

(CIS-542) Digital Forensics Project Report

Network Pattern Analyzer: Visualizing Network Traffic Dynamics through Heatmaps using Wireshark

Group 11:

Sindhuja baikadi (1)

02128756

Table of Contents

S.No	Contents	Page No
1.	Introduction	3
2.	Problem Statement	3
3.	Data Set Overview	4
4.	Softwares used	5
5.	Methodology	5
6.	Results and Outcomes	8
7.	Future Scope	8
8.	Limitations	9
9.	Advantages	10
10.	Conclusion	10
11.	References	10

1. Introduction

The realm of digital forensics is essential for understanding and analyzing the complexities of network interactions, particularly in our increasingly digital world where cybersecurity threats loom large. In this context, our project introduces the "Network Pattern Analyzer," an advanced tool developed to dissect and visualize network traffic to enhance digital security measures. This tool employs Python to scrutinize data captured in PCAP/PCAPNG formats, leveraging the robust features of Wireshark for in-depth network analysis.

Wireshark, a renowned network protocol analyzer, plays a pivotal role in this project, enabling the capture and examination of real-time data flowing across networks. This capability is crucial for professionals in network management, cybersecurity, and forensic analysis, as it aids in identifying anomalies, understanding normal network behavior, and investigating security breaches.

The primary innovation of our project lies in its ability to transform complex network data into intuitive visual representations specifically, heatmaps. These heatmaps provide a clear and immediate visual summary of the data exchanges between network nodes, highlighting potential areas of concern such as unusual data flow volumes or unexpected connections, which are critical for both troubleshooting and forensic investigations.

In summary, the Network Pattern Analyzer is not just a tool but a step forward in digital forensic technology, offering detailed insights and a new layer of security to network management through its advanced analytical capabilities and user-friendly visual outputs.

2. Problem Statement

In the domain of network management and cybersecurity, professionals continually face the challenge of effectively monitoring and analyzing network traffic to detect anomalies, manage performance, and enhance security measures. Traditional tools like Wireshark provide detailed packet capture and analysis capabilities. Still, they require extensive knowledge of network protocols and operations and can be overwhelming due to the sheer volume of data they generate.

Moreover, the complexity of modern network architectures and the prevalence of sophisticated cyber threats necessitate advanced tools that not only capture and filter data but also present it in an easily interpretable format that aids in quick decision-making and effective problem-solving. There is a critical need for a tool that simplifies the analysis of network traffic by visualizing data interactions in a comprehensive yet straightforward manner.

The Network Pattern Analyzer aims to address these issues by offering a solution that:

1. Automates the analysis of network traffic using PCAP/PCAPNG files to identify patterns and anomalies quickly.
2. Provides a heatmap visualization of network data flow, making it easier to spot high-traffic areas or unusual communication patterns that may indicate security risks or network failures.
3. Enhances the capability of network professionals to perform detailed forensic analysis with less dependency on in-depth technical knowledge of network protocols.

4. Bridges the gap between complex data capture and user-friendly analysis, enabling more effective network management and security practices.

This tool is designed to empower network administrators, security professionals, and forensic analysts by streamlining the process of network data analysis and enhancing the visibility of network activities, ultimately improving the security and efficiency of network operations.

3. Dataset Overview

The Network Pattern Analyzer utilizes network traffic data captured in PCAP (Packet Capture) and PCAPNG (Packet Capture Next Generation) formats. These datasets are essential for conducting detailed network analysis and are generated by network monitoring tools like Wireshark. Here is an overview of the characteristics and contents of these datasets:

1. Data Format:

PCAP: The original packet capture format that includes a global header followed by packet records. Each record has a packet header and packet data, detailing when the packet was captured, its size, and the actual bytes that make up the packet.

PCAPNG: An enhanced version of PCAP, offering a more extensible format to include additional information such as interface information, annotations, and options for better analysis and understanding of the captured data.

2. Content of the Data:

Timestamps: Each packet in the dataset includes a timestamp indicating when the packet was captured, which is crucial for analyzing the sequence and timing of network communications.

Packet Header Information: Includes source and destination addresses (IP), ports, protocol type (e.g., TCP, UDP), and other relevant networking layers' information.

Payload Data: The actual data carried within the packet, which can include various types of information depending on the network protocol used and the nature of the traffic (e.g., HTTP requests, DNS queries, file transfers).

3. Usage in Analysis:

Traffic Volume Analysis: By examining the sizes and frequency of packets, analysts can identify high-traffic periods or anomalies in data flow.

Endpoint Communication: Analysis of source and destination addresses helps in mapping the network structure and detecting potential unauthorized accesses or internal threats.

Protocol Usage: Identifying the protocols used across the network can help in understanding the network's operational context and pinpointing unusual protocol activity that might indicate malicious operations.

4. Challenges in Handling the Data:

Volume and Velocity: PCAP/PCAPNG files can become extremely large in high-traffic networks, requiring efficient processing tools to handle and analyze the data effectively.

Security and Privacy: The payload may contain sensitive information, necessitating careful handling to ensure privacy and compliance with data protection regulations.

Complexity of Analysis: The depth of data provided can be overwhelming, requiring sophisticated tools and techniques for meaningful analysis.

This dataset is pivotal for the Network Pattern Analyzer as it allows the tool to generate detailed heatmaps of network traffic, helping to visualize communication patterns and identify anomalies effectively. The data is the foundation for all subsequent analyses, making it a crucial asset for network diagnostics and forensic investigations.

4. Softwares used:

The Network Pattern Analyzer integrates various software tools and libraries to facilitate its functionality in analyzing network traffic and generating visual heatmaps.

1. Wireshark: Used for the initial capture of network traffic and basic analysis before data is imported into the Network Pattern Analyzer.
2. Python: Handles the processing of PCAP/PCAPNG files, executes data analysis, generates heatmaps, and provides an interface for user interaction.
3. Pyshark: Used to parse the PCAP/PCAPNG files within the Python environment, enabling the extraction of necessary attributes like IP addresses, protocols, and timestamps.
4. Pandas: Employs data frames to organize, filter, and analyze large sets of network data efficiently.
5. NumPy: Used for handling large volumes of data and performing numerical operations needed in traffic analysis.
6. Matplotlib: Generates the visual representations of network traffic, specifically heatmaps, to depict the intensity and pattern of network communications visually.
7. Seaborn: Enhances the visualization capabilities of matplotlib, making the generated heatmaps more intuitive and informative for users.
8. Jupyter Notebook: Used as an interactive computational environment for the development, documentation, and execution of the Python codebase for the Network Pattern Analyzer.

These software components collectively provide a robust framework for the Network Pattern Analyzer, enabling comprehensive network analysis, efficient data handling, and effective visualization of network traffic patterns.

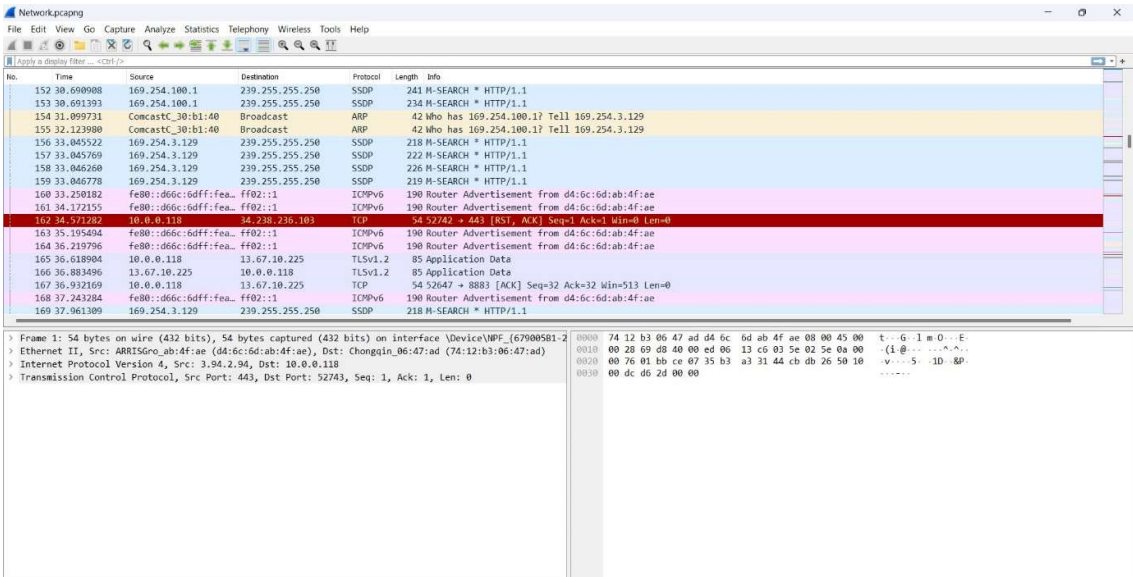
5. Methodology

The Network Pattern Analyzer utilizes a structured approach to analyze network traffic data captured in PCAP/PCAPNG files. This methodology is designed to extract meaningful insights from complex data sets and visualize the results effectively.

1. Data Collection:

Tool Used: Wireshark is employed to capture network traffic data. This tool collects all the packets transmitted over the network during the monitoring period.

Data Captured: The collected data includes packets with detailed information such as timestamps, source and destination IP addresses, protocol types, and payload data.



No.	Time	Source	Destination	Protocol	Length	Info
152	30.690908	169.254.100.1	239.255.255.250	SSDP	241	M-SEARCH * HTTP/1.1
153	30.691393	169.254.100.1	239.255.255.250	SSDP	234	M-SEARCH * HTTP/1.1
154	31.099731	ConcastC_30:b1:40	Broadcast	ARP	42	Who has 169.254.100.1? Tell 169.254.3.129
155	32.123900	ConcastC_30:b1:40	Broadcast	ARP	42	Who has 169.254.100.1? Tell 169.254.3.129
156	33.045522	169.254.3.129	239.255.255.250	SSDP	218	M-SEARCH * HTTP/1.1
157	33.045769	169.254.3.129	239.255.255.250	SSDP	222	M-SEARCH * HTTP/1.1
158	33.046260	169.254.3.129	239.255.255.250	SSDP	226	M-SEARCH * HTTP/1.1
159	33.046778	169.254.3.129	239.255.255.250	SSDP	219	M-SEARCH * HTTP/1.1
160	33.250182	fe80::d66c:6dff:fea::ff02::1		ICMPv6	190	Router Advertisement from d4:6c:6d:ab:4f:ae
161	34.172155	fe80::d66c:6dff:fea::ff02::1		ICMPv6	190	Router Advertisement from d4:6c:6d:ab:4f:ae
162	34.571282	10.0.0.118	34.238.236.103	TCP	54	52742 → 443 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
163	35.195494	fe80::d66c:6dff:fea::ff02::1		ICMPv6	190	Router Advertisement from d4:6c:6d:ab:4f:ae
164	36.219796	fe80::d66c:6dff:fea::ff02::1		ICMPv6	190	Router Advertisement from d4:6c:6d:ab:4f:ae
165	36.018904	10.0.0.118	13.67.10.225	TLSv1.2	85	Application Data
166	36.083496	13.67.10.225	10.0.0.118	TLSv1.2	85	Application Data
167	36.932169	10.0.0.118	13.67.10.225	TCP	54	52647 → 8883 [ACK] Seq=32 Ack=32 Win=513 Len=0
168	37.243284	fe80::d66c:6dff:fea::ff02::1		ICMPv6	190	Router Advertisement from d4:6c:6d:ab:4f:ae
169	37.961309	169.254.3.129	239.255.255.250	SSDP	218	M-SEARCH * HTTP/1.1

2. Data Parsing and Extraction:

Tool Used: Pyshark, a Python wrapper for Wireshark, is utilized to parse the PCAP/PCAPNG files.

Extraction: Relevant data attributes such as timestamps, IP addresses, packet lengths, and protocol types are extracted from each packet for further analysis.

3. Data Processing:

Tool Used: Python, along with libraries like Pandas and NumPy, is used for data processing.

Operations Performed: The extracted data is processed to clean, organize, and transform it into a structured format suitable for analysis. This may involve handling missing data, filtering out irrelevant packets, and aggregating data based on certain criteria such as time intervals or IP pairs.

4. Traffic Analysis:

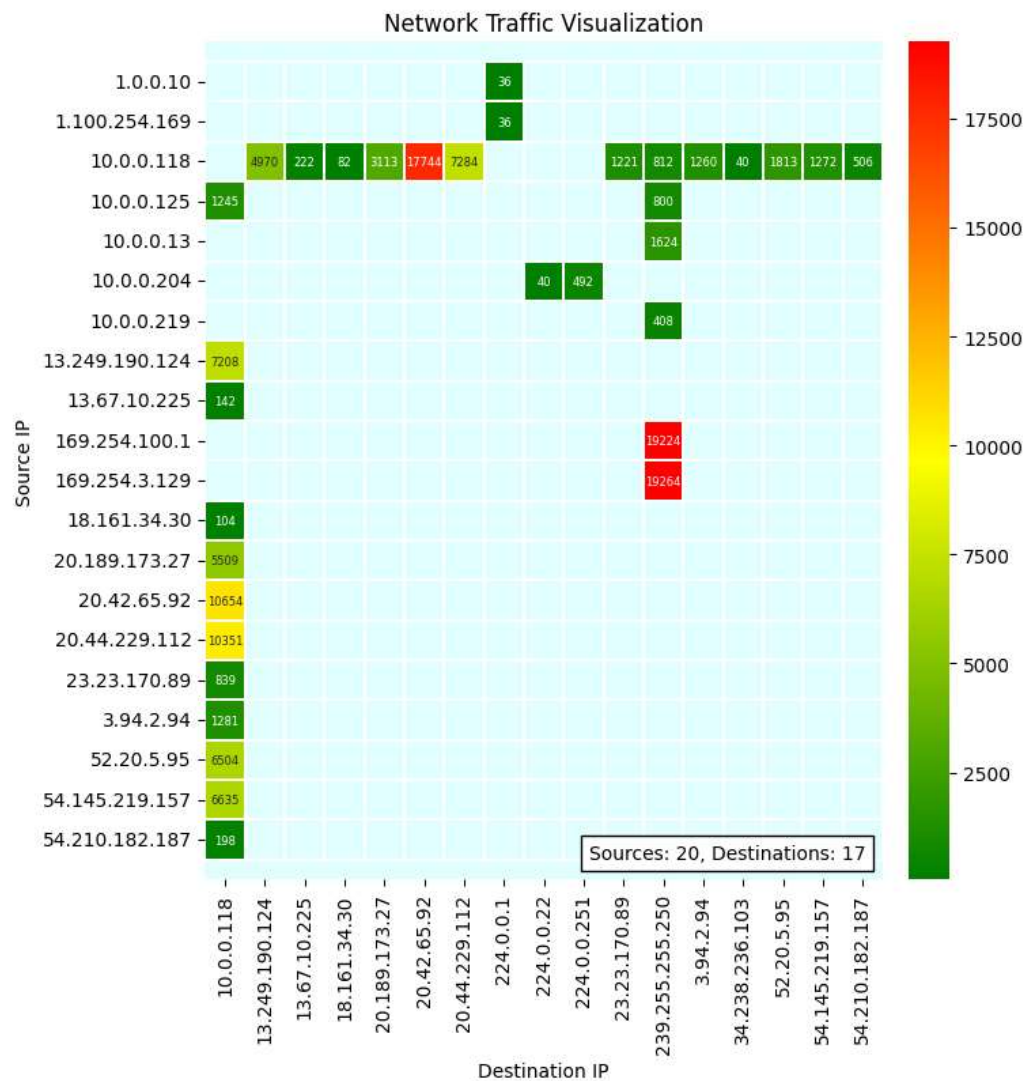
Analysis Techniques: Statistical analysis and pattern recognition techniques are applied to the processed data to identify common patterns, anomalies, or specific characteristics of the network traffic.

Purpose: The objective is to understand the normal operational baseline of the network and identify deviations that might indicate issues like network failures, unauthorized access, or potential security threats.

5. Visualization:

Tools Used: Matplotlib and Seaborn are used for creating visualizations.

Heatmaps: One of the primary outputs is a heatmap, which visually represents the intensity of data transfer between different network nodes. The color intensity in the heatmap correlates with the volume of traffic, providing an intuitive view of network activity patterns.



6. Interpretation and Reporting:

Process: The visualizations and analysis results are interpreted to make informed conclusions about the network's behavior.

Reporting: Findings are documented in a report or presented through an interactive dashboard, providing actionable insights to network administrators or security professionals.

7. Feedback and Iteration:

Continuous Improvement: Based on feedback from users and the initial outcomes, the methodology and the tool itself are refined and improved. Adjustments may include tuning the data collection settings, modifying the analysis parameters, or enhancing the visualization features.

This methodology ensures that the Network Pattern Analyzer not only provides detailed and accurate analysis of network traffic but also presents the data in a user-friendly manner that aids in quick decision-making and effective network management.

6. Results and Outcomes

The Network Pattern Analyzer offers crucial insights for managing network traffic, vital for network administrators, security analysts, and digital forensics professionals.

1. **Heatmaps of Network Traffic:** These visual representations highlight data volumes transferred between nodes, aiding in identifying traffic patterns and anomalies.
2. **Statistical Analysis Reports:** Provide data on traffic patterns, including load averages and peak times, useful for network planning and forensic documentation.
3. **Anomaly Detection Alerts:** Automated alerts flag deviations from normal traffic patterns, helping to swiftly address potential security threats or network issues.
4. **Protocol and Service Analysis:** Analyzes traffic by protocol and service, enhancing network infrastructure optimization and security.
5. **Network Forensics Data:** Captures detailed logs of network transactions, crucial for investigating security incidents and tracing attack origins.
6. **Performance Metrics and Trends:** Offers a longitudinal analysis of network performance, supporting predictive planning and operational adjustments.

These functionalities collectively enhance network monitoring, management, and security, providing essential insights for maintaining network integrity in a complex digital landscape.

7. Future Scope:

The Network Pattern Analyzer, while robust in its current form, offers a wide array of opportunities for future enhancements and expansion. These improvements can extend its capabilities, improve user experience, and address emerging challenges in network management and security.

1. Integration with Machine Learning Models:

Purpose: To develop predictive models that can analyze patterns over time to predict future network conditions or identify anomalous behavior more accurately.

Impact: This could lead to more proactive network management, with the ability to alert administrators about potential issues before they become critical.

2. Real-time Data Processing and Analysis:

Purpose: To enable the analyzer to process data in real-time rather than relying solely on historical data.

Impact: Real-time analysis can dramatically decrease the time to detect and respond to network threats or failures, enhancing security and operational efficiency.

3. Enhanced Encryption and Privacy Features:

Purpose: To implement advanced encryption measures for data in transit and at rest, and to provide more robust privacy controls.

Impact: This would address data security and privacy concerns, making the tool suitable for use in more regulated industries like finance and healthcare.

4. Cloud-Based Analytics and Scalability:

Purpose: To migrate the tool to a cloud-based platform, allowing for better scalability and remote analysis capabilities.

Impact: Cloud-based operations could support larger datasets and more complex analysis, as well as provide flexibility and accessibility for users across multiple locations.

5. Advanced Visualization Techniques:

Purpose: To incorporate more sophisticated visualization tools and dashboards that provide deeper insights and customizable views.

Impact: Improved visualizations can help in better understanding the data, tailoring the user interface to specific needs, and aiding in quicker decision-making.

6. Expanded Protocol Support and Analysis:

Purpose: To extend the range of protocols analyzed, especially newer or less common ones, to keep pace with the evolving network environment.

Impact: This would ensure the tool remains relevant and capable of analyzing modern network traffic, including IoT device communications and other emerging technologies.

7. Automated Response Capabilities:

Purpose: To develop features that not only detect issues but also initiate predefined responses automatically.

Impact: Automation can help in managing routine tasks or responding to incidents faster, reducing the workload on network administrators and enhancing network security.

8. Interoperability with Other IT Management Tools:

Purpose: To ensure seamless integration with existing network management and security tools to provide a more comprehensive IT management suite.

Impact: Interoperability can simplify IT operations, reduce the learning curve for new tools, and provide a unified approach to managing IT infrastructure.

8. Limitations:

1. Complexity of Setup and Use: Although the Network Pattern Analyzer provides powerful analysis capabilities, it requires a certain level of expertise in network protocols and analysis techniques, which might be challenging for novice users.

2. Resource Intensity: Processing large volumes of network data, especially in high-traffic environments, can be resource-intensive and may require significant computational power, potentially limiting its use in smaller or resource-constrained setups.

3. **Data Privacy and Security:** Handling sensitive network data raises concerns about privacy and security. Ensuring data is securely stored and processed, and complies with privacy regulations, can be complex and costly.
4. **Real-Time Analysis Limitations:** Currently, the tool focuses more on the analysis of historical data, and real-time capabilities are limited, which might not suffice for environments where immediate response is critical.
5. **Dependence on External Tools:** The tool relies heavily on external software like Wireshark and Python libraries, which might pose integration challenges and dependency risks if these tools are updated or their development is discontinued.

9. Advantages:

1. **Detailed Network Insights:** The tool provides deep insights into network traffic patterns, helping organizations to better understand their network activities and detect anomalies effectively.
2. **Visualization Capabilities:** With advanced visualization options like heatmaps, the tool helps in quickly identifying traffic patterns and potential issues, enhancing the decision-making process.
3. **Customizability:** Thanks to its Python-based architecture, the tool is highly customizable, allowing users to adapt it to their specific needs by modifying the code or integrating additional Python libraries.
4. **Educational Value:** It serves as a valuable educational tool for teaching network analysis and security, providing a practical platform for students to learn about real-world network traffic scenarios.

10. Conclusion:

The Network Pattern Analyzer stands out as a significant advancement in the field of network management and digital forensics. By leveraging the strengths of Wireshark and Python, it offers comprehensive analysis and visualization of network traffic, providing valuable insights into network performance and security. While there are challenges related to complexity, resource demands, and data privacy, the benefits of enhanced visibility and detailed network analysis outweigh these limitations. As network environments continue to grow in complexity and scale, tools like the Network Pattern Analyzer will become increasingly crucial in ensuring network integrity and security. The planned future enhancements and expansion into real-time analysis and cloud-based capabilities further promise to broaden its applicability and effectiveness, making it an indispensable tool for network administrators and security professionals in the digital age.

11. References:

1. <https://www.comptia.org/content/articles/what-is-wireshark-and-how-to-use-it#school>
2. <https://www.wireshark.org/#>
3. <https://www.comparitech.com/net-admin/wireshark-cheat-sheet/>