Analysis and Detection of Distributed Denial of Service Attacks using National Institute of Standards and Technology Method

Muhammad Fauzan Taufiqurrahman
Department of Informatics
Universitas Ahmad Dahlan
Yogyakarta of Indonesia

Imam Riadi
Department of Information System
Universitas Ahmad Dahlan
Yogyakarta of Indonesia

ABSTRACT
This research analyzes Distributed Denial of Service (DDoS) attacks of the UDP Flood type on a Linux-based server using a live forensic approach, conducted in a closed network environment with a Linux server as the target and a client as the attacker using Low Orbit Ion Cannon (LOIC). Evidence acquisition was performed using tcpdump in .pcap format, and the captured data was analyzed using Wireshark based on the NIST method stages: collection, examination, analysis, and reporting. The analysis parameters included protocol type, source and destination IP addresses, destination port, packet size, packet rate, time interval, and traffic distribution pattern. The results showed that 2,046,672 packets were captured in approximately 26 seconds, with average rate of 78,015 packets per second. The traffic was dominated by small UDP packets of 62 bytes, transmitted massively, repeatedly, and directly toward the target server, indicating the characteristics of a UDP Flood attack. Recommended mitigation strategies include filtering UDP traffic using iptables, restricting port access, and conducting regular network monitoring. This research demonstrates that live forensic analysis based on the NIST framework is effective for systematically identifying and analyzing UDP Flood attacks.

Keywords
Distributed Denial of Service, Forensics Digital, NIST, UDP Flood

1. INTRODUCTION
Distributed Denial of Service (DDoS) attacks are serious threats to network security because they flood servers with large amounts of data traffic that exceed normal capacity. The impacts may include performance degradation, request errors, system failure, and even hardware damage due to excessive load [1]. Attackers conduct these attacks in a distributed manner by controlling multiple computers to attack a target simultaneously [2]. In normal communication, the client and server perform a handshake process to establish a connection [3].

Linux is an open-source operating system that is widely used as a server because it supports various client-server-based network services, such as DHCP, email, HTTP, FTP, and DNS [4][5]. Its support for raw sockets also enables the development of DDoS detection systems in Linux environments, with promising experimental results [6]. This makes Linux relevant for DDoS forensic research.

User Datagram Protocol (UDP) is a fast and simple data transmission protocol [7]. One type of DDoS attack that exploits this protocol is UDP Flood, which is an attack that sends excessive UDP packets to overload the network and the target server [8][9][10]. Because UDP does not require connection verification, attackers can send packets massively without waiting for a response, causing disruption to the target service [11]. Therefore, effective mitigation strategies are needed to maintain service availability and network security [12].

This research uses the National Institute of Standards and Technology (NIST) framework because it is considered practical for handling network security incidents. NIST includes the stages of identification, protection, detection, response, and recovery. Compared with the Digital Forensic Research Workshop (DFRWS) method, which is more technical and academic, NIST is more suitable for the operational needs of DDoS attack analysis [13][14].

This research aims to apply live forensic analysis to analyze UDP Flood DDoS attacks on a server. Digital evidence is collected in real time without disrupting the running system, so that attack patterns can be identified before the data changes or is lost. The results of this research are expected to provide an understanding of UDP Flood patterns, appropriate mitigation steps, and contributions to strengthening network defense systems. In addition, this research can be used as a reference for forensic investigation procedures in handling network-based cyberattacks [15].

2. LITERATURE STUDY
2.1 Distributed Denial of Service
Distributed Denial of Service is an activity that sends a large number of data packets to a network to flood its traffic, causing the host to become inaccessible to legitimate users [16]. This attack floods the target server or network from multiple sources simultaneously, usually through devices controlled by the attacker in a botnet network. As a result, the server becomes overloaded, services slow down, or access becomes unavailable. This condition can disrupt normal network operations and reduce service reliability [17].

2.2 Digital Forensics
Digital forensics is a systematic process for collecting, preserving, analyzing, and reporting digital evidence from computers, networks, communication devices, or storage media. This process combines legal science and information technology to maintain the integrity of evidence [18].

Digital forensics consists of five main stages, namely identification, preservation, analysis, data filtering, and reporting. These stages aim to locate evidence, maintain data integrity, analyze information, select relevant data, and prepare accountable reports [19]. The stages are shown in Figure 1.

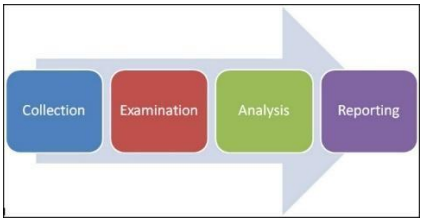

Figure 1 : Digital Forensics Step

2.3 National Institute of Standards and Technology

The National Institute of Standards and Technology is a standard guideline of the United States Government for managing risks in the implementation of information technology systems [20]. NIST is known through the Cybersecurity Framework (CSF), which helps organizations address information security threats effectively and continuously.

This framework has five main functions, namely identify, protect, detect, respond, and recover. These five functions are used to understand risks, protect critical services, detect incidents, respond to threats, and recover services so that organizations can return to normal operations. This framework can be applied across an entire organization or specifically to protect critical services. Organizations can also use it to prepare security reports and develop cybersecurity profiles according to their needs [21].

2.4 UDP Flooding

UDP Flooding is a type of Distributed Denial of Service (DDoS) attack that exploits the User Datagram Protocol (UDP) to flood a server with a large number of data packets, thereby overloading its resources and causing services to become unavailable to legitimate users. This attack is carried out by sending UDP packets to random ports without a connection verification process.

If a packet is received on a closed port, the server responds with an ICMP message, which further increases the system load. When the traffic exceeds capacity, bandwidth and processing resources are depleted, resulting in performance degradation and service disruption. This attack is generally carried out using a network of infected devices.

2.5 Live Forensic

Live forensic is a digital forensic approach that investigators conduct on an operating system to obtain evidence directly without stopping its services. This method enables investigators to acquire volatile data, such as memory contents, running processes, network connections, and other temporary activities, which may be lost if the system is shut down.

Investigators must implement this approach using careful procedures and non-invasive tools to avoid altering the original condition of the system. This approach is highly relevant to modern cyber incident handling, particularly in network attack cases such as DDoS attacks, because it enables investigators to identify attack patterns and respond quickly.

Investigators use live forensic as an important strategy to maintain service continuity and support an accurate and reliable investigation process.

2.6 Linux

Linux is a Unix-based operating system that is freely available under the GNU General Public License (GPL). Linux provides various distributions with software packages that users can update to obtain new features, bug fixes, and security updates. Linux is widely used on servers because it can manage data access security, maintain network stability, and support various applications [22].

2.7 Firewall

Network administrators design a firewall as a network security system to monitor, filter, and control incoming and outgoing data traffic based on predefined security rules. Network administrators use a firewall as an effective tool to improve computer network security [23].

2.8 Packet Sniffing

Packet sniffing is an activity that monitors data packets passing through a computer network. The system divides data from the sender into small packets, and the receiver reassembles the packets into complete data [24].

2.9 Botnet

A botnet is a collection of internet-connected devices that attackers have compromised to control remotely without the owners' knowledge [25].

3. RESEARCH METHOD

This research uses a Linux-based server as the target for a UDP Flood DDoS attack simulation in a closed network. The researcher configured the server with SSH, HTTP, and DNS services to represent real operational conditions. The researcher monitored the server in real time using a live forensic approach.

The simulation uses one server and one client in a local network. The researcher used Ubuntu Server, tcpdump, Wireshark, and LOIC as the attack simulation tools. The research methodology refers to the NIST framework, which consists of the collection, examination, analysis, and reporting stages. The researcher used these stages to collect evidence, examine data, analyze attack patterns, and prepare reports and mitigation recommendations systematically.

3.1 Research Scenario

This research scenario is designed to simulate and analyze a UDP Flood Distributed Denial of Service attack and detect it using a digital forensic approach based on the National Institute of Standards and Technology framework. The researcher conducted the simulation in a fully controlled closed network environment to ensure security and data integrity during the experimental process.

The researcher divided the research implementation into three main phases, namely pre-incident, incident, and post-incident phases. These three phases represent systematic forensic investigation stages, starting from preparation, data collection and observation during the attack, to analysis and reporting of the results. The research scenario sequence is illustrated in Figure 2. This phase division helps ensure that each activity is performed systematically, supports evidence traceability, and improves the reliability of forensic findings.

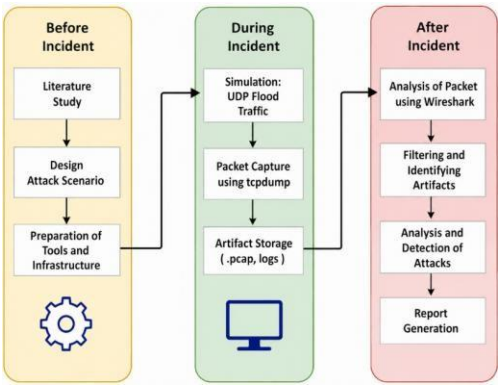

Figure 2 : Case Scenario

The pre-incident stage includes preparation before the attack simulation is conducted. The researcher conducted a literature review on UDP Flood DDoS attacks and digital forensic principles as the research foundation.

The researcher then designed the testing environment in a closed network consisting of a target server and an attacking client. The researcher configured the server with SSH, HTTP, and DNS services. The researcher also equipped the server with Wireshark and tcpdump to monitor traffic. The researcher prepared LOIC on the client side as the attack simulation tool. The researcher also enabled logging and monitoring to record activities during the research.

The incident stage is the phase in which the researcher conducts the UDP Flood attack simulation against the target server using LOIC. The attacker sends a large number of UDP packets to increase traffic load and disrupt service performance.

The researcher recorded network traffic using tcpdump and Wireshark during the attack. Tcpdump captured packets entering the server and stored them in .pcap format. The .pcap file contained information about source and destination IP addresses, ports, protocols, packet sizes, and transmission frequency. The researcher used the data as digital artifacts to analyze attack patterns in the post-incident stage based on the NIST method.

The researcher conducted the post-incident stage by analyzing the .pcap file using Wireshark to identify relevant digital artifacts. The researcher began the analysis by filtering packets based on the UDP protocol to separate suspicious traffic from normal traffic. The analyzed parameters included source and destination IP addresses, destination ports, packet sizes, and packet transmission frequency.

The researcher then compared the traffic results during the attack with normal conditions to identify anomaly patterns. These anomaly patterns included high packet intensity, source IP distribution, and traffic concentration on specific ports. The researcher documented all findings in a forensic report containing the chronology, digital artifacts, attack patterns, and security recommendations.

3.2 Stages of Forensic Acquisition

3.2.1 Pre-Acquisition

The pre-acquisition stage focuses on planning before digital evidence collection. This stage includes a literature review on UDP-based DDoS attacks and digital forensic methods, testing environment preparation, target server and attacking client configuration, and LOIC, Wireshark, and tcpdump installation. The logging system records activities during the research, and time synchronization between devices maintains consistency in the records. This stage also determines the digital artifacts for analysis, such as IP addresses, destination ports, timestamps, and packet transmission frequency.

3.2.2 Core Acquisition

The core acquisition stage collects digital evidence of the UDP Flood attack during the incident. The simulation sends a large number of packets to the target server, which increases the system load. Tcpdump and Wireshark record network traffic in

.pcap format. The .pcap file contains IP addresses, ports, timestamps, packet sizes, and transmission frequency. System logs also monitor system performance during the simulation. Structured storage organizes all collected data, and hashing validates the data to maintain the integrity of digital evidence.

4. RESULT AND DISCUSSION

This section presents the results and discussion of the UDP Flood DDoS attack simulation in a controlled network using a digital forensic approach based on the NIST framework. The analysis identifies attack characteristics and traffic patterns based on the obtained digital evidence.

The discussion follows a systematic structure covering needs analysis, simulation scenarios in the pre-incident, incident, and post-incident stages, and analysis based on the collection, examination, analysis, and reporting stages. This approach produces a structured, comprehensive, and scientifically valid investigation process.

4.1 Requirements Analysis

The researchers conducted a requirements analysis to ensure readiness for the UDP Flood attack simulation and digital forensic investigation in accordance with the NIST framework. The researchers identified requirements for hardware, software, network configuration, and forensic aspects.

The researcher used a single host computer to run the target server and attacker client virtual machines. The server ran Linux and hosted SSH, HTTP, and DNS services as standard services. The client ran Windows and executed LOIC for the attack simulation. The server used tcpdump to capture traffic, while the forensic analyst used Wireshark to analyze packets.

The network uses bridged adapter mode so that the server and client are on the same network segment. Researchers collected digital evidence in the form of .pcap files and verified their integrity using SHA-256 hashes. Researchers determined test parameters that included attack duration, number of packets, and destination ports. Researchers determined success indicators based on the system's ability to detect UDP traffic spikes and UDP Flood attack patterns.

4.2 Research Simulation

4.2.1 Pre Incident

The pre-incident phase is the preparation phase before the UDP Flood simulation, as shown in Figure 3.In this stage, the Linux server and attacker client are configured on the same network. The server runs SSH, HTTP, and DNS services, while the client uses LOIC as the attack simulation tool. The server is equipped with tcpdump and logging to record network activity. Before the attack is launched, traffic is observed as a baseline. This stage ensures that all components are ready for the simulation and evidence collection. This preparation supports accurate analysis and reliable forensic results. It minimizes configuration errors and provides a reference for comparing normal and attack traffic conditions during investigation and supports more accurate forensic interpretation results.

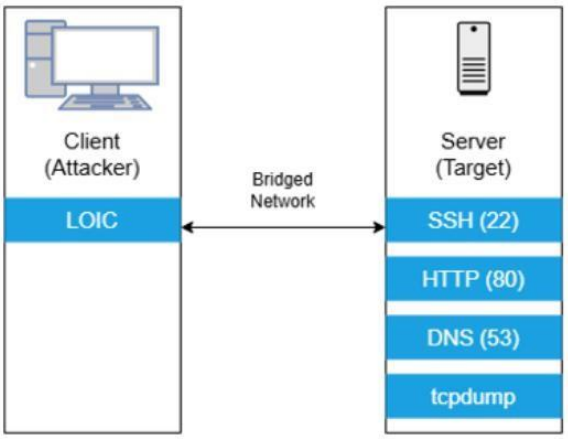

Figure 3 : Pre-Incident Scenario

4.2.2 Incident

The researcher conducted a UDP flood attack simulation against the target server during the incident phase, as shown in Figure 4. The attacker's client continuously sent a large number of UDP packets to overload the network and server services. The attacker's client ran LOIC with specified destination port and packet transmission intensity parameters.

The target server records network traffic in real time using tcpdump. The target server saves all packets into a .pcap file as digital evidence for forensic analysis. Researchers monitor system conditions to observe traffic spikes and potential service performance degradation. This phase generates data representing the actual conditions at the time the attack occurred.

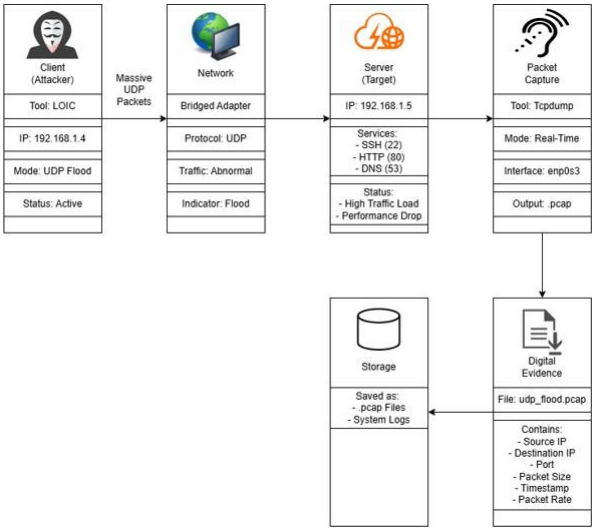

Figure 4 : Attack Implementation

4.2.3 Post Incident

The post-incident phase involved analyzing .pcap files using Wireshark to identify the characteristics of a UDP flood attack, as shown in Figure 5. The analysis began by filtering packets based on the UDP protocol so that the examination could focus on relevant traffic. The parameters examined included the source and destination IP addresses, port, packet size, time interval between packets, and payload. Analysis is also performed to identify abnormal patterns, such as a sudden surge of packets, high-intensity repeated transmissions, and traffic concentration on specific ports. The analysis results are used to confirm the presence of a UDP Flood attack based on the digital evidence obtained. All findings are then documented as part of the forensic investigation in accordance with the NIST framework stages.

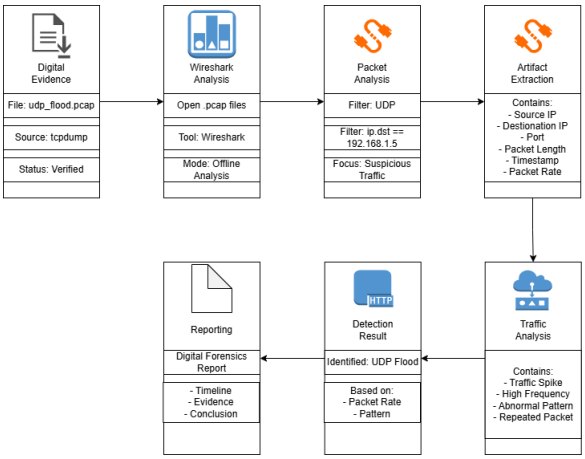

Figure 5 : Analysis Process

4.2.4 Roles in the Research Scenario

In this simulation, each component plays a role in representing a UDP Flood DDoS attack and supporting digital forensic investigations. The client acts as the attacker, sending large amounts of UDP traffic to the target server using LOIC. The server acts as the attack target, running SSH, HTTP, and DNS services, and recording network traffic in real-time using tcpdump in .pcap format. The forensic analyst examines the .pcap file using Wireshark by observing the source and destination IPs, ports, packet size, transmission frequency, and traffic patterns. The analyst also verifies the hash to maintain the integrity of the digital evidence and compiles an investigation report. The division of roles is shown in Table 1.

Table 1 : Roles in the Research Scenario

Component	Roles	Description
Client	Attacker	Sending large numbers of UDP packets using LOIC to overload the server
Server	Victim	Capturing attack traffic and running network services and logging using tcpdump
Forensics Analyzer	Investigator	Analyzing .pcap files using Wireshark and compiling an investigation report

The role division in Table 1 clarifies the function of each component in the UDP Flood DDoS simulation. This separation supports a structured forensic process, from attack traffic generation and evidence acquisition to .pcap analysis and report preparation. This arrangement helps maintain clear responsibilities, improve evidence traceability, reduce analysis errors, and support reliable conclusions during investigation.

4.3 Results and Discussion

4.3.1 Collection

The collection phase gathers digital evidence related to the UDP Flood DDoS attack. In this phase, network traffic is recorded using a packet capture tool, focusing on UDP packets

sent to the target server. The collected evidence becomes the basis for the examination and analysis phases according to the NIST method.

A. Digital Evidence Acquisition

Digital evidence was collected by recording the target server's network traffic during the UDP Flood simulation, as shown in Figure 6. The recording was performed in real time using the command tcpdump -i enp0s3 -w udp_flood.pcap. The -i parameter specifies the network interface, while -w saves the captured traffic into a .pcap file.

The resulting udp_flood.pcap file contains network activity data, including IP addresses, protocols, ports, packet sizes, timestamps, and payloads. This file was used as the main digital artifact for forensic examination and analysis of the UDP Flood DDoS attack.

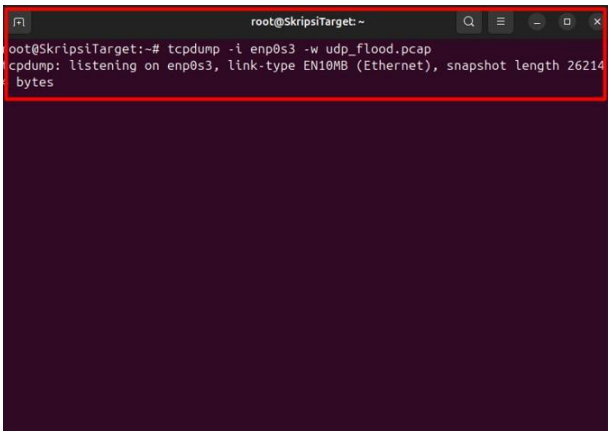

Figure 6: Data Collection

B. Verification of Evidence Integrity

The researcher verifies the integrity of the captured file after acquisition, as shown in Figure 7. The verification uses the SHA-256 algorithm with the sha256sum udp_flood.pcap command. The generated hash value serves as the file's digital identity and ensures that the digital evidence remains unchanged, valid, and reliable during the investigation process. This step strengthens the authenticity and credibility of the forensic analysis results.

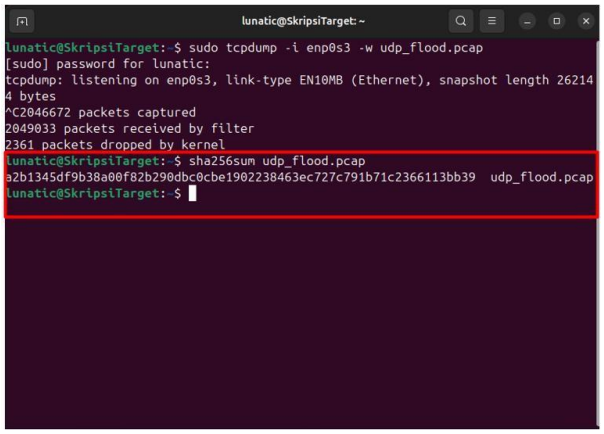

Figure 7: Verification Result

4.3.2 Examination

The examination phase involves reviewing the .pcap files obtained during the collection phase. This review is conducted to identify and filter data relevant to UDP flood DDoS attacks. The review focuses on the UDP protocol, source and destination IP addresses, ports, packet size, and packet transmission frequency.

A. Identification of Digital Evidence Files

The researcher conducted an initial analysis on the udp_flood.pcap file using the Statistics Summary feature in Wireshark, as shown in Figure 8. The results showed that the file contained 2,046,672 packets, with a recording duration of 26 seconds and an average rate of approximately 78,015 packets per second. The extremely high number of packets within a short period indicates abnormal traffic activity and suggests a UDP Flood attack.

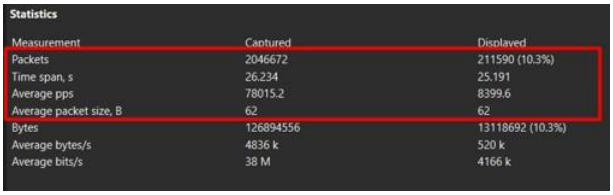

Figure 8 : File Summary

B. General Network Traffic Information

General traffic information was analyzed to obtain an overview of the data characteristics during the attack. Based on the analysis, the average packet size recorded was 62 bytes. The relatively small packet size, combined with the large volume of packets sent, is a common characteristic of UDP flood attacks, which overload networks and servers by sending packets at a high frequency.

C. Network Protocol Analysis

The protocol analysis was performed using the Protocol Hierarchy feature to identify the most dominant protocol in the capture file, as shown in Figure 9. The results of the analysis show that the network traffic is dominated by the UDP protocol. This dominance reinforces the indication that the recorded traffic is related to a UDP flood attack, since the UDP protocol is connectionless and can be exploited to send packets rapidly without an initial connection process.

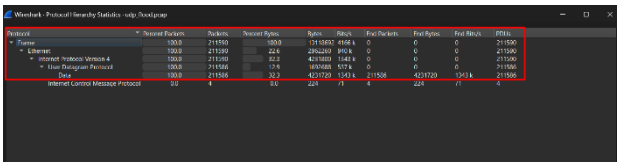

Figure 9 : Protocol Analysis

D. Identify the Source and Destination IP Addresses

IP addresses were identified using the Endpoints and Conversations features in Wireshark, as shown in Figure 10. The results of the analysis show that most UDP packets originate from IP address 192.168.1.4 and are sent to IP address 192.168.1.5. This pattern indicates that 192.168.1.4 acts as the source of the attack, while 192.168.1.5 is the target server receiving a large volume of traffic. This finding supports the identification of communication direction and attack source during the simulation.

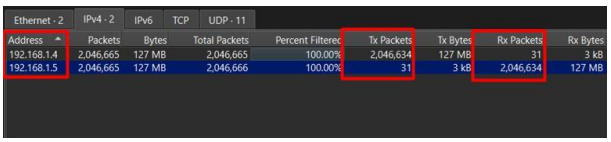

Figure 10 : Source and Destination IP

E. Analysis of Timestamps and Inter-Packet Intervals

The analysis of timestamps and inter-packet intervals was performed using the I/O Graph feature in Wireshark. The visualization results show that packets were sent densely and

continuously at very short intervals. The absence of significant pauses in packet transmission indicates a high-intensity traffic pattern that does not reflect normal network communication.

F. Packet Rate Analysis

Packet rate analysis is performed to measure the number of packets received within a time interval, as shown in Figure 11. With a 50-miliseconds interval, the graph shows that the number of packets is in the range of thousands per interval. When converted, the packet rate can reach tens of thousands of packets per second. A high and consistent packet rate is a strong indicator of flooding activity on the network. This condition indicates abnormal traffic behavior, which can reduce server performance and support the identification of a UDP Flood DDoS attack.

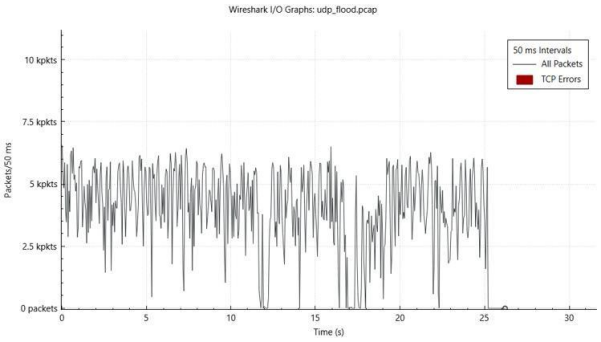

Figure 11 : I/O Graph Rate

G. Network Traffic Pattern Analysis

Network traffic patterns were observed using I/O Graph visualization, as shown in Figure 12. The results showed a continuous, dense, and repetitive packet distribution throughout the recording period. Although there were fluctuations at certain points, the traffic generally remained at a high level without significant interruptions. This condition indicates that packet transmission occurred intensively and consistently during the simulation.

The dense traffic pattern also shows that the network received a large number of packets in a short period of time. Such behavior differs from normal network communication, which usually has more varied and irregular traffic patterns. Therefore, the I/O Graph results provide an initial indication of abnormal network activity and become an important reference for identifying UDP Flood attack characteristics in the subsequent analysis phase.

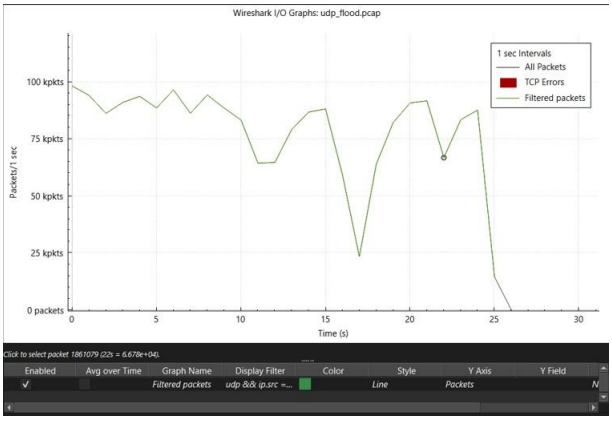

Figure 12 : Network Traffic Pattern

H. Packet Payload Analysis

The next examination stage analyzes the payload of captured UDP packets to identify transmitted data patterns. The researcher selected one UDP packet in Wireshark and observed the Payload section, which displays data in hexadecimal and ASCII formats. The result showed a simple repeated text payload, such as “UDP FLOOD ATTACK!!!!”. Indicates that the packets contain minimal data and supports the identification, as shown in in Figure 13.

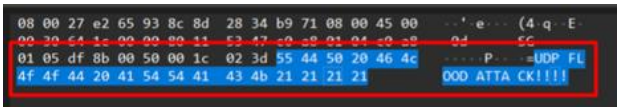

Figure 13 : Payload Data

4.3.3 Analysis

The analysis phase interprets the digital artifacts obtained during the examination process to determine whether the captured network activity corresponds to a UDP Flood attack. Several network parameters, including protocol type, traffic intensity, communication patterns, and the distribution of source and destination IP addresses, are correlated to identify attack characteristics and distinguish malicious activity from legitimate network traffic.

A. Analysis Based on Network Protocols

The examination results show that the udp_flood.pcap capture is dominated by UDP traffic, as shown in Figure 14. Since UDP is a connectionless protocol that does not require session establishment or acknowledgements, it enables attackers to transmit packets at a very high rate. Packet inspection also shows that a large volume of UDP packets was directed to port 80, which is commonly used for TCP based HTTP services. This abnormal protocol usage indicates that the attacker attempted to overwhelm the target server rather than establish legitimate communication, providing strong evidence of a UDP Flood attack.

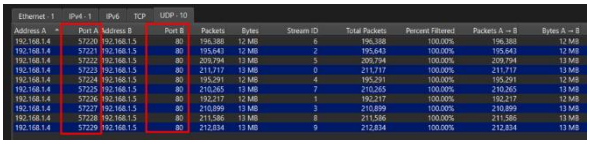

Figure 14 : Traffic Port

B. Network Traffic Intensity Analysis

The intensity analysis shows that the server received more than two million packets in approximately 26 seconds, averaging 78,000 packets per second. The I/O Graph visualization shows continuous and dense traffic, with an interval between packets of approximately 0.00008 seconds. These conditions indicate that packet transmission is occurring very rapidly without significant pauses, suggesting flooding activity that is overloading the server in a short period of time.

C. Network Traffic Pattern Analysis

Network traffic patterns were analyzed to determine the distribution of packets during the observation period. The visualization results show that packets were sent continuously, densely, and repeatedly throughout the recording. This pattern differs from normal traffic, which typically varies according to user activity. The consistency of high-intensity traffic indicates the presence of repeated automated packet transmission, consistent with the characteristics of a flooding attack.

Analysis of the Distribution of Sources and Destinations Wireshark Endpoints analysis showed the distribution of source and destination traffic during the attack. The results showed that IP address 192.168.1.4 generated high-volume traffic toward IP address 192.168.1.5. This one-way traffic pattern indicates that 192.168.1.4 acted as the attacker and

192.168.1.5 acted as the target server. The concentrated traffic distribution supports the indication of a UDP Flood attack rather than normal network communication.

D. Conclusions of the Analysis

Based on the correlation of all parameters, the network activity observed exhibits the characteristics of a UDP Flood attack. This is indicated by the dominance of the UDP protocol, extremely high traffic intensity, tight packet transmission intervals, continuous and repetitive traffic patterns, and traffic distribution concentrated from a single source IP address to a single destination IP address. These findings serve as the basis for the reporting phase to systematically compile a timeline, digital evidence, and mitigation recommendations.

4.3.4 Reporting

Based on the simulation results, the attack occurred when the client continuously sent UDP packets to the target server using LOIC. The traffic was directed to port 80, causing a spike in network traffic within approximately 26 seconds. The attack activity was recorded using tcpdump on the target server. All incoming packets were saved in a .pcap file and used as primary digital evidence during the forensic examination and analysis phase.

A. Chronology of Events

Based on the simulation results, the attack commenced when the client sent continuous UDP packets to the target server using LOIC. The traffic was directed to port 80, resulting in a network traffic surge within approximately 26 seconds. The system performed traffic recording using tcpdump on the target server. All incoming packets during the attack were stored in a .pcap file, which subsequently served as the primary digital evidence for the forensic examination and analysis process.

B. Digital Evidence

Based on the completed acquisition process, digital evidence was obtained in the form of network traffic capture files. This evidence contains records of network activity during the simulated attack, including packet details such as source and destination IP addresses, protocols, ports, packet sizes, and timestamps. The specific details are presented in Table 2.

Table 2: Digital Evidence

Analysis Parameter	Value
Packet Total	2.046.672 packets
Recording Time	26 seconds
Packet Rate	±78.015 packets/seconds
Packet Interval	±0.00008 seconds
Dominant Protocol	UDP (>90%)
Packet Size	62 byte
IP Source	192.168.1.4
IP Destination	192.168.1.5
Port Destination	80

Based on Table 2, the total number of packets, dominance of the UDP protocol, high packet rate, small packet size, and repeated traffic from one source IP to the target server indicate abnormal network activity during the simulation process. These characteristics support the identification of a UDP Flood DDoS attack and strengthen the validity of the captured .pcap file as digital evidence for further forensic analysis.

C. Main Findings

The examination and analysis stages showed that the UDP protocol dominated the network traffic with very high intensity, reaching tens of thousands of packets per second. The very small interval between packets showed that packet transmission occurred rapidly, repeatedly, and continuously without significant pauses. The attack traffic was concentrated from the source IP address 192.168.1.4 to the destination IP address 192.168.1.5, with packets directed to port 80 as the HTTP service port. The repeated simple payload in several packets further strengthened the indication of a UDP Flood attack.

D. Activity Identification

The overall findings showed that the network activity had the characteristics of a connectionless protocol, continuous transmission of large numbers of packets, absence of normal two-way communication, and centralized one-way traffic distribution. These characteristics were consistent with the pattern of a UDP Flood DDoS attack, which aims to overload the target system by sending massive UDP packets and disrupting service performance.

E. Attack Handling and Mitigation Strategies

Firewall rate limiting can mitigate UDP Flood attacks by restricting UDP traffic through tools such as iptables. Traffic filtering can also control packets based on IP addresses, ports, and protocols. Intrusion Detection Systems, continuous monitoring, and logging can detect attacks in real time. System hardening can strengthen server security against repeated attacks. Incident response can reduce the impact of attacks by blocking attack sources and isolating affected systems. Post-incident evaluation can improve system resilience against future attacks.

5. CONCLUSIONS

The research results show that UDP Flood DDoS attacks have several characteristics, including UDP traffic domination, high-intensity packet transmission, relatively small packet sizes, and continuous transmission patterns without significant pauses, which increase the load on the target server and can potentially reduce service performance. The implementation of the NIST framework through the collection, examination, analysis, and reporting stages supports a systematic investigation process, while the live forensic approach enables evidence collection while the system is still running. Tcpdump and Wireshark support the recording, examination, and analysis of network traffic. Attack mitigation can be performed through UDP traffic rate limiting, firewall implementation based on IP addresses, ports, and protocols, and IDS usage for real-time detection. Monitoring, logging, system hardening, and incident response, such as blocking attack sources and limiting traffic, are also required to reduce attack impacts and improve system resilience.

6. REFERENCES

[1] M. Adam, E. I. Alwi, and I. As'ad, "Analisis Forensik terhadap Serangan DDoS Ping of Death pada Server," 2022.

[2] A. A. R. Nisa, A. D. Wijayanto, A. P. J. Priana, and A. Setiawan, "Analisis Log Server untuk mendeteksi Serang DDoS pada Keamaan Jaringan di Website," *J. Internet Softw. Eng.*, vol. 1, no. 3, p. 17, 2024, doi: 10.47134/pjise.v1i3.2612.

[3] A. Nurudin and I. Mubariq, "Intrusion Prevention System to Protect Dos Attack," vol. 2, no. 1, pp. 49–59, 2024, doi: 10.62885/improsci.v2i1.441.

[4] A. R. P. Anisa, "Literature Review Jurnal Memahami Faktor-Faktor Yang Mempengaruhi Popularitas Windows Dibandingkan Linux," *Jamastika*, vol. 3, no. April, pp. 28–33, 2024.

[5] H. Alfidzar and B. P. Zen, "Implementasi HoneyPy Dengan Malicious Traffic Detection System (Maltrail) Menggunakan Analisis Deskriptif Guna Untuk Mendeteksi Serangan DDOS Pada Server," *J. Informatics, Inf. Syst. Softw. Eng. Appl.*, vol. 4, no. 2,

pp. 32–45, 2022, doi: 10.20895/inista.v4i2.534.

[6] M. M. Shafi, A. H. Lashkari, V. Rodriguez, and R. Nevo, "Toward Generating a New Cloud-Based Distributed Denial of Service (DDoS) Dataset and Cloud Intrusion Traffic Characterization," *Inf.*, vol. 15, no. 4, 2024, doi: 10.3390/info15040195.

[7] A. H. Jatmika and A. Zubaidi, "Analisis Perbandingan Performa Mode Trafik Tcp Dan Udp Menggunakan Protokol Routing Aodv Dan Dsr Pada Jaringan Manet(Comparisonal Analysis of TCP And UDP Traffic Mode Performance Using AODV And DSR Routing Protocols On Manet Networks)," *J. Teknol. Informasi, Komput. dan Apl.*, vol. 4, no. 1, pp. 21–26, 2022.

[8] W. Warcita, "Deteksi Serangan DDoS UDP Flood pada Jaringan IoT Menggunakan Recurrent Neural Network (RNN)," no. 2504, pp. 1–9, 2024.

[9] F. Nova, M. D. Pratama, and D. Prayama, "Wazuh sebagai Log Event Management dan Deteksi Celah Keamanan pada Server dari Serangan Dos," *JITSI J. Ilm. Teknol. Sist. Inf.*, vol. 3, no. 1, pp. 1–7, 2022, doi: 10.30630/jitsi.3.1.59.

[10] K. F. I. Ilham, E. I. Alwi, and F. Fattah, "Penerapan dan Analisis Network Security Snort Menggunakan Intrusion Detection System pada Serangan UDP Flood," *INFORMAL Informatics J.*, vol. 8, no. 1, p. 94, 2023, doi: 10.19184/isj.v8i1.34003.

[11] H. W. Aripradono, "JTIM : Jurnal Teknologi Informasi dan Multimedia Perbandingan Support Vector Machine, Random Forest Classi- fier, dan K-Nearest Neighbour dalam Pendeteksian Anomali," vol. 7, no. 1, pp. 23–33, 2025.

[12] W. Haniyah, M. C. Hidayat, Z. F. I. Putra, V. A. Pertama, and A. Setiawan, "Simulasi Serangan Denial of Service (DoS) menggunakan Hping3 melalui Kali Linux," *J. Internet Softw. Eng.*, vol. 1, no. 2, p. 8, 2024, doi: 10.47134/pjise.v1i2.2654.

[13] W. Y. Sulistyo, S. A. Pratiwi, M. Haedar, and Z. Hidayatullah, "Analisis Forensik Citra di Platform X Menggunakan Metode Digital Forensic Research Workshop (DFRWS)," vol. 8, pp. 10–20, 2025.

[14] A. M. Kurniawan, D. G. Purnama, and B. Al Ghiffari, "Analisis Keamanan Jaringan Pada Rumah Menggunakan Metode Nist," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 2, pp. 1659–1664, 2024, doi: 10.36040/jati.v8i2.9071.

[15] A. Syahputra, Y. Yuhandri, and S. Arlis, "Jurnal KomtekInfo Penerapan Metode Live Forensik untuk Analisis Serangan DoS pada Router Mikrotik," vol. 11, no. 4, pp.247–255, 2024, doi: 10.35134/komtekinfo.v11i4.

[16] M. Ilman Aqilaa, D. Firdaus, and N. Naofal, "Identifikasi Serangan Lowrate Distributed Denial Of Services Dalam Jaringan Dengan Menggunakan Algoritma Adaboost," *Simpatik J. Sist. Inf. dan Inform.*, vol. 3, no. 1, pp. 34–41, 2023, doi: 10.31294/simpatik.v3i1.1829.

[17] S. E. Prasetyo, H. Haeruddin, and K. Ariesryo, "Website Security System from Denial of Service attacks, SQL Injection, Cross Site Scripting using Web Application Firewall," *Antivirus J. Ilm. Tek. Inform.*, vol. 18, no. 1, pp. 27–36, 2024, doi: 10.35457/antivirus.v18i1.3339.

[18] S. R. N. Suwaryo, I. Nawangsih, "Deteksi Serangan Pada Intrusion Detection System (Ids) Untuk Klasifikasi Serangan Dengan Algoritma Naïve Bayes, C.45 Dan K-Nn Dalam Meminimalisasi Resiko Terhadap Pengguna," 2021.

[19] D. R. Anggraeni and M. Salsabila, "Analisis Yuridis Peran Digital Forensik Dalam Pembuktian Tindak Pidana di Indonesia," *Media Huk. Indones.*, vol. 2, no. 2, pp. 593–600, 2024.

[20] R. J. A. Saputra, "Analisis Sistem Inventori dengan Metode National Institute of Standards and Technology NIST Studi Kasus Lotte Mart Grosir Palembang," 2023, [Online]. Available: http://repository.binadarma.ac.id/id/eprint/9263

[21] O. Aljumaiah, W. Jiang, S. R. Addula, and M. A. Almaiah, "Analyzing Cybersecurity Risks and Threats in IT Infrastructure based on NIST Framework," vol. 2025, no. 2, 2025.

[22] M. R. H. Tambunan and S. N. Neyman, "Implementasi Firewall pada Linux untuk Pencegahan Dari Serangan DoS," *J. Technol. Syst. Inf.*, vol. 1, no. 4, p. 10, 2024, doi: 10.47134/jtsi.v1i4.2648.

[23] A. B. Pratomo, "Pengembangan Sistem Firewall Pada Jaringan Komputer Berbasis Mikrotik Routeros," *Bull. Netw. Eng. Informatics*, vol. 1, no. 2, p. 51, 2023, doi: 10.59688/bufnets.v1i2.10.

[24] M. Syaffiq, A. Malek, and A. R. Amran, "A Study of Packet Sniffing as an Imperative Security Solution in Cybersecurity," *J. Eng. Technol.*, vol. 9, no. 1, pp. 96–101, 2021.

[25] W. Yunanri, "Analisis Serangan Botnet pada Website Aplikasi Facebook dengan Menggunakan Metode National Institute of Standards and Technology (NIST)," vol. 6, no. 2, 2024.