

A PROJECT REPORT
on
“NETWORK INTRUSION DETECTION SYSTEM USING
NEURAL NETWORKS, AIDED BY BLOCKCHAIN”

Submitted to
KIIT Deemed to be University

In Partial Fulfillment of the Requirement for the Award of

BACHELOR’S DEGREE IN
COMPUTER SCIENCE AND ENGINEERING

BY

ANANYA DUTTA	1805011
ARISMITA BANERJEE	1805021
ARKAPRABHA SAMANTA	1805106
SAYANTANI BALA	1805245

UNDER THE GUIDANCE OF
PROF. PRADEEP KR. MALLICK



SCHOOL OF COMPUTER ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESWAR, ODISHA -751024
May 2021

KIIT Deemed to be University

**School of Computer Engineering
Bhubaneswar, ODISHA 751024**



CERTIFICATE

This is certify that the project entitled

**“NETWORK INTRUSION DETECTION SYSTEM USING
NEURAL NETWORKS, AIDED BY BLOCKCHAIN”**

submitted by

ANANYA DUTTA	1805011
ARISMITA BANERJEE	1805021
ARKAPRABHA SAMANTA	1805106
SAYANTANI BALA	1805245

is a record of bonafide work carried out by them, in the partial fulfillment of the requirement for the award of Degree of Bachelor of Engineering (Computer Science & Engineering OR Information Technology) at KIIT Deemed to be university, Bhubaneswar. This work is done during year 2019-2020, under our guidance.

Date: 15 / 05/ 2021

Prof. Pradeep Kr. Mallick
Project Guide

Acknowledgment

We are deeply obliged to Prof. PRADEEP KR. MALLICK, our project guide and Dr. Jagannath Singh, Coordinator, Project FIC for their persistent support and guidance all throughout to see that this project meets its target since its initiation to its fulfillment.




ANANYA DUTTA



ARISMITA BANERJEE



ARKAPRABHA SAMANTA



SAYANTANI BALA

ABSTRACT

The current world scenario tells that we all are going through a tough time. In the rise of the Global Pandemic, every individual has become dependent on the internet for day to day activities. Right from children to adults, students to professionals use the internet for various online activities. In order to perform these online activities one needs a safe and secure network so that the data is not stolen by third parties. To ensure a safe and secure internet services, NIDS is used. Network Intrusion Detection System is an effective module to distinguish and protect against unwanted third parties before the framework is influenced.

However, there are certain limitations. The biggest security issue faced by this neural network is that it is not tamper resistant. One can make a change in one of the layers of the model. This will result in a wrong prediction output of the model, thus drastically reducing its efficiency, and posing a threat to the authenticity of the result. So, to ensure the security of this NIDS model, we have used Blockchain.

Keywords : Blockchain, Blocks, Chain, Hashing, Decentralization, Nodes, Bias, Agreements, Mining, Security, Authenticity, Dataset, Neurons, Weights, Activation Function, Neural Network.

CONTENTS

1	Introduction	8
1.1	Project Motivation	8
1.2	Problem Statement	8
1.3	Core Concepts Used	9
1.3.1	Neural Network	9
1.3.2	Blockchain	10
2	Requirement Analysis	12
3	Software Requirements Specification	13
3.1	Functional Requirements	13
3.1.1	Import data from network	13
3.1.2	IDS	13
3.1.3	Result	13
3.1.4	Response	14
3.1.5	Blockchain Security Check	14
3.1.6	Update	14
3.2	Non-Functional Requirements	15
3.2.1	Performance Requirements	15
3.2.2	Security Requirements	15
3.2.3	Reliability	15
3.2.4	Maintainability	15
3.2.5	Integrity	16
3.2.6	Correctness	16
3.2.7	Test-ability	16
3.2.8	Flexibility	16
4	Project Planning and System Design	17
4.1	Project Scope And Objectives	17
4.2	Establish Project Infrastructure	17
4.3	Analysis of Project Characteristics	18
4.4	Project Product and Activities	18
4.4.1	Product Flow Diagram	18
4.4.2	Activity Network Diagram	19
4.5	Effort Estimation for Each Activity	19

4.6	Identify Activity Risks	19
4.7	Allocate Resources and Gantt Chart	19
4.7.1	Pert Table	20
4.7.2	Project Network Diagram	20
4.7.3	Gantt Chart	20
5	Project Implementation	21
5.1	Neural Networks	21
5.1.1	Data Cleaning	22
5.1.2	Classification Techniques	23
5.1.3	Neural Networks	24
5.1.4	Clustering Techniques	24
5.1.5	Feature Selection	25
5.2	Blockchain in Neural Network	26
5.2.1	Blockchain Creation	27
5.2.2	Creation of New Block	28
5.2.3	Proof of Work Algorithm	28
5.2.4	Integration of neural network with blockchain	28
6	Screenshots of the Project	30
7	Results and Discussions	33
8	Conclusion	34
9	Future Scope	35
10	References	36

List of figures

Fig 1.1. Deep Neural Network	9
Fig 1.2. Blocks in a blockchain linked cryptographically through hash	10
Fig 4.1. Product Flow Diagram	18
Fig 4.2. Activity Network Diagram	19
Fig 4.3. Compressed Activity Network Diagram	19
Fig 4.4. Pert Table	20
Fig 4.5. Project Network Diagram	20
Fig 4.6. Gantt Chart	20
Fig 5.1. Intrusion Detection System using NN	22
Fig 5.2. Data Cleaning	23
Fig 5.3. Data Classification	23
Fig 5.4. Neural Network	24
Fig 5.5. Clustering	25
Fig 5.6. Feature Selection	25
Fig 5.7. Sequential Model	26
Fig 5.8. transition of a neural network architecture to a deep ring architecture	27
Fig 5.9. Creation of Blocks	27
Fig 5.10. Proof of Work	28
Fig 5.11. Layer i of neural network represented as block j of deep ring	29
Fig 5.12. Ouroboros block: the starting and ending block of the deep ring	29
Fig 6.1. Data	30
Fig 6.2. Data	30
Fig 6.3. Data	30
Fig 6.4. Heat map	31
Fig 6.5. Value count of two classes under different flags	31
Fig 6.6. Value count of two classes under different services	32
Fig 6.7. Accuracy of the model	32

Chapter 1

Introduction

1.1 PROJECT MOTIVATION

In the 21st century, the network has become one of the essential parts of life, and lots of primary activities are performed by using the network. Also, network security plays an important role in the administrator and monitors the operation of the system. The world security context is changing more than ever. Military interest has shifted from the conventional means of warfare to that of cyber warfare. The most potent nations have entire armies that are watching the international cyberspace for anomalies. And these forces are ready to intervene for keeping peace at home or for an enemy nation. The international interest in exploit development has risen significantly.

In this ever-changing world of computer technology and rapidly evolving cyber threats, attackers and defenders are in a constant loop of cat-and-mouse. On the defender's side, a defence-in-depth strategy must be added to their security layer to effectively detect advanced persistent cyber threats, where Intrusion Detection System (IDS) is a key element. The network intrusion detection system (NIDS) is a crucial module to detect and defend against malicious traffics before the system is affected. This system can extract the information from the network system and quickly indicate the reaction which provides real-time protection for the protected system.

However, detecting malicious traffics is very complicating because of their large quantity and variants. Also, the accuracy of detection and execution time are the challenges of some detection methods. Hence, we are aiming to use an IDS platform based on a neural network, to detect DOS attack, web attacks, brute-force attacks over the SSH and FTP protocol, etc. Also, as cybersecurity is a major issue that is to be addressed, so the security of the neural network model also plays a crucial role, as a small change in the neural network model can affect the model's output to a large extent. Hence, the neural network model is thought to be securely set up in a blockchain network for further security purpose, thus making the system more robust, along with having high precision.

1.2 PROBLEM STATEMENT

The aim is to use the IDS platform, based on neural network in order to detect DOS attack, web attacks over SSH and FTP protocols, etc. We are using the NIDS model- Deep Ring to detect such malicious activities. But the biggest drawback of neural network model is that it is not tamper resistant. One can make changes in any layer of the model.

The idea is to prevent this tampering of data. In order to do this we have used Blockchain. Blockchain will secure each layer and will prevent any third party attack.

1.3 CORE CONCEPTS USED

1.3.1. Neural Network

The central thought behind distributed portrayal is the sharing of statistical qualities where various segments of engineering are re-utilized for various purposes. Deep neural models are made out of various layers using non-direct activities, for example, in neural nets with many hidden layers.

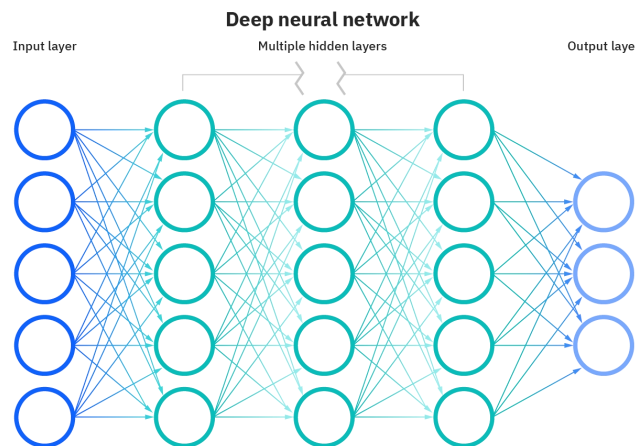


Fig 1.1. Deep Neural Network

1) Neuron - Just like a neuron frames the essential component of our cerebrum, a neuron shapes the fundamental design of a neural organization. At the point when we get the data, we measure it and afterwards, we produce a yield. Also, if there happens to be an occurrence of a neural organization, a neuron gets a piece of information, measures it and creates a yield which is either sent to different neurons for additional preparation or it is the last yield.

2) Weights - When information enters the neuron, it is increased by weight. For instance, in a situation where a neuron has two sources of info, each information will have a related weight appointed to it. We initiate the loads arbitrarily and these loads are refreshed during the model preparing measure. The neural organization then subsequently prepares and allocates a higher load to the information it considers more significant when contrasted with the ones which are viewed as less significant. A load of zero means that the specific component is insignificant. Let's expect the contribution to be 'a', and the weight-related to be 'W1'. At that point in the wake of going through the hub, the information becomes $a * W1$.

3) Bias – Besides the loads, another direct part is applied to the info, called the inclination. It is added to the aftereffect of weight increase to the information. The inclination is essentially added to change the scope of the weight increased information. Subsequently adding the inclination, the outcome would look like $a * W1 + \text{bias}$. This is the last linear segment of the information change.

4) Activation Function – Once the direct part is applied to the information, a non-linear capacity is applied to it. This is finished by applying the enactment capacity to the direct combination. The initiation work makes an interpretation of the information signs to yield signals. The yield after utilization of the actuation capacity would look something like $f(a*W1+b)$ where $f()$ is the enactment work.

1.3.2. Blockchain

Blockchain Technology- Blockchain is a decentralized, distributed, shared and immutable ledger, generally used to store various records, in the form of blocks. The blocks are connected with each other using hash functions, such that if one block is altered, the next block will not be accessible. There is a consensus algorithm that verifies the transactions, hence making the blockchain a highly verified and secure system.

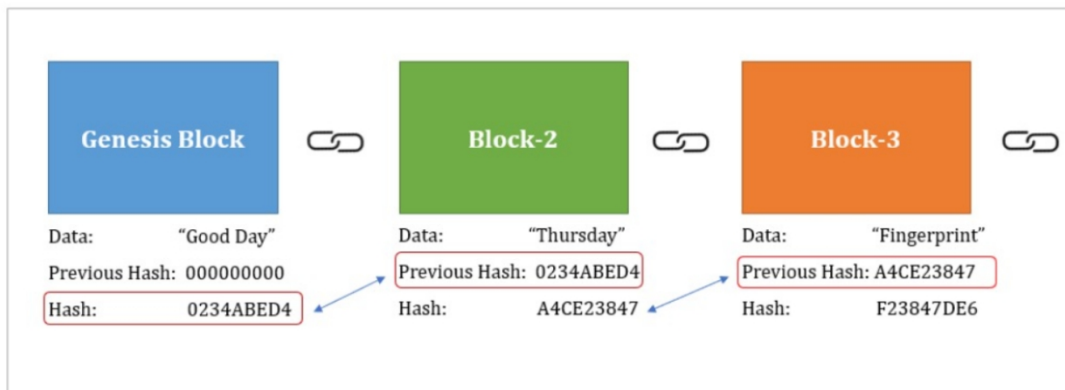


Fig 1.2. Blocks in a blockchain linked cryptographically through hash

- Blocks-** Each block consists of the hashes of the previous and the current block, the private and public keys of the current block the public key of the previous and next blocks, the parameters of the present and next block, and the AES key. The hash of a block is is a function of the previous block's hash, parameters of the current and next layer. There is a common ledger that is maintained which stores the state of the blockchain.
- Decentralized -** A blockchain is supposed to be decentralized as it isn't put away in one spot and doesn't have its centre. All things being equal, the information saved in the blockchain is distributed across a wide range of PCs, called nodes. Since no single substance has command over the information, clients connect with one another straightforwardly without the inclusion of an outsider.
- Decentralized Consensus -** A blockchain is a decentralized distributed framework that has no focal power to control the trading of data. In an ordinary unified model, a focal power or a leading group of chiefs take every one of the necessary choices. In any case, it is unimaginable on account of blockchain as it has no pioneer. The individuals from a blockchain network

need to go to an agreement through "agreement instruments" to decide. We will examine a portion of the huge agreement calculations in detail.

- d) Smart Contracts - Smart agreements are the structure blocks for blockchain-based applications. The idea driving keen agreements is the authoritative administration of exchanges between at least two members. It tends to be checked automatically with the blockchain, rather than a focal authority. Also, keen agreements permit clients to control proprietorship by offering controlled information exposure.

Chapter 2

Requirement Analysis

The requirements for the NIDS model are as follows:

1. Network inflow and outflow to the model to detect intrusion in the system using the parameters of the network.
2. The network admins should be able to monitor the components and functionalities of the Intrusion Detection System.
3. The network admins should be able to monitor the blockchain for any flaws.
4. If at all the Consensus Algorithm gets violated, the admins will get notified immediately.
5. If any intrusion happens in the network, the users must get notified so that he/she can take up necessary actions.
6. The admins should regularly update the neural network model whenever they receive new datasets.
7. The admins must generate a monthly/yearly report depending upon the user's requirements regarding the probabilistic analysis of the network.

Chapter 3

Software Requirements Specification

3.1. FUNCTIONAL REQUIREMENTS

3.1.1 Import data from network

Description: The IDS ought to have the option to persistently screen and report any interruption. The IDS should give sufficient data to patch up the framework, and along these lines decide the scope of harm and set up obligation regarding the framework. A portion of the network based IDS have issues in managing network based assaults that include dividing bundles. These deformed bundles cause the IDS to get insecure and in the long run crash. [1].

Input: Take information from the network on user-side.

Processing: Scan the information, scale up or scale down accordingly to feed input in IDS.

Output: Fragmented packets of information.

3.1.2. IDS

Description: The IDS should be equipped for performing information combination and ought to have the option to deal with data from various and appropriated information sources like firewalls, switches and routers. As ongoing identification needs push network-based arrangements, to re-programmable equipment gadgets, the IDS needs to communicate with hardware-based devices.[1].

Input: Take fragmented data packets input in the Intrusion Detection system.

Processing: The system checks whether the input matches with any of the threats of which the system is trained.

Output: The system give outcome of precise threats.

3.1.3. Result

Description: Determine the degree of harm. The anomaly detection systems ought to have a very low bogus alert rate. The extended expansion in the network availability and the traffic, which essentially diminishes the level of generally speaking bogus cautions may be insufficient as their supreme number may proceed to rise[1].

Input: Information on Threat (Outcome from IDS)

Processing: Stores the Threat(type,description and other details) in the database.

Output: Displays the threat details to user. Either of the two scenario will occur -

- a) Informs the user that no threat has been detected.
- b) Informs the user in case of threat.

3.1.4. Response

Description: The IDS is modified to identify danger and respond to dispersed or composed attacks. The IDS should start a reaction to any dubious activity [1].

Input: Information on Threat (Outcome from IDS)

Processing: The decision support tools will be very important for helping the framework managers react to different assaults. Here, the IDS which will be required not solely to distinguish bizarre events, yet additionally taking computerized restorative actions [1].

Output: The arrangement of activities are regularly gathered into dynamic and inactive measures, with dynamic estimates including some computerized intercession with respect to the framework, and aloof measures including revealing IDS discoveries to people, who are then expected to make a move dependent on those reports. [2].

3.1.5. Blockchain Security check

Description: As the IDS plays an effective role in checking the security part of a network, and the IDS becomes the honeypot aim for the attacks to occur. Hence, the IDS should be robust enough for operating in unfriendly and outrageous processing environment, exhibiting fault tolerance to a high level [1].

Input: A set of N Nodes participating in the network [3].

Processing: At some interval, check the functioning of the deep ring model at each and every layer of the neural network model in a block.

Output: Review the inspection. Any one scenario will occur -

- a) Informs the Admin that there is no abnormalities in blockchain.
- b) Informs the Admin about the exception is observed and take necessary action to secure the neural network.

3.1.6. Update

Description: Regular updates should be arranged by the admin, so that it can detect more efficiently. The test cases that the admin provides should be regularly upgraded and replaced with new test cases. The IDS ought to have the option to gain from past encounters and hence improve the identification exactness. It ought to have the option to gain from past encounters of false alarms, guided by the admin so that it will not possibly repeat a false alarm [1].

Input: The dataset in training phase of Neural Network.

Processing: Admin has gathered new advanced dataset.

The system is again trained and tested.

Output: The adapted system could detect

1)The threat more accurately.

2)Newly known threats.

3.2. NON - FUNCTIONAL REQUIREMENTS

3.2.1. Performance Requirements

- The user selects an input compatible to the system.
- Identify the threat.
- The accuracy of the detected threat.
- Real-time execution(detecting and reporting)
- Large quantity and variations of malicious traffics makes their detection very complicated.
- IDS must not interfere with the normal operation or provide unnecessary burden on the network.
- Minimum damage of the network must be ensured and corruption or the loss of information must be avoided.

3.2.2. Security Requirements

- Unauthorized activities shall be continuously monitored and reported.
- Securely set up blockchain network on the block; making the system more robust, along with having high precision.
- Minute change in the neural network model can affect the model's output to a large extent. Leading to incorrect detection and unrequired hassle on User.

3.2.3. Reliability

- Easier to be used for the input preparation, it's operation, and the interpretation of obtained output.
- Reliability of this product depends on the working environment of the system. It gives the most accurate results when used in an environment with less or no noise.

3.2.4. Maintainability

- Different variations of the said product are available for maintenance.
- For the development, it will be easier to add the required code to the existing IDS system, easier to be upgraded to include new features and technologies regularly.

- The maintenance and correction of defects, making changes in the IDS software is easy.

3.2.5. Integrity

- System integrity prevents unauthorized access to the system functions, prevents information loss.

3.2.6. Correctness

- The system adheres to the functional requirements.

3.2.7. Test-ability

- There are features on Admin side to test and find defects. The different modules used in the system can be separately tested as well.

3.2.8. Flexibility

- The system code can be updated from time to time and be made available to the users.

Chapter 4

Project Planning and System Design

4.1. PROJECT SCOPE AND OBJECTIVES

4.1.1. Network is one of the most essential part of our everyday life nowadays, and many primary activities are being performed by using the network. In addition to it, the security of the network also plays a very important role in the administration and monitoring the operation of the mentioned system. The Network Intrusion Detection System (NIDS) is a very crucial and sensitive module to detect the intrusion and defend the network against the malicious traffic that may be flowing through, before the system is affected. The IDS system can successfully extract the required information from the network of the system and quickly predict the reaction that provides real-time protection for the implied protected system. Hence, we are aiming to use an IDS platform based on neural network, to detect DoS attack, web attacks, brute-forcing attacks over the FTP and SSL protocols, etc. Further, the neural network model is thought to be securely set up in a blockchain network for security purpose, thus making the system more robust, along with having high precision.

4.1.2. As this project is a minor project activity, it is led by our project guide.

4.1.3. Stakeholders of this project includes the members who are responsible for the research and development of the model, the project guide, who is responsible for the evaluation and refinement of the model and the end users who will be able to use it.

4.1.4. In the backdrop of the current COVID situation, the development of the model will be done and monitored online.

4.2. ESTABLISH PROJECT INFRASTRUCTURE

4.2.1. With the growth in the need for distribution of workload of the IDS among various computing devices, security related hazards has have continued to be persistently present in various organizations. The absence of a relatively strong security system, integrated with the network can thereby lead to a significant loss of important personal information. Un-monitored networks can in turn lead to an intrusion through various types of malicious attacks on the system, due to which the concerned organizations may incur various heavy losses. As the need to protect the network and detect the possible malware

attacks on the computing system surfaces, the demand for intrusion detection and prevention system is expected to remain very high in various organizations.

4.2.2. Project team organization consists of the programmers who are broadly divided into neural network team and blockchain team.

4.3. ANALYSIS OF PROJECT CHARACTERISTICS

4.3.1. The project mainly aims to create a high precision neural network model that can detect the intrusion in networks, and further securing the model with blockchain, namely implementing the deep ring model.

4.3.2. There are many neural network models proposed for IDS, but our model aims to integrate the prevailing models with blockchain, so as to create a robust and secure neural network model.

4.3.3. We are aiming to follow the waterfall model.

4.4. PROJECT PRODUCTS AND ACTIVITIES

4.4.1. Product flow diagram

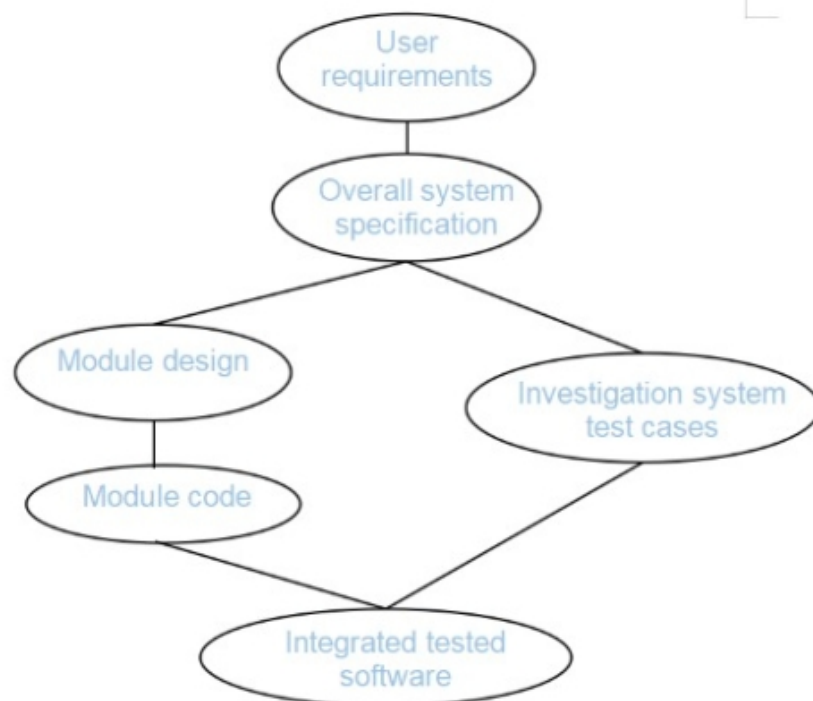


Fig 4.1. Product Flow Diagram

4.4.2. Activity network diagram

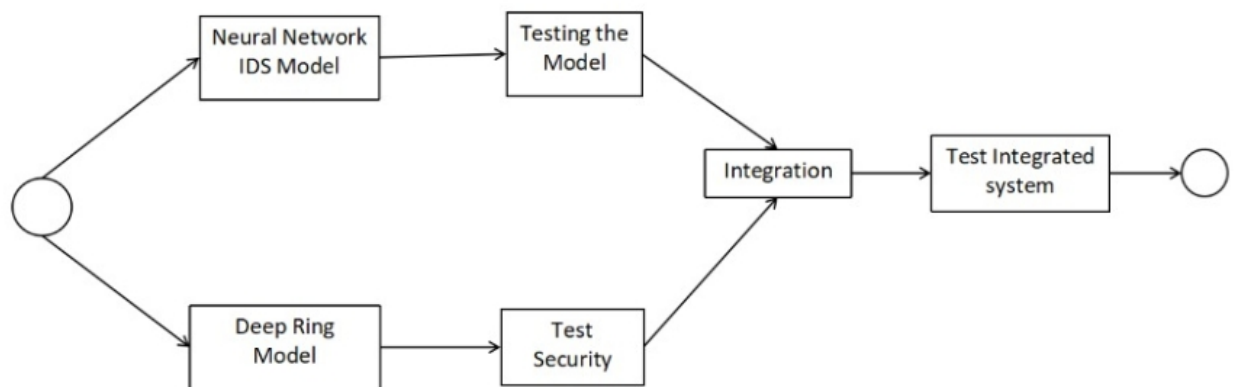


Fig 4.2. Activity Network Diagram

4.5. EFFORT ESTIMATION FOR EACH ACTIVITY

4.5.1. The effort required to complete the project can be calculated as: 15 days for neural network model building, 25 days for deep ring model building, 7 days for combining the two, and 4 days for testing. So, an effort of total 47 days is required to complete the project, as estimated by us, if all the conditions remain favorable.

4.6. IDENTIFY ACTIVITY RISKS

4.6.1. Risks of the project include a new type of intrusion introduced in the networks, leading to re-training the neural network model and adapting the corresponding changes in the deep ring model. The chances of the risks occurring are very less as statistically, it takes much time to device and initiate a new type of attack on a network.

4.7. ALLOCATE RESOURCES AND GANTT CHART

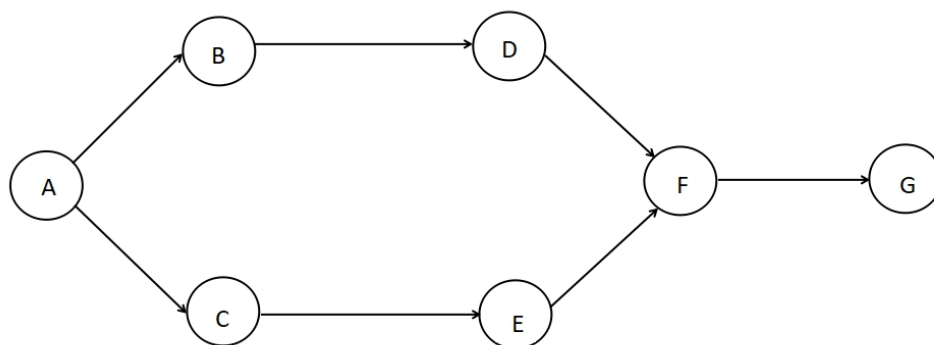


Fig 4.3. Compressed Activity Network Diagram

4.7.1. Pert Table

Activity	Predecessor	Duration (in days)		
		Optimistic	Most likely	Pessimistic
A	--	2	3	4
B	A	13	15	20
C	A	22	25	32
D	B	4	5	7
E	C	4	5	7
F	D,E	6	7	9
G	F	2	4	6

Fig 4.4. Pert Table

4.7.2. Project Network Diagram

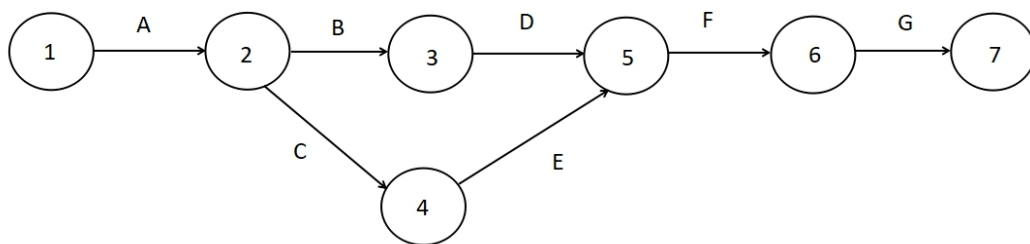


Fig 4.5. Project Network Diagram

4.7.3. Gantt Chart (considering most likely time)

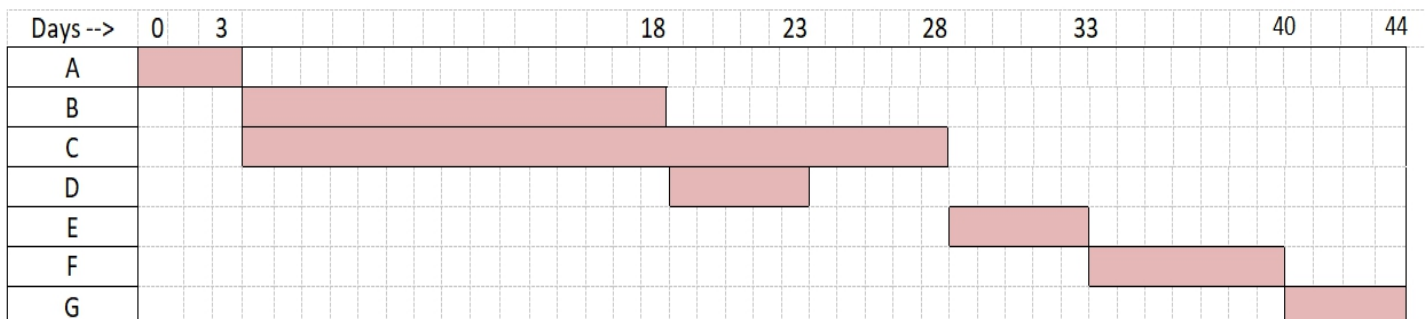


Fig 4.6. Gantt Chart

Chapter 5

Project Implementation

5.1. NEURAL NETWORK

Intrusion detection plans can be grouped into two classifications: abuse and abnormality interruption detection. Oddity implies uncommon action overall that could show an interruption. On the off chance that the noticed movement of a client strays from the normal conduct, an inconsistency is said to happen. Abuse identification can be incredible on those assaults that have been customized in to the location framework. Notwithstanding, it is beyond the realm of imagination to expect to expect every one of the various assaults that could happen, and surprisingly the endeavor is laborious. Some sort of peculiarity identification is at last essential. One issue with peculiarity identification is that it is probably going to raise numerous bogus cautions[4]. Strange yet real use might at times be considered as irregular. The test would be to construct a model of genuine conduct which might acknowledge a unique authentic use. It is tough to assemble such a model for the very explanation that it is difficult to construct a complete abuse recognition framework: it is unimaginable to expect to expect all potential varieties of such conduct[5].

Machine Learning is the detailed examination of computer calculations that improve naturally by experience[6]. Applications span from information mining programs which find general guidelines at huge informational collections, to data sifting frameworks which naturally gain proficiency with clients' inclinations. Rather than measurable strategies, Machine Learning procedures are appropriate to learning designs with no deduced information on what those examples might be. Clustering and Classification are likely the two most famous AI issues. Methods which address both these issues have been implemented in IDSs.

In our project various features are used-

Duration, protocol_type, service, flag, src_bytes, dst_bytes, land, wrong_fragment, urgent, hot, num_failed_logins, logged_in, lnum_compromised, lroot_shell, lsu_attempted, lnum_root, lnum_file_creations, lnum_shells, lnum_access_files, lnum_outbound_cmds, is_host_login, is_guest_login, count, srv_count, serror_rate, srv_error_rate, error_rate, srv_error_rate, same_srv_rate, diff_srv_rate, srv_diff_host_rate, dst_host_count, dst_host_srv_count, dst_host_same_srv_rate, dst_host_diff_srv_rate, dst_host_same_src_port_rate, dst_host_srv_diff_host_rate, dst_host_error_rate, dst_host_srv_error_rate, dst_host_error_rate, dst_host_srv_error_rate, label.

TCP Flags wrong data packet size rate.

Timestamp variance of packet count to keys.

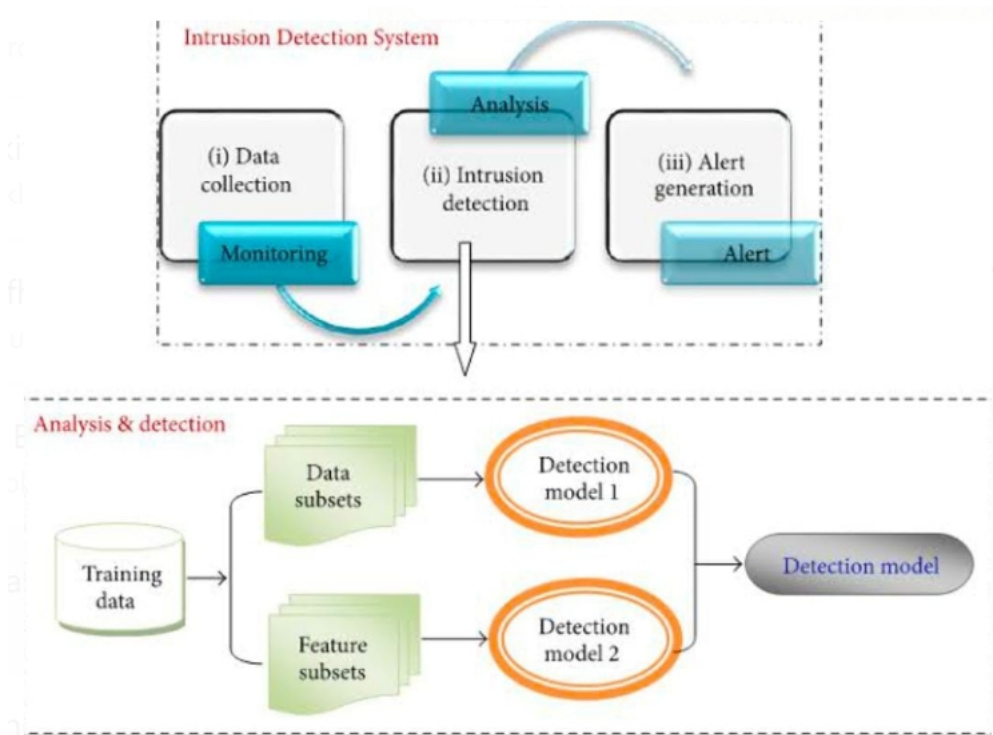


Fig 5.1. Intrusion Detection System using NN

5.1.1. Data Cleaning

Data cleaning is the route toward preparing data for assessment by taking out or changing data that is misguided, divided, unessential, replicated, or improperly coordinated. This data is typically over the top or strong with respect to analyzing data since it may disturb the cooperation or give mistaken results. There are a couple of strategies for cleaning data depending upon how it is taken care of close by the proper reactions being searched for. Data cleaning isn't just about annihilating information to represent new data, however rather sorting out some way to intensify an enlightening assortment's accuracy without basically deleting information. Information cleaning is viewed as a focal segment of the data science stray pieces, as it accepts a huge part in the logical interaction[7].

In this project, first the information is gathered from the site. After the assortment of the new dataset, the dataset is isolated into the preparation and the test set. The preparing set comprises of 80% of the entire dataset and the test set contains 20% of this whole dataset. After that the uniqueness of all the features are taken into consideration. If the unique value become equivalent to one, it says that only a single value exist. So, in this way if we get most of the values similar, then the cleaning occurs. After that the correlation matrix is formed, if we get any value nearer to 1

or 0, we can drop out those features since we can say that those can lead to duplicate values.

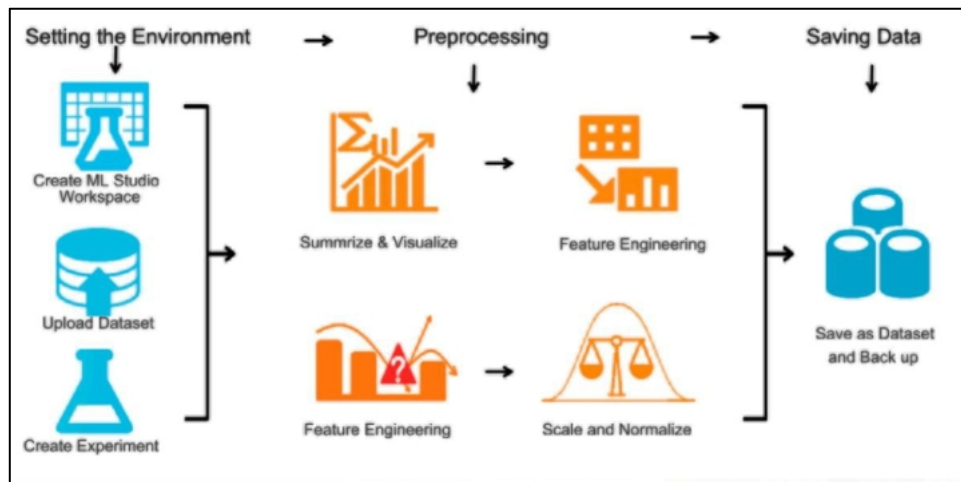


Fig 5.2. Data Cleaning

5.1.2. The Classification Techniques

In the characterization work in AI, the undertaking is to pick each occasion of the dataset and relegate that to a specific class. An order-based IDS endeavors at arranging all the traffic as one or other ordinary or malignant. The objective will be to limit the amount of bogus positives (characterizing typical traffic as noxious) and bogus negatives (order by pernicious traffic as ordinary).

Mapping of the label class id done. Normal to '0' and anomaly to '1'. Since our basic interest is on anomaly as in the intrusion detection our focus is mainly on the attacks ,which comes under anomaly.

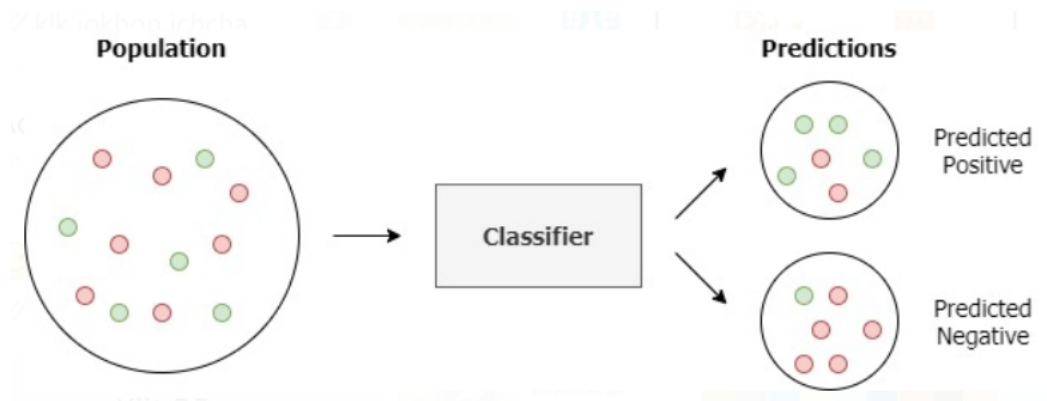


Fig 5.3. Data Classification

5.1.3. Neural Networks

The utilization of neural associations for IDSs has been investigated by various analysts. Neural organizations give an answer for the issue of displaying the clients' conduct in abnormality discovery since they don't need any express client model[8]. Neural organizations for interruption discovery were first acquainted as an option with factual procedures in the IDES interruption recognition master framework to display. Specifically, the common grouping of orders executed by every client is learned. Various activities have utilized neural nets for interruption identification utilizing information from singular hosts, like BSM information[9].

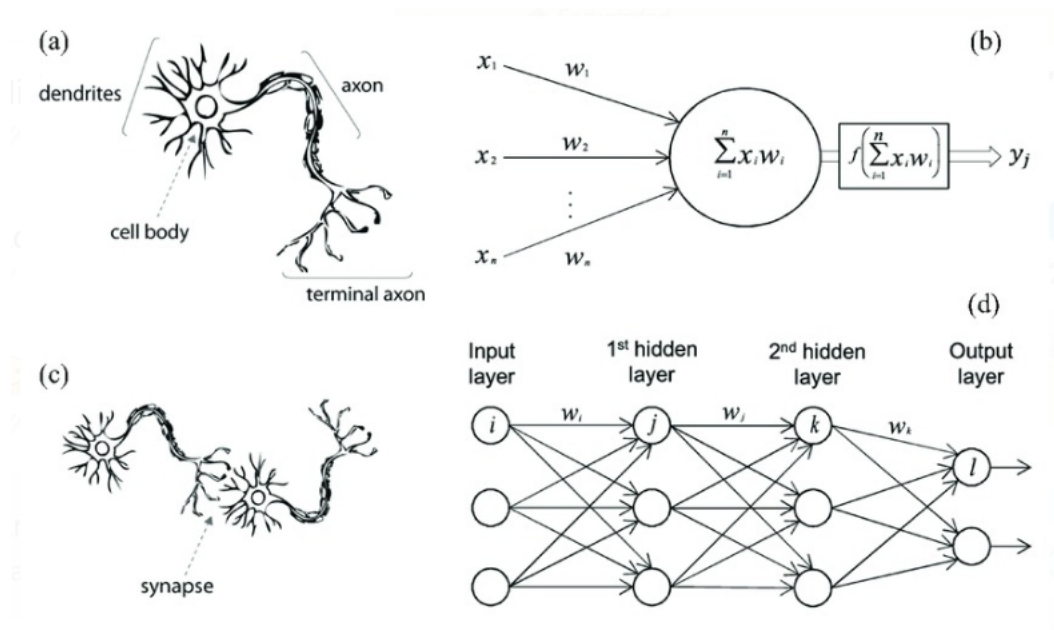


Fig 5.4. Neural Network

5.1.4. Clustering Techniques

Data grouping is a common strategy to measure and examine information, which is then used in many fields, which includes Machine Learning, Artificial Intelligence, Design Acknowledgment, Information Mining, and bio-informatics and picture investigation. Clustering can be defined as arranging comparative articles into various groups, or in other words, the parceling of an informational collection into clusters, in such a way that the information in every cluster share some normal attribute - often nearness according to some characterized distance measure. Machine adapting regularly views information grouping as a type of solo learning. Clustering is valuable in interruption location as malevolent action should group together, isolating itself from non-malignant movement. Bunching gives some critical benefits over the order procedures previously talked about, in

that it doesn't need the utilization of a marked informational index for preparing.

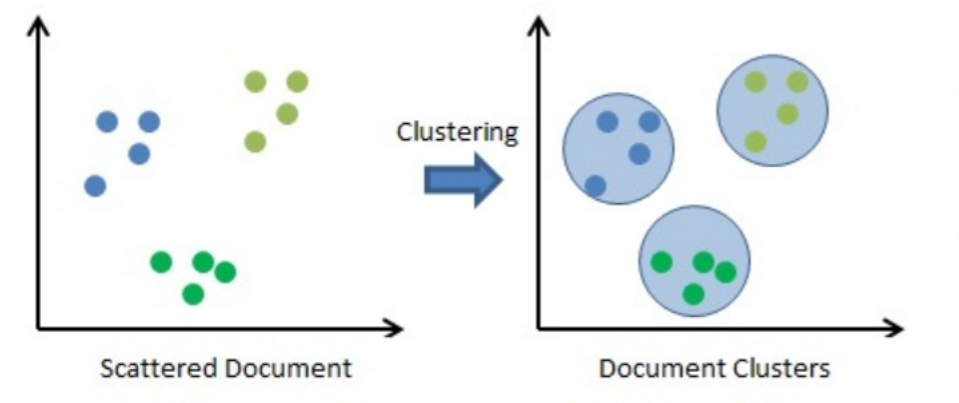


Fig 5.5. Clustering

5.1.5. Feature selection

It is called subset choice or variable choice, is a cycle generally utilized in AI, in which a sub-set of all the features accessible from the information is chosen to be used in a learning calculation. Feature selection plays a vital role either on the grounds that it is computationally in-feasible to utilize every single accessible component, or due to issues of assessment when restricted information tests (however an enormous number of the features) are available. Feature determination from accessible information is fundamental to the adequacy of the utilized strategies. Analysts apply different examination methods to the amassed information, to choose the arrangement of highlights that they think, expands the viability of their information mining techniques. Extracted features can be positioned concerning their commitment.



Fig 5.6. Feature Selection

Here, MLP Classifier and Decision Tree Algorithms are used for Feature selection method. Here the Sequential method is used for achieving the output. A Sequential model is proper for a plain heap of layers where each layer has precisely one info tensor and one yield tensor. Here, each layers are added accordingly to achieve the output of every layers as a whole. Here, we got the best precision and results where the accuracy is 98%. Correlation in Machine Learning is finished in regards to most precision and least number of features picked. Most extreme exactness implies more information ordered accurately. While least number of highlight implies least memory required and diminished calculation complexity.

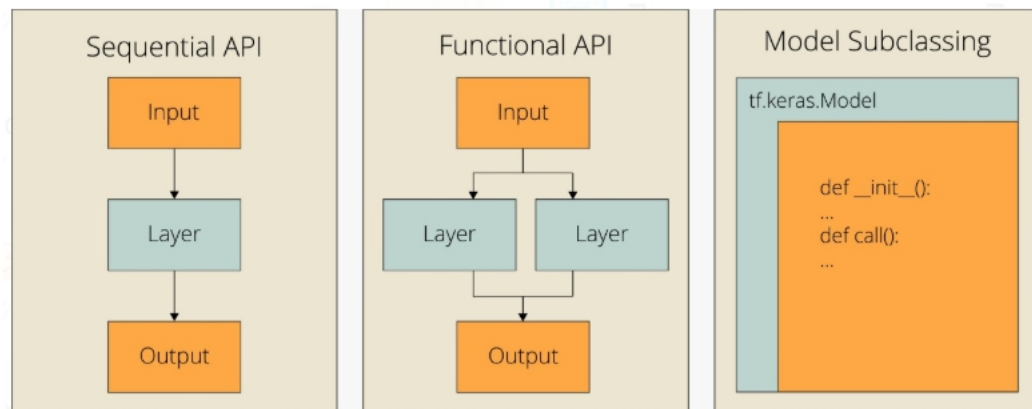


Fig 5.7. Sequential Model

5.2. BLOCKCHAIN IN NEURAL NETWORK

Main security issue that is faced by the neural network is that it is not tamper resistant. One can make a change in one of the layers of the model. This will result in a wrong prediction output of the model, thus drastically reducing its efficiency, and posing a threat to the authenticity of the result. The security of the neural network can be enhanced by blockchain in various ways:

1. To secure the data: here, the data is to be secured by using blockchain, to provide a layer of anonymity and tamper resistance to the data used.
2. Secure the neural network model: implement each layer of the neural network model in a block of a blockchain- the deep ring model

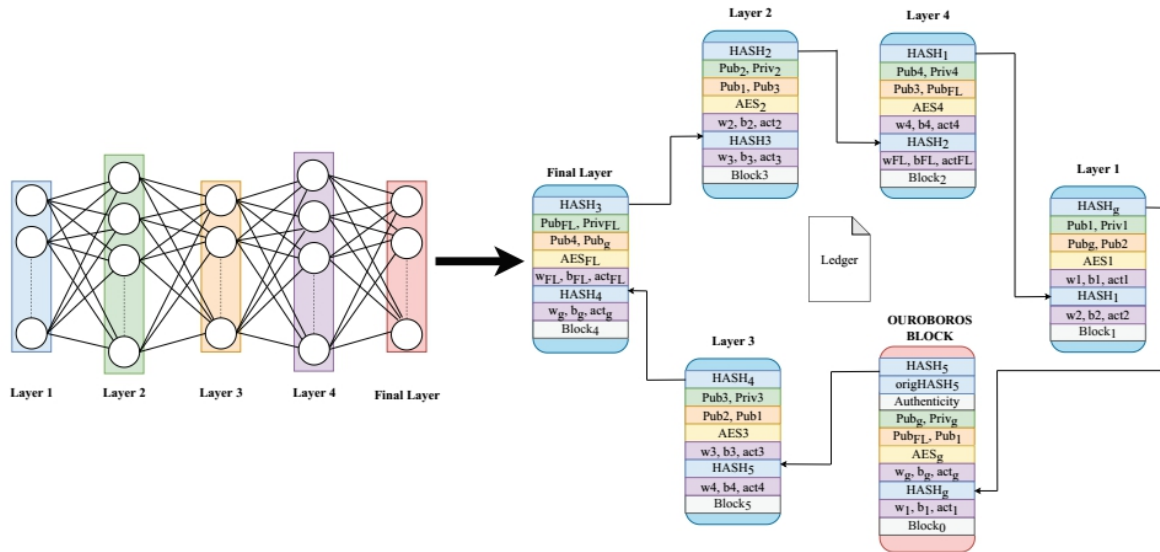


Fig 5.8. transition of a neural network architecture to a deep ring architecture

3. Decentralize the neural network model, thus distributing the computational overhead.

Here, we have only focused on (2), that is, securing the neural network model, each layer is inserted in one block and a ring is made to keep the blocks connected, while being secured.

5.2.1. Blockchain Creation

We have created a Blockchain class which has a constructor that is responsible for creating an empty list `self.chain = []` to store the blockchain (ring). This class is basically responsible for managing the chain of blocks of the deep ring and for adding new blocks to the same with the help of some methods. Each block consists of an index, a time stamp, the hash value of the preceding block, and one layer of the neural network.

At this point, each block must contain the hash value of the previous block within itself. This is extremely important as this provides blockchain's immutability i.e, if an attacker corrupts a previous block present in the chain, then all the blocks following that block will get affected and the blocks will contain incorrect hashes[10].

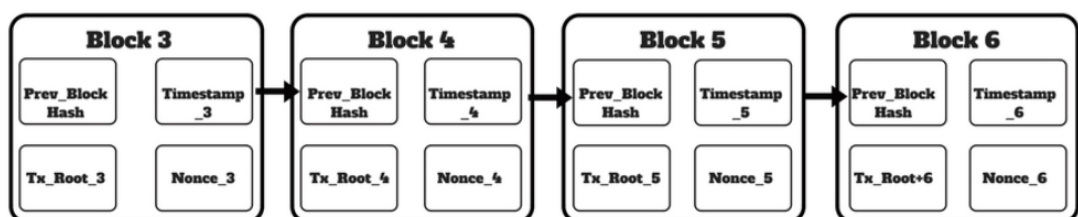


Fig 5.9. Creation of Blocks

5.2.2. Creation of New Block

When the Blockchain is finally instantiated, it will be seeded with the genesis block (block having no predecessors). Besides the creation of the genesis block, in our constructor, we also define the methods `new_block()`, and `hash()`. In the `new_block()` method, we pass the parameters `self` and the previous hash value[11].

In the `hash()` method, we will pass the entire block as a parameter and it will be responsible for creating a SHA- 256 hash value of the block. Here we need to make sure that the dictionary of the block is ordered else we will be having inconsistent hashes. At the end it will return the hash and will be secured using `hashlib`.

5.2.3. Proof of Work Algorithm

This algorithm basically defines how each block is created and linked into the blockchain. It helps to ensure that the block that is attached is indeed a valid block.

Taking an example:

Let us assume that the hash value of an integer ‘x’, multiplied by another integer ‘y’, must end with 0.

Implementing this entire thing in Python, we get $y = 21$. So, the produce hash value is `2053e9873e....5e3600255e860`. Implementing this in our program, we find a number ‘p’, which when hashed with the preceding block’s solution, a hash with four leading 0s is thus produced[12].

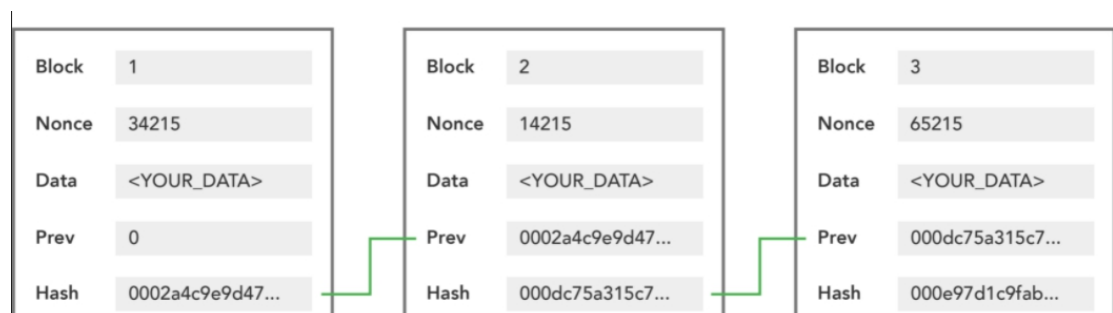


Fig 5.10. Proof of Work

5.2.4. Integration of neural network with blockchain

Once these methods are completed, now, for implementing deep ring, we need to make the number of blocks as constant, the number of blocks will be equal to one more than the number of layers neural network. The

starting block is named the ouroboros block and is the starting and ending point of the ring. Now, each of the blocks will contain each layer of the neural network and they will pass their output to the next layer.

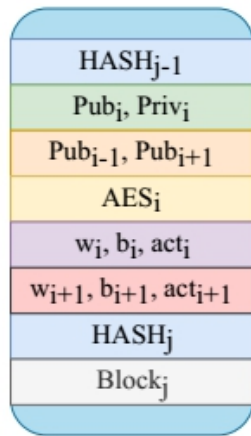


Fig 5.11. layer i of neural network represented as block j of deep ring

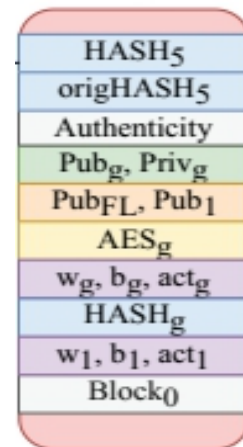


Fig 5.12. ouroboros block: the starting and ending block of the deep

When a set of parameters of the network will be fed to the NN model, it will invoke the ouroboros block. The ouroboros block will check the authenticity of the query and then call the block that contains the first layer of NN. After the block verifies the invoking call, if the call is not authentic, it raises a concern that the model needs to be checked for tampering, if it is authentic, it forwards the control flow of the program to the next block (layer)[13]. In this manner, when all the layers are visited, the last block redirects the flow and the control returns to the ouroboros block. The ouroboros block considers the input and determines that it is from the last block and is authentic, and hence it displays the result that is sent back, that is, whether intrusion is there or not, and terminates the query and waits for the next query. In this manner, the neural network model can be integrated with the blockchain and implemented.

Chapter 6

Screenshots of the Project

	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot	...	dst_host_srv_count
0	0	tcp	http	SF	181	5450	0	0	0	0	...	9
1	0	tcp	http	SF	239	486	0	0	0	0	...	19
2	0	tcp	http	SF	235	1337	0	0	0	0	...	29
3	0	tcp	http	SF	219	1337	0	0	0	0	...	39
4	0	tcp	http	SF	217	2032	0	0	0	0	...	49

5 rows x 42 columns

Fig 6.1. Data

dst_host_same_srv_rate	dst_host_diff_srv_rate	dst_host_same_src_port_rate	dst_host_srv_diff_host_rate	dst_host_serror_rate
1.0	0.0	0.11	0.0	0.0
1.0	0.0	0.05	0.0	0.0
1.0	0.0	0.03	0.0	0.0
1.0	0.0	0.03	0.0	0.0
1.0	0.0	0.02	0.0	0.0

Fig 6.2. Data

dst_host_srv_serror_rate	dst_host_error_rate	dst_host_srv_error_rate	label
0.0	0.0	0.0	normal
0.0	0.0	0.0	normal
0.0	0.0	0.0	normal
0.0	0.0	0.0	normal
0.0	0.0	0.0	normal

Fig 6.3. Data

Chapter 7

Results and Discussions

In this project, we have proposed a tamper-proof NN model, secured with blockchain. The characteristic properties of blockchain such as security is provided to the NN model by employing cryptographic techniques. The layers of the NN model are placed inside blocks of the blockchain. Hiding the network architecture and parameters from an adversary prevents the threat of any white-box adversarial attack. Tampering in any block changes the hash of the current and the subsequent blocks thereby highlighting the performed attack. In this way, the transparency between the blocks and the entire network is increased. However, this enhanced version of security comes at a price of increased computational complexity of performing expensive cryptographic functions and protocols.

In this project, we have trained an ANN model, secured with blockchain, having training accuracy 98% and validation accuracy 98%. In the future, we will extend the approach to make it efficient in terms of computational complexity and defend models against input image perturbation.

Attacking the model using the tampering attack:

If we apply the parameter tampering attack on NN architecture and the Blockchain architecture. For the case of Blockchain, tampering the parameters alerts the ouroboros block and informs the user.

Compromising a block by changing its input leads to the failure of the validation clause which indicates compromise. the proposed DeepRing model is fault free because of multiple authentication blocks such as validation/consensus and Hash functions. Therefore, we do not observe any reduction in performance for the proposed DeepRing. In case of compromising a block by changing its input, DeepRing model will also trigger a violation and therefore, it can inherently provide adversarial attack detection mechanism. In our research, we observe that the proposed DeepRing yields 100% accuracy for detecting perturbations of input to a block.

Chapter 8

Conclusion

In our project, we have created a model of a network intrusion detection system, that uses neural network to detect the possible intrusion in a network, by feeding the network parameters to the model. The accuracy of the model is taken into account and have striven to obtain great accuracy, as the intrusion detection is a sensitive work. In order to further secure the model, so that it can be tamper-proof, the neural network model is modified by using a deep ring model, where, each layer of the neural network is embedded in a block of a blockchain and connected with each other using hash functions. This blockchain model makes sure that the neural network model is not tampered with, and raises a concern if there is any modification in the blocks containing the layers of the neural network. Thus in this project, the security of the neural network is ensured using blockchain and the IDS model is built to provide the user a near accurate prediction of the possible intrusion detected in a network, thus making it possible for the user to take any action regarding the incident response for such activity at an early stage.

Chapter 9

Future Scopes

Currently, the model built only detects the intrusion in the network, but doesn't recognize and display which type of intrusion has occurred. Hence, the project can be scaled to classify and output the type of intrusion that has happened. The knowledge of the type of intrusion that took place can be used to reduce the response time and handle the situation accordingly.

The software can be improved by equipping it with administrator control. For instance, once an intrusion has been detected, it will be reported to the administrator dashboard to decide an action. The administrator will decide what to do with that connection. The neural network in our project has been built using previously collected data that are not up-to-date as time goes by, and is currently not able to learn from experience. Thus, we can incorporate the ability of learning from experience into our model thus making it more up-to-date and robust.

Expanding on this idea, we aim to develop a complete Intrusion Detection and Management Software, secured by Blockchain, an integrated system, which provides one-stop and tamper-proof solution for protecting the network as well as provides options for customizing the IDS to some extent.

Chapter 10

References

1. <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.685.626&rep=rep1&type=pdf>
2. <https://ranger.uta.edu/~dliu/courses/cse6392-ids-spring2007/papers/NIST-IntrusionDetection-2001.pdf>
3. <https://www.colleaga.org/sites/default/files/12-55-blockchain-based-approach-final.pdf>
4. Debar, H., Becker, M., and Siboni, D. (1992). A neural network component for an intrusion detection system. In *Proceedings of the 1992 IEEE Computer Society Symposium on Research in Computer Security and Privacy*, 240-250.
5. Denning, D. E. (1987). An intrusion detection model. *IEEE Transactions on Software Engineering*, SE- 13:222-232.
6. Fox, K. L., Henning, L. T., Leed, J. H., and Simonian, R. (1990). A neural network approach towards intrusion detection. In *Proceedings of the 13th International Computer Security Conference*, 125-134.
7. <https://www.sisense.com/glossary/data-cleaning/>
8. Debar, H., Becker, M., and Siboni, D., "A Neural Network Component for an Intrusion Detection System", IEEE Computer Society Symposium on Research in Security and Privacy, Los Alamitos, CA, pp. 240–250, Oakland, CA, May 1992.
9. Ryan, J., M.-J. Lin, and R. Miikkulainen, "Intrusion detection with neural networks", In M. I. Jordan, M. J. Kearns, and S. A. Solla (Eds.), *Advances in Neural Information Processing Systems*, Volume 10, Cambridge, MA.
10. <https://medium.com/@vanflymen/learn-blockchains-by-building-one-117428612f46>
11. <https://github.com/dvf/blockchain/blob/master/blockchain.py>
12. <https://medium.com/analytics-vidhya/intrusion-detection-system-using-artificial-neural-networks-14afc2112be>
13. <https://imiblockchain.com/blockchain-coding/implementation/>

NETWORK INTRUSION DETECTION SYSTEM USING NEURAL NETWORKS, AIDED BY BLOCKCHAIN

ANANYA DUTTA
1805011

Abstract:

In the rise of the global pandemic, almost every individual has become dependent on the internet to carry out their day-to-day activities. In order to perform these online activities, one needs a safe and secure network to prevent the data from getting stolen by third parties. To ensure safe and secure internet services, the NIDS model is used. The Network Intrusion Detection System (NIDS) is a crucial model to detect and defend against malicious traffics before the system is affected. NIDS not being tamper resistant, we have used blockchain to ensure the security the model.

Individual contribution and findings:

Neural Network:

In this project, first the information is gathered from the site. After the assortment of the new dataset, the dataset is isolated into the preparation and the test set. The preparing set comprises of 80% of the entire dataset and the test set contains 20% of this whole dataset. After that the uniqueness of all the features are taken into consideration. If the unique value become equivalent to one, it says that only a single value exist. So, in this way if we get most of the values similar, then the cleaning occurs. After that the correlation matrix is formed, if we get any value nearer to 1 or 0, we can drop out those features since we can say that those can lead to duplicate values.

Mapping of the label class id done. Normal to '0' and anomaly to '1'. Since our basic interest is on anomaly as in the intrusion detection our focus is mainly on the attacks, which comes under anomaly.

MLP Classifier and Decision Tree Algorithms are used for Feature selection method. Here the Sequential method is used for achieving the output. A Sequential model is proper for a plain heap of layers where each layer has precisely one info tensor and one yield tensor. Here, each layers are added accordingly to achieve the output of every layers as a whole. Here, we got the best precision and results where the accuracy is 98%.

Individual contribution to project report preparation:

Abstract

Chapter 1 : Introduction

1.3 CORE CONCEPTS USED - 1.3.1. Neural Network

Chapter 3 : Software Requirements Specification

3.1. FUNCTIONAL REQUIREMENTS

3.2. NON - FUNCTIONAL REQUIREMENTS

Chapter 4 : Project Planning and System Design

4.1. Project scope and objectives

4.2. Establish project infrastructure

Chapter 5 : Project Implementation - 5.1 Neural Networks -

5.1.1 -Data Cleaning

5.1.2. The Classification Techniques

5.1.3. Neural Networks

5.1.4. Clustering Techniques

5.1.5. Feature selection

Chapter 6 : Screenshots of the Project

Chapter 7 : Conclusion

Chapter 8 : Future Scopes

Chapter 9 : References

Full Signature of Supervisor



Full Signature of the Student

NETWORK INTRUSION DETECTION SYSTEM USING NEURAL NETWORKS, AIDED BY BLOCKCHAIN

ARISMITA BANERJEE
1805021

Abstract:

In the rise of the global pandemic, almost every individual has become dependent on the internet to carry out their day-to-day activities. In order to perform these online activities, one needs a safe and secure network to prevent the data from getting stolen by third parties. To ensure safe and secure internet services, the NIDS model is used. The Network Intrusion Detection System (NIDS) is a crucial model to detect and defend against malicious traffics before the system is affected. NIDS not being tamper resistant, we have used blockchain to ensure the security the model.

Individual contribution and findings:

Blockchain and Neural Network:

In this project, the main motive was to integrate the concept of blockchain with the neural networks, to make a robust model. For implementing blockchain, the already available sources presented only the blockchains that are used for bitcoin mining. But, to bring in live the concept in neural network, the blockchain needed to be modified a little, that is, instead of creating an infinite chain of blocks for each transaction, there would be a finite number of blocks that would be connected to each other in the form of a ring. There would be a common ledger, that would keep the track records of all the 'transactions' or processing of each block. Each block will receive the output from previous block, encrypted by AES, and once received, proof of work would also be conducted to make sure that the system is not tampered with. This concept is derived through numerous research, from a research paper called 'deep ring'.

Once this setup is made, the neural network is to be fit inside the deep ring model this can be done by two methods. One, by hardcoding all the layers, and thus sending the output to the layers following, which is a bit complex, and the other being that calling each 'add' functions in each block and making the 'model' parameter global. Thus opting for the second method, the aim is achieved. In this manner, the built neural network model was incorporated in the deep ring.

Individual contribution to project report preparation:

Chapter 1 : Introduction

1.3 CORE CONCEPTS USED - 1.3.2. Blockchain

Chapter 2 : Requirement Analysis

Chapter 3 : Software Requirements Specification

3.1. FUNCTIONAL REQUIREMENTS

Chapter 4 : Project Planning and System Design

4.1. Project scope and objectives

4.2. Establish project infrastructure

4.3. Analysis of project characteristics

4.4. Project products and activities

4.5. Effort estimation for each activity

4.6. Identify activity risks

4.7. allocate resources and gantt chart

Chapter 5 : Project Implementation - 5.2. Blockchain in neural network -

5.2.1. Blockchain creation

5.2.2. Creation of new blocks

5.2.4. Integration of neural network with blockchain

Chapter 9 : References

A handwritten signature in blue ink that reads "Aismita Banerjee". The signature is written in a cursive style and is underlined with two parallel lines.

Full Signature of Supervisor

Full Signature of the Student

NETWORK INTRUSION DETECTION SYSTEM USING NEURAL NETWORKS, AIDED BY BLOCKCHAIN

ARKAPRABHA SAMANTA
1805106

Abstract:

In the rise of the global pandemic, almost every individual has become dependent on the internet to carry out their day-to-day activities. In order to perform these online activities, one needs a safe and secure network to prevent the data from getting stolen by third parties. To ensure safe and secure internet services, the NIDS model is used. The Network Intrusion Detection System (NIDS) is a crucial model to detect and defend against malicious traffics before the system is affected. NIDS not being tamper resistant, we have used blockchain to ensure the security the model.

Individual Contribution and Findings:

Blockchain in Neural Network:

The main aim was to secure the neural network and is secured using Blockchain. For this, we have created a chain of blocks, where each block contains the hash of the previous block. It is extremely important as it provides blockchain's immutability i.e, if an attacker corrupts a previous block present in the chain, then all the blocks following that block will get affected and the blocks will contain incorrect hashes.

Creation of a new block comprises of two methods, `new_block()` and `hash()`. In the `new_block()` method, we pass two parameters, `self` and the previous hash value. In the `hash()` method, we passed the entire block as a parameter and this method created a SHA- 256 hash value of the block.

The Proof of Work algorithm basically defines how each block is created and linked into the blockchain. It helps to ensure that the block that is attached is indeed a valid block.

Once these methods are completed, now, for implementing deep ring, the number of blocks are made constant and, the number of blocks will be equal to one more than the number of layers neural network. The starting block is named the ouroboros block and is the starting and ending point of the ring. Now, each of the blocks will contain each layer of the neural network and they will pass their output to the next layer. In this way the entire neural network is secured.

Individual contribution to project report preparation:

Abstract

Chapter 1 : Introduction

1.1 PROJECT MOTIVATION

1.2 PROBLEM STATEMENT

Chapter 2 : Requirement Analysis

Chapter 3 : Software Requirements Specification

3.1 FUNCTIONAL REQUIREMENTS

Chapter 4 : Project Planning and System Design

4.1. Project scope and objectives

Chapter 5 :Project Implementation - 5.2. BLOCKCHAIN IN NEURAL NETWORK

5.2.1. Blockchain creation

5.2.2. Creation of new blocks

5.2.3. Proof of Work Algorithm

Chapter 9 : References

Full Signature of Supervisor

Arkaprabha Samanta

Full Signature of the Student

NETWORK INTRUSION DETECTION SYSTEM USING NEURAL NETWORKS, AIDED BY BLOCKCHAIN

SAYANTANI BALA
1805245

Abstract:

In the rise of the global pandemic, almost every individual has become dependent on the internet to carry out their day-to-day activities. In order to perform these online activities, one needs a safe and secure network to prevent the data from getting stolen by third parties. To ensure safe and secure internet services, the NIDS model is used. The Network Intrusion Detection System (NIDS) is a crucial model to detect and defend against malicious traffics before the system is affected. NIDS not being tamper resistant, we have used blockchain to ensure the security the model.

Individual contribution and findings:

Integration:

Possibilities of different components in the Intrusion Detection System. Distinguish the number of functional components required for the successful detection of the Threat. Identify The actions to be taken to detect threat or Intrusion via networks. The large scale information should be fragmented to fit in the IDS. The large scale information should be fragmented to fit in the IDS . If you acknowledge the threat then proceed the required steps to taken in response (i) by the System itself (ii) by the User after the result of the threat is declared to them.

Individual contribution to project report preparation:

Chapter 3 : Software Requirements Specification

3.1 FUNCTIONAL REQUIREMENTS

- 3.1.1 Import data from network
- 3.1.2 IDS
- 3.1.3 Result
- 3.1.4 Response
- 3.1.5 Block chain Security check
- 3.1.6 Update

3.2 NON-FUNCTIONAL REQUIREMENTS

3.2.1 Performance Requirements

3.2.2 Security Requirements

3.2.3 Reliability

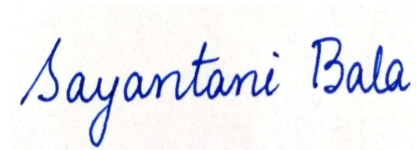
3.2.4 Maintainability

3.2.5 Integrity

3.2.6 Correctness

3.2.7 Test-ability

3.2.8 Flexibility



Full Signature of Supervisor

Full Signature of the Student

TURNITIN PLAGIARISM REPORT

NETWORK INTRUSION DETECTION SYSTEM USING NEURAL
NETWORKS, AIDED BY BLOCKCHAIN

9%

SIMILARITY INDEX

7%

INTERNET SOURCES

2%

PUBLICATIONS

4%

STUDENT PAPERS

PRIMARY SOURCES**1****cps-vo.org**

Internet Source

3%**2****Submitted to The British College**

Student Paper

1%**3**

**Mihai-Gabriel Ionita, Victor-Valeriu Patriciu.
"Cyber Incident Response Aided by Neural
Networks and Visual Analytics", 2015 20th
International Conference on Control Systems
and Computer Science, 2015**

Publication

1%**4****medium.com**

Internet Source

1%**5****ukprwire.com**

Internet Source

<1%**6**

**Daniel van Flymen. "Chapter 4 Proof of Work",
Springer Science and Business Media LLC,
2020**

Publication

<1%**7****Submitted to Informatics Education Limited**

Student Paper

<1%

8	Submitted to University of Brighton Student Paper	<1 %
9	Submitted to University of Queensland Student Paper	<1 %
10	senior.ceng.metu.edu.tr Internet Source	<1 %
11	hdl.handle.net Internet Source	<1 %
12	mafiadoc.com Internet Source	<1 %
13	trap.ncirl.ie Internet Source	<1 %

Exclude quotes Off

Exclude matches Off

Exclude bibliography Off