World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:15, No:1, 2021

Performance Analysis of Traffic Classification with Machine Learning

Htay Htay Yi, Zin May Aye

Abstract—Network security is role of the ICT environment because malicious users are continually growing that realm of education, business, and then related with ICT. The network security contravention is typically described and examined centrally based on a security event management system. The firewalls, Intrusion Detection System (IDS), and Intrusion Prevention System are becoming essential to monitor or prevent of potential violations, incidents attack, and imminent threats. In this system, the firewall rules are set only for where the system policies are needed. Dataset deployed in this system are derived from the testbed environment. The traffic as in DoS and PortScan traffics are applied in the testbed with firewall and IDS implementation. The network traffics are classified as normal or attacks in the existing testbed environment based on six machine learning classification methods applied in the system. It is required to be tested to get datasets and applied for DoS and PortScan. The dataset is based on CICIDS2017 and some features have been added. This system tested 26 features from the applied dataset. The system is to reduce false positive rates and to improve accuracy in the implemented testbed design. The system also proves good performance by selecting important features and comparing existing a dataset by machine learning classifiers.

Keywords—False negative rate, intrusion detection system, machine learning methods, performance.

I. Introduction

NOWDAYS, network security becomes very important role in data and network system. Because malicious users and attackers are more and more increasable, especially in business and education. If the traditional hardware-based firewalls implement, these can vendor lock and higher cost. The system applies software-based open source firewall and it reduces complexity, time, often adaptive in configuration, and especially in cost [18]. When setting a rule on a firewall, the rule may be out of order, and the admin configuration error as typing may be a system vulnerability [2]. The protect system is main factors to be reliable, and robustness and also now focuses on IDS rather than firewall.

An IDS collects a variety of incoming data traffic and analyzes which data are what kind of attacks. The Intrusion Detection System has two main types. The first type is signature-based that can detect malicious attack with specific byte patterns to know attack. The second is anomaly-based that is a statistical monitor the network traffic instead of particular pattern. The system applies open source Snort-IDS to analysis protocols and detect for matching content. Intrusion detection is needed as an additional barrier for network protecting systems. Moreover, this Intrusion detection is applied to detect intrusions and also provided important data for countermeasures [19].

The main research areas of this paper are: 1) Creating the firewall rules on the software-based firewall interfaces as Outgoing traffic, IPCop Access, Internal Traffic, Port Forwarding and External IPCop Access based on services. 2) Providing IDS signature-based policy and proving with machine learning. 3) Proposed dataset implemented to improve the performance of the system.

The rest of the paper is composed of as follows. Section II summaries of the related works of the previous authors. Section III presents the research methodology. Section IV introduces the propose dataset and system setup. Section V approves the implementation and evaluation of the system with proposed dataset. Section VI is the conclusion and future work of this paper.

II. Related Works

The researchers are assisted to it to plan more effective NIDSs. In [8], that presented the detection procedures, attitudes and knowledge of IDSs. The authors acquaint with two prominent and open source tools for learning IDSs. The virtualization technology is used to study of IDS matters on Virtual Machine. In [1], the joint technique is used to Network Intrusion Detection Systems NIDS. They approached on determining the effectiveness and the performance of Snort IDS and the new one of Suricata IDS.

The researcher [9] proposed the types of network attacks. The paper described the firewall that is limited the access between networks in order of rules to prevent attack and impossible signal an attack from inside the network. The author is classified of IDS based on methodology as architecture, decision making, locality, reaction or response, decision methods.

Reference [5] described two Machine Learning approach neural network and Support Vector Machine (SVM) with a set of benchmark data from 1998 DARPA. The result compared the performance of neural network and SVM with intrusion detection. In this work, SVM is faster training time and running time. Tao et al. [7] also compared with other SVM-based Intrusion detection and the detection rate is so high. This paper proposed feature selection, weight, and parameter.

Reference [19] modified old Logistic Regression Algorithm to reduce training time. Hwang et al. [20] proposed a classification method using statistic signatures as direct sequence of packet size based on SVM (Support Vector

Htay Htay Yi is with the Information and Communication Technology Research Center, Myanmar (e-mail: htayhtayyee@ictresearch.edu.mm).

Machine) for application traffic. In [22], the authors applied Kyoto2006+ to judge the performance, accuracy, false positive rate and detection rate with SVM.

III. Research Methodology

This section contains the Firewall, Intrusion Detection System (IDS) that are applied in this work. It will talk about using software-based firewall as IPCop and how to use the rules related to Snort-IDS.

A. Firewall

Today, firewalls are such a mainstream technology that are often considered a panacea for many security issues. In network security system, the design of firewall is to prevent malicious attack to or from a local network. Firewall is limited the damage that will spread from one subnetwork to another by divide a network into different subnetworks [2]. A firewall is enforced a firewall policy to access control between two more networks. The firewall policy is a compose of filtering fields as network fields and also includes protocol type, Source IP address, Source port, Destination IP address and Destination port that perform as action field.

When choosing the firewall to adjust with the system, there may be some considerations for the following facts: (i) the features that are given by firewall, (ii) the rate of wages to be adjusted for the users in the organization, and (iii) the budget to be spent when implementing the system. There are many types of firewall, some are hardware based and some are software-based. In this system software-based firewall IPCoP is deployed in the testbed environment. IPCop offers the outshines features and the free available for the users.

B. IPCop Firewall

The system can be implemented firewall as software-based or hardware-based or both. This paper applies software-based open source firewall. There have many software-based firewalls in firewall devices. Among them, some firewall gives free even commercial. A proposed system, firewall implements software firewall instead of a hardware firewall by using IPCop version 2.1.8, the last stable version, though it has a limited functionality, however, it is flexible enough to allow installation of various add-ons to enhance it to commercial grade firewalls.

IPCop is an open source Linux Firewall Distribution and supports a secure and stable. IPCop firewall amidst those firewalls can get free and firewall policy rules can be set their service depended on their respective network. Moreover, add-on packages can be added easily if it is needed. It composed of four types of network interfaces as Green, Red, Blue, and Orange. A good design of IPCop firewall provides a web interface that can manage the firewall. The firewall filtering rules create in four interfaces such as outgoing traffic, IPCop access, internal traffic, external IPCop access and port forwarding. These four interfaces can assign the firewall filtering rules to manage the desire system. The examples of filter rules that applied in IPCop interface in internal traffic as shown in Table I.

TABLE I
Example of Internate Traffic In IPCop

R-ule	Pr-oto	Src-IP	Src-Port	Dest-IP	Dest-Port	Act-ion
rl	UDP	192.168.235.50	any	192.168.137.100	53	allow
r2	TCP	192.168.235.*	any	192.168.137.*	443	deny
r3	UDP	192.168.235.*	any	192.168.137.100	22	deny
r4	UDP	192.168.235.50	any	192.168.137.100	443	allow
r5	UDP	*.*.*.*	any	192.168.137.*	22	deny
r6	ICMP	192.168.235.*	any	192.168.137.*	Ping	deny
r7	ICMP	192.168.235.50	any	192.168.*.*	Ping	allow
r8	UDP	*.*.*.*	any	*.*.*.*	53	deny

C. Snort Intrusion Detection System

An intrusion detection system (IDS) monitors communication pursuant to certain rules. If the rule on the network connection is complied with, the system evaluates whether it is intrusion and reports it to the relevant administrator or user [2], [6]. There are three modes in snort: sniffer mode, packet recording mode and intrusion detection system. In sniffer mode, the program will read network packets and display them on the console. In packet logger mode, the program will log packets to the disk. This system uses intrusion detection mode to monitor network traffic and analyze it against a rule set defined by the user. Some of the powerful features of Snort depends on the signature-based rule through the plug-in and also preprocessors. Snort is depended on feasible of content analysis and a pattern matching. The Snort rule has two portions: the rule header and rule option. The example of a snort rule is
Rule Header –> alert udp any any –> 192.168.235.0/24 53
Rule Options –> (msg: ”Domain access”, sid=1000005;)
The general form of a Snort rule:
action proto src_ip src_port direction dst_ip dst_port (option)
Actions: Snort supports several assemble actions. A rule is matched with the directly log the packet, used the log actions. The alert action creates an alert by using the method defined in the file as configuration or on the command line, to logging the packet.
Protocols: The next field is operated to define the protocol in the rule applies. The values of this field are IP, ICMP, TCP, and UDP.
IP addresses: This field is specified the source IP addresses, destination IP addresses and ports in a rule.
Ports: The port filed will accept single ports as ranges with IP address. A range is defined to separate from upper to lower bound with a colon character.
Options: A snort plug-in used in each option field that consists of two potions as a keyword and an argument. This field scans incoming packets as opposed to snort plug-in.

Snort is a very useful reporting mechanism and that allows alerts to be recognized as a log, as well as by sending alerts to log servers such as syslog or a database. The Snort intrusion detection system is included in IPCop Firewall and can be detected attacks on internal servers. The added benefit of an IDS is that we can see what is passing through our network and attempt to isolate any traffic that appears malicious.

Open Science Index, Computer and Information Engineering Vol:15, No:1, 2021 publications.waset.org/10011782.pdf

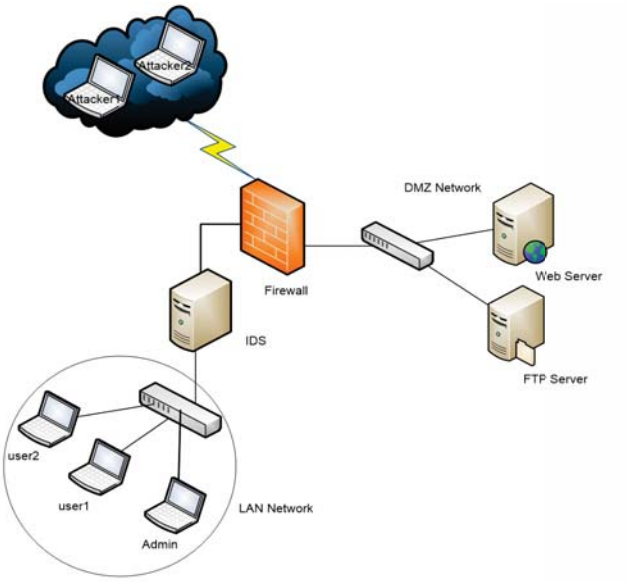

Fig. 1 Network System Design

IV. Proposed Dataset and System Setup

The proposed dataset creates a testbed that takes from the traffic of firewall, IDS, web server, and public attacks.

A. Proposed Network Testbed

The proposed testbed network design uses software-based firewall IPCop. The firewall is configured for External Network, Local Area Network (LAN) and De-Militarized Zone (DMZ) for public and local users access in Fig. 1. De-Militarized Zone (DMZ) is also added as an additional security layer for the LAN network. Web server and file server are accessed from local and public users in DMZ. The Firewall defines the rules for the three main zones. For public user access, the forwarding rules are required for web server access within the DMZ network. To make the LAN secure, rules are set to prevent malicious attacks from invading the public and the DMZ network. The firewall rule creates only what is needed and focuses not only on security but also on performance. In the system implementation, the IDS is deployed with two NIC cards, one for external and the other for LAN card. The predefined rules related to firewalls are also applied in this IDS infrastructure.

The system testbed contains two ubuntu 20.04 machines as attacker1 and attacker2 for public network. The web server and ftp server are operated with OpenSuSE 15.1 in the system implementation. Admin and User PCs are setup with OpenSuSE 13.2 in the LAN. The number of services such as DNS, HTTP, and SSH servers are deployed and implemented in the Web Server.

B. Network Traffic in Testbed

DoS traffic are created by using hping3 tool for the network traffic between the public network and Web server. The public network to web server for DoS attack traffic using **hping3** tool. For normal traffic, traffic is captured by accessing Google, Facebook, and Amazon sites. Attack or normal traffic is captured by **tcpDump** tool on IDS VM to create a pcap file.

The pcap file is loaded to Wireshark that selected filter out traffic of TCP. The comma specified file format (.csv) is created by manually aggregating the values of features depending on the destination host of the package range. DoS attack traffic is captured at 3s, 5s, 10s, and 15s time and is generated according to different DoS instances weight and package range for csv file. Traffic analysis of performance obtained based on DoS time and Machine Learning Classifiers.

C. Applied Machine Learning Classifier in System

The system used six classifiers as Support Vector Machine (SVM), Logistic Regression (Logistic), J48, JRip, Random Tree, and Multiclass Classifier. Category of Classification: Classification belongs to the category of supervised learning where the targets also provided with the input data. SVM is an efficient tool widely used in the multiclass classification [15]. The first sequential minimal optimization algorithm for SVM is implemented by John for training a support vector classifier. Le Cessie and Van Houwelingen (1992) [4] illustrated Logistic Regression. Some are modified compare to [4] because Logistic Regression not divided with instance weight [16].

J48 is developed by Weka project team. C4.5 is an extension of ID3 (Iterative Dichotomiser 3) algorithm. It is applied to improve accuracy and performance in anomaly detection [21].

J48 and Random Tree are the decision tree algorithms that widely used in machine learning [23]. Random Tree is allowed to estimate of class probabilities and constructed a tree that considers k randomly chosen attributes at each node. The Machine Learning tasks applied Multiclass classifier that use to get valid output code and to improve accuracy. JRip is one of the data mining algorithms and is developed by Chohen to classify accuracy. The limitation of JRip has memory consumption (from Weka).

Most models of machine learning have over-fitting problems, which are conducted to prevent this from happening in k-fold cross validation. The dataset is randomly partitioned into k mutually exclusive subsets those are approximately equal size in each and one is kept for testing while others are used for training. This process is iterated throughout the whole k folds. The system is k = 10.

D. Overview of Existing Dataset

In intrusion detection field, KDD Cup 99 dataset [10], [11] has been used for a long time as evaluation data of intrusions. It contains 41 features labeled as normal or attack. However, there is a fatal problem in that the KDD Cup 99 dataset cannot reflect current network situations and the latest attack trends [3]. It was developed over a virtual network environment. Four types of attacks as Dos, R2L, U2R, Probe are used in KDD Cup 99.

Kyoto 2006+ has a total of 24 features, 14 of which are selected by KDD Cup 99 dataset and 10 features are further included in the analysis of NIDSs [3]. Kyoto 2006+ datasets on real network traffic and ignores the inclusion of redundant

TABLE II
DATASET FEATURES APPLIED IN SYSTEM

No.	Feature	Description
1	Dst_port	Destination port
2	Dst_IP	Target IP address
3	Total_Inpkt	Total Inbound packages to destination host
4	Total_Outpkt	Total Outbound packages from destination host
5	Inpkt_bytes	Inbound packages bytes to Destination
6	Outpkt_bytes	Outbound packages bytes from destination
7	Total_InOut_pkt	Total packages to/from destination host
8	Inpkt_bits/s	Inbound packet bits/s
9	Outpkt_bit/s	Outbound packet bits/s
10	Protocol	Protocol as TCP or UDP
11	Service	Service types as http, ftp
12	Min_pktlen	Minimum packet length in the packet range
13	Max_pktlen	Maximum packet length in the packet range
14	Avg_pktlen	Average length of packet that fall in the range
15	Inout_count	Number of packets count with source and destination IP in this range
16	Class	Describe normal or attack

features. It composed two types of traffics such as normal and attack [14], [17].

NSL-KDD (2009) dataset features extract selected from KDD Cap 99 to improve the accuracy of IDS [3], [12]. It has 41 features that not included redundant duplicate record for training and testing data and not perfect for representing for existing real network. NSL-KDD Cup 99 dataset are composed of five main classes [13], [17]. There are Normal, Denial of Service (DoS), Remote to User (R2L), User to Root (U2R), and Probing (Probe).

The CICIDS-2017 dataset obtains a huge of traffic and a large number of 78 features to be observed for anomalies detection [13]. It composed of two traffics normal (Benign) and attack that is complexed type and improved performance of IDS on this dataset [12]. CICIDS-2017 included 7 attack types as Brute force, PortScan, Botnet, Dos, DDoS, Web, Infiltration [14].

E. Dataset with Extract Feature

The proposed dataset now included 16 keys features in Table 2. The dataset derived by extracting some features as destination port, minimum packet length and maximum packet length [12], [14] from CICIDS-2017 and added other features to reduce false positive rate. These features are considered depending on the destination according to the packet range, such as destination ports, destination inbound/outbound packets and, etc. Features are not specifically designed for the flag feature. Adding six TCP flag feature does not significantly improve the performance. Therefore, instead of applying those features, synchronous(syn), syn_ack, retransmission, reset(rst) are categorized into package ranges and are considered with respective features in normal and attack traffic.

V. IMPLEMENTATION AND EVALUATION

For huge network traffic, it is normally difficult to analyze the data. The developed system applied the WEKA (Waikato Environment for Knowledge Analysis) data mining tool to prove the performance of the system. The proposed dataset applied 10-folds cross validation of the training and testing to classify better performance.

TABLE III
PERFORMANCE WITH CLASSIFIERS IN PORTSCAN ATTACK

Detection Classifier	PortScan			
	TP	FP	PRC	REC
Logistic	0.989	0.001	0.99	0.989
SVM	1.00	0.00	1.00	1.00
J48	0.996	0.179	0.965	0.966
JRiP	1.00	0.09	1.00	1.00
Random Tree	0.977	0.09	0.977	0.977
Multiclass Classifier	0.989	0.001	0.990	0.989

The proposed system implement performance with IDS by machine learning classifier as Logistic Regression (Logistic), Support Vector Machine (SVM), J48, JRip, Random Tree, and Multiclass Classifier in Tables III and IV. The True Positive (TP), False Positive (FP), Precision (PRC), Recall (REC) proved performance with each classifier. Due to the high false positive rate, it is possible that the actual attack could not be detected and the important attacks were not recognized. So, the proposed system can reduce the false positive rate in all classifiers except J48 and Random Tree, when calculating the false positive rate to detect normal and PortScan attack.

TABLE IV
PERFORMANCE WITH CLASSIFIERS IN DOS ATTACK

Detection Classifier	DoS			
	TP	FP	PRC	REC
Logistic	0.993	0.004	0.993	0.993
SVM	0.990	0.008	0.990	0.989
J48	0.993	0.009	0.993	0.993
JRiP	0.994	0.004	0.994	0.994
Random Tree	0.993	0.002	0.994	0.993
Multiclass Classifier	0.992	0.005	0.992	0.992

In Table IV, the higher false positive rate is only 9% in J48 and SVM, and significantly lower in the other classifiers. Table V uses the following equation to calculate the accuracy of DoS and PortScan attack.

$$Accuracy = \frac{TruePositive}{TruePositive + FalsePositive} \tag{1}$$

In the DoS attack, the Random Tree has the highest accuracy of 99.8%, followed by Logistic and JRiP at 99.6% in Table V. The proposed system can see that the J48 and Random Tree classifier has the lowest accuracy 84.8% and the rest of the classifier has good accuracy in PostScan attack.

A. Comparison of Proposed Dataset and Existing Dataset

The proposed dataset is based on the CICIDS2017 dataset. Some features are taken from CICIDS2017 and some features

TABLE V
ACCURACY OF DETECTION ON DOS AND PORTSCAN ATTACK

Detection Classifier	Accuracy %	
	DoS	PortScan
Logistic	99.598	99.89
SVM	99.198	100
J48	99.1	84.766
JRiP	99.599	100
Random Tree	99.799	91.57
Multiclass Classifier	99.499	99.899

have been added. This paper tested 26 features of CICIDS2017 with two classifiers to know performance. The 26 features shows in Table VI.

TABLE VI
EXTRACTED 26 FEATURES FROM CICIDS2017

No.	Feature Name Extracted from CICIDS2017
1	Destination Port
2	Flow Duration
3	Total Fwd Packets
4	Total Backward Packets
5	Total Length of Fwd Packets
6	Total Length of Bwd Packets
7	Fwd Packet Length Max
8	Fwd Packet Length Min
9	Fwd Packet Length Mean
10	Fwd Packet Length Std
11	Bwd Packet Length Max
12	Bwd Packet Length Min
13	Bwd Packet Length Mean
14	Bwd Packet Length Std
15	Bwd Header Length
16	Fwd Packetss
17	Bwd Packetss
18	Min Packet Length
19	Max Packet Length
20	Packet Length Mean
21	Packet Length Std
22	Packet Length Variance
23	Average Packet Size
24	Avg Fwd Segment Size
25	Avg Bwd Segment Size
26	Fwd Header Length

The data with a high number of features require a lot of time, as well as resource consumption and computational complexity for data analytics [12]. Table VII proved the performance of FP, FN, PRC, and REL with two classifiers. Based on the 26 features of the CICIDS2017 dataset, the Logistic accuracy for the PortScan attack is 98.6%, and SVM is 94.3%. In this paper, when we compare Table VIII with Table III, we can see that the false positive rate is significantly higher.

TABLE VII
PERFORMANCE OF TWO CLASSIFIERS WITH CICIDS2017

Detection Classifier	PortScan			
	TP	FP	PRC	REC
Logistic	0.988	0.014	0.988	0.988
SVM	0.953	0.058	0.957	0.953

In summary, when calculating performance using six classifiers in the proposed dataset, only one J48 classifier has low accuracy and the other classifiers Logistic, SVM, JRiP, Random Tree and Multiclass Classifier have more than 99%, respectively. And also, the true positive rate is good with all classifiers.

VI. CONCLUSIONS

The proposed system implemented the dataset with a network testbed to prove performance. The system testbed included especially devices as firewall and IDS. This paper focuses on reducing false positive rates and improving accuracy, as false positives may not be known to the real attack. The system analyzes detection performance using six machine learning classifiers as Logistic Regression, Support Vector Machine, J48, JRiP, Multiclass Classification and, Random Tree. The system analyzes the performance of 16 features with six classifiers. The later work will focus on choosing some machine learning classifiers, adding other attacks, and using the python language to improve performance.

ACKNOWLEDGMENT

We are extremely grateful to our supervisor Prof: Dr Zin May Aye, University of Computer Studies, Yangon (UCSY), for her invaluable guidance, supervision, patience, encouragement during the period of this research time. We would like to special thank Information and Communication Technology Research Center (ICTRC) for supporting of research materials.

REFERENCES

[1] A. Alhomoud, R. Munir, J. P. Disso, I. Awan, "Performance Evaluation Study of Intrusion Detection Systems", Procedia Computer Science 5, published by Elsevier Ltd, pp. 173-180, 2011.
[2] H. H. Yi, Z. M. Aye, "Awareness of Policy Anomalies with Ruled-Based Firewall", ProMAC 2019, pp. 678-686.
[3] S. Jungsuk, T, Hiroki, and O. Yasuo, "Statistical nalysis of Honeypot Data and Building of Kyoto 2006+ Dataset for NIDS Evaluation", 1st Workshop on Building Analysis Datasets and Gathering Experience Returns for Security (BADGERS 2011), April, 2011.
[4] le Cessie, S. and van Houwelingen, J.C. (1992). "Journal of the Royal Statistical Society. Series C (Applied Statistics)", Ridge Estimators in Logistic Regression. Applied Statistics, Vol. 41, pp. 191-201, 1992.
[5] S. Mukkamala, G. Janoski, A. Sung "Network Intrusion Detection with Feature Selection Techniques using Machine-Learning Algorithms", International Journal of Computer Applications, Vol.150, no.12, 2016.
[6] H. H. Yi, Z. M. Aye, "Security Awareness of Network Infrastructure: Real-time Intrusion Detection and Prevention System with Storage Log Server", The 16th International Conference on Computer Application, 2018, pp. 678-686.
[7] P. Tao, Z. Sun, and et. al, "An improved intrusion detection algorithm based on GA and SVM", IEEE, 2018.
[8] H. Liao, C.R. Lin, and Y. Lin, K. Tung, "Intrusion detection system: A comprehensive review", Journal of Network and Computer Applications 36, pp 16-24, 2013.
[9] M. Bijone,"A Survey on Secure Network Intrusion Detection & Prevention Approaches", American Journal of Information System, vol. 4, No.3, pp. 69-88, 2016.
[10] M. Urvashi, and A. Jain, "A survey of IDS classification using KDD CUP 99 dataset in WEKA", International Journal of Scientific & Engineering Research, Vol.6, Issue 11, Nov, 2015.
[11] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, 1999.
[12] Kurniabudi, D. Stiawan, and et al. "CICIDS-2017 Dataset Feature Analysis with Information Gain for Anomaly Detection", IEEE, July, 2019.

[13] P. S. Pervez and D. M. Farid, “Feature selection and intrusion classification in NSL-KDD cup 99 dataset employing SVMs”, The 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2014), Dec, 2014.

[14] A. Thakkar and R. Lohiya. “A Review of the Advancement in Intrusion detection Datasets”, Procedia Computer Science, Vol-167, pp. 636-645, 2020.

[15] Y. Li a, J. Xia, et. al “An efficient intrusion detection system based on support vector machines and gradually feature removal method”, Expert System with Applications, pp. 424-430, 2012.

[16] https://www.dbs.ifi.lmu.de/ zimek/diplomathesis/implementations/ EHNDs/doc/weka/classifiers/functions/Logistic.html, Extract from Dec-6, 2020.

[17] D. Protic, “Review of KDD Cup ’99, NSL-KDD and Kyoto 2006+ datasets”, Vojnotehnicki Glasnik/ Military technical Courier, Vol. 66, pp. 560-596, 2018.

[18] N. Akhyari, and S. Fahmy, “Design of a Network Security Tool Using Open-Source Applications”, Australian Journal of Basic and Applied Sciences, pp. 40-46, 2014.

[19] M. Sumner, E. Frank, and M. Hall, “Speeding Up Logistic Model Tree Induction”, European Conference on Principles of Data Mining Knowledge Discovery (KDPP), pp. 675-683, 2005.

[20] S. Hwang, K. Cho, and et.al “Traffic Classification Approach Based on Support Vector Machine and Statistic Signature”, Springer, pp. 332-339, 2013.

[21] S. Aljawarneh, M. B. Yassein, and M. Aljundi, “An enhanced J48 classification algorithm for the anomaly intrusion detection systems”, Cluster Comput., pp. 117, 2017.

[22] R. Chitrakar and H. Chuanhe, “Anomaly detection using Support Vector Machine classification with k-Medoids clustering”, 2012 Third Asian Himalayas International Conference, pp. 1-5, 2012.

[23] S. Mulay, and P. R. Davale, “Intrusion Detection System Using Support Vector Machine and Decision Tree”, International Journal of Computer Applications, vol 3, no.3, 2010.

Htay Htay Yi received the first masters degree, M.Sc. (physics) from Pathein University, in 2001 and second masters degree, M.A.Sc (Computer Engineer) from University of Computer Studies, Yangon (UCSY), in 2003. She is a lecturer in the Information and Communication Technology Training Institute (ICTTI), ICT Research Center. She certified CCNA and CCNA Security. She is currently working as an instructor of Network Development Course and Cisco Network Academy at ICTTI. And also she is a researcher at UCSY.

Zin May Aye received M.S. (2002) from Yangon Technological University, and the Ph.D. (Computer Hardware Technology) (2005) from UCSY. She is a Professor in the Faculty of Computer Hardware Technology, UCSY. She joined Cisco Net Academy in 2013 and passed the ITQ exam in 2016. She is now in charge of Cisco Network Lab in UCSY. She has been supervised Master thesis students for about ten years and published authorized and coauthored research papers based on the networking fields. In the network science area, she has now focused on the security tools and network exploring with routing and switching with Cisco devices.