Detect and Classify the Unpredictable Cyber-Attacks by using DNN Model

G. Swarnalatha[1], Rajaram Jatothu[2], E. Shirisha[3]

[1]PG Student,
[2]Professor, (drjrajaram16@tkrec.ac.in)
[3]PG Student.
Department of CSE,Teegala Krishna Reddy Engineering College, Hyderabad,Telangana,India.

Article History:Received:11 november 2020; Accepted: 27 December 2020; Published online: 05 April 2021

Abstract - Machine learning techniques are often used to develop IDS by detecting and deploying fast and automated network attacks to torpedoes and host standards. However, there are many problems, as severe attacks change all the time and occur at very high levels that require a lot of resolution. There are many malicious packages available for further investigation by the cybersecurity community. However, one completed study did not provide a complete analysis to apply different machine learning algorithms on different media packages. Because of the persistent methods of attack and the dynamic nature of malware, it is important to systematically update and approve malicious packages that are available to the public. This paper explores the DNN, a type of comprehensive learning model, promoting flexible and appropriate IDS for detecting and deploying expected and unpredictable online attacks. Sustainable industrial development and rapid development of attacks need evaluation for some data developed over the years using static and dynamic methods. This type of research can help determine the best algorithm to identify future attacks. Comparative data for some commonly available malware provides a comprehensive comparison of DNN experiences with other class machine learning classifications. The best network parameters and network topologies for DNN are selected using the KDDCup 99 package with this hyperparameter selection method. The DNN model, which works well on KDDCup 99, works on other data, such as the NSL-KDD memory test. Our DNN model teaches how to transfer IDS information functions from multicultural.

Multidisciplinary representations in a variety of encryption. Complex tests have shown that DNN performs better than conventional machine learning classification. Finally, we present a large and hybrid DNN torrent structure called Scale-Hybrid-IDS-AlertNet, which can be used to effectively monitor the impact of network traffic and host-level events to warn directly about cyber-attacks.

Keywords: Cyber security, intrusion detection, malware, big data, machine learning, deep learning, deep neural networks, cyberattacks, cybercrime.

1. INTRODUCTION

1.1 Introduction

Information and communication technology systems and networks perform a variety of confidential user information processes, which are edited to coordinate to update this document.They are sensitive to multiple attacks that creep in and out. These attacks can be done manually and with the help of machines and can lead to slow corruption and data corruption. For example, Yahoo data breaches lost $ 350 million, but Bitcoin logins lost around $ 70 million. Academic research allows translation and content, and such online visuals can only be developed using sophisticated algorithms that include the latest advances in technology, software, and network topology, as well as the Internet of Things (IoT). Malicious cyber-attacks are serious security breaches that require a new, simple, and secure access identification system (IDS). Identification intrusion detection tools work to automatically detect and track these points, security attacks or violations, and torpedo levels and infrastructure. According interactive traffic, this Detection is divided into NIDS and HIDS. IDS systems that use internet traffic are called NIDID. Network characteristics are collected using network equipment including modifications to network devices, routers, and network devices, and analyzed to identify hidden attacks and potential threats to network traffic. The IDS system is called HID which attacks using system actions of some type of log file running on a nearby computer. Memory files are collected by internal sensors. When NID monitors the contents of each packet in network traffic, it relies on HID memory file information, including audited logs, system logs, program logs, file systems, disk resources, user account information, and other information about each system. Many organizations use NID and HID hybrids.

1.2 Purpose of the Research Paper

Malicious cyber-attacks are serious security breaches that require new, simple, and IDS. The basic mobile operating system is divided into NID and HID. IDS systems that use internet traffic are called NIDID. Network characteristics are collected using network equipment including modifications to network devices, routers, and network devices, and analyzed to identify hidden attacks and potential threats to network traffic. The IDS

*Corresponding author : G. Swarnalatha
PG Student,
Department of CSE,Teegala Krishna Reddy Engineering College, Hyderabad,Telangana,India.

system that detects an attack using system logs of multiple memory files running on a local host computer is called an HID. Memory files are collected by internal sensors. When NID monitors the contents of each packet in network traffic, it relies on HID memory file information, including audited logs, system logs, program logs, file systems, disk resources, user account information, and other information about each system. Many organizations use NID and HID hybrids. Humans use machine learning techniques to detect attacks

1.3 Problem statement

- The model rejects the main influence of lies and positive attacks
- It is not possible to change the model, because the current research report reports the effect of the machine learning model with only one set of data
- Then the model that has been tested so far has passed through the current internet traffic

1.4 Solution for the problem statement:

By combining NID and HID, a more accurate approach is proposed for deep detection of cyber-attacks for deep science by a DNN. In this study, the behavioral effects of learning behaviors with different DNN algorithms were evaluated using multiple sets of NID and HID data when they determined whether network movement was associated with normal attacks or not. Related attack groups.

Its function provides a unique DNN structure for NID and HID, consisting of an insert layer, 5 hidden layers, and an output layer. Layers in DNN enable very complex job separation and the ability to identify systems in IDS data. Each class evaluates the soft properties sent to the second class and implements the final class in the DNN partition. Section submits 41 neurons for KDD Cup 99, 41 neurons for NSL-KDD, Output column 1 neuron for binary classification for each data type and 5 neurons for classification in multiple classes up to 99 columns in KDDC, 5 neurons from NSL Input Unit KDD is fully connected to the layer that is hidden by the layer. DNN has several tools.

2. LITERATURE SURVEY

Internet traffic flow analysis is done by detecting fraud, abuse, and analysis of government protocols. Fraud detection uses signatures and filters to identify attacks. To continue updating the signature statement, it depends on the opinion of the person. This method appears to be close to a well-defined attack, but it is certainly not effective against an unknown attack. Incorrect findings use arithmetic to detect unknown behavior. In most cases, anomaly detection leads to an increase in false positives. To combat this problem, many organizations use an unusual combination of detection systems in their business systems. Government protocol analysis is more appropriate based on the above identification method, as approval protocol analysis is used in industrial class, application class, and transportation class. It uses enterprise-specific settings to set the settings of the usage-related protocol. Considering the new in-depth training methods to improve the intelligence of such access detection methods, it is not sufficient to compare such machine learning algorithms with general data. The most common problems and solutions based on the machine learning model are: first, the vertical model with false-positive levels and major attacks; second, it is not possible to include an example because the current study has reported the effect of a machine learning model with only one set of data; third, the model being tested so far has missed the current mass movement; and finally, solutions are needed to maintain the speed, speed, and dynamics of rapidly increasing torpedoes. Although this difficulty is the main motivation for this work, the research aims to evaluate the effects of different learning machines and DNNs that work on NIDs and HIDs. They perform this function;

- The attacker tries to hide because the average user remains hidden from the access system. However, the pattern of aggressive behavior varies in several ways. This is for attacking specific purposes, such as unauthorized access to computers and Internet resources.
- You can map out the use of online resources, but the available methods lead to an increase in false-positive results.
- Appropriate instructions are usually long, and very brief.
- In general, this study contributes to cybersecurity as follows:
- A more in-depth training strategy for the effects of NID and HID is proposed to DNN.

Examining the effects of different behavioral machines and DNN algorithms on different NID and HID data sets, this study can determine whether the movement of network traffic as a result of normal or abnormal attacks and will be divided into appropriate attack groups. Bicycle. The mother tongue activity is added to the systemic dialogue about gender and semantic equality and the semantic context and the continuous information of the call system, namely the Native Language Text Representation method.

STAGES OF COMPROMISE

An Attacker’s View

Violations are often initiated by unauthorized users, also known as attackers. Attackers can try to access computers remotely from the Internet or prevent them from working remotely. Recognizing records requires an understanding of how the system attacks. Typically, attacks can be divided into five stages. It is exploration, exploitation, unity, unity, and plunder. Attacks can be detected in the first three stages, but by the fourth or fifth stage, the system is completely stolen. Therefore, it is very difficult to distinguish the normal movement from the attack. During the research phase, the attacker attempts to gather information about the value and function and version of the operating system and usage. During the operation phase, the attacker uses special tools to reach the target computer. Employment can be divided into crime, cancellation, or violation of the law. The login service contains stolen keywords or glossary attacks and corrupts SQL queries. After unauthorized entry into the system, the attacker monitors hidden operations and installs and other devices to take advantage of the reinforcement phase. Hackers try to access all of your systems based on the user account you are using. Finally, attackers use programs that can be accessed through user accounts. The attacker manages all systems in the integration phase and the back door is set to work for communication in the integration phase. The last step is robbery, which includes potentially harmful actions by attackers who steal information and CPU processing and fraud. Because computers and networks are designed and configured by humans, it is thought to have hardware and software errors. The mistakes and errors of these people cause security problems. Confidentiality, accuracy, and access to information are the most important pillars of security.

Accuracy and accountability also play an important role in ensuring the security of information. In general, sexual assault is an attack such as listening, device attacks, such as system surveillance, is "detecting" and access restrictions related to network resource restrictions because they are not visible to normal users. The IDS system's ability to detect calls is limited. Disruptive attacks can be posted online or found. Attacks can now be defined as a series of actions that threaten corporate privacy, data integrity, access, or any security policies. First, the IDS system is designed to detect such attacks to prevent malicious activity from computers and networks.

3. OVERVIEW OF THE SYSTEM

3.1 Study of The System

1. Numpy
2. Pandas
3. Matplotlib
4. Scikit –learn

3.2. Proposed System

A combination of NID and HID is proposed for the practical definition of torpedo attacks as an effective training method for the deep network model (DNN). In this study, the effect of machine behavior training was different from the DNN algorithm measured using multiple NID and HID data sets to determine whether network traffic was normal due to attacks that could be divided into multiple groups. or was abnormal. gun attacks.

Features for NID and HID provide a single DNN structure, which consists of an initial layer, 5 hidden layers, and an external layer. Classification in DNN allows more complex tasks to be developed in IDS data and improved composite identification functions. Each layer evaluates the non-soft properties distributed by the last layer, and the last layer forms a subset in the DNN. The entry line for KDD Cup 99 is 41 neurons for NSL-KDD, 41 neurons for binary classification for each data in the start line, and 5 neurons in many classes up to 99.5 NSL neurons in KDDC. -KDD Typically, the input device for the output layer is hidden and fully integrated. DNN has several tools.

3.3. INPUT AND OUTPUT

Inputs and Outputs

The following are the project's inputs and outputs.

Inputs:

- Covers all the essentials like num numpy, panda, lib, sci-kit - learn how to use the algorithm kits you need to learn the machine.
- Graphics Adjust graphic size.
- Download and edit data exchange and edit.

Outputs:

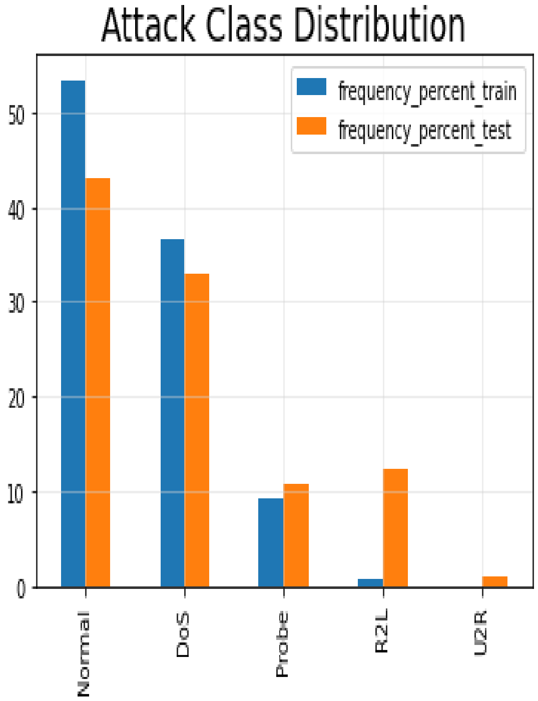
- Enter a data box to enter the zero value and related information.
- Encamp All results are filtered for display.
- Machine After using the machine learning algorithm, it gives good results with visual animation.

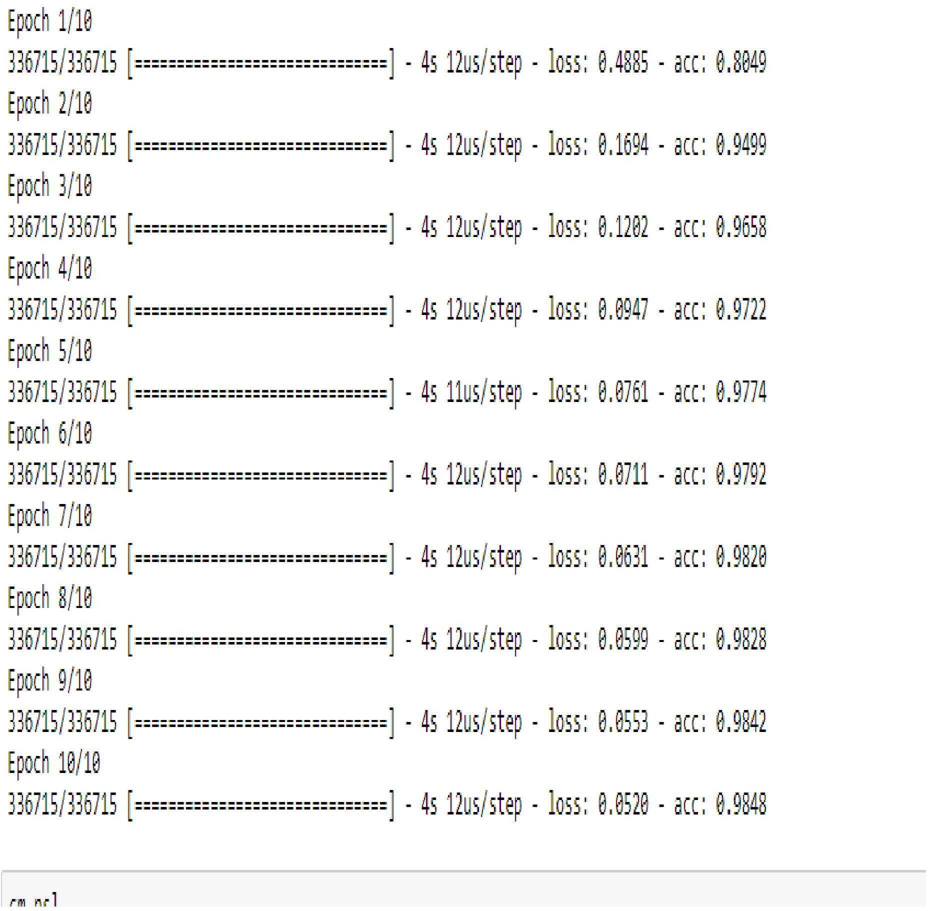
3.4 FUNCTIONAL REQUIREMENTS

User

- Load data
- Data analysis
- Data preprocessing
- Model building
- Prediction

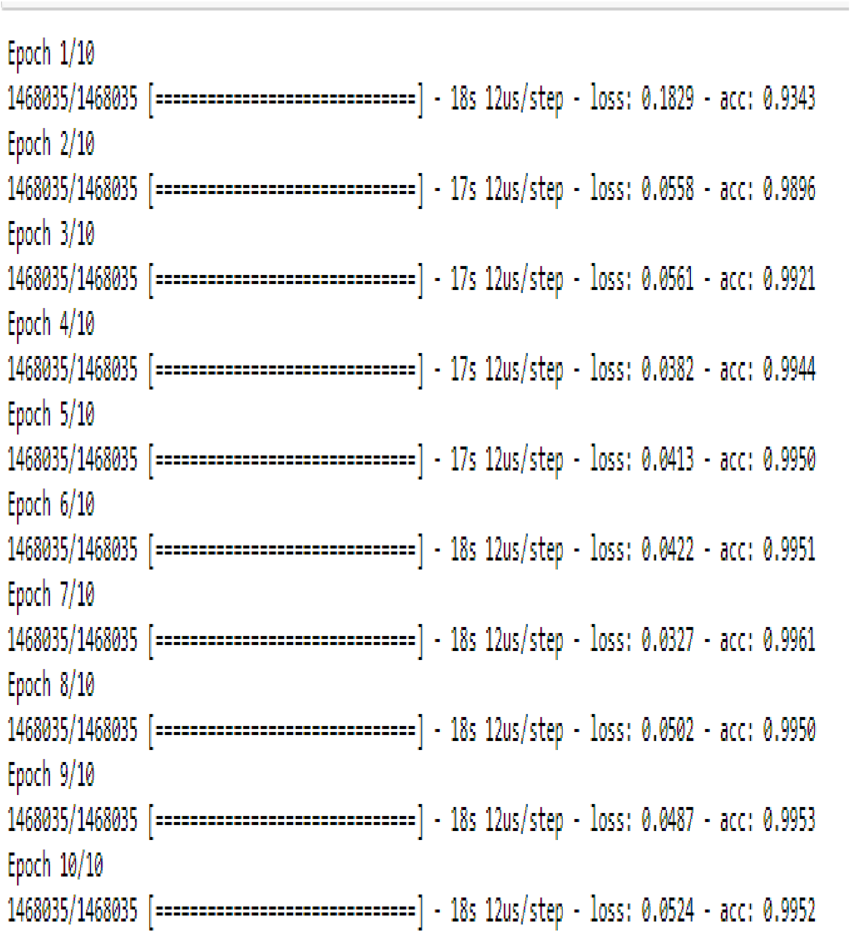
4. OUTPUT



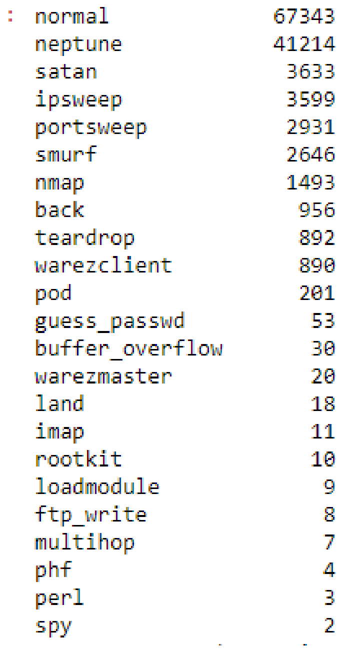


```
Epoch 1/10
336715/336715 [==============================] - 4s 12us/step - loss: 0.4885 - acc: 0.8049
Epoch 2/10
336715/336715 [==============================] - 4s 12us/step - loss: 0.1694 - acc: 0.9499
Epoch 3/10
336715/336715 [==============================] - 4s 12us/step - loss: 0.1202 - acc: 0.9658
Epoch 4/10
336715/336715 [==============================] - 4s 12us/step - loss: 0.0947 - acc: 0.9722
Epoch 5/10
336715/336715 [==============================] - 4s 11us/step - loss: 0.0761 - acc: 0.9774
Epoch 6/10
336715/336715 [==============================] - 4s 12us/step - loss: 0.0711 - acc: 0.9792
Epoch 7/10
336715/336715 [==============================] - 4s 12us/step - loss: 0.0631 - acc: 0.9820
Epoch 8/10
336715/336715 [==============================] - 4s 12us/step - loss: 0.0599 - acc: 0.9828
Epoch 9/10
336715/336715 [==============================] - 4s 12us/step - loss: 0.0553 - acc: 0.9842
Epoch 10/10
336715/336715 [==============================] - 4s 12us/step - loss: 0.0520 - acc: 0.9848
```

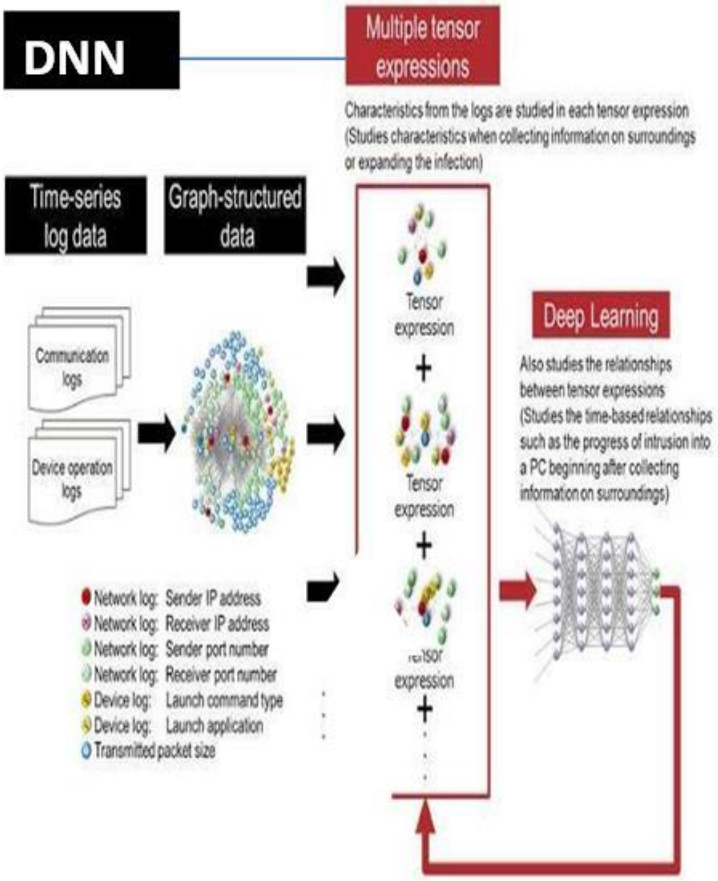
```
cm_nr]
```



```
Epoch 1/10
1468035/1468035 [==============================] - 18s 12us/step - loss: 0.1829 - acc: 0.9343
Epoch 2/10
1468035/1468035 [==============================] - 17s 12us/step - loss: 0.0558 - acc: 0.9896
Epoch 3/10
1468035/1468035 [==============================] - 17s 12us/step - loss: 0.0561 - acc: 0.9921
Epoch 4/10
1468035/1468035 [==============================] - 17s 12us/step - loss: 0.0382 - acc: 0.9944
Epoch 5/10
1468035/1468035 [==============================] - 17s 12us/step - loss: 0.0413 - acc: 0.9950
Epoch 6/10
1468035/1468035 [==============================] - 18s 12us/step - loss: 0.0422 - acc: 0.9951
Epoch 7/10
1468035/1468035 [==============================] - 18s 12us/step - loss: 0.0327 - acc: 0.9961
Epoch 8/10
1468035/1468035 [==============================] - 18s 12us/step - loss: 0.0502 - acc: 0.9950
Epoch 9/10
1468035/1468035 [==============================] - 18s 12us/step - loss: 0.0487 - acc: 0.9953
Epoch 10/10
1468035/1468035 [==============================] - 18s 12us/step - loss: 0.0524 - acc: 0.9952
```



```
:  normal             67343
   neptune            41214
   satan               3633
   ipsweep             3599
   portsweep           2931
   smurf               2646
   nmap                1493
   back                 956
   teardrop             892
   warezclient          890
   pod                  201
   guess_passwd          53
   buffer_overflow       30
   warezmaster           20
   land                  18
   imap                  11
   rootkit               10
   loadmodule             9
   ftp_write              8
   multihop               7
   phf                    4
   perl                   3
   spy                    2
```


DNN
Multiple tensor expressions
Characteristics from the logs are studied in each tensor expression (Studies characteristics when collecting information on surroundings or expanding the infection)
Time-series log data
Graph-structured data
Communication logs
Device operation logs
Tensor expression
Tensor expression
Tensor expression
Deep Learning
Also studies the relationships between tensor expressions (Studies the time-based relationships such as the progress of intrusion into a PC beginning after collecting information on surroundings)
Network log: Sender IP address
Network log: Receiver IP address
Network log: Sender port number
Network log: Receiver port number
Device log: Launch command type
Device log: Launch application
Transmitted packet size

DEEP NEURAL NETWORK (DNN)

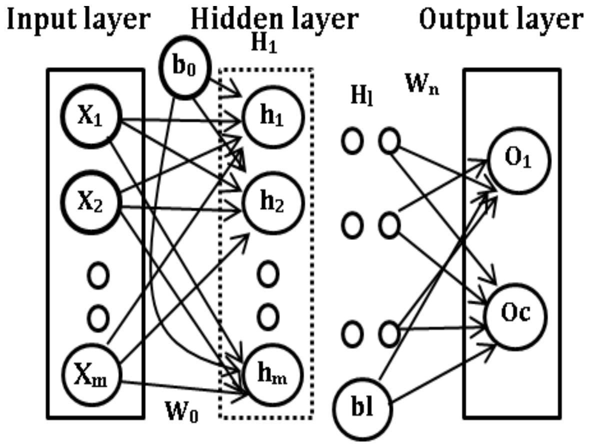
We employ an artificial neural network (ANN) approach as the computational model since it is influenced by the characteristics of biological neural networks to incorporate intelligence in our proposed

Table 1: Training and testing connection

Attack category	Description	Data instances - 10 % data			
		KDDCup 99		NSL-KDD	
		Train	Test	Train	Test
Normal	Normal connection records	97,278	60,593	67,343	9,710
DoS	Attacker aims at making network resources down	391,458	229,853	45,927	7,458
Probe	Obtaining detailed statistics of system and network configuration details	4,107	4,166	11,656	2,422
R2L	Illegal access from remote computer	1,126	16,189	995	2,887
U2R	Obtaining the root or super-user access on a particular computer	52	228	52	67
Total		**494,021**	**311,029**	**125,973**	**22,544**

method. Feed forward neural network (FFN), a type of ANN is represented as a directed graph to pass various system information along edges from one node to another without forming a cycle. We adopt a multilayer perceptron (MLP) model which is a type of FFN having three or more layers with one input layer, one or more hidden layers and an output layer in which each layer has many neurons or units in mathematical notation. We select the number of hidden layers by following a hyper parameter selection method. The information is transformed from one layer to another layer in a forward direction with neurons in each layer being fully connected.

FIGURE 1. Architecture of a deep neural network (DNN).



5.CONCLUSION AND FUTURE SCOPE

CONCLUSION

In this paper, I propose a hybrid access detection system that uses a test tool on a regular host server to analyze host behavior and performance. In this context, a comprehensive model of in-depth training, process, and data analysis are conducted directly by DNN. The DNN model was selected through a comprehensive evaluation of its performance about the class machine learning classification in different IDS memory packs. Also, we set up host and network assets in real-time and use the recommended DNN model to detect attacks and raids. In any case, we find that DNN is more than just a standard machine learning class. The architecture we offer works better than the HID and NID classification standards. As far as we know, this is the only system that can use DNN networks and host activities on a distributed network to detect true attacks.

FUTURE WORK

Besides, the proposed system efficiency can be improved by adding DNS and BGP monitoring modules to torpedoes. The proposed system working time can be increased by adding more nodes to existing groups. Besides, the proposed system does not contain complete information about the structure and function of malware. In general, the device can be further enhanced by using the device DNN method on advanced devices. Due to the high inventory costs associated with the complex development of DNNs, they are not taught using the IDS references mentioned in this study. This can be an important task in a stressful situation and one of the key areas for future work.

6. REFERENCES

B. Mukherjee, L. T. Heberlein, and K. N. Levitt, ``Network intrusion detection,'' IEEE Netw., vol. 8, no. 3, pp. 2641, May 1994.

D. Larson, ``Distributed denial of service attacks holding back the good,'' Netw. Secur., vol. 2016, no. 3, pp. 57, 2016.

R. C. Staudemeyer, ``Applying long short-term memory recurrent neural networks to intrusion detection,'' South Afr. Comput. J., vol. 56, no. 1, pp. 136154, 2015.

S. Venkatraman and M. Alazab, ``Use of data visualization for zero-day Malware detection,'' Secure. Commun. Netw., vol. 2018, Dec. 2018, Art. no. 1728303. [Online]. Available: https://doi.org/10.1155/ 2018/1728303

P. Mishra, V. Varadharajan, U. Tupakula, and E. S. Pilli, ``A detailed investigation and analysis of using machine learning techniques for intrusion detection,'' IEEE Commun. Surveys Tuts., to be published. DOI: 10.1109/comst.2018.2847722.

A. Azab, M. Alazab, and M. Aiash, ``Machine learning-based botnet identification traffic," in Proc. 15th IEEE Int. Conf. Trust, Secure. Privacy Comput. Commun. (Trustcom), Tianjin, China, Aug. 2016, pp. 17881794.

R. Vinayakumar. (Jan. 2, 2019). Vijayakumar/Intrusion-Detection V1 (Version V1). [Online]. Available: http://doi.org/10.5281/zenodo.2544036