https://doi.org/10.26682/csjuod.2023.26.2.43

Journal of University of Duhok., Vol. 26, No.2 (Pure and Engineering Sciences), Pp 464 - 472, 2023
4th International Conference on Recent Innovations in Engineering (ICRIE 2023) (Special issue)

A NOVEL NETWORK ANALYZING TOOL (NNAT) FOR COMPLEX NETWORK INFORMATION VISUALIZATION

ASMA **B**ARZHAWAND **I**SMAIL, **R**OZHGAR **R**ZGAR **S**ABR and **S**ALAR **K**HEDER **S**HAIKHAH
Erbil Technical Engineering College, Erbil Polytechnic University, Kurdistan Region- Iraq

(Accepted for Publication: November 27, 2023)

ABSTRACT

Analyzing and monitoring networks can be difficult, especially when large volumes of packets fly back and forth between devices. Therefore, visual network analysis provides a solution by allowing users to visualize and interpret complex network data more intuitively and interactively. Traditional tools have limited visualization capabilities and mostly represent data in a tabular format that is difficult to analyze and time-consuming for the users. In this paper, a novel network analyzing tool (NNAT) is proposed to integrate packet sniffing, visual analysis of packets, and live network scanning within the same tool. Also, presenting the analyzed data via different interfaces with the capability of filtering specific information. The function variety of the proposed tool reduces the number of tools to be used by network admins. Also, the flexibility of the proposed tool in the data visualization and presentation with proper diagrams and drawings makes it to be user-friendly, in additional to its inherent outperformance explore and analyze large datasets compared to traditional tools.

KEYWORDS: Information visualization; network security; network data analyzing; network management; network sniffing

1. INTRODUCTION

Large amounts of traffic are generated in a network, making it challenging for the network engineers and cybersecurity analysts to interpret and analyze this data. As visualization is an efficient way of analyzing large datasets [2], researchers have proposed various visualization tools to address this issue [1]. Network administrators would need to analyze raw data collected in packet captures to detect anomalies within a network. However, analyzing packet captures can be challenging due to the vast volume of packets transmitted in computer networks. Examining each packet is a time-consuming process. Visual analysis has become crucial because data size has increased in recent years [4].

However, research community has proposed several tools for data analyzing, but most of them focused on one side of the process. To the best of our knowledge, there is no comprehensive tool covers end-to-end processes of data capturing, then analyzing the content of the captured data, and finally present to the users efficiently and user friendly. Traditional tools such as Wireshark [3] have limited visualization capabilities. Whereas Recent proposed tools that support visualization do not support actions needed after or before data analysis, such as packet sniffing.

In [5], a visual network analyzing tool is proposed to provide awareness about how user data is shared over the web. The tool supports packet sniffing and collect users' packets and demonstrates them through numerous visuals. What limits the tool is specifically designed for social web pages. In [6], a web application tool was designed to facilitate rapid and

asma.bismail11@gmail.com
rozhgar.rzgar.sabr@gmail.com
salar.shaikhah@epu.edu.iq

comprehensive event detection in network security through the integration of visual analytics techniques and collaborative functionalities. It enables its users to contribute by submitting feedback, comments, or re-analysis of events for enhanced collaboration. A web-based visual analyzing tool was proposed to analyze PCAP files in [7]. The tool shows protocol information, node communication, and extracts SSL/TLS session information. The uploaded PCAP file will be publicly available. In [8], DynamiteLab was proposed that provides a visually engaging platform through various analytical perspectives, which include network behavior graphs and timeline analyses. Similar to [7], the uploaded PCAP will be publicly available, and the upload size limit is 75 MB which makes the tool to be suitable only for small data analysis. Further tools are analyzed and presented in [9, 10]. Thirteen common tools are reviewed in [9] in the field of cyber security. The tools were analyzed, and their potential contributions are illustrated. In [10], PCAP Funnel was proposed to analyze packets visually by making network charts, packets, and connections. Also, it allows users to select the part they want to further analyze from the chart interactively to go into a more detailed view. Though, it doesn't support further steps that are needed in the analysis process. NetCapVis tool was proposed by [11]. It is a packet capture analysis tool that consists of timeline, protocol, graph, source, destination, and filter status views. It also supports progressive packet capture data pre-processing. The ability of the tool for the analysis and presentation data is limited. A prototype tool was introduced to provide visual guidance for cyber forensics. Its purpose is to gather data for live digital forensics and showcase visuals that display the links between events, progress over time, and extensive details about incidents [12].

In this work, a Novel Network Analyzing Tool (NNAT) is proposed and developed a comprehensive solution to the challenges facing network users by offering an interactive and user-friendly interface that makes it easy for users to visualize packet captures efficiently. NNAT covers end-to-end processes which are data capturing, then analyzing the data, and finally present in an efficient and user-friendly format.
It enables users to collect the data flowing in the network, which is then visualized interactively, allowing for easy identification of potential network issues. It also supports network scanning to find active devices in a network or to find open ports in a device. What makes the proposed tool, NNAT to be unique is ability and flexibility in the presentation of the analyzed data, as well as it covers end-to-end processes of data capturing, analyzing, and then presentation within the same platform.

The paper is organized as follows: Section 2 describes data and user groups. In section 3, the implementation processes and results are presented. The work is concluded in section 4.

2. DATA DESCRIPTION AND USER GROUP

The main data source for network traffic analysis is PCAP files. PCAP or Packet Capture File is a file that contains packets captured on the network. Network analysts rely on packet captures as their primary data source for network traffic analysis. Packet Captures contain valuable information that can be used by network analysts for network troubleshooting, identifying threats, and performance monitoring.

Packets are chunks of data exchanged across the network and can be viewed using these PCAP files. Packet sniffing programs like Wireshark or TCPdump can be used to capture packets and build a PCAP file from the collected packets. PCAP analysis is essential for network administrators and security researchers to find network intrusions and other unusual behaviors [13]. For instance, it can be seen a source delivers a lot of malicious traffic to the network and then

takes the necessary steps to halt the threat.

Useful information could be extracted from the packets, the packet header is the beginning part of a packet, and it contains control information. The header contains information such as version, protocol, source, and destination IP and MAC addresses, timestamps, etc. A packet also contains payloads, which is the data sent. This information can be used for network analysis and forensics.

Visual analysis makes network analysis easier, hence, a large group of users could benefit from it, such as network administrators. However, security analysts and penetration testers could benefit from it the most to make reconnaissance and forensics tasks easier. Such application tools could also be used for educational purposes because of its user-friendly interface, users would be able to learn to use the application quickly.

2.1. NNAT SYSTEM ARCHITECTURE

The architecture of the proposed system tool is shown in figure 1. The system consists of three main components: the packet sniffer, PCAP visualizer, and the scanner. The sniffer works by capturing the packets that are flowing in the network and making a PCAP file from them that can be used for network analysis. The PCAP visualizer component takes a PCAP file as its input, parses the packet headers, and generates useful charts and graphs as an output. The scanner scans the networks for discovering unauthorized hosts and for finding open ports that might cause a security risk.

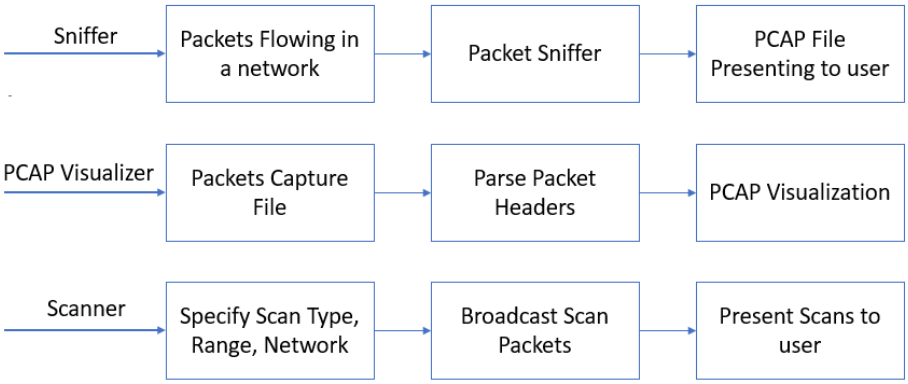


Fig.(1):- NNAT System Architecture

2.2. Background

In this section, the technologies used for the proposed tool's design and implementation are introduced. The proposed application NNAT has four main sections. The first section is the Dashboard, and it contains statistics about the application usage. The second section is for live packet sniffing, which allows users to sniff packets and apply sniffing filters. The third one is the PCAP analyzer which gives a general visual overview of the provided PCAP file. The last section of the proposed application tool NNAT is live scanner, and it supports two types of scans; host and port discovery scans.

The front end of the tool is implemented through React.js and Chart.js. React is a JavaScript framework used to build single paged user interfaces through components [14]. Chart.js is another JavaScript framework based on d3.js, it is used for making data visualizations [15].

For the backend implementation, Flask has been used. Flask is a Python micro framework used for making web applications [16]. The proposed application NNAT mainly relies on Scapy. Scapy is a Python framework for packet crafting and manipulation, network scanning, packet sniffing, and packet dissecting. The scanning data is saved in a JavaScript Object

Notation file (JSON).

3. NNAT Implementation and Results

In this section implementation of the proposed tool NNAT is presented. The first page that a user would interact with is the dashboard page as shown in figure 2. The dashboard shows information about the application's usage. Live packet sniffing allows user to collect the packets that are flowing in the network. Several filters are introduced for the proposed tool to sniff a specific type of packet. The view provides a basic filter that allows user to filter packets based on protocol, address, or port number. Once the filter is chosen, user can toggle the sniffing process and the collected packets will be displayed in a table containing each packet's information as seen in table 1. Results of the live packet sniffer are returned in a tabular form that can be saved as a PCAP file for later visual analysis. Live packet sniffing is a valuable technique in network forensics that enables the collection of packets flowing within a network.

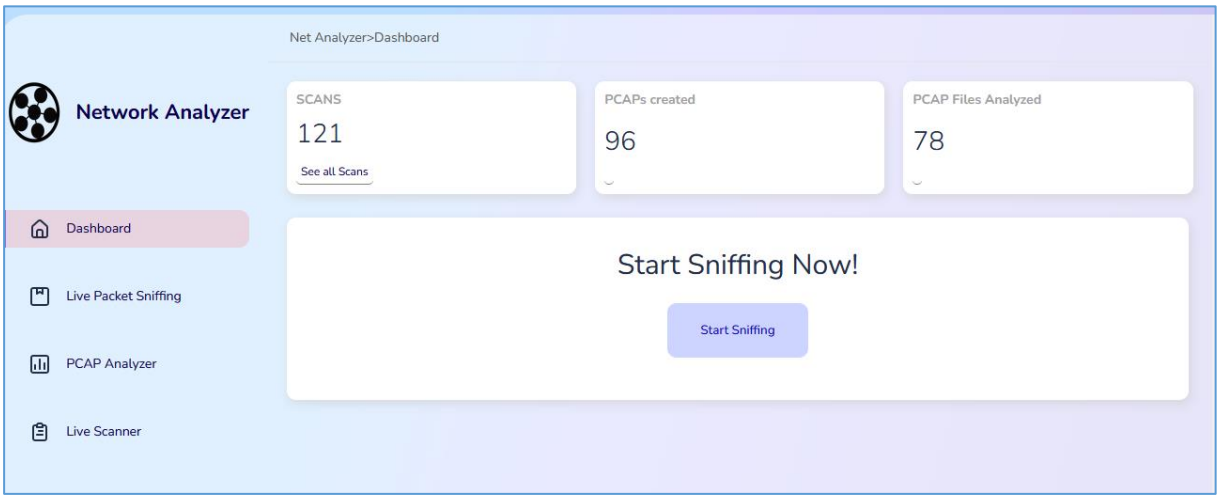


Fig.(2):- The NNAT application dashboard.

Table (1):- Main headers of each packet.

Save PCAP

Source	Destination	Src Port	Dst Port	Timestamp	Protocol	Payload Length
192.168.100.6	192.168.100.97	8009	60557	07/05/2023, 06:49:36	TCP	110 B
192.168.100.97	192.168.100.6	60557	8009	07/05/2023, 06:49:36	TCP	0 B
192.168.100.97	142.250.184.206	51917	https	07/05/2023, 06:49:37	UDP	1246 B
192.168.100.97	142.250.184.206	51917	https	07/05/2023, 06:49:37	UDP	1188 B
142.250.184.206	192.168.100.97	https	51917	07/05/2023, 06:49:37	UDP	29 B
142.250.184.206	192.168.100.97	https	51917	07/05/2023, 06:49:37	UDP	25 B
142.250.184.206	192.168.100.97	https	51917	07/05/2023, 06:49:37	UDP	674 B

3.1. PCAP Analyzer, Protocol Count:

The protocol count graph shows the count of packet headers of each layer as shown in figure 3. It allows user to navigate through layers and see the count of protocol packet headers in each layer. The chart gives a clear insight into the packets, representing the content of each layer through the pie chart.

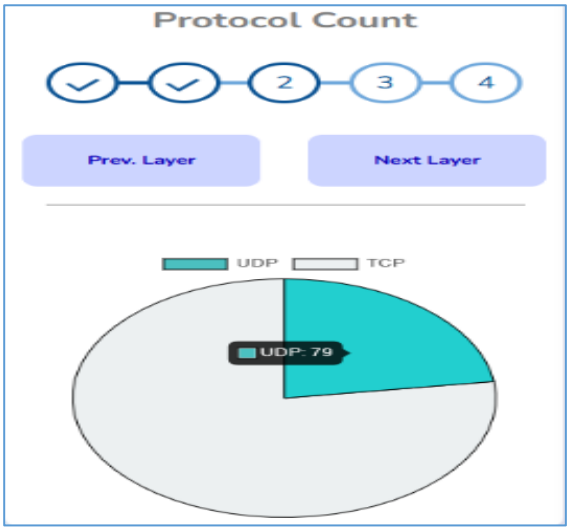


Fig.(3):- The protocol count in each layer

3.2. Connection Graph:

Connection graphs represent the connection between devices, it allows you to see the connections of each IP address. User can select the connections of a specific host by choosing the host in the drop-down menu and then clicking submit to see all the IP addresses that contacted the selected host as shown in the figure 4. This graph gives valuable information about which addresses have communicated; it can help security researchers to identify the connections between devices.

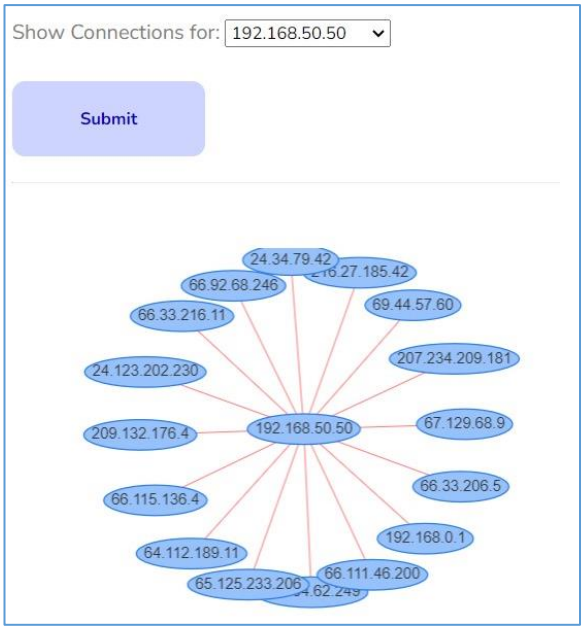


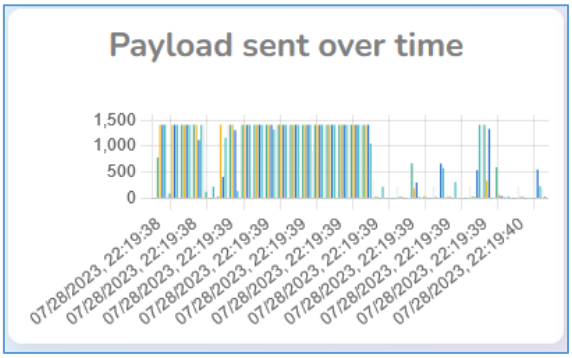
Fig.(4):- Connection Graph

asma.bismail11@gmail.com
rozhgar.rzgar.sabr@gmail.com
salar.shaikhah@epu.edu.iq

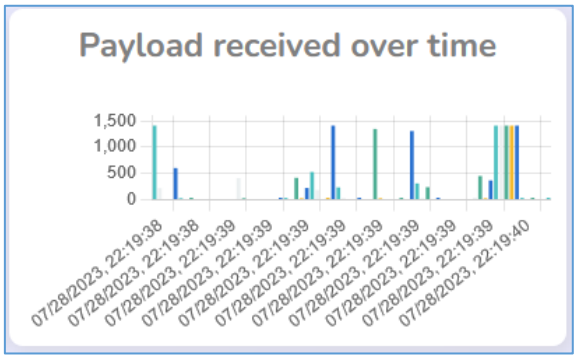
3.3. Timeline and Port Graphs:

In the PCAP Analyzer, there are two timeline graphs as demonstrated in figure 5. These timeline graphs provide a visual representation of various network traffic statistics extracted from the PCAP data. The first graph. Figure 5 (a) shows the payload sent over time in bytes. While figure 5 (b) demonstrates the data received over time.

Figure 6 demonstrates two other graphs that show the bytes sent and received over each port number. The graphs provide valuable insights into network traffic and can help identify patterns, anomalies, and potentially malicious activities. In the Payload sent/Received by port/Address: these graphs represent the data sent or received by each address or port in bytes. This will help in identifying which IP addresses are sending or receiving the most payloads. Payloads sent or received by the port, on the other hand, it allows investigators to find which processes are transferring or receiving the most data.

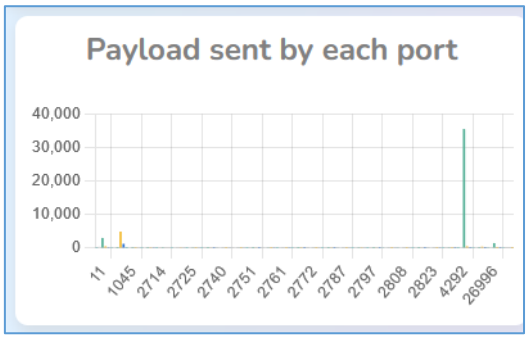


(a)



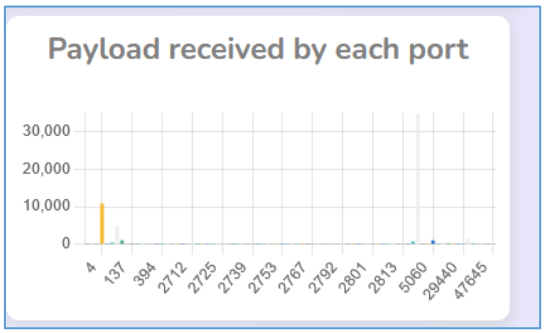
(b)

Fig.(5):- Flow of Payload data overtime, (a) Sent, (b) Received.



(a)

asma.bismail11@gmail.com
rozhgar.rzgar.sabr@gmail.com
salar.shaikhah@epu.edu.iq



(b)

Fig.(6):-Flow of Payload by each port, (a) Sent, (b) Received.

3.4. Live Network Scanning:

Live network scanning is another capability of the NNAT that allow users to scan the hosts or ports in a network. It supports four types of protocols that can be used for scanning. Address Resolution Protocol (ARP) and Internet Control Message Protocol (ICMP) scans are used for host discovery while Transport Control Protocol (TCP) and User Datagram Protocol (UDP) protocols are used for port scanning. Figure 7 demonstrates the form for information required to start a TCP port discovery scan. After submitting the required information, the results will be shown in a table containing each port's status. The saved scanning results will be available on the scan history page for later analysis. Scanning networks can be crucial for security analyzers. For instance, a user may be able to discover unauthorized devices on a network through scanning, or user might want to scan one of the hosts in the network for open ports that might cause a security risk.

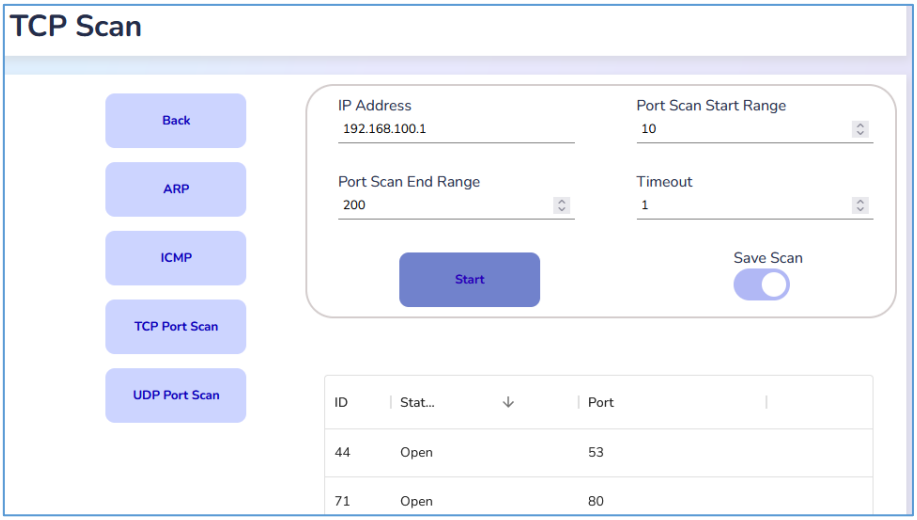


Fig.(7):-The results of TCP port scan are returned in the table indicating open and close ports.

3.5. nnat Capability

Unlike traditional tools that only support one step in the network analysis process, NNAT provides a comprehensive approach that covers the procedures needed for effective network analysis from packet sniffing to network scanning. NNAT empowers users to gain a deeper understanding of their networks and detect potential security threats more effectively. It is an advanced tool that supports multiple steps in the network analysis process, NNAT offers a robust and comprehensive approach to network analysis,

enabling users to gather valuable insights and effectively detect potential security threats. Table 2 presents a comparison among three of the most popular tools in the related field with the proposed tool, NNAT. It is obvious that NNAT covers end-to-end processes required for data analysis of a network compared to other that include only one or two processes of NNAT.

Table (2):-Comparison between NNAT and other available tools

Tool	NNAT	Reference [3]	Reference [11]	Reference [5]
Packet Sniffing	✓	✓	✗	✓
VisualPCAP Analysis	✓	✗	✓	✓
Network Scanning	✓	✗	✗	✗

4. CONCLUSION

Network data analyzing can be a difficult and time-wasting process. The use of proposed tool, NNAT gives users high flexibility and varies ability to capture, analyze and present network information. The NNAT allows users to view and analyze packet contents, providing insights into the headers and payloads of network traffic. NNAT provides an interactive and user-friendly interface that makes it easy for users to visualize network data. It enables analysts to identify anomalies with ease while ensuring that the users can analyze datasets easier than traditional tools. Moreover, providing a live network sniffer, and host and port scanning to minimize the number of tools required during network analysis. It is concluded that the NNAT outperforms the related tools in terms of ability of doing multitasks within the same platform. The proposed tool will have significant potential in advancing research development efforts aimed at improving cybersecurity, as well as, it could be used as a lab tool for teaching purpose.

5. REFERENCES

S.-Y. Ji, B.-K. Jeong, and D. H. Jeong, “Evaluating visualization approaches to detect abnormal activities in network traffic data,” International Journal of Information Security, May 2020, doi: https://doi.org/10.1007/s10207-020-00504-9.

P. Godfrey, J. Gryz, and P. Lasek, “Interactive Visualization of Large Data Sets,” IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 8, pp. 2142–2157, Aug. 2016, doi: https://doi.org/10.1109/tkde.2016.2557324.

Wireshark Foundation, “Wireshark,” Wireshark.org, 2016. https://www.wireshark.org/

V. T. Guimaraes, C. M. D. S. Freitas, R. Sadre, L. M. R. Tarouco, and L. Z. Granville, “A Survey on Information Visualization for Network and Service Management,” IEEE Communications Surveys & Tutorials, vol. 18, no. 1, pp. 285–323, 2016, doi: https://doi.org/10.1109/comst.2015.2450538.

S. Cirillo, D. Desiato, and B. Breve, “CHRAVAT - Chronology Awareness Visual Analytic Tool,” IEEE Xplore, Jul. 01, 2019. https://ieeexplore.ieee.org/abstract/document/8811994

S. Chen, C. Guo, X. Yuan, F. Merkle, H. Schaefer, and T. Ertl, “OCEANS,” Proceedings of the Eleventh Workshop on Visualization for Cyber Security, Nov. 2014, doi: https://doi.org/10.1145/2671491.2671493.

"Online PCAP file analyzer designed to visualize HTTP, Telnet, FTP," A-Packets. https://apackets.com/

"DynamiteLab," lab.dynamite.ai. https://lab.dynamite.ai/ (accessed Jul. 07, 2023).

A. Attipoe, J. Yan, C. Turner, and D. Richards, "Visualization Tools for Network Security," Electronic Imaging, vol. 2016, no. 1, pp. 1–8, Feb. 2016, doi: https://doi.org/10.2352/issn.2470-1173.2016.1.vda-489.

J. Uhlár, M. Holkovič, and V. Rusňák, "PCAPFunnel: A Tool for Rapid Exploration of Packet Capture Files," IEEE Xplore, Jul. 01, 2021. https://ieeexplore.ieee.org/abstract/document/9582696/metrics#metrics

A. Ulmer, D. Sessler, and J. Kohlhammer, "NetCapVis: Web-based Progressive Visual Analytics for Network Packet Captures," IEEE Xplore, Oct. 01, 2019. https://ieeexplore.ieee.org/abstract/document/9161633

F. Böhm, Ludwig Englbrecht, S. Friedl, and Günther Pernul, "Visual Decision-Support for Live Digital Forensics," Oct. 2021, doi: https://doi.org/10.1109/vizsec53666.2021.00012.

J. Biswas and A. Ashutosh, "An Insight into Network Traffic Analysis using Packet Sniffer," International Journal of Computer Applications, vol. 94, no. 11, pp. 39–44, May 2014, doi: https://doi.org/10.5120/16391-5975.

Meta Open Source, "React," react.dev, 2023. https://react.dev/

"Chart.js | Open source HTML5 Charts for your website," Chartjs.org, 2019. https://www.chartjs.org/

Flask, "Welcome to Flask — Flask Documentation (2.3.x)," flask.palletsprojects.com. https://flask.palletsprojects.com/en/2.3.x/

Appendix:

The proposed application tool NNAT is available on: https://github.com/asma-ismail1/NNAT.