Artificial Intelligence Model for Network Security Analysis

Haider Alshalah
Department of Computer Networks and Communications, College of Computer Science and Information Technology, King Faisal University, P.O. Box 400, Al-Ahsa 31982, Saudi Arabia
217021452@student.kfu.edu.sa

Heider A. M. Wahsheh
Department of Information Systems, College of Computer Science and Information Technology, King Faisal University, P.O. Box 400, Al-Ahsa 31982, Saudi Arabia

*Corresponding author: hwahsheh@kfu.edu.sa

Abstract:

Network traffic analysis (NTA) means packet sniffing, which is the procedure of gathering and tracking network activities to recognize it is behavior.. NTA holds a real-time and documented record or log of the activities occurring inside the network and identifies the vulnerable or weak protocols and ciphers. Commonly, packet analysis or packet sniffing is conducted by a packet sniffer tool that is utilized to capture raw network traffic. There are various existing tools, either free or commercial, based on the command-line interface (CLI) or graphical user interface (GUI). NetworkMiner is one of the widespread network forensics tools that can parse the Packet Capture (pcap) files and conduct live sniffing of the traffic. This paper utilized NetworkMiner to explore network packets among various attack scenarios. A dataset of 20,000 instances of various protocols packets was captured and collected in an education network environment, extracting the features, and labeling each instance as normal or anomaly. A decision tree model was employed to detect the network behavior in real-time, and results showed it as the most appropriate model for predicting with an accuracy of 96.29%.

Keywords: Packet Sniffer, Network Security, Network Forensics, Artificial Intelligence, Suspicious behavior.

DOI Number: 10.14704/nq.2022.20.13.NQ88254 Neuro Quantology 2022; 20(13):2036-2044

I. INTRODUCTION

Network traffic analysis (NTA) is a technique for tracking network availability and actions to determine security and operational problems. NTA learns the network properties, investigates protocol behaviors, troubleshoots packet forwarding points, determines susceptibilities in the network protocols, recognizes malicious actions, and collects real-time details on activities between different network components. More useful network visibility authorizes administrators to enhance performance and improve security protection methods [1, 2].

NetworkMiner is an open-source, Graphical User Interface (GUI) network forensic tool that can be used to detect and capture various network activities such as open ports, operating systems, hostnames, and sessions without putting any traffic on the network [3]. It can perform off-line analysis and generate certificates and extract related documents by parsing packets from a Packet Capture (PCAP) file, email files, or directly from the network traffic [4].

NetworkMiner was released in 2007 and became a popular tool among network analysis teams, such as incident response teams [5]. It supports several operating systems, such as Windows, Linux, and macOS. There are two available versions of NetworkMiner, the first version is free, and the other is the professional version that you need to purchase, which contains more features than the free version [3-5]. Also, it is effortless for network forensics investigators to make an Advanced Network Analysis (NAT) on NetworkMiner, because of its simplicity that organizes different components in

simple and readable taps. NetworkMiner has different capabilities, such as live sniffing of packets, parsing PCAP files, Operating System Fingerprinting, etc. NetworkMiner is used in forensics cases such as network discovery, investigating intruder hosts, reassembling transferred files, data leakage, malicious activities, and vulnerability discovery [6].

Learning how to track and monitor network traffic is not sufficient; it's essential to develop a real-time solution capable of continuously monitoring network traffic, providing the required insight to enhance network performance, prevent and detect possible attacks, and manage network resources.

This paper utilized NetworkMiner to investigate network packets among various attack scenarios. A dataset of 20,000 instances of Transmission Control Protocol (TCP), User Datagram Protocol (UDP) and other protocols packets was captured and collected in an education network environment, extracting the features and labeling each instance as normal or anomaly. A decision tree classifier was employed to detect the network behavior in real-time, and results showed it as the most appropriate model for predicting the network behavior with high accuracy results.

The rest of this paper is organized as follows: Section 2 illustrates the insights of how NetworkMiner works.. Section 3 explores the analysis of network attacks. Section 4 presents the artificial intelligence model and the performance evaluation. Section 5 draws the concluding remarks.

II. NetworkMiner Insight

NetworkMiner can be installed primarily on Windows via its official [7] as zip file that can be extracted and use the tool directly on Windows without going to the installing procedures like other programs [7].

NetworkMiner used to capture and analyze network traffic via PCAP file or directly by using the interfaces available on the operating system, such as wi-fi and ethernet. Packet capture can be done by sniffing network data traffic. Packet sniffing is the process of gathering information via capturing network packets from an interface of the network [8]. There are two techniques of packet sniffing: active sniffing and passive sniffing. The active sniffing technique is done using one of the available interfaces by sending a request over the network and then calculating packets sent and received through the network [8]. On the other hand, the passive sniffing technique is done by scanning network traffic without being detected over the network. This technique is helpful for governmental systems such as military systems and medical systems. It helps the administrator of the network to analyze deep down on the network without affecting the network traffic [2, 8].

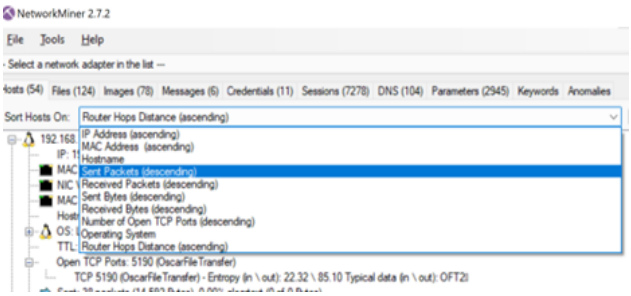
NetworkMiner can automatically reconstruct documents, images, and other files. NetworkMiner can list each file into specified taps based on how the network investigator wants. It consists of different taps such as Hosts, Files, Images, Messages, Credentials, Sessions, DNS, Parameters, and keywords. Each tap has its own details, which network investigators can select based on what is suitable to analyze first. Now, we will discuss the features of NetworkMiner as GUI (Graphical User Interface). It has various analysis features such as analyzing PCAP files, port identification protocol, and extracting files from different protocols (e.g... FTP, HTTP, TFTP, SMTP, etc.). Table 1 explores NetworkMiner free version features.

Table 1. NetworkMiner Features of the Free version

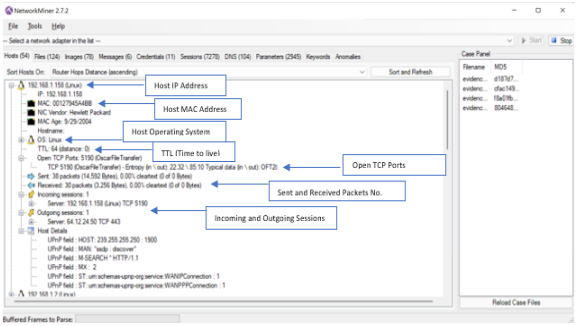
Feature	Availability
An open source.	✓
Works on Windows Primarily.	✓
It can work on Linux, macOS.	✓
It works as Graphical User Interface (GUI).	✓
It identifies Operating System type.	✓
Parse PCAP file.	✓
Live Sniffing.	✓
Receive PCAP file over IP.	✓
Extract files from different protocol such as FTP, HTTP, HTTPS, SMTP traffic.	✓
Reconstruct certificate from SSL encrypted traffic such as HTTP, SMTP, etc.	✓

A. The Hosts tap

The host tap feature in NetworkMiner can sort hosts of the sniffed packets based on different ways such as IP Address, MAC address, Hostname, Sent Packets, Received Packets, Sent Bytes, Received Bytes, Number of Open TCP Ports, Operating System, and Router Hops Distance. After the sort of packets, we can know more information about each host that passed through the network. The host's information contains the IP address, MAC address, type of Operating System, TTL (Time to live), Open TCP ports, Sent and received Packets, Incoming and Outgoing Sessions, and Hosts details. The operating system could be Windows, Linux, macOS, and others Operating systems. The TTL is used to know how packets travel in and out of the network before it is discarded if the time of the packet reaches the limited time required in the buffer [9]. As shown in figure 1 (A and B).



(A)



(B)

Figure 1: The host tap feature in NetworkMiner

B. The Files tap

The Files tap consists of several components such as Frame no, Filename, Extension, Size, Source host, Source Port no., Destination host, Destination Port no., Protocol used, Timestamp, and the path of where files are stored. When you click on any of the icon components, it will sort the packets based on that component. The filename icon specifies the name of each file. The Extensions icon specifies the type of files, such as word file (.docx), image file (.pig), and Hypertext Markup Language file (.html). The Source and Destination host icons specify the source IP Address and the Destination IP address of each file. The Source Port no. and Destination Port no. icon specifies the port number in which packets are sent and received. The Timestamp specifies the date and time of the received file. Also, the Filter keyword is useful when you are searching for a specific file. Furthermore, we can see the content of the file by right click on the file and then opening the file. Figure 2 presents the NetworkMiner File tap feature.

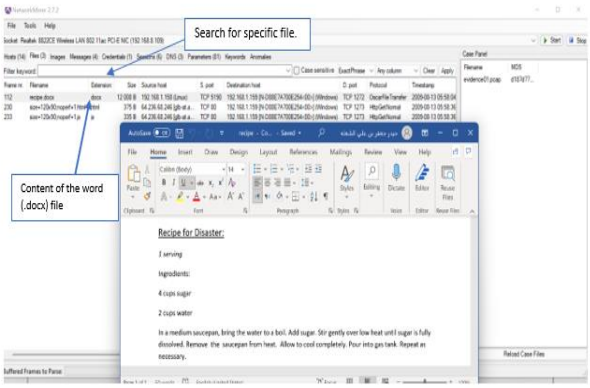


Figure 2. The NetworkMiner File tap feature.

C. Other taps

Other taps are categorized as the following:

The Images tap consists of images that were captured from the network. We can click on any image to open it and see its content.

- The Messages tap specifies each message with Frame No., Source host, Destination host, from and to which host the message was delivered, Subject of the message, Protocol, Timestamp, and Size of the message. The message content can be seen by clicking on one of the hosts. Also, the

conversation of each host can be seen through the opened message. Figure 3 illustrates each host message from who and to which host the message was delivered.

- The Credentials tap consists of several components such as Client, Server, Protocol, Username, Password, Valid login, and login timestamp. The Client icon specifies the IP address of the client that connects to a specific server over a protocol like HTTP. The Username and Password icons show the used username and password on the visited websites.
- The Session tap consists of several components such as Frame no., Client host, Client port, Server host, Server port, Protocol, and start time. The session tap is responsible for recording the session connection between the client and the server through a specific port number.
- DNS indicates to Domain Name System, which is translating a domain name into an IP address. For instance, the Google website "www.google.com" will be converted into IP address like 100.60.1.72 to let the machine know the address of the google website because machines do not understand the name system like google.com. So, The DNS tap consists of Frame no., Timestamp, Client, Client Port, Server, Server Port, IP TTL, DNS TTL, DNS Query, and DNS Answer.
- The Keywords tap is used as a filter to search for specific words, whether in hexadecimal or text format. Also, you can add keywords from an external text file to search about it.
- The Anomalies tap is used to detect any suspicious behavior or error that occur during packet capturing and parsing, such as ARP spoofing may happen if the MAC address of a device change [10].

III. Analysis of Network Attacks

In this section, we will analyze two PCAP files that I already downloaded on a device. These PCAP files will be analyzed to detect possible attacks that may happen to them. The first PCAP file will be used to sniff credentials such as usernames and passwords from a host machine. The second PCAP file will be extracted from Ettercap sniffing tool to analyze the ARP spoofing attack by NetworkMiner.

A. Credentials Attack

In this section, we will use a PCAP file that has been downloaded from GitHub about some network forensics cases, such as credentials attacks. The credentials attack is an attack where host data traffic is sniffed by using packet sniffing tools like Wireshark to gather host information, such as the username and password of a specific website that the host logged on to it before. The performed analysis shows that the sniffed packets use the SMTP protocol. SMTP stands for Simple Mail Transfer Protocol, which is responsible for transmitting and receiving mail transmission from the client to the mail server. This type of attack could be critical if it were a bank account username and password. That would be massive financial damage to companies or individuals. Figure 3 illustrates the sniffed credentials from the PCAP file.

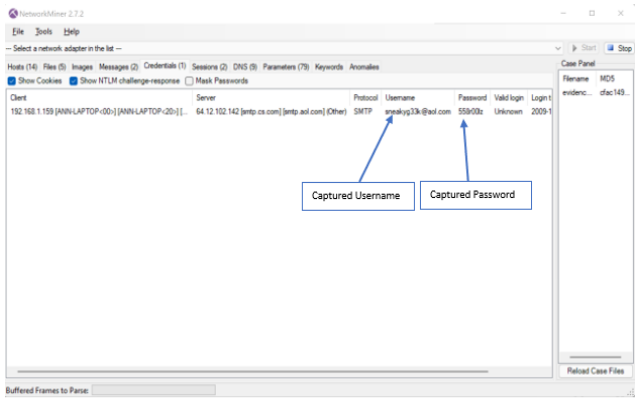


Figure 3 The sniffed credentials from the PCAP file.

B. ARP poisoning attack

Address Resolution Protocol (ARP) is responsible for mapping the IP address to a physical machine address called The MAC address in the local network. The ARP cache handles the mapping between the MAC address with its corresponding IP address. ARP spoofing or poisoning can be done by corrupting the ARP cache or ARP table of

the host machine. This method is used to allow the attacker to redirect all the host machine traffic to his machine. When an attacker sends a bogus ARP reply to the host machine, the ARP poisoning happens after getting the replies [10]. Figure 4 shows how ARP spoofing is performed.

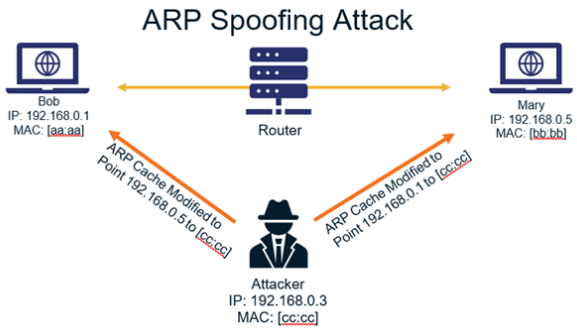


Figure 4. ARP Spoofing.

The Man-In-The-Middle (MITM) is a type of attack in which a third person on the network pretends to be the router to take control of all the network communication between network devices [11]. The attacker can manipulate, intercept, modify, and replace the victim's communication across the network. For instance, Bob wants to communicate and exchange messages with Mary. Bob knows the IP address of Mary, which is 192.168.0.5. But Bob did not know where Mary was located on the network. So, Bob will send a request to the router to redirect it to Mary's machine because the router stores the MAC address of each machine and where it's located on the network. The attacker pretends to be the router by sending it the MAC address as the router MAC Address. So, now Bob and Mary exchange messages over the attacker machine that pretends to be a router.

The MITM attack was performed on the VMware Workstation. VMware is software that can install several virtual operating systems. There are two virtual machines installed on VMware to perform the MITM attack. The Windows 7 machine was the victim machine that we will perform the ARP poisoning attack on it to obtain all victim machine traffic. The Kali machine was the attacker that used to perform the ARP poisoning attack on the Windows 7 machine to obtain all traffic that passed from it.

IV. Artificial Intelligence Model and Evaluation

Artificial Intelligence by machine learning (ML) methods was widely employed in network security analysis and attack predictions [12-18]. In our study, we analyze particular network traffic from the education environment by NetworkMiner and export the data into a CSV file.

The artificial Intelligence model includes the following main steps:

1. Data collection: we have extracted 20 000 instances from the NetworkMiner tool, divided equally into 10 000 normal and 10 000 anomaly ones.
2. Network Traffic Features extraction and selection: we used a feature selection method to consider the most optimal features.
3. Machine learning prediction and comparisons: we utilized three machine learning models, Decision Tree (J48) and support vector machine, to find and adopt the best method to predict the network behavior in real-time.

The dataset is balanced, so if the instances from one of the classes outnumber the other (imbalance), the data is skewed in favor of one class. This may lead to training issues because the majority class can affect the minority class, and the model may only learn one concept alternatively of two different concepts [16-20]. The feature selection method is essential for enhancing the performance of machine learning in both accuracy and execution time. It can select the main features and reduce the unrelated features that decompose the model efficiency. The information gain (IG) technique was used in our experiments, which depends on a threshold value, where the IG value is greater than 0, and all the features are dropped if the IG value is equal to 0 [21]. Table 2 summarizes the selected features in this study that correspond to [22-28].

Table 2. The list of selected features

#	Feature Name	Description
1	time	time of traffic capturing
2	app	honeypot captured the traffic
3	dest	dest ip

4	dest_port	dest port
5	dionaea_action	either Dionaea honeypot accept or reject the connection
6	direction	the direction of the captured traffic either in or out
7	eth_dst	the dest mac address
8	eth_src	the source mac address
9	host	Splunk server ip or hostname
10	ids_type	the type of the used ids
11	ip_id	the packet id
12	ip_len	packet length
13	ip_tos	packet type of service
14	ip_ttl	packet time to live
15	linecount	the number of lines of the captured traffic
16	p0f_app	protocol used by P0f for fingerprinting
17	p0f_link	the connection type at the attacker side like modem or dsl
18	p0f_os	the operating system of the machine generating the attack
19	p0f_uptime	how long since the attacking machine is up
20	protocol	tcp or udp
21	sensor	id assigned by MHN per honeypot
22	severity	severity rank of the attack
23	signature	the signature of the attack as matched by snort
24	snort_classification	a number given by snort to classify the traffic
25	snort_header	the rule header
26	snort_priority	assigns a severity level to rules
27	source	input data source (needed by Splunk)
28	sourcetype	input data type (needed by Splunk)
29	splunk_server	Splunk ip or hostname
30	src	attack src ip
31	src_port	attack source port
32	ssh_password	password used by the attacker trying to get ssh access
33	ssh_username	username used by the attacker trying to get ssh access
34	ssh_version	attacker ssh client version
35	tcp_flags	indicate a particular connection state or provide additional information
36	tcp_len	packet length
37	timeendpos	at which byte into the event the timestamp ends
38	timestartpos	at which byte the timestamp starts
39	transport	transport protocol type tcp or udp
40	type	honeypot event type
41	udp_len	packet length
42	vendor_product	name of the honeypot that captures the traffic
43	raw	raw (not parsed) event

Two machine learning is used Decision Tree (J48) and support vector machine [29, 30]. A Decision Tree is a popular and effective supervised learning method that is widely employed for prediction. A Decision tree is a visualization tree design where each internal node represents a test on a characteristic, each branch means a result of the test, and each leaf node represents a class label [29]. Support vector machine (SVM) is a supervised learning technique that transforms the original training data into a higher-dimensional space so that data points can be linearly divided. SVM searches for the linear optimal separating hyperplane that is utilized for determining the class of new incoming testing instances [30].

We adopted 66% of the data as a training dataset and the remaining 34% as a testing dataset. We compared the yielded results of J48 and SVM among the prediction quality measurements; True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN), Precision (P), Recall (R) and F-Measure (F-M), as shown in formulas 1-4 [19].

$$Accuracy_i = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

$$Recall_i = \frac{TP}{TP + FN} \quad (2)$$

$$Precision_i = \frac{TP}{TP + FP} \quad (3)$$

$$F - measure = \frac{(2 \times TP)}{(2 \times TP) + FP + FN} \quad (4)$$

When comparing the models, we investigate the most elevated significances of TP, precision, recall, and F-measure. On the other hand, the FP rates should be minimized toward the optimal value, which is zero.

SMV model yielded an accuracy result of 93.22% with an error rate of 6.77%, Table 3 explores the detailed results.

Table 3. Detailed Accuracy Results for SVM Model.

Class	TP Rate	FP Rate	Precision	Recall	F-M
Normal	0.968	0.104	0.904	0.968	0.935
Anomaly	0.896	0.032	0.965	0.896	0.930
Weighted Avg.	0.932	0.068	0.934	0.932	0.932

As shown in Table 3, the SVM model yielded high prediction results for Normal behavior with an accuracy of 96.8%. The accuracy result for anomaly detection yielded 89.6%, which needs to be enhanced compared to the Normal detection accuracy result.

A confusion matrix is a particular table design that provides visualization of an algorithm's performance, typically in supervised learning models. Every row presents the actual instances class label in the confusion matrix, while every column shows the instances' predicated class label.

Among the 0.34% (6800 instances) of the total dataset, the SVM model correctly classified 6339 instances and failed to classify 461 samples to their correct class label, as shown in Table 4.

Table 4. Confusion Matrix of SVM Model.

Actual Class	Predicted Class	
	Normal	Anomaly
Normal	3297	109
Anomaly	352	3042

J48 enhanced the accuracy results of SVM model and obtained 96.29% with an error rate of 3.70%, where Table 5 shows the detailed results.

Table 5. Detailed Accuracy Results for J48 Model.

Class	TP Rate	FP Rate	Precision	Recall	F-M
Normal	0.996	0.070	0.935	0.996	0.964
Anomaly	0.930	0.004	0.995	0.930	0.962
Weighted Avg.	0.963	0.037	0.965	0.963	0.963

Among the 0.34% (6800 instances) of the total dataset, the J48 model correctly classified 6548 instances and failed to classify 252 samples to their correct class label, as shown in Table 6.

Table 6. Confusion Matrix of J48 Model.

Actual Class	Predicted Class	
	Normal	Anomaly
Normal	3391	15
Anomaly	237	3157

The results showed that Decision Tree (J48) yielded the best results, and it is the recommended model for being used as an early warning tool for detecting anomaly behavior on the network traffic.

V. CONCLUSION AND DISCUSSION

NetworkMiner is a beneficial tool for monitoring and analyzing wired and wireless networks. Administrators use this tool to analyze the network traffic and troubleshoot any unusual behavior detected. As attacks from anonymous individuals increase, such as DDoS attacks, administrators must be aware of how attacks are performed to demonstrate a solution to solve them quickly if they happen. These attacks may damage the company's reputation if the company cannot prevent them. The best approach to get the advantage of NetworkMiner is when an administrator wants to check and troubleshoot the network. The administrators can use other sniffing tools. The PCAP file used instead of live sniffing is because NetworkMiner lives to sniff and cannot capture all types of protocols from network traffic.

This study used NetworkMiner to analyze network packets among various attack scenarios. A dataset of 20,000 Transmission Control Protocol (TCP) and other protocols packets was captured and collected in an education network

environment, extracting the features and labeling each instance as normal or anomaly. A Decision Tree (J48) model was used to predict the network behavior in real-time, and results showed it as the most appropriate model with high accurate results.

ACKNOWLEDGMENT

This work was supported by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [Grant No. GRANT987]'.

REFERENCES

[1] A. Shahraki, M., Abbasi,A., Taherkordi, A., A. Jurcut. A comparative study on online machine learning techniques for network traffic streams analysis. Computer Networks, 207, 108836, 2022

[2] V. Jain, "Wireshark Fundamentals: A Network Engineer's Handbook to Analyzing Network Traffic", Apress; 1st ed. edition (March 4, 2022), pp. 1-276

[3] S. Qureshi, S. Tunio, F. Akhtar, A. Wajahat, A. Nazir, F. Ullah. "Network Forensics: A Comprehensive Review of Tools and Techniques", International Journal of Advanced Computer Science and Applications, Vol. 12, No. 5, 2021.

[4] I. Volarević, M. Tomić, I. Milohanić,. Network forensics. In 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO) (pp. 1025-1030). IEEE, 2022.

[5] A. Bijalwan. "Network Forensics: The Privacy and Security". Chapman and Hall/CRC, 2021.

[6] A. Javed, W. Ahmed, M. Alazab, Z. Jalil, K. Kifayat, T. Gadekallu. A comprehensive survey on computer forensics: State-of-the-art, tools, techniques, challenges, and future directions. IEEE Access, 2022.

[7] Network Forensics and Network Security Monitoring , https://www.netresec.com/, 2022.

[8] A., Siswanto, A., Syukur, E., Kadir, E. A. Network traffic monitoring and analysis using packet sniffer. In 2019 International Conference on Advanced Communication Technologies and Networking (CommNet) (pp. 1-4). IEEE, 2019.

[9] M., Sioutis, Y., Tan. Open Source Implementation of HTIP for Embedded Devices. In 2019 IEEE 8th Global Conference on Consumer Electronics (GCCE) (pp. 106-110). IEEE, 2019.

[10] N. Ahuja, G. Singal, D. Mukhopadhyay, A. Nehra. "Ascertain the efficient machine learning approach to detect different ARP attacks". Computers and Electrical Engineering, Vol. 99, 107757, 2022.

[11] A. Mallik. Man-in-the-middle-attack: Understanding in simple words. Cyberspace: Jurnal Pendidikan Teknologi Informasi, 2(2), 109-134, 2019.

[12] S., Patil,V., Varadarajan, S., Mazhar,A., Sahibzada, N., Ahmed,O. , Sinha,K., Kotecha. Explainable Artificial Intelligence for Intrusion Detection System. Electronics, 11(19), 3079, 2022.

[13] L., Zhao, D., Zhu, W., Shafik, S., Matinkhah, Z., Ahmad, L., Sharif, A., Craig. Artificial intelligence analysis in cyber domain: A review. International Journal of Distributed Sensor Networks, 18(4), 2022.

[14] N., Kumar, U., Kumar. Artificial intelligence for classification and regression tree based feature selection method for network intrusion detection system in various telecommunication technologies. Computational Intelligence.2022.

[15] Z., Abou El Houda, B., Brik, L., Khoukhi. "Why Should I Trust Your IDS?": An Explainable Deep Learning Framework for Intrusion Detection Systems in Internet of Things Networks. IEEE Open Journal of the Communications Society, 3, 1164-117,, 2022.

[16] S., Naseem, A., Alhudhaif, M., Anwar, K., Qureshi, G., Jeon. Artificial general intelligence-based rational behavior detection using cognitive correlates for tracking online harms. Personal and Ubiquitous Computing, 1-19., 2022.

[17] V., Rani, M., Kumar, A., Mittal, K., Kumar. Artificial Intelligence for Cybersecurity: Recent Advancements, Challenges and Opportunities. Robotics and AI for Cybersecurity and Critical Infrastructure in Smart Cities, 73-88, 2022.

[18] M., Rath, S., Mishra. Advanced-level security in network and real-time applications using

machine learning approaches. In Research Anthology on Machine Learning Techniques, Methods, and Applications (pp. 664-680). IGI Global., 2022

[19] H. Wahsheh, M. Al-Zahrani. Lightweight Cryptographic and Artificial Intelligence Models for Anti-smishing. In International Conference on Emerging Technologies and Intelligent Systems (pp. 483-496). Springer, Cham., 2021.

[20] M. Al-Zahrani, H. Wahsheh, F. Alsaade. Secure Real-Time Artificial Intelligence System against Malicious QR Code Links. Security and Communication Networks, 2021.

[21] D. Fernando, N. Komninos. FeSA: Feature selection architecture for ransomware detection under concept drift. Computers & Security, 116, 102659., 2022.

[22] N., Moustafa, J., Slay. The significant features of the UNSW-NB15 and the KDD99 data sets for network intrusion detection systems. In 2015 4th international workshop on building analysis datasets and gathering experience returns for security (BADGERS) (pp. 25-31). IEEE, 2015.

[23] A., Alshaibi, M., Al-Ani, A., Al-Azzawi, A., Konev, A., Shelupanov.The Comparison of Cybersecurity Datasets. Data, 7(2), 22, 2022.

[24] J., Fuhr, F., Wang, Y., Tang. MOCA: A Network Intrusion Monitoring and Classification System. Journal of Cybersecurity and Privacy, 2(3), 629-639., 2022.

[25] H., Ahmed, A., Hameed, N. Bawany. Network intrusion detection using oversampling technique and machine learning algorithms. PeerJ Computer Science 8:e820, 2022.

[26] M., Alani. Implementation-Oriented Feature Selection in UNSW-NB15 Intrusion Detection Dataset. In International Conference on Intelligent Systems Design and Applications (pp. 548-558). Springer, Cham., 2022.

[27] M., Sarhan, S., Layeghy, M., Portmann. Feature Analysis for Machine Learning-based IoT Intrusion Detection.2022 ,.

[28] A., Mahfouz, A., Abuhussein, F., Alsubaei, S. Shiva. Toward A Holistic, Efficient, Stacking Ensemble Intrusion Detection System using a Real Cloud-based Dataset, International Journal of Advanced Computer Science and Applications(IJACSA), 13(9), 2022.

[29] B., Amrutha, I., Meghana, R., Tejas, R., Pilare, D., Annapurna. An Efficient Automated Intrusion Detection System Using Hybrid Decision Tree. In Inventive Systems and Control (pp. 703-716). Springer, Singapore, 2022

[30] Z., Long, W., Jinsong. A hybrid method of entropy and SSAE-SVM based DDoS detection and mitigation mechanism in SDN. Computers & Security, 115, 102604, 2022.