	GET /updates.rss HTTP/1.1	139.60.163.41:443
10.9.29.134	139.60.163.41	HTTP	429	65333	80	GET /ptj HTTP/1.1	139.60.163.41

Figure 8. Advanced Network Traffic Log

Figure 8 provides evidence of more complex network activity originating from device 10.9.29.134. The device issued multiple requests to different servers, including an HTTP GET request to IP 50.17.226.156 to obtain its public IP address, and POST requests to /8/forum.php at IP 194.147.115.132. Additional connections were observed to IP 8.209.76.110 and IP 139.60.163.41 across various endpoints. Most of these communications were conducted without encryption, raising the risk of sensitive data leakage. The POST request to /8/forum.php is particularly suspicious, as it may indicate unauthorized data transmission to an untrusted server.

Further investigation revealed that IP 194.147.115.132 is associated with the domain forkineler.com and located in Lelystad, Flevoland, Netherlands; IP 8.209.76.110 corresponds to the domain 4maurpont.ru and is located in Frankfurt am Main, Hesse, Germany; and IP 139.60.163.41 is located in New York City, United States. These findings warrant deeper analysis to determine whether the domains are linked to malicious activity such as malware distribution or phishing.

4.4. Reporting

Based on the file testing conducted through VirusTotal (Figure 8), the analyzed file was identified as a Win32 DLL with detailed technical properties. The file includes multiple hash values for verification and identification, such as MD5 (7e3ef3feb2939316d7395acbdc6f5e99), SHA-1 (7b12236565453b2a6f38f82c204cc8a1b0f16275), and SHA-256 (0ab628d05eebe781cbb8f811652dbb6f3170c662fb906b21b16ad866626f86b2) in Figure 9.

Figure 9. IoCs

Figure 9 show that the file is classified as a PE32 executable (DLL) designed for Microsoft Windows systems with Intel 80386 32-bit architecture, and has a size of 267.59 KB (274,016 bytes). Compatibility analysis indicates that the file can execute across multiple platforms with varying degrees of compatibility: Win64 Executable (40.3%), Win16 NE Executable (19.3%), Win32 Executable (17.2%), and OS/2 Executable. Additional hash values, including Vhash, Authentihash, Imphash, Rich PE header hash, SSDEEP, and TLSH, were also identified, which are commonly used in malware classification and forensic verification. The technical characteristics of the file confirm that it is a Windows-based component, typically analyzed within the context of digital forensic investigations or system security assessments.

5. Conclusion

The network traffic analysis revealed the presence of malicious activity, characterized by the transmission of form data from a client device to a remote server. Virus scanning results confirmed that files associated with this activity were identified as malware by multiple antivirus engines. Further examination of the traffic logs uncovered communication with several external domains, most of which occurred without encryption, thereby increasing the risk of sensitive data leakage. File analysis indicated that the suspicious artifact was a Windows executable designed to operate across multiple platforms. The technical characteristics of the file, combined with the observed unencrypted communications, highlight a significant security threat. Collectively, these findings underscore the existence of potential malicious activity that warrants further investigation and mitigation to safeguard organizational systems and data assets.

Reference

Firdonsyah, A. (2022). Analisis forensik rekayasa dokumen digital dengan metode NIST. Informatics Journal, 7(2), 121–127.

Putra, M. A. D. (2024). Analisis forensik pada Instagram dan TikTok dalam mendapatkan bukti digital dengan menggunakan metode NIST 800-86. Jurnal Sistem Informasi Galuh, 2(1), 44–54.

Riadi, I. (2022). Buku ajar forensik jaringan & cloud. Yogyakarta: Diandra Kreatif.

Kurniawan, A. M. (2024). Analisis keamanan jaringan pada rumah menggunakan metode NIST. Jurnal Mahasiswa Teknik Informatika, 8(2), 1659–1664.

Fitriana, M. (2020). Penerapan metode National Institute of Standards and Technology (NIST) dalam analisis forensik digital untuk penanganan cyber crime. Jurnal Pendidikan Teknologi Informasi, 4(1), 29–39.

Qureshi, S. (2021). Analysis of challenges in modern network forensic framework. Security and Communication Networks, 2021, 3–12.

Riadi, S. I. (2020). Perbandingan tool forensik data recovery berbasis Android menggunakan metode NIST. Jurnal Teknologi Informasi dan Ilmu Komputer, 7(1), 199–207.

Umar, R., Riadi, I., & Kusuma, R. S. (2021). Analysis of Conti ransomware attack on computer network with live forensic method. International Journal on Informatics for Development, 10(1), 54–61.

Duncan, B. (2021, April 1). Malware. Palo Alto Networks. https://unit42.paloaltonetworks.com/hancitor-infections-cobalt-strike/

Duncan, B. (2021, April 7). Cybersecurity tutorials. Palo Alto Networks. https://unit42.paloaltonetworks.com/wireshark-tutorial-hancitor-followup-malware/

Cybereason Global SOC Team. (2022, February 8). Threat analysis. Cybereason. https://www.cybereason.com

P. S. Informatika. (2023, March 27). Wireshark. TIF UAD. https://tif.uad.ac.id/wireshark-perangkat-lunak-analisis-jaringan/

Sharma, M. (2024). Ethical hacking and network analysis with Wireshark. London: BPB Publications.

Johnson, R. (2024). The Wireshark handbook: Practical guide for packet capture and analysis. Framingham, MA: HiTeX Press.

Huda, M. (2020). Keamanan informasi. Nulisbuku.

Alshalah, H. (2022). Artificial intelligence model for network security analysis. NeuroQuantology, 20(13), 2637–2645.

VTDOC. (2024, October). How it works. https://docs.virustotal.com/docs/how-it-works

Julian, T. S. D. (2023). Analisa kinerja aplikasi digital forensik Autopsy untuk pengembalian data menggunakan metode NIST SP 800-86. Jurnal Informatika Terpadu, 9(2), 138–145.

Duncan, B. (2021, September 29). 2021-09-29 (Wednesday) – Hancitor with Cobalt Strike. Malware-Traffic-Analysis.net. https://malware-traffic-analysis.net/2021/09/29/index.html

Asiedu. (2022). Packet analysis for network forensics. Advances in Multidisciplinary & Scientific Research Journal Publication, 1(1), 91–98.

Porambage, P. (2021). The roadmap to 6G security and privacy. IEEE Open Journal of the Communications Society, 2, 1094–1122.

Wijayanto, A. (2023). TAARA method to processing on the network forensics in the event of an ARP spoofing attack. Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), 7(2), 208–217.

Akinbi. (2022). Digital forensics challenges and readiness for 6G Internet of Things (IoT) networks. Wiley Interdisciplinary Reviews: Forensic Science, 1(1), e1.