International Journal of Innovative Information Systems
& Technology Research 9(2):75-89, April-June, 2021

© SEAHI PUBLICATIONS, 2021 www.seahipaj.org ISSN: 2467-8562

An Improved Cyber Attack Mitigation Model for Detecting Malicious Network Traffic using DNNs

Mission Franklin[1] & Friday E.Onuodu[2]

School of Post Graduate Studies, Department of Computer Science[1],
Faculty of Natural and Applied Sciences, Ignatius Ajuru University of Education, Port Harcourt[1]
Department of Computer Science, Faculty of Science[2],
University of Port Harcourt, Choba, Port Harcourt[2]
Email: franklinmission@yahoo.com[1] ; gonuodu@gmail.com[2]

ABSTRACT

Cyber-attacks are a great challenge to the business community, military, health, education and other industry services providers whose infrastructure and national assets depends on IT assets and resources for their effective services delivery. And cyber protection systems deployed to prevent these attacks are limited to understanding signatures of attack vectors. This research focus is on the use of deep learning to study the available dataset of cyber-attack vectors and to model a system to predict, defend and prevent cyber-attack operations of information systems. The research explored available literatures on cyber attack vectors, systems and algorithms within the cyber space, as well as mitigation strategies adopted in artificial intelligence. A study of machine language techniques, artificial neural network and deep learning techniques and their suitability was investigated. Deep neural network algorithm and TensorFlow were used for the computational model, to aid the system to learn the dataset and test the performance of the model. Datasets were taken from NSL-KDD and KDD'99 and the samples where imported and classified for encoding with MATLAB before classification, training and testing. Conclusion on the model design and suitability of deep learning neural network for, predictability, detectability and preventability of malicious attack vectors in network traffic was made with recommendations for further research.

Keywords: Cyber Attacks, Mitigation Model, Deep Neural Networks, TensorFlow, MATLAB

INTRODUCTION

Today's world is a fully interconnected world. The world is so interconnected that things, people, devices and media are all interconnected. It is this concept of interconnection that has necessitated the ideology of internet of things. The interconnectivity of things and the internet has made the world truly a global village. With interconnectivity comes the business leveraging on technology to move businesses and services to other parts of the world (Hodo et al, 2017). This is what gives rise to globalization. The connectivity has made life easier and even more meaningful. Services can be provided and consumed from any part of the world as long as one is simply connected to the internet, through any of the internet aware devices, sensors, or things that has the capability to access the internet with other resources of the internet (El-Alfy et al, 2015).

As connectivity improves with increased bandwidth and cheaper data bundles, services become relatively cheaper and activities dramatically improves on the internet. This dramatic rise actually leads to an exponential increase in the community of users. This community of users with over a billion of connected users, devices and things have also sparked off criminal tendencies on the part of the user community. What is actually behind this tendency is traceable to many factors. The vulnerability of the internet architecture and its security protocols, availability of valuable resources, availability of system tools designed to manage sections of the internet and the ease of tweaking such tools for criminality.

The internet is now a source vulnerabilities, threats and risks to the user community, components, devices, people and resources. Criminal activities are now very common on the internet (El-Alfy et al, 2015). Different forms of attacks are perpetrated by cyber criminals who hack into different systems, networks and resources to expose the weaknesses of the system, causing breaches with potential losses in billions of dollars, mostly to financial institutions, military organisation and health institutions; violating privacy laws and causing problems to institutional IT assets, making these organisations suffer different kinds of losses ranging from financial disrepute and compliance issues (Kravchik et al 2018).

The cyber threats faced by these institutions are heightened by the enormity of the attacks that the institutions face daily. These risks associated with these institutions increases every day. Currently, cyber-attacks are of different proportions from different actors, both state and non-state actors, some are being classified as a state-sponsored tools for terrorism or warfare. Generally, cyber-attacks on internet-aware systems span; malware, phishing, password attack, DoS attacks, man in the middle (MITN), malvertising, eavesdropping, click jacking, - all these are intended to degrade the IT assets and resources of an organization, causing breaches and financial losses, even destroy the entire function of any resource. In some cases, state control resources like infrastructure grids for power systems, health services records, or credit card information for financial systems are compromised.

Being aware of the existence of these threats or attacks, there is need to provide a solution to mitigate these attacks, either reactively in detection or proactively in prevention of these threats.

Several methods for solving these problems have been proposed. However most of them have shown to be reactive. The intrusion detection system, firewalling, other reactive mechanism, etc. has not proven to be effective in proactively identifying the attacks and the motivations before the attacker eventually strikes (Wang, 2018). Every attacker adopts an approach before fully executing the plans, although there is not accurate patterns identification. But we do understand that attacks are in stages. Exploration, flick through, arbitrary code killing, access and intensification of data collection, exfiltration and exploitation and finally cleaning up, where the attacker cleverly does so. However, no matter the identification of the named steps, understanding the intentions and stopping the attacker before the strike is the ultimate goal of this paper. The proposition in this paper is a technique that is based on an deep artificial neural network (DANN) that would proactively analyse the flow of packets and pay loads as they transverse the network and effectively identify the eminent attacks on the network and its resources; so that network administrators and managers can foil the intended attack before it eventually occurs.

The concept of predict, detect and prevent (PDP) is a secure paradigm using various mechanism to achieve it; including MLA (Machine Learning Algorithm), EA, (Evolutionary Algorithm), SA (Statistical Approach), NTA (Network Traffic Analysis). In addition, relevant secondary data would be used to train the system to ensure proper training of the system to act proactively, and to predict and detect the intent of the attackers for possible prevention.

Artificial Neural Networks (ANN)

The research shall use the capability of Artificial Neural Networks (ANNs). ANNs are motivations by biological behaviours of neurones that are used to design algorithms. ANNs are machines learning algorithms. They were designed to model after the nerves of the central nervous system and the brain. ANNs are simplified models of network neurons that occur naturally in the human brain (Gurney, 1997; Graupe, 2007). ANN receives inputs (dendrites) feeds into the artificial neurons which are transmitted through one to many hidden layers that are weighted. The signals are processed to determine the next layer output (Gershenson, 2003). ANNs uses self-adaptive technique that enables it in capturing non-linear complex interactions without prior knowledge between variables of the system to be learned (Jacobson, 2013a; Jacobson, 2013b). ANNs are shown to be successful when used in a variety of domains to perform classification (unsupervised learning) such as discriminant analysis and regression (Wang, 2018). Figure 1 shows the structure of an artificial neuron showing fundamental components of an artificial neural network with inputs, weights, the product of which is summed in a transfer function and feed into the activation function to generate the output.

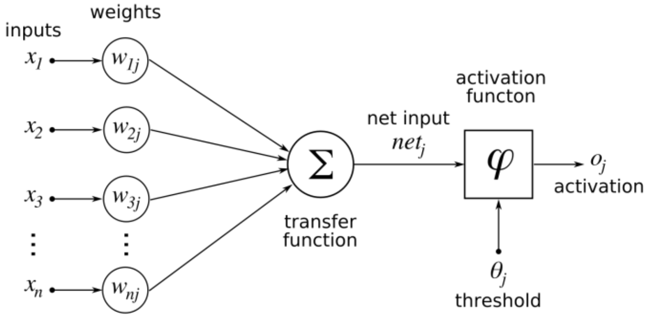

Figure 1: Organisation an artificial neuron of artificial neural networks
(Source: Wikipedia, 2020)

ANNs are primarily 'black boxes' with the capability of adaptation to the underlying system, such an algorithm requires a reasonable level of understanding of the principal assumption of a probabilistic system; a model which produced the data (Gershenson, 2003). The feature of adaptability in the understanding of the underlying system makes ANNs suitable for forecast and taxonomy of internet traffic, signature certification with adaptable difficulties, decision support for concealed item detection, as well as overcoming challenges of model structure related with orthodox taxonomy such as decision tree and K-nearest neighbour (K-NN) algorithms. Due to its efficiency, it is used in classification as a result of its adaptability and detection capability. ANNs have also been used in cyber security in detecting computer software design flaws, viruses and malware detection.

Statement of Problem

Networks are challenged with malicious attacks, creating breaches and vulnerabilities. This is a major concern for businesses, network administrators and global IT stakeholders. Finding solution to these threat vectors is fundamental to the safety of a network and its resources. To solve this problem is to proactively mitigate the cyber-attack vectors with a model to accurately predict, detect, and prevent (PDP) the network system with a deep learning neural network algorithm. The motivation is to adopt a non-signature oriented recognition technique for identification of malicious codes traffic that transverse the network centred on Artificial Neural Network (ANN) (Shenfield et al., 2018).

Aim and Objectives

This research purposes to model a system with a capability to predict, detect and prevent (PDP) malicious network attacks due to threats from the network of users and connected devices. The model would learn the behaviour of attack vectors from datasets in training and predict new model attacks from traversing traffic of the network. The detection modules shall be activated to deter malevolent traffic and cut off traffic from mischievous sources to prevent attacks that may ensue from such sources. The objectives of this paper are:

1. Design and develop a model for mitigation of cyber-attack vectors.
2. Training of the model for identification and classification of attack vectors.
3. Implementation of the model
4. Testing the model for efficient classification.
5. Evaluation of the model operations for its accuracy

Related Works

Cyber security is a major concern now as we get more and more connected to each other. The internet has made connectivity to resources in different geographies, resources, devices easier and this has in some way improved the way we do things. Businesses have easy access to market, people, resources, and that has opened up more spaces in diverse facets of our lives. As the network of people, resources and devices grow, so the interconnectivity shows vulnerability. These vulnerabilities created an unusual opportunity for criminal elements to exploit the network and its system of connected financial assets, infrastructure assets, and other IT based resources. The exploitation causes major problem in the IT-Business ecospace. Security system experts are well aware of the need to protect IT assets and resources and help to mitigate these dangerous and malicious attacks against IT assets and infrastructure.

The challenge has been how the proposed solutions can proactively and effectively predict the behaviour of the attacker? Most systems already deployed have generated false positives that alert the system wrongly. How do we create systems that monitor and effectively characterised the behaviour of the attacker and the motives, in order to stop the incidences of attack even before the attacker strikes? The use of network intrusion discovery systems (NIDS) were proposed to observe and recognise undesirable and malevolent network traffic (Shenfield et al, 2018), though NIDS are effective against known threats, but are ineffective when the signatures of attack vector is unknown or are modified (Kravchik et al 2018).

Another key challenge is the false positive plague associated with NIDS. For instance NIDS is problematic and discovery of mischievous shell code is a challenge. Of course shell code is used commonly to burden system penetration tools by an improved access and power they offer an attacking system (Shenfield et al, 2018). Intrusion Discovery Systems (IDS) are designed with the intention to identify where a malefactor is making an attempt to mislead the normal operation of the system. IDS ensures that abnormal operation are detected and reported, ensuring that confidentiality, integrity and availability of the system and its associated devices are maintained. IDS are of two categories, which are Host-IDS (HIDS) and Network based-IDS (NIDS) (Veeramachaneni et al, 2016). Essentially HIDS detects a compromise on a device, while NIDS detects a compromise in transit over a network. NIDS can further be classed into abnormal (ANIDS) and signature oriented (SNIDS). Most commercially available NIDS are signature based, while ANIDS are still at a stage of research concept. IDS usually generate incident information which serves as an input in security information and event profile management system (SIEPMS) with other feeds and logs. The concept is to allow a complete view of potential incidences to be captured for recording (Wu et al, 2020).

Several researchers have attempted to proffer solutions to the issues of cyber security and recommended various solutions to mitigate these threats. In this study, an extensive review of related literatures within the last five years was carried out to determine the solutions that attempt to provide solutions and tool that address the challenge. Below are some of the authors, and a descriptive, critical and tool based analysis of their works.

RELATED WORKS REVIEWED

- **Neethu B.(2014)** presented Principle Component Analysis Architecture (PCAA) for the Naive Bayes collection of features to build a network intrusion detection program. KDDCup 1999 dataset was used for intrusion detection experiments in the study. The result shows the efficiency in the technique and achieves a greater discovery rate, low time consumption and low cost factor, with accuracy of 94%, when compared to neural network and tree algorithms approaches.
- **Naseer et al.(2018)** suggested, effected and also educated intrusion detection models using diverse deep neural network frameworks such as: RNNs, Autoencoders, and CNNs; also trained were their models and evaluated with NSLKDD datasets. Performance of 85% and 89% accuracy were recorded on DCNN and LSTM models respectively.
- **Zhang et al.(2017)** suggested network intrusion detection of two kinds: direct: uses one algorithm; while the combination: using a mixture of different techniques. Their proposition is a directed acyclic graph (DAG), detection model that is novel, based on belief rule base (BRB). When in comparison with orthodox detection models, the results show that the model of DAG-BRB combinational had a rate detection that is greater.

- **Wang et al.(2017)** suggested a hierarchically spatial system; The system learns traffic features of networks spontaneously. The system is an intrusive detection system, which uses deep CNNs to learn spatial components, as well as the features of LSTM networks.
- **Chowdhury et al.(2017)** suggested a novel method of botnet detection, node-centered topology within a network. The method would detect unusualness by looking for at small number of nodes, and was based on a clustering of self-organizing maps (SOM), which is a part of an unsupervised system. CTU-13 database was used for the analysis; with the largest dataset containing nodes labeled with bot. Comparison was done with another detection algorithm, supporting the vector machine (SVM).
- **Vinayakumar et al.(2019)** hybrid intrusion detection system(HIDS) was developed by the authors with a capacity to Investigate the web links and host level undertakings. HIDS uses deep learning model that is distributed with DNN, for analysis and treatment of big data instantaneously. The model of DNN was carefully selected by comprehensive evaluation of its strength when compared to orthodox machine learning taxonomy. In comparison by various standards, UNSW-NB15 and NSLKDD used as IDS dataset out performed others.
- **Khan et al.(2019)** proposed a deep learning model that is unique in two-stage. The model is an efficient network intrusion detection system that is a stacked auto-encoder with a classification by soft-max. Numerous tests were conducted on: UNSW-NB15 and KDD99 which are public datasets. The study results achieved 89.1% for UNSW-NB15 and 99.9% for KDD99.
- **Du et al. (2017)** suggested a firsthand algorithm based on the k-NN classification. The algorithm models program performance in invasive discovery regarding system calls. Text categorization techniques are implemented to transmute each data on system call to an attack vector and evaluate the resemblance between activities of system calls in programs. The researchers report that in the field of malware detection, the intrusion detection based on k-NN taxonomy appears to be more suitable in intrusion detection sphere
- **Kozik et al.(2014)** commended a new method for identification web applications targeting cyberattacks. The method adopted machine-learning algorithms like Naive Bayes, AdaBoost, Part, and J48. Dataset of HTTP from CSIC 2010 was used to test the model. The major area of the study focused was on HTTP protocols. Enabling server clients to communicate. They assumed higher percentage of detection while getting false positive rate that are lower. The results indicated that J48 strategy is the most viable solution, with about 0.04 rate as the true-positive value.
- **Hoque et al. (2012)** developed genetic-algorithm centered on intrusion detection system (IDS) that accurately identify various types of network intrusion. The model used knowledge evolution theory for filtering traffic data; decreasing the complexity. Dataset from KDD99 was used to test model performance. The experimental results indicate a fair detection rate.
- **Haddadi et al. (2015)** evaluated several approaches to botnet identification. Bot Hunter and Snort are two methods focused on public rule schemes. Other methods used are Data processing methods, including packet payload and traffic Flow-based strategies. Several experiments were conducted using Bayesian networks, SVM, C4.5, KNN. Findings indicate flow-based system performance is by value higher in comparison with findings in published works.
- **Wijesinghe et al. (2015)** focus on identification by examining network traffic flows of botnet families. They suggested a twofold model: (i) they identified appropriate dataset with specific features to detect botnet from IP flows. (ii) The use of IP flow data to detect unlabeled botnet behaviours. Publicly accessible IPFIX dataset was used in the analysis and the concept led to botnet detection based on IP flow data.
- **Shen et al. (2018)** proposed a system for foretelling malicious activities by means of deep rooted learning was developed, named Tiresias. To forecast happenings on a machine, the system leverages on RNNs based on surveillance previously. The testing was done on a 3.4 billion commercial IPS collection of dataset of security events. The result indicated effectiveness of the approach in forecasting the next actions may occur on a system with a correctness of 93%, even a complex circumstance, high precision and stable results were maintained.

- **Veeramachaneni et al.(2016)** developed a complete machine learning techniques. The procedure significantly forecasts cyber-attacks superior to existing systems, by continuous incorporation of human intervention as input. Several months of data directly labeled with ordered metric was used. The supervised learning module received the labeled data as input to forecast an occurrence of bout. The technique uses six methods of anomaly detection, with a detection rate of 85 percent of attacks, in the same time reduced false positives by a factor of 5. Test data generated by millions of users, over 3.6 billion for a period of three months was used to test the system.
- **Oprea et al.(2015)** designed a system to timely discover from DNS logs enterprise infection using belief propagation. They proposed a framework was inspired from graph theory, which is a technique demonstrated on two large datasets have to performed well. The algorithms was applied to data collected at the circumference of a large corporate, about 38TB of web proxy logs, which showed a high accuracy on DNS logs for two months.
- **Zhang et al. (2016)** suggested new system that instantaneously analyses flooded console data and identify timely cautions of failure forecast for IT systems. The scheme employed LSTM in the training process to deal with specifically labeled data. A computerization approach with text mining techniques, like term frequency - inverse document frequency (TF-IDF) was adopted.The proposed technology was equated with current machine learning approaches, and showed an edge over current techniques in prediction of IT failures in complex system.
- **Liao and Vemuri (2002)** suggested a deep neural network model, named DeepLog. The model adopted LSTM to learn data configuration from normal execution. This work uses frequency inverse document frequency (FIDF) vector to collect data key. In addition to the collection of data, models of detection for recognizing atypical record entries and parameter value anomaly were used. In comparison, DeepLog outclassed current log-based anomaly detection mechanisms, attaining an F-measure of 96% and 98% with HDFS and OpenStack data respectively.
- **Wang (2017)** investigated the performances of current attack algorithms alongside Intrusion Detection System (IDS) on the NSL-KDD data based on deep data learning algorithm. The IDS validated vulnerabilities with deep learning neural networks. in generating adversarial, the roles of individual features examples were explored. The research shows applicability and feasibility of the attack methodologies.
- **Ding et al. (2016)** suggested the use DBN to detect malware and malware as opcode sequences, where a multilayer generative model used unsupervised learning to train. With this training, overfitting problem was solved by DBN. The proposed DBN achieved a higher accuracy of 96%, with more unlabeled data, which outdoes models such as: decision tree kNN, SVM.
- **Tan et al**. **(2015)** suggested an ad hoc network intrusion detection structure based a deep belief network (DBN). Their typical model contains 6 modules: module with data fusion to inject useful data, wireless observatory node for data input, redundancy removal, intrusion module and training module. This model trains and detects whether intrusion exist and responds to the intrusion that expresses results showing minimum accuracy of 97.6%.
- **Zhao et al.(2017)** suggested a network attack identification archetypal that incorporate deep learning and flow calculation that consist of instantaneous detection algorithm and classification algorithm. A flooded data processing can actualize instantaneous detection; with a procedure can improve taxonomy exactness. With dataset of CICIDS2017, several experiments were performed and comparative drawn. Result showed an instantaneous detection efficacy which is higher when compared to conventional procedures.

Information safety and security clarifications are categorized into dual sets: (i) Driven by analyst, (ii) Driven by machine learning solutions. Solutions by Analyst depend on guidelines made by security specialists (Veeramachaneni et al., 2016), while machine learning driven solutions does not depend on human analysis but on the machine, learning a big data about the trends and activities and figuring out the behaviour pattern of data and actions when new data flows in. However, Zhang et al. (2017) classed network intrusion discovery into two, such as direct method and combination method. The direct method uses single algorithm, while the combination method uses several methods in combination. Others proposed recognition models are directed acyclic graph oriented (DAGO) and a belief rule oriented (BRO). Solutions driven by deep learning are commonly used to detect unusual patterns that can improve recognition of new computer-generated threats. Existing learning method and mechanism have some limitations (Veeramachaneni et al, 2016). This is because obtaining labelled data at a higher scale is difficult; therefore getting accurate training of a model is a challenge. The use of anomaly detection would help detect unknown cyber threats. Tiresias developed a system through deep learning for predicting security events. The system leverages on RNNs, based on previous observations on a machine, to predict future events (Shen et al., 2018). Tiresias work was focused on anomaly detection, which predicted events in a noisy situation with a wide variability. The dataset was from a collection of an intrusion inhibition system (IIS) records. The prediction was found to be highly accurate with a precision of 93% for a multifarious conditions and sustained steady results.

METHODOLOGY

The method adopted in this research was to study existing industry and academic literatures in the identification and classification of threats and determine active threat indicators, as inputs into the neural networks algorithm, to determine the pattern mutated packets were transmitted and predict the behaviour of attack vectors. The method extracted and processed downloaded datasets containing attack vectors. The datasets were structured with fields (features) that serve as input into the deep neural network (DNN) model. The data was then encoded to ensure all text strings are converted into numbers. The numbers are fed into the engine as input, which will in-turn determine threat and non-threat rows (true positives and true negatives) (Wu et al, 2020). After pre-processing, selection and setup were carried out, and then an activated function was initiated to create the model with neural network layers. A model was generated and the accuracy, applicability, and identify modifications or fine tuning needed in any of the data channel parts was examined. Data extraction and categorization into columns was done that require prediction.

The ANN is designed and structured to have an input layers, a set of hidden layers depending on the complexity of the network and an output layer (Graupe, 2007). The process of information flowing from an earlier layer to a later layer is called feed forward as shown in Figure 2 below. The network can have as many input variables as possible depending of the parameters feed in for detection. The hidden layers could also be more depending on complexity, ordering and process of data from the inputs. The output layers are designed to send out take processed information to the external environment. The number of input variables must not necessarily be equal to the output variables. In some cases, an error correction mechanism is built into the system for fine tuning the algorithm, which is to achieve the expected output form the network, called error backpropagation (Hodo et al, 2017).

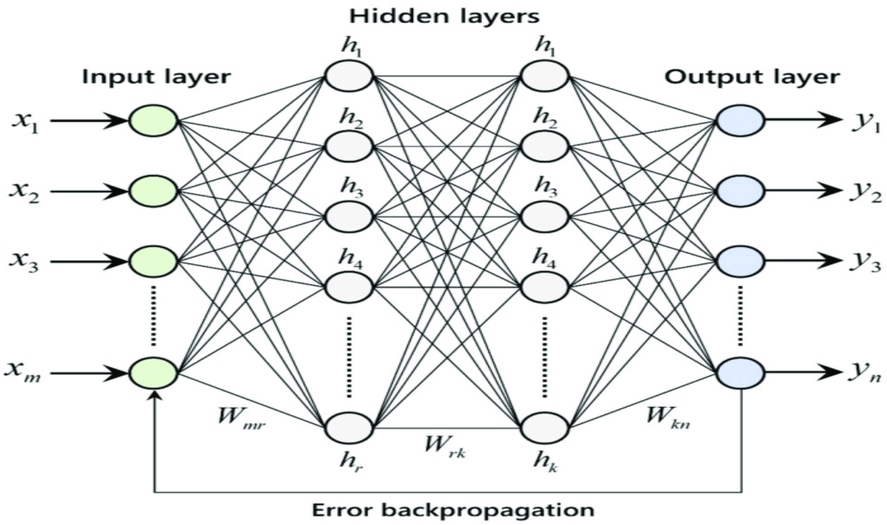

Figure 2. A model of Artificial Neural Network with Backpropagation (Fernández-Cabán et al, 2018)

$Y_1 = X_1W_{1r}+ h_1W_{1k}+ h_1W_{1n}$ ……………………………………………(1)

$Y_2 = X_2W_{2r}+ h_2W_{2k}+ h_2W_{2n}$ ……………………………………………(2)

$Y_3 = X_3W_{3r}+ h_3W_{3k}+ h_3W_{3n}$ ……………………………………………(3)

Where :

$Y_i = \{Y_1, Y_2, Y_3,\ldots\ldots\ldots\ldots,Y_n\}$ are the outputs from the output layer of the neural network

$X_i = \{X_1, X_2, X_3,\ldots\ldots\ldots\ldots,X_n\}$ are the inputs into the input layer of the neural network

$W_i = \{W_1, W_2, W_3,\ldots\ldots\ldots,W_n\}$ are the weights of the connection links to the nodes in the layers of the neural network.

$W_{ik} = \{W_{1k}, W_{2k}, W_{3k},\ldots\ldots,W_{nk}\}$ are the weights of the connection links to the nodes in the layers of the neural network with respect to the nodal numbers.

Using the compact summation to model the relationship between the inputs, weighted links (parts) and the output is shown in equation 4.

$\mathrm{Y} = \sum_{ik}^{n} W_i X_k$ ……………………………………………………………(4)

A dataset containing information of cyber-attack variables of interest was collected to help unveil the features of interest that relates to intrusion attacks on data records. Principally, two universal datasets on cyber-attacks were investigated among others: KDD99 and NSL-KDD dataset.

a) KDD99 Dataset has complex attribute and dependencies was used. KDD99 dataset was from an intrusion detection evaluation program to develop a network. The program was created by (DARPA) and has capability to discriminate between ‘good’ and ‘bad’ connection. The dataset was imported, read, pre-processed, pre-set and programmed to produce two or multiclass grouping of cyber-attacks (Wu et al, 2020). The data covers major attacks in the cyberspace including IoT, DoS, channelside (Probe), R2L (root to local), and as well as the user to root(U2R). The file type downloadable was in .txt/.csv containing non-redundant records. The dataset correlates high level network traffic structure and cyber-attacks that can be customised, expanded and regenerated (Al-Haija et al., 2020).

b) NSL-KDD Dataset is a well-developed high level assorted interpretation of training data. The dataset covers data on network for abnormal and/or normal traffic. NSL-KDD dataset is a restructured version of KDDCUP'99(Aiyanyo et al., 2020). Abnormal data are transfigured data packets attained by considerable mutation of the network packets header configuration enhanced from previous development (Hodo et al, 2017). NSL-KDD original dataset is available in grouping of: (i) Binary labels with two class traffic dataset, (ii) Multiclass traffic dataset that include difficulty level and attack-type labels. (Al-Haija et al., 2020).

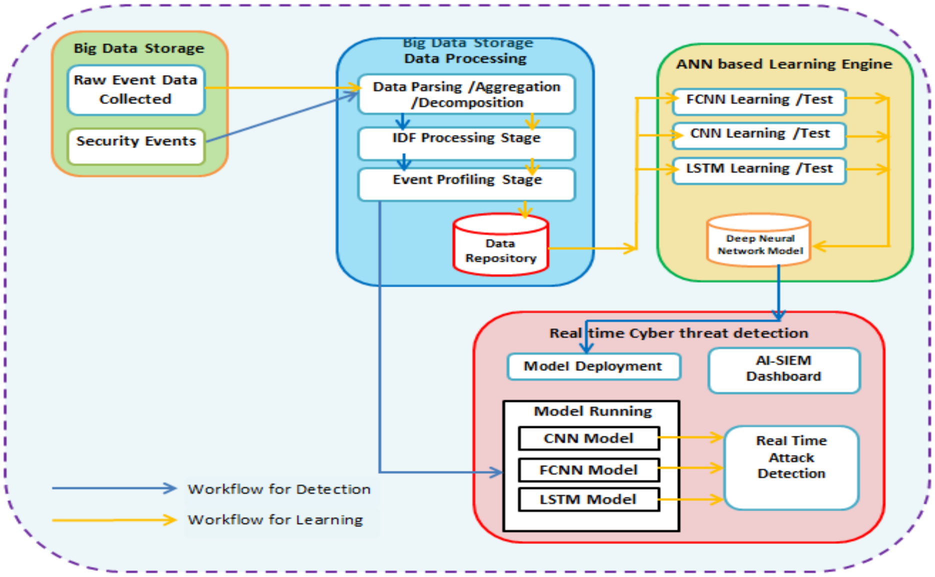

Figure 3: Existing Architecture of an AI-SIEM Model for Cyber Attack Detection

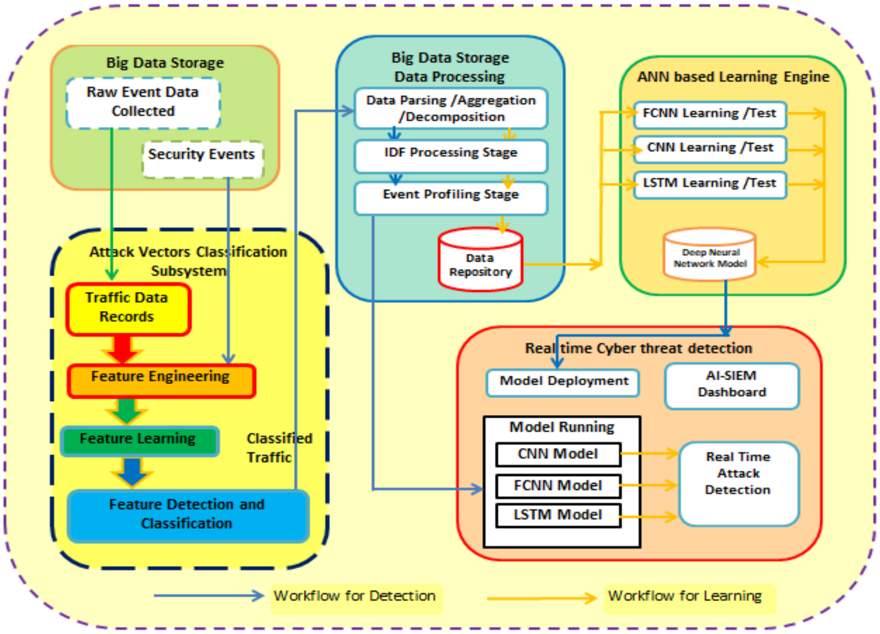

Figure 4: Architecture of the proposed system: (A modification of the AI-SIEM Model)

Proposed System: Modelling of the System Processes

Our system design was subdivided into distinctive component subsystems, implemented with numerous modules as shown in Figure 4. The dataset downloaded showed three different protocol features. The three identified protocol features were (TCP, UDP, ICMP). The service features found within the dataset were 65 different types, Flag features were – 11 different flags, labelling the target features.(Zhang et al., 2017). The dataset was imported into MATLAB using the MATLAB import toolkit. The imported data has 65 features in 65 columns represented in a tabulated form. These features were categorised into features per column. The table of 65 columns with several rows representing records were converted from table dataset to a double data type matrix. After the dataset importation, the dataset was encoded and labelled (categorised) , the process is shown in the proposed architectural model in Figure 4 .

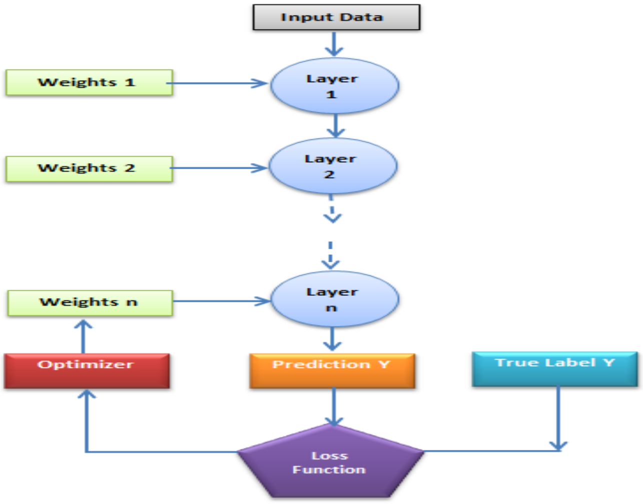

Figure 5: Process of detection and classification subsystem

The detail of the processing with the detection and classification subsystem is shown in figure 5. Inputs are feed into layer 1 alongside the weights, and the output is feed into layer 2 with weights of layer 2 and the process continues with other hidden layers until the last hidden layer (layer n).

The output of the last layer is the forecasting outcome of the neural network. A comparison is done with true label Y to determine the loss function, so that the system can be fine-tuned. The difference is feed into the system optimizer which then optimizes the value as feed back to the system through back propagation to correct the system and reduce the value of the function loss.

The reduction of the value of the function makes the prediction tends to the value of the true label Y for accurate classification.

System Evaluation and verification

The verification and evaluation was done using the dataset with the following parameters. The system was tested for suitability after the process of learning from the dataset. Testing was done with the following metrics to evaluate and verify the performance characteristics of the algorithm (Wu et al, 2020).

$$\text{Classification Accuracy}(\%) = \frac{\text{Correctly Predicted Samples}}{\text{Number of Testing Samples}}\ x\ 100\% \ldots\ldots\ldots\ldots(5)$$

$$\text{Classification Error}(\%) = \frac{\text{Incorrectly Predicted Samples}}{\text{Number of Testing Samples}}\ x\ 100\% \ldots\ldots\ldots\ldots(6)$$

$$\text{Classification Time}(\text{ms}) = \sum_{i=1}^{No.of\ Runs} Execution\ time\ (i) x \frac{1000}{No\ of\ Runs} \ldots\ldots\ldots\ldots(7)$$

Equation 8 was used to test for the classification accuracy of the model. Equation 9 to test for the classification error due to incorrectly predicted samples, and equation 10 to determine the classification time (for complete execution of the runs that is suitable for a classification). Other parameters and metrics used in the measurement to determine efficacy where precision, recall and alarm rate. The formulation for each of these parameters is shown in equations 8, 9, 10 and 11. Precision is a function of True positive (TP) and False positive (FP) to determine the accuracy of the model. Recall is a function of TP and False negative (FN) to determine the rate of recall expressed as a percentage. False alarm rate is due to false positive (FP) and false negative to determine the rate of false alarm expressed in percentage (Choi et al, 2020).

$$\text{Precision} = \frac{TP}{TP+FP}\ x\ 100\% \ldots\ldots(8)$$

$$\text{Recall} = \frac{TP}{TP+FN}\ x\ 100\% \ldots\ldots(9)$$

$$\text{F} = 2\ x\ \frac{Recall\ x\ Precision}{Recall+Precison}\ x\ 100\% \ldots\ldots(10)$$

$$\text{False Alarm Rate} = \frac{FP}{TN+FP}\ x\ 100\% \ldots\ldots(11)$$

RESULTS AND DISCUSSION

The detection of attack vectors in cyberspace usually adopts different strategies. Defensive methods used reactive approaches that are focused on detection, prevention and responses, while offensive approaches used Machine Learning (ML) techniques: supervised and unsupervised learning methods such as clustering, frequent pattern mining and association rule mining. And in some cases a mixture of unsupervised and supervised, and semi-supervised learning were adopted (Aiyanyo et al., 2020). Defensive cyber security approaches with the aid of Machine Learning includes neural networks, Support Vector Machine (SVM), Hybrid methods, Decision trees, Naïve Bayes, Logistic regression, Random forests (Aiyanyo et al., 2020; Choi et al., 2020).

Table 1: Traffic circulation Statistics of NSL-KDD dataset

	Two Class DS		**Multi-Class DS**				
Data Group	**Normal**	**Attack**	**Normal**	**DoS**	**Probe**	**R2L**	**U2R**
Training	65,829	56,829	76,289	53,930	12,920	879	67
Testing	8,791	12,920	9,872	8,728	3,728	3,201	189
Total	**74,620**	**69,749**	**86,161**	**62,658**	**16,648**	**4,080**	**256**

The detection of attack vectors using neural network was found to be effective through the use deep learning algorithm that effectively learned the datasets available in training and then use the learned algorithm to classify and detect the attack vectors (Lee, et al, 2019). The traffic distribution statistics of the NSL-KDD for the analysis is shown in two classes of “two class distribution” and “multiclass distribution” as shown in Table 1 for the different attack vectors, for training and testing.

We found that most of the current attack vectors are slightly different from the commonly known vectors. This difference has made it difficult to identify and effectively classify the attacks using traditional machine learning tools and methods (Prabhu1 et al., 2020). Offensive mechanisms are counteractive points to defensive approaches that proactively predict and remove threats in the system (Aiyanyo et al., 2020). Therefore we have adopted newer approaches of machine learning to achieve this through the application of deep learning neural network as an effective detection and classification tool of attack vectors (Al-Haija et al., 2020).

Deep learning (DL) cyber security solutions (defensive and offensive) can handle and analyse large amounts of data with detection logic that is complex where conventional methods have experienced difficulty (Wu et al., 2020). This is because there are a several perceptions and difficulties that can be recognised in the use of DL techniques. These were due to legitimacy of data over the feature space, network traffic data-high dimensionality, general entropy of information, and class overlap between threats. Although, techniques of Machine Learning, particularly deep learning is a growing trend in use in cyber security; the future points to more utilization (Aiyanyo et al., 2020).

Machine Learning procedures are increasingly been adopted to solve cyber-attack vectors, such as Remote to Local (R2L), Denial of Service (DoS), User to Root (U2R), and probe attacks. The datasets used contains attacks signatures in the learning process (Veeramachaneni et al, 2016). DoS attacks result to the limitation of network resources availability to service consumers as host become overwhelmed with service requests. U2R attack is an attempt at getting access to a target system without user privileged authentication and authorisation (Wu et al, 2020). R2L attack is an exploitation of vulnerabilities, including guessing of usernames and passwords. The essence was to take advantageous control of a remote computing machine. Probe attacks involve exploration of system vulnerabilities to obtain important information for possible attacks. Distributed Denial of Service (DDoS) is an improved attack vector of DoS spell, using the distributed computing techniques (Aiyanyo et al., 2020). Even the anti-malware software product uses signature-oriented detection mechanisms that involve the mining of distinctive signatures features from identified malevolent files. And again with executable file identification as a malevolent cipher code (Wang, 2018). Therefore where a signature file match exists, the mutated attack vectors are identified (Wu et al, 2020). New detection techniques to address attacks and traffic behaviour analysis were currently proposed; as machine learning techniques and deep neural network to classify and detect the presence, activities, and network traffic behaviour of most attack vectors. The analysis of actions of the packet traffic can function normally even with encrypted network communication rules as it may not be depended of the payload of the packet (Aiyanyo et al., 2020).

A model of the deep convoluted network is shown above in Figure 2 that learned the system with the dataset provided. The system classified the packet according to the **normal, attack, DoS, Probe, U2R, R2L** (Wang, 2018). The classification mechanism used the various input and the complex hidden layers to learn the data and provided the output for the dataset from which the model learns. The implementation of the processes was achieved with the aid of TensorFlow adapted code models.

CONCLUSION

The detection of malicious traffic has been numerously attempted by several practitioners using programmable techniques. Some are signature based and others were purely hard coded classification. While these techniques are in some cases effective and at other times ineffective; these methods were incorporated into the intrusion detection systems. However, a lot of false positives and false negative were generated. Machine learning techniques (MLT) have also been used to improve the system. Nonetheless, MLT has its limitation of classification and identification of malicious mutative traffic packets that transverse the network. Deep learning neural network was adopted, which is a subset of machine learning toolset and algorithm to help learn the data and predict the system for better operation of the network and its applications that rely on its architecture. Deep learning required huge amount of learning data for its accuracy and effectiveness. The system to learned from the downloaded dataset, and was tested for its accuracy in classification with the test data. The detection and predictability of the deep learning neural network have given rise to predictability of the system, which enables the system to flag on whenever a mutated packet is transmitted within the perimeter of the network.

RECOMMENDATION

Deep learning algorithms and implementation were used to learn the dataset and have tested the model with the training data. Further work is required to reduce the training data quantity and time taken, to ensure speed up the training process. Since the network was not feed with any labelled data, the training time is high. The system though capable of training with CPU based machine, but was more effective with system with both CPU and GPU. The time and space complexity required with large data may crash conventional system configurations. The design of simpler algorithm with less training dataset may be able to achieve deep learning with lower time and space complexity.

REFERENCES

Aiyanyo, I.D., Samuel,H., Heuiseok Lim,H.(2020). A Systematic Review of Defensive and Offensive Cybersecurity with Machine Learning.Appl. Sci. 2020, 10, 5811; oi:10.3390/app10175811

Al-Haija, Q.A., Zein-Sabatto, S.(2020).An Efficient Deep-Learning-Based Detection and Classification System for Cyber-Attacks in IoT Communication Networks. Electronics 2020, 9, 2152; doi:10.3390/electronics9122152 www.mdpi.com/journal/electronics

Choi,H.Y., Sadollah, A., Kim, H.J.(2020). Improvement of Cyber-Attack Detection Accuracy From Urban Water Systems Using Extreme Learning Machine

Chowdhury, S., Khanzadeh, M., et al. (2017). Botnet detection using graph-based feature clustering. Journal of Big Data, 4(1).https://doi.org/10.1186/s40537-017-0074-7

Ding,Y., Chen,S., Xu, J. (2016).Application of deep belief networks for opcode based malware detection. in Proceedings of 2016 International Joint Conference on Neural Networks (IJCNN), pp. 3901–3908, Vancouver, British, July 2016.

Du,M., Li, F., Zheng,G., Srikumar,V.(2017).DeepLog: Anomaly detection and diagnosis
from system logs through deep learning," in *Proc. ACM CCS*, Dallas, TX, USA, vol. 17, Nov. 2017, pp. 1285_1298.

El-Alfy, E.M, Al-Obeidat, F.(2015).Detecting Cyber-Attacks on Wireless Mobile Networks Using Multicriterion Fuzzy Classifier with Genetic Attribute Selection. Hindawi Publishing Corporation.Mobile Information Systems. Volume 2015, Article ID 585432, 13 pages. http://dx.doi.org/10.1155/2015/585432

Fernández-Cabán,P.L., Masters,J.F., Phillips, B.M.(2018).Predicting Roof Pressures on a Low-Rise Structure From Freestream Turbulence Using Artificial Neural Networks. Frontiers in Built Environment. www.frontiersin.org 1 November 2018 | Volume 4 | Article 68

Gershenson, C.(2003). Artificial Neural Networks for Beginners

Graupe,D.(2007).Principles of Artificial Neural Networks (2nd Edition). Advanced Series on Circuits and Systems – Vol. 6. World Scientific Publishing Co. Pte. Ltd.

Gurney,K.(1997). An introduction to neural networks. University of Sheffield,UCL Press Limited

Haddadi, F,C., Le, D.(2015). On the Effectiveness of Different Botnet Detection Approaches. Lecture Notes in Computer Science, 9065, 421–436. https://doi.org/10.1007/978-3-319-17533-1

Hodo, E., Bellekens,X., Hamilton,A., Dubouilh,P., Iorkyase,E., Tachtatzis,C., Atkinson, R.(2017). Threat analysis of IoT networks Using Artificial Neural Network Intrusion Detection System

Hoque, S, A., Naser, A. (2012). An Implementation of Intrusion Detection System Using Genetic Algorithm. International Journal of Network Security & Its Applications, 4(2), 109–
120. https://doi.org/10.5121/ijnsa.2012.4208

Jacobson, L.(2013a).Introduction to Artificial Neural Networks - Part 1.
https://www.theprojectspot.com/tutorial-post/introduction-to-artificial-neural-networks-part-1/7

JacobSon, L.(2013b).Introduction to Artificial Neural Networks Part 2 – Learning.
https://www.theprojectspot.com/tutorial-post/introduction-to-artificial-neural-networks-part-2- learning/8

Khan, F. A., Gumaei, A., Derhab, A., Hussain, A.(2019).A novel two stage deep learning model for efficient network intrusion detection," *IEEE Access*, vol. 7, pp. 30373_30385, 2019.

Kozik, R., Choraś, M., Renk, R., & Hołubowicz, W. (2014). A Proposal of Algorithm for Web Applications Cyber Attack Detection. IFIP International Conference on Computer Information Systems and Industrial Management, 8838. https://doi.org/10.1007/978-3-662-45237-0_61

Kravchik,M. Shabtai, A.(2018).Detecting Cyber Attacks in Industrial Control Systems Using

Convolutional Neural Networks. Session 4: Industrial Control and SCADA Systems CPS-SPC'18, October 19, 2018, Toronto, ON, Canada

Lee, J., Kim,J., Kim, I., Han, K.(2019).Cyber Threat Detection Based on Artificial Neural Networks Using Event Profiles. Special Section on Artificial Intelligence In Cybersecurity

Liao, Y. and Vemuri,V. (2002).Use of K-nearest neighbor classi_er for intrusion detection. *Computer. Security.*, vol. 21, no. 5, pp. 439_448, Oct. 2002.

Naseer, S., Saleem, Y., Khalid, S., Bashir, M. K., Han, J., Iqbal, M. M., and Han, K.(2018).

Enhanced network anomaly detection based on deep neural networks. *IEEE Access*, vol. 6, pp. 48231_48246, 2018.

Neethu, B. (2014). Classification of Intrusion Detection Dataset using machine learning Approaches. International Journal of Electronics and Computer Science Engineering, 34(3), 1044–1051. https://doi.org/10.3969/j.issn.0253-2417.2014.03.013

Oprea, A., Li, Z., Yen,T.F., Chin, S.H., Alrwais,S. (2015).Detection of early stage enterprise infection by mining large-scale log data," in Proc. 45th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw., Rio de Janeiro, Brazil, Jun. 2015, pp. 45_56.

Ostertag,C.(2019).The Shortest Introduction To Deep Learning You Will Find On The Web.

http: https://medium.com/analytics-vidhya/the-shortest-introduction-to-deep-learning-you-will-find-on-the-web-25a9975bbe1d

Prabhu1, S., Bhat,S. (2020). Cyber Attacks Mitigation: Detecting Malicious Activities in Network Traffic -A Review of Literature. International Journal of Case Studies in Business, IT, and Education. (IJCSBE), ISSN: 2581-6942, Vol. 4, No. 2, August 2020. Srinivas Publication www.srinivaspublication.com PAGE 40

Rungta, K. (2018). TensorFlow in 1 Day: Make your own Neural Network

Shen, Y., Mariconti, E., Vervier, P.A., Stringhini, G.(2018). Tiresias: Predicting security events through deep learning. in Proc. ACM CCS, Toronto, ON, Canada, Oct. 2018, pp. 592_605.

Shenfield, A., Day, D., Ayesh, A. (2018).Intelligent intrusion detection systems using artificial neural networks, ICT Express (2018), https://doi.org/10.1016/j.icte.2018.04.003

Tan,Q., Huang, W., Li, Q. (2015).An intrusion detection method based on dbn in ad hoc networks," in Proceedings of Wireless Communication and Sensor Network: International Conference on Wireless Communication and Sensor Network (WCSN, World Scientific, Wuhan, China, pp. 477–485, December 2015.

Veeramachaneni, K., Arnaldo, I., Korrapati, V., Bassias, C., Li, K.(2016). AI2: Training a big

data machine to defend, in Proc. IEEE Big Data Security, HPSC IDS, New York, NY, USA, Apr. 2016, pp. 49-54.

Vinayakumar,R., Alazab, M., Soman,K., Poornachandran, P., et al.(2019). Deep learning approach for intelligent intrusion detection system *IEEE Access*, vol. 7, pp. 41525_41550, 2019.

Wang, J., & Paschalidis, I. C. (2017). Botnet Detection Based on Anomaly and Community Detection. IEEE Transactions on Control of Network Systems, 4(2), 392–404. https://doi.org/10.1109/TCNS.2016.2532804

Wang,W., Zhu, M., Zeng,X., Ye, X., Sheng, Y.(2017) .Malware traffic classification using convolutional neural network for representation learning. in *Proc. Int. Conf. Infor. Netw. (ICOIN)*, Da Nang, Vietnam, Jan. 2017, pp. 712_717.

Wang, Z.(2018).Deep Learning Based Intrusion Detection With Adversaries. Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Wijesinghe, U., Tupakula, U., & Varadharajan, V. (2015). An enhanced model for network flow based botnet detection. Conferences in Research and Practice in Information Technology Series,159(January), 101–110.

Wu,Y., Wei,D.,Feng,J.(2020).Network Attacks Detection Methods Based on Deep Learning Techniques: A Survey. Hindawi. Security and Communication Networks. Volume 2020, Article ID 8872923, 17 pages. https://doi.org/10.1155/2020/8872923

Zhang, B., Hu, G., Zhou, Z., Zhang, Y., Qiao, P., Chang, L.(2017). Network Intrusion Detection Based on Directed Acyclic Graph and Belief Rule Base. ETRI Journal, Volume 39, Number 4, August, 2017 592. http://etrij.etri.re.kr. https://doi.org/10.4218/etrij.17.0116.0305

Zhang,K., Xu,J., Min, M.R., Jiang,G., Pelechrinis, K., Zhang,H.(2016).Automated IT system failure prediction: A deep learning approach," in *Proc. IEEE Int. Conf. Big Data (IEEE BigData)*, Washington, DC, USA, Dec. 2016, pp. 1291_1300.

Zhao,G., Zhang,C., Zheng,L.(2017).Intrusion detection using deep belief network and probabilistic neural network. in Proceedings of 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Taipei, Taiwan, December 2017.