# Machine Learning Based DoS Traffic Analysis on the Testbed Environment

Htay Htay Yi
*Information and Communication Technology Training Institude (ICTTI)*
Information and Communication Technology Research Center (ICTRC)
Yangon, Myanmar
htayhtayyee@ictresearch.edu.mm

Zin May Aye
*Faculty of Computer Systems and Technologies*
University of Computer Studies, Yangon (UCSY)
Yangon, Myanmar
zinmayaye@ucsy.edu.mm

***Abstract*— Today, malicious users are widespread and are frequently lengthening worldwide. So, network security becomes crucial in the domain of education, government, business, and other sectors with related network connections. The firewall filtering rules itself might cause network vulnerability due to the misconfiguration and order them. The system builds a network testbed using a firewall, and Intrusion Detection System (IDS) and then implements a dataset using DoS traffic and normal traffic from that testbed environment. It is needed to be tested various requirements as features, false positive rates, and accuracy based on datasets apply and built for DoS. The importance of features in the proposed dataset was tested using attribute evaluators and methods. The focus of this work is to improve the performance with two classifiers as Logistic Regression and Support Vector Machine. The system also selects the important features by classifying traffics according to times by machine learning methods.**

***Keywords— features, Intrusion Detection System, machine learning classifier, performance***

## I. INTRODUCTION

Network security becomes very important role in data and network system. Because malicious users and attacker are more and more increasable, especially in business and education. The network security violation is typically reported and analyzed centrally with a security event management system. The Firewall, Intrusion Detection System (IDS), and Intrusion Prevention System applies to have a way to monitor for signs of latent contravention, exploit, and imminent threats. An attacker applied DoS/DDoS to attack an authorized user from accessing a network service, accessing the internet [21]. Many researchers are using Classifier of Machine Learning to detect DoS/DDoS network traffic and sampling techniques to find the most useful methods for detection [21, 22]. In this work, the system creates a network testbed environment that include firewall, IDS and other devices to analysis DoS traffic. 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 is 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. 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 2 summaries of the related works of the previous authors. Section 3 presents the research methodology. Section 4 introduces the propose dataset and system setup. Section 5 approve the implementation and evaluation of the system with proposed dataset. Section 6 is the conclusion and future work of this paper.

## II. RELATED WORKS

The researchers are assisted to plan more effective NIDS. 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.

The [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. The author Peiying Tao, Zhe Sun and 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.

The [19] is modified old Logistic Regression Algorithm to reduce training time. S. Hwang, et al. [20] proposed a classification method using statistic signatures as direct

sequence of packet size based on SVM (Support Vector Machine) for application traffic.

## 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 composed 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 that will depend on the following topics: (i) What the features are those of the firewall give, (ii) What wage will be adjusted with the user's organization, (iii) How many budgets/funds (How much money) can be spent by organization when implementing the system. In this system, software-based firewall, as IPCop is chosen because its feature outshines and it is free cost.

### *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 sufficient to allow installation of various add-ons to strengthen 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 1.

TABLE 1. EXAMPLE OF INTERNAL TRAFFIC IN IPCOP

| Rule | Proto: | Src_Ip | Src_Port | Dest_IP | Dest_Port | Action |
|---|---|---|---|---|---|---|
| r1 | UDP | 192.168.235.50 | any | 192.168.137.100 | 53 | allow |
| r2 | TCP | 192.168.235.* | any | 192.168.137.* | 443 | deny |
| r3 | TCP | 192.168.235.* | any | 192.168.137.100 | 22 | deny |
| r4 | TCP | 192.168.235.50 | any | 192.168.137.100 | 443 | allow |
| r5 | TCP | *.*.*.* | 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 communication is found matching a rule, the system judges a critical event related to intrusion and reports an alert to the administrator or user [2]. The three modes that sniffer mode, packet recording mode, and intrusion detection system are applied in snort. The network packet is read by the program in the sniffer mode and present them on the console. The packet logger mode log packets to the disk from the program. 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 dependent on feasible of content analysis and pattern matching. The Snort rule has two portions: the rule header and rule option. An 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_post (option)

**Actions**: Snort supports several assemble actions. A rule is matched with the direct log of the packet, using 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 field 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 each option that activated in a snort rule and runs through the equivalent with plug-in to scan against the packet.

In addition, Snort has a reporting mechanism that collects alerts from those reports and sends them to a Syslog server or a database [6]. In IPCop firewall, Snort is altogether useful to detect a server on DMZ and internal network. The convenience of an IDS is added in the network system that we can know which is flowing on the network and attempt of any malicious traffic.

## IV. PROPOSED DATASET AND SYSTEM SETUP

The proposed dataset deployed in the testbed 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 user's access in figure 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.

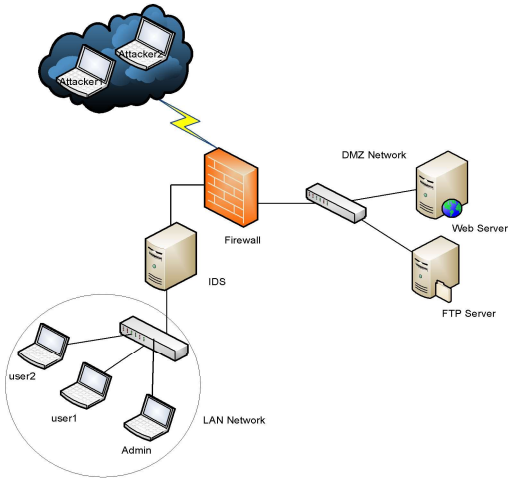


Fig 1. Network System Design

### 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 Virtual Machine 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 two classifiers as Support Vector Machine (SVM) and Logistic Regression (Logistic). 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 the author John for training a support vector classifier. The authors S. Le Cessie and J. C. Van Houwelingen (1992) [4] illustrated Logistic Regression. Some are modified that compared to the paper [4] because Logistic Regression not divided with instance weight [16].

In most models of machine learning, the common problem is Over-fitting. k-fold cross-validation can be supervised to check for Over-fitted problem. The dataset is divided into desultory into k, one of which is kept for training while the other is applied for testing. This process is repeated over an entire k- folds. k=10 is used for the system.

### D. Overview of Existing Dataset

In intrusion detection field, KDD Cup 99' dataset [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 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 Features

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, synchronization

(syn), synchronization and acknowledgement (syn_ack), retransmission, reset (rst) are categorized into package ranges and are considered with respective features based on time in normal and attack traffic [23].

TABLE 2. 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 |

## V. IMPLEMENTATION AND EVALUATION

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

TABLE 3. NORMAL AND ATTACK DATA IN TRAINING DATASET

| Time | Normal (%) | Attack (%) |
|---|---|---|
| 3s | 78 | 22 |
| 5s | 70 | 30 |
| 10s | 73 | 27 |
| 15s | 86 | 14 |

In the above Table 3, show the percentage of normal and attack training data with EM (expectation maximization) cluster. The performance describes with SVM and Logistic Regression in Table 4 and 5.

TABLE 4. LOGISTIC CLASSIFIER RESULT WITH PERFORMANCE

| Time | False Positive | Precision | Recall | True Positive | Accuracy (%) |
|---|---|---|---|---|---|
| 3s | 0.061 | 0.953 | 0.946 | 0.946 | 94 |
| 5s | 0.004 | 0.991 | 0.991 | 0.991 | 99.5 |
| 10s | 0.012 | 0.97 | 0.96 | 0.96 | 98.7 |
| 15s | 0.006 | 0.973 | 0.967 | 0.967 | 99.3 |

With 5s time, false positive rate is 0.4%, accuracy is approximately 99.6% in SVM and Logistic regression result but other time is little bit difference. False positive rates are lower when SVM is compared to Logistic regression in other time. When classifying with SVM classifier using KDD-CUP-99 dataset [10], the false positive rate is 1.11% and can further reduce the false positive rate from the proposed Dataset.

TABLE 5. SVM CLASSIFIER RESULT WITH PERFORMANCE

| Time | False Positive | Precision | Recall | True Positive | Accuracy (%) |
|---|---|---|---|---|---|
| 3s | 0.052 | 0.989 | 0.989 | 0.989 | 95 |
| 5s | 0.004 | 0.991 | 0.991 | 0.991 | 99.6 |
| 10s | 0.00 | 1.00 | 1.00 | 1.00 | 100 |
| 15s | 0.00 | 1.00 | 1.00 | 1.00 | 100 |

Figure 2 shows accuracy using two machine learning classifiers, SVM and Logistic regression, on normal and attack traffic taken from each DoS time.

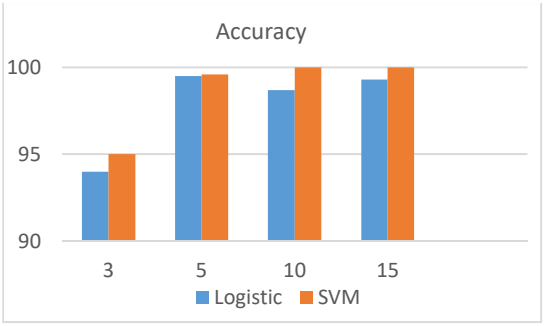


Fig 2 Accuracy output of testing data

The performance of these classifiers analyzed with the help of classified instances is correct or not and the result can be shown in Table 6. Both classifiers have a lot of valid data classified and SVM probably classified instance is better.

TABLE 6. DATA CLASSIFIED RESULT OF TWO CLASSIFIER

| Stratified cross-validation | Logistic (%) | | | | SVM (%) | | | |
|---|---|---|---|---|---|---|---|---|
| | 3s | 5s | 10s | 15s | 3s | 5s | 10s | 15s |
| Correctly Classified Instances | 94.62 | 99.07 | 96.04 | 96.67 | 98.92 | 99.07 | 100 | 100 |
| Incorrectly Classified Instances | 5.38 | 0.93 | 3.96 | 3.33 | 1.08 | 0.93 | 0 | 0 |

The system calculates the probability of training set with various Attribute Evaluators and two method as BestFirst and Ranker in Table 7. Details of each features listed in Table 2.

TABLE 7. PROPERLY ATTRIBUTES WITH METHODS AND ATTRIBUTE EVALUATOR

| Attribute Evaluator + Method | Selected Attributes |
|---|---|
| CfsSubsetEval + BestFirst | 1,4,8,9,12,13,14,15 |
| ClassifierAttributeEval + Ranker | 15,4,5,3,7,2,6,8,14,12,13,11,9,10,1 |
| CorrelationAttributeEval + Ranker | 1,15,11,13,7,3,12,14,4,2,5,8,6,9,10 |
| GainRatioAttributeEval + Ranker | 8,13,14,9,1,4,3,11,15,12,5,7,6,2,10 |
| InfoGainAttributeEval + Ranker | 8,2,13,14,9,15,11,1,4,7,3,12,5,6,10 |
| OneRAttributeEval + Ranker | 13,2,8,14,9,1,11,3,15,4,12,7,5,6,10 |
| ReliefFAttributeEval + Ranker | 11,1,2,12,15,13,7,4,3,14,5,6,8,9,10 |
| SymmetricalUncertAttributeEval + Ranker | 8,13,14,9,1,15,11,4,3,12,7,5,6,2,10 |

Of the features selected by the Attribute Evaluators and Methods in the Table 6. The 10 features as 1,4,7,8,9,11,13,14,15 are the most selectively identical with full training set. However, when analyzing the most identical attributes in 5s, SVM classifier may not be noticeable. In Logistics, the false positive rate increased by 2% and the accuracy decreased by 1.1%. Considering the average of the two classifiers, the system performance does not decrease significantly. In this system, it will be considered the full features to reduce false positive rate and good accuracy for all classification methods.

TABLE 8. AVERAGE PERFORMANCE ON TWO CLASSIFIERS

| Time | False Positive Rate | Accuracy |
|---|---|---|
| 3s | 0.06 | 94.5 |
| 5s | 0.004 | 99.6 |
| 10s | 0.006 | 99.4 |
| 15s | 0.003 | 99.7 |

The current system's proposed dataset uses the same features and analyzes the value of the features at different times. In Table 8, The average values of false positive rate and accuracy on these two classifiers. Traffic of 5s and 15s time have the lowest false positive rate of 0.004 and 0.003, while highest accuracy of 99.6% and 99.7, respectively. In this work, the quality of a feature selected rather than the instance involved in implementing a dataset depends on the value of that feature. In addition, the performance of the system is determined by the value difference and validity of the selected features on normal and DoS traffics.

## VI. 14CONCLUSION AND FUTURE WORK

The system proposed dataset by using some features of existing dataset and adding some features on network testbed as firewall and IDS. This work classifies the network traffic by using machine learning. The core of the system is now proven for performance using the proposed dataset, for example, higher accuracy and lower false positive rate on DoS and normal traffic. In the network system, we extend the instances of the dataset and then create a good dataset based on features and their values in the future work. Also, we will add other attacks and algorithms to improve the performance of Intrusion Detection.

## 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] A. Fadil, I. Riadi, and et.al "Review of Detection DDoS Attack Detection Using Naive Bayes Classifier for Network Forencisc", Bulletin of Electrical Engineering and Informatics, Vol, 6, No. 2, pp. 140-148, 2017.

[22] H. Hadian. Jazi, H. Gonzalez, and et. al, "Detecting HTTP-based application layer DoS attacks on web servers in the presence of sampling", Computer Networks (International Journal of Computer and Telecommunications Networking), Vol. 121, pp. 26-36, 2017.

[23] A. H. Lashkari, G. D. Gil, M. S. I. Mamun, and Ali A. Ghorbani ,"Characterization of Tor Traffic using Time based Features", In Proceedings of the 3rd International Conference on Information Systems Security and Privacy (ICISSP 2017), pp. 253-262, 2017.

## ACKNOWLEDGMENT

We would like to special thank our department ICTRC for its dedicated support of research materials. We are extremely thankful to our supervisor, Prof: Dr Zin May Aye (UCSY), for her encouragement, guidance, perseverance on the research path.