International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 3, Issue 7, pp 22-28, July-2019

Machine Learning Model for Denial of Service Network Intrusion Detection

[1]Stephen Ngure Gitonga, [2]Otanga Ananda Daniel, [3]Stephen Makau Mutua

[1,2]Information Technology Department, Masinde Muliro University of Science and Technology, Kakamega, Kenya
[3]Computer Science Department, Meru University of Science and Technology, Meru, Kenya

***Abstract* - Every organization strives to ensure its network is secured from all sorts of attack and is available to its intended users all the time. Network Intrusion Detection Systems (IDS) is one of the techniques used to detect and classify abnormal network access. Therefore, IDS should always be up and up to date with intrusion types and techniques. The most common network intrusion is denial of service (DOS). This study seeks to find out the best machine learning tools that can be used to detect a DOS attack. Using Knowledge Discovery and Data Mining (Knowledge Discovery in Databases) KDD dataset, three machine learning tools are evaluated to find out their performance. The findings show that MLP performs better compared to SVM and KNN.**

Keywords: IDS, Neural Network, DOS, SVM, KNN.

I. INTRODUCTION

An intrusion is an effort targeted towards compromising the confidentiality, integrity, availability of a computer or network resource, or to circumvent security instruments of a computer network or system. Network intrusion is a hazardous activity since it grants unauthorized access to important organizational data. Network security personnel are responsible for providing the required protection to network users to avert any potential loss of data or organizations integrity[1]. Therefore, there is need for a well configured security system that is robust and dependable. Network administrators focus in providing protected networked environment for all user accounts, passwords, personal files and network resources like printers, fax machines among others. Neutralizing unwanted network activities is core to ensuring a secure network. Apart from ensuring data security and enhancing network privacy activities, it also helps in avoiding situations that are hazardous[2].Network Intrusion Detection Systems (IDS) are implemented to ensure security. IDS monitors network traffic and identify any malicious network access or use and alert network administrators about any malicious network access.

IDS help in discovery, determination and identification of unauthorized network use, replication of services, and alteration of network resources and destruction of information systems[3]. Cyber security is a set of technologies, techniques and processes designed to protect networks, computers, data and programs from unauthorized access, attack, alteration or destruction. Network security breach could be internal (from within the organization), or external (from outside the organization). IDS is a component of a cyber security system that help in ensuring a secure network[4].

IDS is supported by three major cyber analytics namely; hybrid, signature based and anomaly based[2]- [5]. Signature based detection system involves analysing network traffic for a series of bytes or packet sequences known to be an anomaly[6].Signature based techniques are responsible for detecting attacks that are known using the signatures of those attacks. They are effective in detecting attacks that are known without generating plenty of false alarms[7]. Database needs to be frequently updated with signatures and rules. Signature based method cannot detect novel attacks. The major Signature based detection system limitations is, relatively easier to understand and develop especially with the knowledge of the network behaviour required to be identified[8]. A signature is created for every attack and only detects that particular attack only. Detecting new attacks in signature based technique that have never occurred in training data is difficult.

The anomaly based detection scheme analyses the network behaviour[9]. This scheme is capable of detecting anomaly behavior by analyzing the high volume traffic, a surge in traffic from a specific host or to a specific host, load imbalance in the network. Ability to detect novel (zero-day) attack makes it more appealing. Customizing network, systems or application makes it difficult for unauthorized users from gaining access[10]. The limitation of anomaly based scheme is that if the malicious behavior falls within normal network behavior then it is not detected as an anomaly. The major advantage is that it easily detects a new attack that behaves differently from normal traffic patterns. Neural

ISSN (online): 2581-3048
Volume 3, Issue 7, pp 22-28, July-2019

network applications has been taken into account of both[11] the anomaly detection and signature based model.

Hybrid techniques combine anomaly detection and signature based. They increases the detection rates of the attacks that are known and reduces the unknown intrusions false positive rate (FP) many pure anomaly detection schemes were not discovered when an intense literature review was done[12]. Instead, most methods discovered were hybrid. In data mining and ML both hybrid and anomaly schemes are classified together. Any other IDS division, host-based or network-based is basically based on where they look for intrusive behavior[13]. A host-based IDS monitors process and file activities related to the software environment associated with a specific host. A network-based IDS identifies intrusions by monitoring traffic through network devices.

Although the most common network intrusion is Denial of Service (DOS), IDS researchers have not investigated how to best detect whether an attack is a DOS or not[14]. In this paper, we investigate three machine learning tools and how they can detect a DOS attack. A DOS attack can be classified as back, land, Neptune, pod, smurf, teardrop, mailbomb, processtable, udpstorm, apache2 or worm[15]. In KDD dataset, the most common DOS attacks are smurf, Neptune, back and teardrop[16]. This study investigates how to detect and classify a DOS attack as either of the four common attacks in the experimental dataset.

II. RELATED WORK

A) Instrusion Detection Systems

Intrusion Detection System (IDS) is defined as a protection system that monitors computers or networks for unauthorized activities based on network traffic or system usage behaviors, thereby detecting if a system is targeted by a network attack such as a denial of service attack[17]. In response to the attacks identified on the network, IDS can alert the relevant authorities to take corrective actions. Neutralizing unwanted network activities is core to ensuring a secure network. Apart from ensuring data security and enhancing network privacy activities, it also helps in avoiding situations that are hazardous

Software and hardware are combined together by IDS to help in detecting network attacks to protect systems access from unauthorised users. Network intrusion detection is categorised as either signature based or anomaly based detection system. Signature based detection system involves analysing network traffic for a series of bytes or packet sequences known to be an anomaly[18]. The major Signature based detection system limitations is, relatively easier to understand and develop especially with the knowledge of the network behaviour required to be identified. A signature is created for every attack and only detects that particular attack only. Detecting new attacks in signature based technique that have never occurred in training data is difficult[19].

The anomaly based detection scheme analyses the network behaviour. This scheme is capable to detect anomaly behavior by analyzing the high volume traffic, a surge in traffic from a specific host or to a specific host, load imbalance in the network[20]. The limitation of anomaly based scheme is that if the malicious behavior falls within normal network behavior then it's not detected as an anomaly. The major advantage is that it easily detects a new attack that behaves differently from normal traffic patterns. Neural network applications have been taken into account of both the anomaly detection and signature based model.

Intrusion detection system (IDS) help determine, discover and identify alteration, access from unauthorized users, duplication and destruction of information systems[21]. To identify security attacks on the network both from within and outside organization, IDS are used to help analysing and gathering network data. IDS provides important network defence to most network security architectures. Some of the limitations IDS face includes; high false alarm rates and low detection rates that can miss intrusion attacks and sometimes classifying normal connection as an attack[22]. This denies authorized users from accessing network resources when required. These problems are as a result of attacks sophistication and their intended similarities to normal behaviour.

Most IDS use machine learning (ML) tools to achieve detection capability that is high for automation to ease user from task of constructing signatures of attacks and novel attacks[23]. Theoretically, it is possible for a ML algorithm to achieve the best performance, that is, it can maximize detection accuracy and minimize the false alarm rate. The commonly used supervised learning algorithms in IDS are decision trees due to its accuracy, simplicity, fast adaption and high detection. Artificial Neural Networks (ANN) is another method that models linear and non-linear patterns and performs highly well[24]. The resulting model can generate a probability estimate of whether given data matches the characteristics that it has been trained to recognize. IDS that are based on ANN usually achieve a lot in detecting difficult tasks.

Support Vector Machines (SVMs) are also a good candidate for intrusion detection systems which can provide real-time detection capability, deal with large dimensionality

ISSN (online): 2581-3048
Volume 3, Issue 7, pp 22-28, July-2019

of data. SVMs plot the training vectors in high dimensional feature space through nonlinear mapping and labeling each vector by its class[25]. The data is then classified by determining a set of support vectors, which are members of the set of training inputs that outline a hyper plane in the feature space. SVMs are scalable as they are relatively insensitive to the number of data points.

B) Machine Learning Tools for Intrusion Detection

1) Artificial Neural Networks

Artificial Neural Networks (ANNs) are brain composed and inspired artificial neurons which are interconnected together, and able to compute on their inputs. In the first layer of network, neurons activate the input data whose output is the input to the second layer of neurons in the network[26]. The output of every layer is passed to the next layer and the outcomes are passed to the last layer. Hidden layers are the layers that separate input and output layers. The final classification category of the output layer is generated when ANN is used as a classifier.

The popularity of ANN classifiers ended in 1990s when SVMs were invented. ANNs were perception based. ANNs always experience long runtimes caused by local minima. An increase in the number of ANN features increases its learning runtime unlike an SVM. Nonlinear models are easily generated by the ANN with one or more hidden layers. Multi-Layer perception (MLP) is an improved ANN alternative[27]. MLP has made ANN IDS tools more efficient and accurate in terms of normal and detection communication. Compared to the traditional mechanisms, MLP-ANN shows detection outcomes better and overcomes the limitation of low-frequency attacks[28]. MLP-ANN can easily define the type of attacks and classify them. This feature allows system to predefine actions against similar future attacks.

2) Support Vector Machines (SVM)

The SVM is a classifier based on finding a separating hyperplane in the feature space between two classes in such a way that the distance between the hyperplane and the closest data points of each class is maximized[29]. The approach is based on a minimized classification risk rather than on optimal classification. SVMs are well known for their generalization ability and are particularly useful when the number of features, m, is high and the number of data points, n, is low (m >>n)[30].

When the two classes are not separable, slack variables are added and a cost parameter is assigned for the overlapping data points. The maximum margin and the place of the hyperplane is determined by a quadratic optimization with a practical runtime of O(n2), placing the SVM among fast algorithms even when the number of attributes is high. Various types of dividing classification surfaces can be realized by applying a kernel, such as linear, polynomial, Gaussian Radial Basis Function (RBF), or hyperbolic tangent[31]. SVMs are binary classifiers and multi-class classification is realized by developing an SVM for each pair of classes.

3) J48

It is an extension of the algorithm, the most common classifier used to manage the database for supervised learning that gives a prediction about new unlabeled data, J48 creates Univariate Decision Trees[32]. J48 based used attribute correlation based on entropy and information gain for each attributes. J48 has been utilized in various field of study that includes; pattern recognition, machine learning, information extraction and data mining. J48 is capable of dealing with various data types' inputs; nominal, numerical and textual. It can build small trees and follows depth-first strategy, and a divide-and conquer approach[33].

4) K-Nearest Neighbour

K-Nearest Neighbors (KNN) is an algorithms used in Machine Learning for regression and classification problem. It utilizes a data and classifies new data points based on a similarity measures. For example, distance function. Classification is done by a majority vote to its neighbors. It uses data.

III. METHODOLOGY

This section describes the methodology used to detect DOS attack. The NSL-KDD dataset used in the experiment is described together with the technique used to sample both training and testing datasets. NSL-KDD dataset has some entries as text. Data pre-processing technique used to convert the textual data to numeric is also presented.

A) NSL-KDD dataset

This study used NSL-KDD dataset for training and testing DOS network intrusion detection. NSL-KDD is an improvement of KDD dataset. Unlike KDD dataset, NSL-KDD;

i. Has no redundant records in the training set as original KDD thereby ensuring that the classifiers are not biased
ii. There are no duplicate records in the test set

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 3, Issue 7, pp 22-28, July-2019

iii. The number of selected records from each difficult level group is inversely proportional to the percentage of records in the original KDD dataset.

The training set is made up of 21 different attacks out of the 37 present in the test set. The attack types are grouped into four categories; DOS, U2R, R2L, and Probe. Table 1 shows major attacks in all the categories.

TABLE 1
Mapping of Attack Class with Attack Type

Attack class	Attack type
DoS	Back, Land, Neptune, Pod, Smurf, Teardrop, Apache2, Udpstorm, Processtable, Worm (10)
probe	Satan, Ipsweep, Nmap, Portsweep, Mscan, Saint (6)
R2L	Guess_Password, Ftp_write, Imap, Phf, Multihop, Warezmaster, Warezclient, Spy, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Httptunnel, Sendmail, Named (16)
U2R	Buffer_overflow, Loadmodule, Rootkit, Perl, Sqlattack, Xterm, Ps (7)

This study focuses on classifying different types of DOS attack. The most common 4 DOS attacks are considered for this study. These are smurf, Neptune, back and teardrop attacks. Table 2 shows NSL-KDD dataset distribution.

TABLE 2
NSL-KDD Dataset Distribution

Category	Training set	Testing set
DOS	45,927	7,458
U2R	52	67
R2L	995	2,887
Probe	11,656	2,422
Normal	67,343	9,710
Total	125,973	22,544

B) NSL-KDD Dataset sampling

Although there are a number of NSL-KDD subsets, i.e the complete dataset, 20% and 10% of complete dataset, we chose 20% of the complete NSL-KDD dataset for training and the whole of the NSL-KDD testing dataset for validation of the DOS IDS model. Table 3 shows the distribution of sample for both training and testing sets used in this study.

TABLE 3
Sample Distribution

	Training	Testing	Total
Neptune	8282	4657	12939
Teardrop	188	12	200
Land	2	7	9
Smurf	529	665	1194
Pod	38	41	79
Back	196	359	555
Mailbomb	0	293	293
Processtable	0	685	685
Worm	0	2	2
Upstrom	0	2	2
Apache2	0	737	737

Note that, only four DOS classes with more than 100 training samples were considered for this study, these are Neptune, teardrop, smurf and back.

C) Data pre-processing and normalization

Classifiers to be trained require that each record in the train and test data be represented as feature vector of real numbers. Therefore, every symbolic (textual) feature in the sample was first converted to a numeric representation. For instance, in the NSL-KDD dataset, there are textual features such as network protocols (UDP, TCP, ICMP), service (FTP, HTTP, Telnet) as well as TCP status flag (RE, SF). All the textual features in the dataset were coded into numerical values. The resultant feature vectors are normalized using min-max normalization as

$$F_{norm} = \frac{x_i - min}{\max - min}$$

Where x_i is a feature at index i, min and max are minimum and maximum features in that particular feature vector. Normalization is done such that all the features have the values between $[0 - 1]$ to ensure that extremely large or small values do not introduce biases during the training. Pre-processing and normalization process is also applied to the test data before the trained model is validated.

D) Feature selection

Although every network connection in the dataset is represented by several features, all these features are not important to build a robust and reliable DOS IDS. It is therefore important to identify and select most informative and discriminative features in order to reduce training and testing

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 3, Issue 7, pp 22-28, July-2019

time as well as achieving higher intrusion detection rates. In this study, we use Correlation-Based Feature Selection (CFS) technique to select most informative and discriminative features that can be used to determine the type of DOS attack. CFS is based on the hypothesis that

"Good feature subsets contain features that are highly correlated with the class, yet uncorrelated with each other"

$$M_s = \frac{kr_{fc}}{\sqrt{k + k(k-1)r_{ff}}}$$

WhereM_s is the merit of a feature subset S containing k features, r_{cf} is the feature class correlation where $f \in s$, and r_{ff} is the feature-feature inter-correlation.

$$r_{fc} = feature\ mean - target\ class\ mean$$
$$r_{ff} = feature\ mean - feature\ correlation$$

We use the best-first search strategy where the CFS starts from an empty set and searches forward best-first with stopping criteria of five consecutive fully expanded non-improving subsets. A total of 17 features [2,3,4,5,6,8.10,12,23,25,29,30,35,36,37,38,40] were selected and used to train and validate a DOS IDS.

E) Classification of DOS Attacks

The experiment was done on a 2.53GHz intel core i3 machine with 4GB of RAM using WEKA 3.8 data mining tool. The memory heap on WEKA was set to be 1048MB with default parameter setting. Although DOS is further split into 11 sub-classes, only 4 sub-classes which had more than 100 records in the feature set were selected for this study. This is because the other sub-classes had very little, or no feature records in the training set, which meant that we could not learn any patterns for them. The selected sub-classes are Neptune, smurf, back and teardrop.

The DOS IDS was modeled as a one-vs-all problem where four models were trained to classify each of the four DOS attacks against the rest. The model is evaluated using accuracy model evaluation metric which is calculated as

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

For each of the four classes investigated. For instance, if an attack is of type Neptune and is classified as of Neptune attack it is considered to be true positive (TP), and if it's classified as any other attack it's considered to be a false positive (FP).

IV. RESULTS AND DISCUSSION

Four classifiers ANN-MLP, SVM, KNN and J48 were evaluated for DOS classification. TABLE **44** shows the accuracy achieved for each of the classifiers using all the 41 features and 17 CFS selected features

TABLE 4
Accuracy for different DOS classes

Classification Algorithm	Class Name	Accuracy (%) 41 Features	Accuracy (%) 17 Features
ANN-MLP	SMURF	95.3	97.2
	NEPTUNE	96.5	98.3
	BACK	94.8	97.4
	TEARDROP	94.7	95.7
SVM	SMURF	93.2	95.4
	NEPTUNE	94.4	95.1
	BACK	91.7	93.5
	TEARDROP	92.5	94.8
KNN	SMURF	74.6	76.5
	NEPTUNE	70.5	73.7
	BACK	71.3	73.6
	TEARDROP	72.7	75.2
J48	SMURF	88.4	91.5
	NEPTUNE	88.6	89.3
	BACK	89.5	91.8
	TEARDROP	87.7	90.3

The results in 4 shows that using 17 CFS selected features achieves better classification accuracies as compared to use of all the 41 features in the NSL-KDD dataset. As shown in 4, ANN-MLP performs better as compared to SVM, KNN and J48 classifiers. The class with highest classification accuracy is Neptune with 98.3% accuracy using ANN-MLP with 17 CFS selected features. Among the four investigated classifiers, KNN performs poorly with the lowest accuracy of 70.5% in classification of Neptune type of DOS attack. The lower accuracies observed in the back and teardrop DOS attacks could be attributed to fewer samples of these classes in the training set of the NSL-KDD dataset.

V. CONCLUSION AND FUTURE WORK

This paper presents a machine learning model for classification of DOS attacks as either Neptune, Smurf, Back or Teardrop. Four machine learning tools (ANN-MLP, SVM, KNN and J48) have been investigated using both 41 features and 17 Best-First CFS selected features. The findings show

ISSN (online): 2581-3048
Volume 3, Issue 7, pp 22-28, July-2019

that ANN-MLP performs better both using 41 and 17 CFS selected features as compared to SVM, KNN and J48. It was also found that SVM performs better compared to KNN and J48 with performing poorly as compared to the other three classifiers. The findings show that using 17 best-first CFS selected features improves classification accuracies by about 2% across all the classifiers.

This study recommends further analysis of NSL-KDD dataset to identify best features that can be used to classify each of the 21 attacks represented in the dataset. This study investigated classification of DOS sub-classes. We recommend further studies to classify sub-classes of the other 3 groups of attacks (U2R, R2L and Probe). A further investigation to determine best feature selection techniques for classification of network attacks is also recommended.

REFERENCES

[1] S. Sharma and R. K. Gupta, "Intrusion detection system: A review," *Int. J. Secur. its Appl., 2015.*

[2] C. F. Tsai, Y. F. Hsu, C. Y. Lin, and W. Y. Lin, "Intrusion detection by machine learning: A review," *Expert Systems with Applications.* 2009.

[3] E. Darra and S. K. Katsikas, "A survey of intrusion detection systems in wireless sensor networks," in *Intrusion Detection and Prevention for Mobile Ecosystems*, 2017.

[4] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, "A survey of intrusion detection techniques in Cloud," *Journal of Network and Computer Applications.* 2013.

[5] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Network Anomaly Detection: Methods, Systems and Tools," *IEEE Commun. Surv. {&} Tutorials*, vol. PP, pp. 1–34, 2013.

[6] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Network Anomaly Detection: Methods, Systems and Tools," *IEEE Commun. Surv. Tutorials*, 2014.

[7] C. Day, "Intrusion prevention and detection systems," *in Managing Information Security: Second Edition*, 2013.

[8] N. Hubballi and V. Suryanarayanan, "False alarm minimization techniques in signature-based intrusion detection systems: A survey," *Computer Communications.* 2014.

[9] V. Jyothsna, V. V Rama Prasad, and K. Munivara Prasad, "A Review of Anomaly based Intrusion Detection Systems," *Int. J. Comput. Appl.*, vol. 28, no. 7, pp. 26–35, 2011.

[10] P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," *Comput. Secur.*, vol. 28, no. 1–2, pp. 18–28, 2009.

[11] T. Shon and J. Moon, "A hybrid machine learning approach to network anomaly detection," *Inf. Sci. (Ny).*, 2007.

[12] M. A. Salama, H. F. Eid, R. A. Ramadan, and A. Darwish, "Hybrid Intelligent Intrusion Detection Scheme," *Springer Berlin Heidelb.*, 2011.

[13] M. A. Aydin, A. H. Zaim, and K. G. Ceylan, "A hybrid intrusion detection system design for computer network security," *Comput. Electr. Eng.*, 2009.

[14] G. G. Liu, "Intrusion Detection Systems," *Appl. Mech. Mater.*, 2014.

[15] E. Hodo et al., "Threat analysis of IoT networks using artificial neural network intrusion detection system," in *2016 International Symposium on Networks, Computers and Communications, ISNCC 2016*, 2016.

[16] M. Govindarajan and R. Chandrasekaran, "Intrusion detection using neural based hybrid classification methods," *Comput. Networks*, 2011.

[17] A.Tajbakhsh, M. Rahmati, and A. Mirzaei, "Intrusion detection using fuzzy association rules," *Appl. Soft Comput. J.*, 2009.

[18] G. P. Rout and S. N. Mohanty, "A hybrid approach for network intrusion detection," in *Proceedings - 2015 5th International Conference on Communication Systems and Network Technologies, CSNT 2015*, 2015.

[19] M. Ahmed, A. Naser Mahmood, and J. Hu, "A survey of network anomaly detection techniques," *Journal of Network and Computer Applications.* 2016.

[20] M. Teng, "Anomaly detection on time series," in *Proceedings of the 2010 IEEE International Conference on Progress in Informatics and Computing, PIC 2010*, 2010.

[21] J. Jabez and B. Muthukumar, "Intrusion detection system (ids): Anomaly detection using outlier detection approach," in *Procedia Computer Science*, 2015.

[22] D. E. Denning, "An intrusion-detection model," in *Proceedings - IEEE Symposium on Security and Privacy*, 2012.

[23] W. Lee and S. J. Stolfo, "A framework for constructing features and models for intrusion detection systems," *ACM Trans. Inf. Syst. Secur.*, 2000.

ISSN (online): 2581-3048
Volume 3, Issue 7, pp 22-28, July-2019

[24] T. Oladipupo, "Types of Machine Learning Algorithms," in *New Advances in Machine Learning*, 2010.

[25] A.Dey and A. S. Learning, "Machine Learning Algorithms : A Review," *Int. J. Comput. Sci. Inf. Technol.*, 2016.

[26] N. Gupta, "Artificial Neural Network," *Netw. Complex Syst.*, 2013.

[27] K. L. Du and M. N. S. Swamy, *Neural networks and statistical learning*. 2014.

[28] G. M. Khan, "Artificial neural network (ANNs)," in *Studies in Computational Intelligence*, 2018.

[29] W. L. Al-Yaseen, Z. A. Othman, and M. Z. A. Nazri, "Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system," *Expert Syst. Appl.*, 2017.

[30] Y. Ma and G. Guo, Support vector machines applications. 2014.

[31] B. Schölkopf, "An Introduction to Support Vector Machines," in *Recent Advances and Trends in Nonparametric Statistics*, 2003.

[32] Bhargava and Sharma, "Decision Tree Analysis on J48 Algorithm for Data Mining," *Int. J. Adv. Res. Decis. Tree Anal. J48 Algorithm Data Min.*, 2013.

[33] S. Drazin and M. Montag, "Decision Tree Analysis using Weka," 2010.

Citation of this Article:

Stephen Ngure Gitonga, Otanga Ananda Daniel, Stephen Makau Mutua, "Machine Learning Model for Denial of Service Network Intrusion Detection" Published in *International Research Journal of Innovations in Engineering and Technology (IRJIET)*, Volume 3, Issue 7, pp 22-28, July 2019.

****

