Deep Convolutional Neural Networks for Intrusion Detection in Automotive Ethernet Networks

Ashwini Kumar
Assistant Professor
Department of Engineering & IT
ARKA JAIN University
Jamshedpur, Jharkhand, India.
ashwini.kumar@arkajainuniversity.ac.in

Dr. Vipul Vekariya
Professor
Department of CSE
PIET, Parul University
Vadodara, Gujarat, India.
vipul.vekariya18435@paruluniveristy.ac.in

2022 5th International Conference on Contemporary Computing and Informatics (IC3I) | 979-8-3503-9826-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/IC3I56241.2022.10073080

***Abstract*—The extensive usage of interconnection and interoperable of computing systems has become an unavoidable requirement for improving our daily lives. Similarly, it paves the way for exploitable flaws that are far beyond human control. Because of the flaws, cyber-security techniques are required in order to conduct communication. To resist concerns, reliable connectivity necessitates security protocols, as well as innovations in protection efforts to control growing security concerns. To identify and categorize networks assaults, this study suggests using deep learning architectures to construct an adaptable and resistant network intrusion detection system (IDS).The focus is about how deep learning or deep convolutional networks (DCNNs) may help adaptable IDS with growing capabilities distinguish known and novel or zero-day networking observable traits, disconnecting the intruder and lowering the risk of exposure. The UNSW-NB15 dataset, which reflects genuine current network interaction complementing synthetically created attack behaviours, was used to illustrate the performance of the model.**

Keywords—Cyber security, Zero-dayattacks, Deep learning, Intrusion detection, global optima and Network anomaly discovery.

I. Introduction

Our relationships with our everyday routines need to be rethought in light of recent developments in information and communication technology (ICT), as well as the growing prevalence of applications that make use of interconnections and compatibility. Because of their reliance on ICT, both individuals and organisations have strengthened their positions. This has made it possible to conduct real-time worldwide business operations, which are continually expanding to offer convenience-related connectivity boundary technologies [1]. A strong network monitoring system is vital for protecting confidence, dependability, and scalability [2], since the exchange of electronic information over networking has resulted in the creation of software defects that may cause harm to both individuals and organisations. Rules for the security of networks serve as the first line of defence in the tiered system of defensive actions that are used to address different types of security vulnerabilities. An intrusion detection system (IDS) is able to inspect network packets and warn an administrator of a breach based on established detection settings that are also changeable. If an intruder is caught quickly, it can be stopped and removed from the server before any information is compromised as a result of the intrusion. IDS believes that the performance of incursions is distinct from that of authorised customers; hence, it measures the activity of intrusions in terms of its design. However, because a precise distinction cannot be formed, there is a blurring between normal and abnormal behaviour. This blurring can be brought to light by employing a smart detection method, which is referenced in [3].

Conventional methods of machine learning, such as Nave Bayes, Decision Trees, Support Vector Machines, and others [4,] are frequently utilised to overcome some of the drawbacks of the created methods. Despite the fact that these technologies have significantly improved detection rates, they still require the expertise and participation of trained professionals in order to manage the copious amounts of data. These methods, which are also known as shallow learners, search for programmes that enable computers to learn without the input of humans; nevertheless, the results that they provide may eventually be subpar for multi-class classification problems that include more features [5]. Research has also made advances in the development of self-learning security mechanisms, which can detect and categorise both known and zero-day intrusions; activities and experiences such as these help preventative approaches recognise and deter hostile network behaviour. Deep learning is a sophisticated model, or a specialised version, of machine learning techniques that overcomes some of the limitations of superficial networking. It is also known as "supervised learning" or "supervised machine learning." Deep neural networks have demonstrated significant potential in a range of applications, including speech signals, image analysis, computational linguistics, and many more [6].

This study examines the efficacy of deep learning approaches for security programmes that detection connection traffic in order to detect and document a contravention based on interference cognitive and behavioural features identified in the data source UNSW-NB15. These features reflect real sophisticated behaviour pattern with synthetic and real-world attack exercises [7]. In the following section, "Section 2," the foundational knowledge for IDS models is presented, which was preceded by "the technique employed in this challenge" shown in Figure 1. The results of the convolution neural model will be presented in the next section. A brief summary of the findings

979-8-3503-9826-7/22/$31.00 ©2022 IEEE

as well as some suggestions for the future are provided at the end of Section 5.

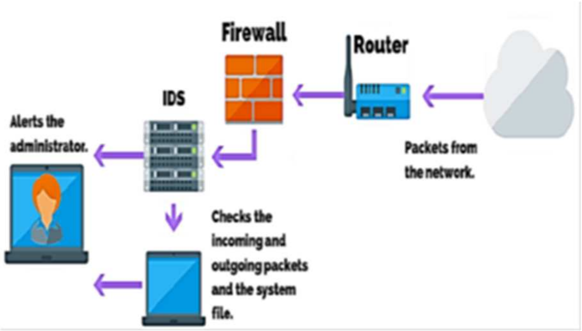
Firewall
Router
IDS
Alerts the administrator.
Packets from the network.
Checks the incoming and outgoing packets and the system file.

Figure 1. Elementary flow of IDS in

II. Literature Review

Attackers use technical advancements for exploitable purposes, which are attributed to technological achievements for the purpose of enriching the existence of digital civilization. Examples of exploits include any attempts to enable remote access to a resource in order to compromise its stability, secrecy, or usability. Those that are damaged suffer unfavorable consequences as a result of the events, while the invader is provided with an opportunity to make a profit. The function of Network IDS is to evaluate a system's capacity to identify malicious behavior on a network that could cause damage to other systems or networks. In spite of the fact that established practices are incorporated into the setups of the identification methods, restricting information that complies with rules may result in a drop in communication speed. As a result, completely convolutional alternatives ought to be investigated. Deep learning methods are multi-layer image retrieval systems that have made it popular and have had a considerable influence on research [8]. They are responsible for making deep learning methods prominent. The current body of research in the field of intrusion detection systems (IDS) has resulted in improved detection performance across a wide range of databases. In addition, due to concerns about the protection of personal information, there is no currently accessible unauthorized access labeled collections. Because of this limitation, investigators are required to simulate internet use through the use of realistic data augmentation. The KDD Cup dataset, which was the result of a research funded by the Defense Advanced Research Projects Agency (DARPA), was the first IDS machine learning library. Lincoln Lab constructed KDD Cup by employing tcp dump and a private network to produce information with human implanted assaults. This was done in order to replicate communications on US Air Force bases. The second version, referred to as NSL-KDD [9], was developed since it was unable to verify whether or not the simulation dataset adequately depicted the actual traffic on the internet, in addition to the fact that there were several redundant records, notably for assaults. The huge sample is still criticized for not having real-time traffic modeling, despite the fact that it is likely the most extensively used and effective benchmark database in NIDS analysis. This is despite the fact that considerable improvements in dependability have been made. Statistics from NSL-KDD, Kyoto, WSN-DS, and CICIDS, in addition to those from other networked IDS, are all available. At the University of New South Wales in Australia, researchers Moustafa and Slay [7, 10] generated the most recent huge dataset, which is referred to as UNSW-NB15. Although there were relatively few basic practices, which enabled database things to get more intricate, the collection was copied and added to the KDD Cup collection utilizing the Association Rule Mining (ARM) technique for feature selection. UNSW-NB15 includes a listing of nine attack families, which are described in [7]. These attack families correlate to the frequent vulnerability database (CVE). CVE is a dictionary of publicly documented types of threats that is updated as new risks are identified [11].

In the same vein as its predecessors, the UNSW-NB15 IDS dataset includes over 40 attributes that accurately reflect real traffic on the network, in addition to over 2.5 million entries and nine modern attack methods [12]. Condensed versions of testing and training databases are provided by the designers and are frequently utilized in investigations. Certain sources, in the event that they are not immediately accessible through university websites, have inverted the segmentation of training and testing sets and used the narrower dataset for training. This has resulted in poorer classification results due to the limited number of cases for certain types of attacks. The segmented samples have really been modified and handled using serious work, editing, compression, and wrangling strategies in order to facilitate application that requires just minimal supervised learning from before the. This was done in order to facilitate application that requires just minimum supervised learning from before the. IDS research on deep learning will begin with the earliest datasets provided by DARPA. These datasets were used to initiate computational investigations using shallow neural networks such as Decision Trees, Support Vector Machines, Random Forests, and so on. These networks produced accurate results but had a significant amount of difficulty with underprivileged threat vectors [13,14]. The UNSW-NB15 dataset, similar to other IDS datasets, offers binary classification, which indicates whether an occurrence is benign (normal) or malicious (attack), as well as descriptive categorization, which indicates the type of attack. This information can be used to determine whether an occurrence is benign or malicious. The majority of academics will construct two different models in order to account for both binary and classified designations. In addition to these models, the KDD Cup dataset provides sub-categorical categorization (24 different types of attacks). Despite the fact that tested information accounts for all classifications, this information is typically not published in studies due to the fact that the training database misrepresents discrepancies in the sub-categories. One technique for dealing with class differences is to combine data and deploy sample procedures that split the information for training and testing sets. This is one example of how to deal with class disparities. In [15], an improved version of the LaNet-5 model used to categories network assaults is discussed for the NSL-KDDD database. The overall accuracy of the suggested deep stochastic neural network technique is 99 percent, with a recognition rate of 97 percent for disadvantaged classifications.

III. Proposed Methodology

The proposed system for the prevention of intrusion can identify attacks by integrating a deep learning algorithm with classification methods. However, in order to protect users' privacy, the system does not gather any information from the datagrams themselves. The procedure is broken down into a number of stages, each of which will be elaborated on in the paragraphs that follow.

A. *Representation of Data*

The NSL KDD collection is extremely extensive and is utilized in the studies that are conducted about intrusion detection. There are around 4.5 million copies in all. There are three basic subcategories of characteristics contained within this database. These are TCP connection factors, Content quality, and Traffic capabilities. This database contains information on 22 distinct types of assaults, which can be categorized into the following three categories:

- DoS - Denial of Service Attack
- UR - User-to-Root attack
- RL - Remote to Local Attack
- Probing – Port scan

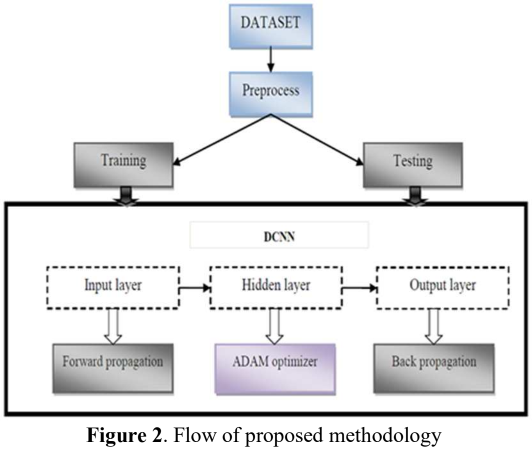

Figure 2. Flow of proposed methodology

B. *Preprocess Step*

The NSL KDD dataset is flawless since it does not contain any noise or missing information, making it an ideal example of a collection. Nevertheless, it is founded on quantitative and linguistic principles, with the quantitative data encompassing a vast amount that delays retraining and makes processing more complicated. In addition, the computing processes of the deep neural network are unable to interpret the text elements. As a direct consequence of this, the data need to be compiled. In this particular paradigm, the two most important aspects of the pretreatment process are the normalization technique and the text projection. The values of the numerical features were decreased, and a new training programme was developed after they were normalized with the Z-score normalization given in Equation 1.

$$Zscore = Y - m/q \qquad (1)$$

Where Z represents the Z-score normalization, Y stands for the variables, m represents the mean of the sample, and q stands for the standard deviations. With the assistance of One Hot encoder’s straightforward mathematical equations interpretation, the conversion of textual features to quantitative data, albeit in the opposite direction, was successfully carried out. The input was processed with anyone hot modulator, which resulted in an increase from 41 to 125 different parameters being available for selection. In this piece, the learning algorithm was provided with 75% of the data, but the system testing was only provided with 25%.

C. *Deep CNN*

Deep learning models are among the most significant examples of computational networking since they are constructed from a large number of hidden nodes that each contain components and the means by which they can be connected to one another shown in Figure 2. The method of deep neural networks that was utilized to develop the model for this investigation may be dissected into three distinct components. The architecture of the model is the first factor to consider. This aspect dictates the layered structure, the number of neurons present in each layer, and the connectivity between the different layers. Second, the transmission of information between neurons uses forwards transmission, and it uses feed-forward neural classifiers and receives training. The third step involves transfer learning accompanied by an error function and algorithm.

D. *Topology of the model*

1. The input layer applies statistical methods that will be utilized by the neural network. The properties of the information contained in the produced code show that the utilized system's input data consists of 125 vertices. These vertices make up the input data.

2. *Hidden layers*: This layer is the layer that sits between the source nodes and the destination nodes, and it is in this layer that all of the computations begin to take place. The utilized system is made up of two secret units: the first of which consists of (50 neural nodes), and the second of which contains of

(30 neural nodes). The instructions served as the basis for the selection of this number.

3. *The output layer*: This layer is responsible for producing the outcome (normal or attack with mention to attack types). In the incoming network, every vertex is connected to every other vertex in the next concealed state, and so on through the rest of the stages. As can be seen in Figure 3, the connections between the nodes are considered to be a linked network.

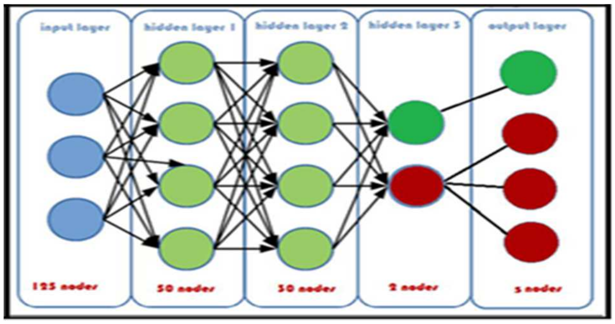

Figure 3. Proposed DCNN topology

E. Forward propagation

Forwards transmission makes an attempt to predict answers by utilizing a classification performed by a recurrent neural network (attacks or ordinary). Two types of supervised machine learning perceptron's are referred to as single-layer perceptron's and multi-layer perceptron's respectively. A multi-layer perceptron is utilized in the machine learning technique, which serves as the fundamental building block for deep neural networks. In Calculation 2, the fundamental equations of the perceptron are outlined and discussed.

$$X = \sum_{i-1}^{n} Yi * Wi + b \qquad (2)$$

Where n is the total number of networks that are contained within the layers, y represents the contents of these vertices (the values of the dataset), Wi stands for Weights (the connected strengths), and b stands for the Bias of these nodes. The discoveries are going to be incorporated into the Training Algorithm. The examination of how a neuron functions inside a neural network gives rise to the concept of a deep neural network in general. This investigation reveals that a neuron becomes activated once it reaches a certain level, which is referred to as the activation potential. This also reduces the number of different outcomes that are possible. The most common training algorithms include Sigmoid, ReLU, softmax, and tanha; this system makes use of ReLU activation capabilities in the convolutional nodes.

F. Back Propagation Process

When training a deep neural network with stochastic gradient descent, adaptation weights and bias are two important components [14]. There are also included algorithms for the loss function and the optimization. The logistic regression, which is a kind of the minimization issue, will cause the price to drop in order to achieve the best values for the parameter value. In a neural network, connection weights are referred to as parameter values. The needs that come with each paradigm are unique to that paradigm, and the architecture of the model is described in terms of those parameter values. The evaluation of the company can be done with the help of a differential equation (loss function). When establishing the best structure, the motivational factor that will be used will be the minimization of the loss function for each parameter. Cross probability characteristics were utilized for the purpose of this investigation. The loss function needs to be used in order to arrive at the optimal values for the model's parameters (weight and bias). The programmer [15] is a method for determining the parameter value that is optimal for the situation. Some of the most common wavelet coefficients include batch optimization techniques, RMS prop, learning algorithms, and Adam. Adam was found to be the most effective optimizer in a competition including a number of different optimizing compilers.

G. NSL KDD Datasets Simulations and Outcome

Windows XP Professional was installed on a machine with an AMD Athon TM64X2 Dual Core Processor 6000 +2.59GMHz processor, 4GB of RAM, and it ran at a frequency of +2.59GHz. MatLab 7.2 was utilized for the testing of the DCNN model that was proposed. During the testing phase, a DCNN model with the specified parameters was trained using 10% of the NSL KDD labeled data. The NSL KDD dataset is comprised of three distinct forms of traffic and six distinct types of DoS attacks, with a total size of approximately four terabytes. Each traffic record in the dataset has 41 features that help differentiate between normal traffic and DoS attack traffic. There are 3, 91,458 attacks traffic records and 97,277 normal traffic records in the NSL KDD dataset that has 10% of its records tagged.

H. Performance Evaluation NSL KDD Datasets

In order to compute the proposed DCNN model's precision, recall, and F-measure, we use the confusion matrix in Table 1 as a data source. This allows us to determine the model's detection accuracy. The following requirements, which are detailed in more detail below, are applicable to both the NSL and KDD datasets.

TABLE I MATRIX OF CONFUSION

	Predictor class	
	Usual	Attacked
Usual	True Positives values (TPV)	False Positives Values (FPV)

Attacked	False Negatives values (FNV)	True Negatives Values (TNV)

Where,

- TPV: the number of usual occurrences that have been successfully labeled as normal.
- FPV: are the counts of regular occurrences that are mistakenly identified as assaults.
- FN values: The count of attack occurrences is projected wrongly as usual.
- TN values: The attack's number of occurrences is accurately anticipated.

The following performance metrics were calculated as a result of these:

$$\text{Recall value } \frac{\text{TPV}}{\text{TPV} + \text{FNV}}$$

$$\text{Precision value } \frac{\text{TPV}}{\text{TPV} + \text{FPV}}$$

$$\text{F} - \text{Measure values} = \frac{2 + \text{Recall value} + \text{Precision value}}{\text{Recall value} + \text{Precision value}}$$

$$\text{on the whole}$$

$$\text{Accurateness} = \frac{\text{TPV} + \text{TNV}}{\text{TPV} + \text{TNV} + \text{FNV} + \text{FPV}}$$

Table II shows the amount of records used in the testing and training phases, as well as the training and testing times for the NSL KDD set.

TABLE II TRAINING AND TESTING– NSL KDD DS

Type of Data based on attacks	Training time in S	Training time in S	Data to train	Data to test
Normal	1.65	0.08	97277	60255
Smurf	0.63	0.04	641	400
Neptune	1.46	0.07	51820	20500
Back	0.9	0.06	994	714
Teardrop	0.7	0.02	918	300
Land	0.1	0.003	19	07
Pod	0.4	0.01	206	101

It is apparent from the above experimental findings that NSLKDD is more practicable since the training and testing duration of the data set is significantly shorter.

I. Testing NSL KDD Dataset against various attacks

The precision, recall, f-measure, and finally accuracy of the proposed DCNN model are depicted in the Table III below.

TABLE III: SIMULATION RESULTS WITH NSL KDD

Metrics (%)		DATATYPE						
	Back	Normal	Land	Neptune	Smurf	Tear drop	Pod	Mail bomb
Precision values	99.30	99.66	100	99.41	99.50	99.67	99.01	99.01
Recall values	100	100	100	100	100	100	100	98.00
F-measure values	99.65	99.83	100	99.51	99.75	99.83	99.50	97.81
Accuracy	99.30	99.66	100	99.71	99.50	99.67	99.50	98.60
Overall DCNN's quality aspect when evaluated with the NSL KDD DS					99.49%			

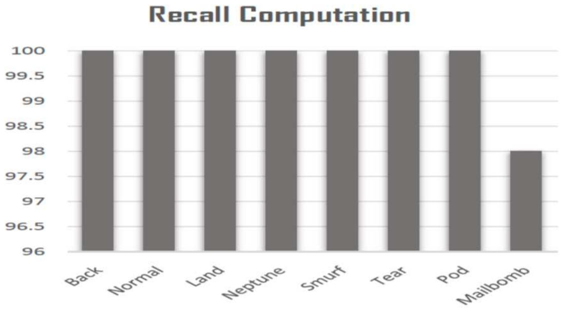

Figure 4. Computation of Precision

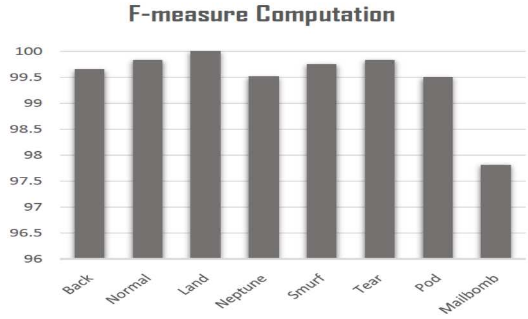

Figure 5. Computation of Recall

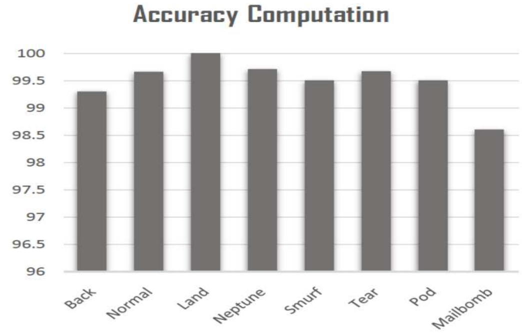

Figure 6. Computation of F-measure

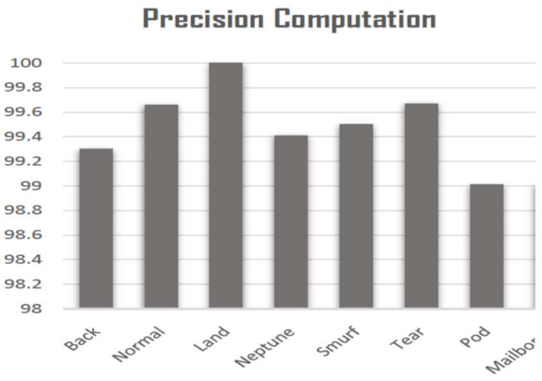

Figure 7. Computation of Accuracy

Figures 4, 5, and 6 illustrate the graphical representations of precision, recall, and f-measure, while Figure 7 shows the graphical representations of accuracy. The average precision of Back, Normal, Land, Neptune, Smurf, Tear drop, and Pod is 99.45 percent, and their average recall is 99.75 percent. Their f-measure average is 99.49 percent, and their accuracy is 99.49 percent. The fact that the DCNN implementation takes into account a resident size of 10 for each age group contributes to an improvement in the trade-off value of the suggested method. Each individual in the population has the ability to serve as a recognition rule. In order to get the population of the future cohort, we set aside two of the greatest people and rules from the current generation. The optimal set of rules is the set of rules that has the highest fitness value. Other members of the subsequent generation are the product of a procedure known as a crossover, which makes use of a method known as uniform random parent selection. We employ the utilization of a single-point crossover while utilizing this strategy. In addition to this, 20% of the new population is made up of people from other countries, and the mutation range is set at 30%. The proposed DCNN model's overall performance has been greatly improved by examining the results of simulation using the NSL KDD set. As was noted before, the model now achieves an average accuracy of 99.49 percent for every type of assault.

IV. Conclusion

Genetic algorithms have the advantage of being able to quickly find solutions that are optimal on a global scale, and they also have the benefit of being able to focus on individuals within a species in a manner that is complementary to that emphasis. The self-tuning features of the DCNN as a classifier make it possible for the patterns to arrive at the best possible solutions. The classification technique improves speed and detection rate while only using a little amount of processing resources, which contributes to the increased efficacy of the DCNN model. This strategy has shown greater detection performance when compared to earlier approaches and is particularly successful at finding different approaches. In addition, this technique can identify approaches more quickly. According to the findings of the testing, the DCNN model has detection results of 99.49% and a false alarm rate of 0.51 percentage compared to NSL KDD. This places it among the enhanced detection approaches used in IDS, and it is ranked as one of the best. On the other hand, zero-day attacks are something that should be looked into because of the huge impact they have on real-time applications. It has been determined which deep learning classifiers or components will be utilized in the future in order to enhance the performance of predictions.

References

[1] Ashiku, Lirim, and CihanDagli. (2019) "Cybersecurity as a Centralized Directed System of Systems using SoS Explorer as a Tool."2019 14th AnnualConference System of Systems Engineering (SoSE), 140-145.IEEE.

[2] Duque, Solane, and MohdNizam bin Omar. (2015) "Using data mining algorithms for developing a model forintrusion detection system (IDS)."Procedia Computer Science, 61: 46-51.

[3] Stallings, William, Lawrie Brown, Michael D. Bauer, and Arup Kumar Bhattacharjee.(2012) Computer security: principles and practice. Upper Saddle River, NJ, USA. Pearson Education.

[4] Whitman, Michael E., and Herbert J. Mattord.(2011) Principles of information security. Cengage Learning.

[5] Shone, Nathan, Tran Nguyen Ngoc, Vu DinhPhai, and Qi Shi. (2018) "A deep learning approach to network intrusion detection." IEEE Transactions on Emerging Topics in ComputationalIntelligence, 2, no. 1: 41-50.

[6] L. Lipton, Zachary C., John Berkowitz, and Charles Elkan. (2015) "A critical review of recurrent neural networks for sequence learning." arXiv preprint arXiv:1506.00019.

[7] Moustafa, Nour, and Jill Slay. (2015) "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)." 2015 military communications and information systems conference (MilCIS), 1-6.IEEE.

[8] Osken, Sinem, EcemNurYildirim, GozdeKaratas, and LeventCuhaci. (2019) "Intrusion Detection Systems with Deep Learning: A Systematic Mapping Study." 2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT), 1-4. IEEE.

[9] McHugh, John. (2000) "Testing intrusion detection systems: a critique of the 1998 and 1999 darpa intrusiondetection system evaluations as performed by lincoln laboratory." ACM Transactions on Information and System Security (TISSEC) 3, no. 4: 262-294.

[10] Moustafa, Nour, and Jill Slay. (2015) "The significant features of the UNSW-NB15 and the KDD99 data sets for network intrusion detection systems." In 2015 4th international workshop on building analysis datasets and gatheringexperience returns for security (BADGERS), 25-31. IEEE.

[11] "Home." CVE. Accessed December12, 2019. http://cve.mitre.org/about/index.html.

[12] Kumar K, Anand S, Yadava RL. Advanced DSP Technique to Remove Baseline Noise from ECG Signal. Int J Electron Comput Sci Eng. , , vol. 1, no. 3, pp. 1013-1019, 2012.

[13] Kumar K, Tanya Aggrawal, Vishal Verma, Suraj Singh, Shivendra Singh, Dr. Lokesh Varshney, “Modeling and Simulation of Hybrid System”, IJAST, vol. 29, no. 4s, pp. 2857 -2867, Jun. 2020.

[14] Kumar K, Varshney L, Ambikapathy, Vrinda, Sachin, Prashant , Namya. Soft Computing and IoT based Solar Tracker. International Journal of Power Electronics and Drive System (IJPEDS). Vol 12, No 3: September 2021. doi.org/10.11591/ijpeds.v12.i3.pp1880-1889.

[15] Lin, Wen-Hui, Hsiao-Chung Lin, Ping Wang, Bao-Hua and Jeng-Ying Tsai. (2018) "Using convolutional neural networks to network intrusion detection for cyber threats." 2018 IEEE International Conference on Applied System Invention (ICASI), 1107-110. IEEE.