Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 218 (2023) 1405–1413

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Machine Learning and Data Engineering

Botnet Detection Using Artificial Intelligence

R. Sri Skandha Moorthy[a] , N. Nathiya[b]

[a]School of Advanced Sciences, Division of Mathematics, Vellore Institute of Technology Chennai,Chennai,600 127,India.
[b]School of Advanced Sciences, Division of Mathematics,Vellore Institute of Technology Chennai, Chennai,600 127,India.

Abstract

In the year 2021 more than 80 million data were breached by cyber attackers. Most of these attacks were executed as a type of ransomware attack and cyber-attack. There are different methods performed by attackers to target individual user, but to breach an organization, they use Botnet forces. A botnet (robot network), is malware infected network that is controlled by a single attacker called bot-herder. Cyber - attacker attacks many users with their malware script by different mediums like emails, spams and takes command and control (CC) of the victim device. Using these devices attacker forms a network and performs large attack like distributed denial-of-service (DDoS), attack on an organization to breach data. The complex analysis for cybersecurity analyst is to find bot-herder and the infected network. The structures of the botnets are become very different now days. Botnets can be found with its peer-peer (p2p) structure, signature detection, behavioral analysis, domain names (DNS) and network traffic. To make this different feature analysis easier, the usage of artificial intelligence (AI) is introduced in cyber security. Data from previous attacks are collected, trained using a model which helps in prediction of future attacks. Detection of DNS of core CC servers using AI are widely used nowadays. This research mainly focuses on detection of botnet malware from the net flows of malware packets. The botnet attack data set are collected from resources like Czech - university (CTU-13), Information security and object technology (ISOT). The bi-directional net flow data and the calculation of the network packets are used. Using algorithms like support vector machine (SVM), decision tree and multi-layer perceptron, the data set is trained and tested. After the training and testing, the decision tree model has good accuracy and performance metrics of 92%. This model is considered as a best fit model and helps in detection the of malware packets. The research's objective is to build an alerting system which reports once a malware packet is intruded into a network.

Keywords: Botnets; Artificial Intelligence; Deep Learning; Cyber-attacks; Command Control; Cyber Security; Bi- directional Flows; Packets

1. Introduction

Internet boom had made the people to access their needs very easily, it has made a tremendous leap in history, culture, technology and future precautions to protect the human race. As the coin has two sides, internet also has a bad side which helps people to know about other people and target them to loot money or information from them. Many different ideas were used by people to attack a person or organization to gain money by threatening them. One of the famous way of attack is ransomware attack that is attacking an individual and locking their information until

10.1016/j.procs.2023.01.119

they settle the ransom asked by an attacker. To attack large organizations, they use botnets. Botnets are the robot network where the attacker takes command and control of the device. The attacker first attacks an individual user and injects his malware to form them into a solider to attack an organization. In 2016 Mirai botnets attacked many important firms and government centers in Europe, and America. Attackers locked their information and asked for ransom to release their information [1]. The main reason for these kinds of individual attacks is less awareness for the users about the attacks. Especially spam based mails like price winning or offer discounts are the honey traps created by attackers to attack individual users. Generally, mass attacks are done to gain ransom from the victims. The identification of botnets has become a hectic job nowadays. Cyber security analysts investigate using many methods to find the botnet, to control receiving packets from the attackers. The structure of the botnet is also changing rapidly like local network connections, analyst will find the botnet based on the p2p structure, network traffics, behavioral analysis, signature detection and DNS. In this paper, the botnet is to be detected based on the network traffic and flows. The existing botnet detection system are mostly based on the intrusion detection and finding CC server using the NLP techniques. The main lag of this existing model is nowadays the name CC server can be changed frequently using some name changing algorithms and the signatures of the intrusion. In this research, the detection botnets is based on the network traffic of the networks. The bi-directional net flows during the botnet attack is used. The bi-directional net flows packets of the data is trained using different AI supervised models like decision tree, SVM, random forest and neural network models like MLP. The data set is trained with these different models and model with the highest accuracy is made into the real time detection system. The proposed model for the detection model is using the bi-directional net flows in the traffic. The packet is labeled either true or false based on the protocols and other features of the packet by using the net flow. This model works instantly to detect whether the receiving packet is a botnet packet or not. The paper consists of literature survey which analyze about previous works done, the methodology consists of collection of data and process in building the model. The results and conclusion elaborate the obtained results and summary of the research's objective & its future work.

2. Literature Survey

The detection of botnet is mainly based on the p2p structure domain name of server, or the signature of the network traffic flows. There are many proposed systems which help in detection but there are some limitations and exceptions in classifying the packets. Shamsul Haq et al. [2] had proposed a model in which accuracy of K means clustering and j48 classification with data set. The total sum of these data set is always equal to the actual data set to compute accuracy. The low positive rate and high rate are based on the classification algorithms instances. Peng J et al. [3] had proposed model which detected the CC domain names. It is mainly based on the analysis of the botnet behavior which focuses on the CC server and domain names. The n-grams feature, name entropy, length features of the collected domain names are trained to confirm the suspected domain name is a botnet server domain name or not. Hasan Alkatani et al. [4] proposed a detection model for IOT devices. The network traffic data from the nine commercial IOT devices infected by Mirai and Bashlite botnets are collected. The model consists of the artificial algorithms like CNN-LSTM to detect the botnet against IOT devices. Afan Alharbi et al. [5] designed a detection model based on the graph –based machine learning approach. Botnets data from the IoT data set and CTU-13 data set, the effective graph-based system is created. Machine learning algorithms like Navies Bayes, decision tree are used as the evaluation measure technique. Network flow data from the data set are ingested and normalize all features for each subset of the data sets. Astha Bhargava et al. [6] designed a AI based framework system to detect the botnet in the sites. In this the P2p signature, IRC signature and CC signature are the main features of the model. This detection system called as extensive botnet detection system, which is an automated process of detection framework capable of generalize the information of malicious community site visitors. Manmeet Singhy et al. [7] presented a survey based on the botnet detection system. Over analysis of 200 papers were done and a compilation about the techniques its flaws were discussed. The paper mainly focuses on the DNS protocol detection system which has many approached like anomaly based, flux based, DGA based. Domain name changing algorithms have resulted in detection of botnet in difficult. Qunyh Chi Nguyen et al. [8] has proposed a system of detection model based on machine learning algorithms using the domain name service query data. Machine algorithms like KNN, decision tree, random forest is used. The detection finds, DNS queries sent to IP address of CC server which is generated automatically is a botnet malware or not using DNS. Schiller et al. [9] presented a study on types of alternative botnets that are advanced and difficult in detection system. Domain names, multihoming, alternative control channels, web-based CC server etc. Mouhamed Fahad Umer et al. [10] proposed a flow based intrusion detection system. The system uses the IP flow records which has number of applications like traffic analysis, network visibility etc. With the IP flow records the machine learning algorithms like SVM, decision tree is used to find the intrusion packets in the network. Vaibhav Nivargi et al. [11] had proposed a detection model based on the machine learning algorithms. The botnet binaries and begin binaries with label are used in the classification system. Also the labeled IRC logs were used in the detection system. Ihsan Ali et al. [12] proposed a study survey on IOT based botnet attack. From 5465 studies the important 34 research study were analyzed completely. The major threaded quoted in the study is the hardware security for the IOT devices and most of the detection system failed in detecting the botnet attacks.

3. Methodology

The statistics by web tribunal says that there are more than 3, 00000 malware created daily, and attacks were executed in various regions of the world. An act of intervention into other individuals' pc or laptop devices without their permission to exploit their data or for some ransom is called as hacking. Hacking consists of a lot of techniques in exploiting the internet users like malware attack, phishing attacks, key logger attacks, DoS attacks, sql injection etc.,. These are essential hacking techniques done to exploiting an individual user, website, or organizations. Using these kinds of attack a botnet network is formed.

3.1. Botnet

Botnet is robot network i.e., collection of networks connected with each other that is handled by single head. The head who controls all this connected network is called as bot-herder. If an attacker decides to attack an organization like google or amazon, he needs a lot of force to break the firewall security of those web servers. Basically, attacker forms an army with his malware network, first he attacks individual user by hacking methods. By phishing attacks or malware attacks attacker takes the command and control of the devices. Attacker takes control of all the devices attacked and form it as a network. With help of this network attacker can perform denial of service attack or brute force attack which will either disrupt the traffic of the network or crash the server of the victim's network [13].

Most recent attack of botnets that took place in last decade are mariposa botnet, methbot, mirai botnet and 3ve botnet.

3.2. C&C Server

Web servers are basically a machine which allows to host the network machines which is accessed from various locations thorough the internet. The web servers are usually hosted by the network admin. If an attacker decides to perform a botnet attack, the network of the botnet should be hosted by server. With the help of the server attacker can access, send, receives packets and information. It is called as Command Control server (CC). This CC server can be physical server systems or it can be the virtual servers which can be a handled remotely [14].

3.3. Control of Botnets

The botnets have two types of most common models, Centralized client server model and Decentralized peer-peer model.

3.3.1. Centralized Client Server Model

Initially the botnet architectures were like the whole botnet is controlled by the bot-herder CC server. There will be no connection between individual bots. This model of communication became the out dated one because of the disadvantage of the peer-peer connections between the channels of the network. For say if a security analyst wants to stop the attack he can perform a SQL injection as a firewall against a centralized herder. It is client and server model that here the server acts as the server and the bots will act as the clients of the server. The most common communication channels of the botnet are IRC and the HTTP protocol.

IRC is Internet Relay Chat botnet, this is most initially type of botnet hosting. The botnets are remotely controlled by a IRC server which is already configured to access the bot network. This server waits for the bot-herder commands, once it receives commands it will start to initiate the attack strategy [15]. HTTP botnet is botnet based on the web. Here the bot-herder uses HTTP protocol to send and receive the commands from the bot-herder. HTTP protocol basically works as request response protocol that is only if the request is received then it will take response to the particular one. The bots will occasionally visits the server to get the updates from it, which is based on the commands of the herder. This protocol allows herder to mask the activities of request-response of servers.

3.3.2. Decentralized Peer-Peer model

Decentralized system is basically the bot network is connected to each other. For E.g.: The bot 1 machine is connected to the bot 2 machine which can also shares the information between them. As the botnets are connected one another if there is a counter attack takes place can also be stopped by the herder. It also has a dis advantage that if one bot is attacked it can be possibly attacks the other bots [15].

Peer-Peer control technique of network that is each node will be act as a server itself so there is no individual or centralized server to handle the network. Each node connected in the network shares the workload equally so that there will be no act of mal-function or crash due to overload.

3.4. Network based Detection

Network based detection are based on the network traffic flow of botnets. There are two types of network based detection system Active monitoring and passive monitoring. •

- Active Monitoring: Active monitoring is the activity of injecting the packets into the network or the server as a test. It will produce the extra traffics on the network, with the help of the injected packet it can be found out whether the session is managed by botnets or not. As most of the botnets works on command and response patterns this method of detection will be very helpful. This strategy can be applied to the real world IRC botnets [16].
- Passive Monitoring: Passive monitoring is a technique of identifying the suspicious traffics in the network. There will be no packets injected to the check the session of the CC server. The data traffic will be collected and analyzed later on.

This research mainly focuses on the network based detection system especially the Passive monitoring method. Using this method that traffic data is collected and bi-directional net flows of the data are obtained to design a detection system. This detection system concentrates on the network based botnet detection system.

3.5. Network Flows and Packets

The network flows, traffic background that flows through the network or between the servers will be in the form of packets. Packet is a small size of data that is sent through the Transmission control protocol or Internet protocol over the networks. Usually the size of the packets will be around 1.5 kilobytes in the Ethernet port and it will 64 kilobytes when it is payload data transmission.

Packet is routed between the origin and destination machines or through the switched or servers. These request is communicated over the network and the large messages will broke down into small bytes. These small bytes will sent through the Transmission control protocol, which makes the transfer of data efficient. A packet holds 1,000 to 1,500 bytes of information during the transfer. Each packet is numbered and has the IP address of the destination, the packets may be sent individual route but at the end of destination it assembles into single information. The capture of these network flows are called as Network packet capture.

3.6. DATA COLLECTION

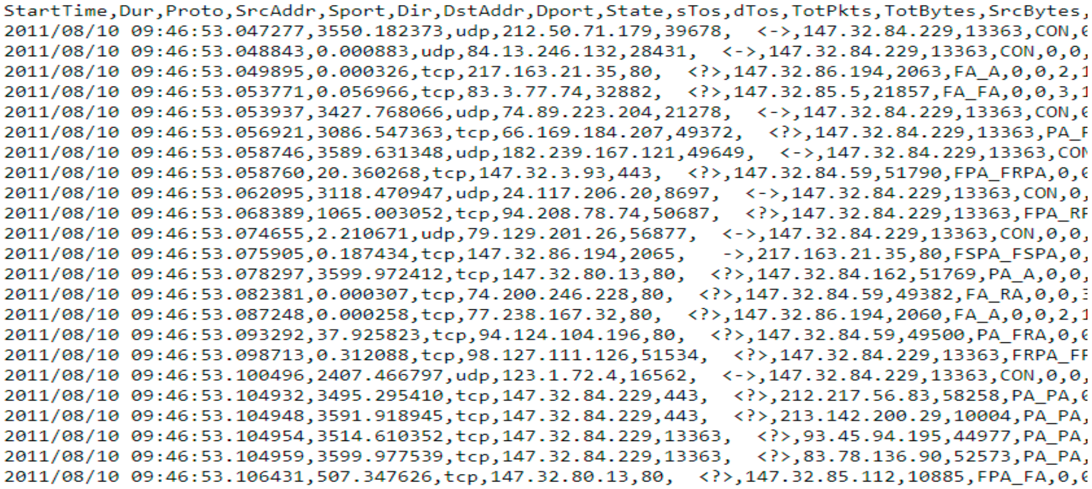
In this research, the detection system designed based on the network flow of data. The network flow data is collected from the stratosphere lab an analysis lab website. The CTU-13 dataset of botnet traffic is a labelled dataset which consists of Normal traffic flows, network packets and the Background traffic. The data was captured by the CTU University, Czech republic in the year of 2011. The dataset has large collection of real botnet traffic combined with the normal, background traffic. The dataset consists of 13 dataset which has data of thirteen different botnet scenarios. Each one scenario was filled with specific malware attack with several protocols and actions. It has 32,000 data instances with 14 features.

Each of the scenario was captured as pcap file that has packets of different protocols. These pcap files can be used extract the information from the packets like bi-directional net flows, web logs, hexa dumps etc. In this detection the main feature used is the bi-directional net flows, it has an advantage rather than single directional net flow. It can easily separate the difference between the client and server, it has more information than one directional and most important is it has labeled details.

The bi-directional net flows from the data set are collected using the scapy and from analysis of the bi – directional net flows files in the Wire shark it is known that all the transfer protocol based on UDP/TCP protocol only.

3.6.1. Parameters of Dataset

The bi –directional net flow data consists of several features. From these features only some features are to be extracted to design a botnet detection system.



```
StartTime,Dur,Proto,SrcAddr,Sport,Dir,DstAddr,Dport,State,sTos,dTos,TotPkts,TotBytes,SrcBytes,
2011/08/10 09:46:53.047277,3550.182373,udp,212.50.71.179,39678,   <->,147.32.84.229,13363,CON,0
2011/08/10 09:46:53.048843,0.000883,udp,84.13.246.132,28431,   <->,147.32.84.229,13363,CON,0,0,
2011/08/10 09:46:53.049895,0.000326,tcp,217.163.21.35,80,   <?>,147.32.86.194,2063,FA_A,0,0,2,1
2011/08/10 09:46:53.053771,0.056966,tcp,83.3.77.74,32882,   <?>,147.32.85.5,21857,FA_FA,0,0,3,1
2011/08/10 09:46:53.053937,3427.768066,udp,74.89.223.204,21278,   <->,147.32.84.229,13363,CON,0
2011/08/10 09:46:53.056921,3086.547363,tcp,66.169.184.207,49372,   <?>,147.32.84.229,13363,PA_F
2011/08/10 09:46:53.058746,3589.631348,udp,182.239.167.121,49649,   <->,147.32.84.229,13363,CON
2011/08/10 09:46:53.058760,20.360268,tcp,147.32.3.93,443,   <?>,147.32.84.59,51790,FPA_FRPA,0,0
2011/08/10 09:46:53.062095,3118.470947,udp,24.117.206.20,8697,   <->,147.32.84.229,13363,CON,0,
2011/08/10 09:46:53.068389,1065.003052,tcp,94.208.78.74,50687,   <?>,147.32.84.229,13363,FPA_RF
2011/08/10 09:46:53.074655,2.210671,udp,79.129.201.26,56877,   <->,147.32.84.229,13363,CON,0,0,
2011/08/10 09:46:53.075905,0.187434,tcp,147.32.86.194,2065,    ->,217.163.21.35,80,FSPA_FSPA,0,
2011/08/10 09:46:53.078297,3599.972412,tcp,147.32.80.13,80,   <?>,147.32.84.162,51769,PA_A,0,0,
2011/08/10 09:46:53.082381,0.000307,tcp,74.200.246.228,80,   <?>,147.32.84.59,49382,FA_RA,0,0,3
2011/08/10 09:46:53.087248,0.000258,tcp,77.238.167.32,80,   <?>,147.32.86.194,2060,FA_A,0,0,2,1
2011/08/10 09:46:53.093292,37.925823,tcp,94.124.104.196,80,   <?>,147.32.84.59,49500,PA_FRA,0,0
2011/08/10 09:46:53.098713,0.312088,tcp,98.127.111.126,51534,   <?>,147.32.84.229,13363,FRPA_FF
2011/08/10 09:46:53.100496,2407.466797,udp,123.1.72.4,16562,   <->,147.32.84.229,13363,CON,0,0,
2011/08/10 09:46:53.104932,3495.295410,tcp,147.32.84.229,443,   <?>,212.217.56.83,58258,PA_PA,0
2011/08/10 09:46:53.104948,3591.918945,tcp,147.32.84.229,443,   <?>,213.142.200.29,10004,PA_PA,
2011/08/10 09:46:53.104954,3514.610352,tcp,147.32.84.229,13363,   <?>,93.45.94.195,44977,PA_PA,
2011/08/10 09:46:53.104959,3599.977539,tcp,147.32.84.229,13363,   <?>,83.78.136.90,52573,PA_PA,
2011/08/10 09:46:53.106431,507.347626,tcp,147.32.80.13,80,   <?>,147.32.85.112,10885,FPA_FA,0,0
```

Fig 1. Parameters of the dataset

In the Fig.1, the bi directional net flow of the network traffic with features.

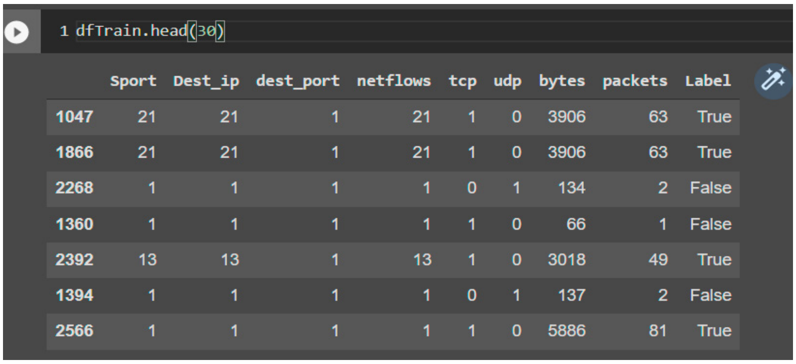
- Start time: Start time is the time duration when the transmission of packet transferred started. It consists of date in the year/month/date format and time stamp of it. For eg: the duration of the first packet is 2011/08/10 and time stamp is 09:46:53.04227
- Duration: Duration feature is about the duration the source and destination machine were connected together through the network.
- Protocol: Protocol is in which protocol the source and destination are connected to each other, whether the protocol is TCP, UDP or IP
- Scr Addr: Source address is the IP address of the source machine from where the packet is transmitted.

- Scr Bytes: Number of bytes carried by source machine.
- Sport: Source port is the feature which has information about from which port that packet has been transmitted out. For example: If the packet is out from port no 256 it should be either TCP/UDP protocol transmission.
- Dst Addr: Destination address is the IP address of the destination port, i.e. where the transmitted packet is reached.
- Dport: Dport is the destination port. Port number of the destination machine
- Tot Pkts: Total packets is the total number of packets send and received duration of the connection between source and destination machine.
- STos: Number of net flows from source machine.
- DTos : Number of net flows received in destination machine
- Dir: Directory of the transmission details.

These are the features of the bi-directional flows.

3.6.2. Feature Extraction and Pre process

Feature Extraction is a method of extracting important features from the data set, which will be helpful in building the model. Some of the features of the data set like start time, duration are neglected. STos and DTos features are summed together into a single feature called net flows. The Total number of packets, Protocols, Total number of bytes, port address of the source and destination machine are converted into a new data frame. The new feature "Label" is added based on the total number of bytes and packets. An average transmission of Ethernet can transfer only up to 1.5 kilobytes of data. If the total number of bytes is greater than the 1.5 kilobytes it is considered as the malware packets. In some cases the Directory feature has the details in which some of the server name are registered as botnet. If it has greater than 1.5 kilobytes of data or in the directory name it is registered as "Botnet" it is considered as malware packet and labeled as "True" else "False". This is the image of Fig.2, the new data set after the feature extraction is done.



```
dfTrain.head(30)
```

	Sport	Dest_ip	dest_port	netflows	tcp	udp	bytes	packets	Label
1047	21	21	1	21	1	0	3906	63	True
1866	21	21	1	21	1	0	3906	63	True
2268	1	1	1	1	0	1	134	2	False
1360	1	1	1	1	1	0	66	1	False
2392	13	13	1	13	1	0	3018	49	True
1394	1	1	1	1	0	1	137	2	False
2566	1	1	1	1	1	0	5886	81	True

Fig 2. Labelled Dataset

3.7. Process

Artificial Intelligence consists of machine learning and deep learning models as inter disciplinary of it. So these ML and DL models are used to construct the detection system models considered for this research are decision tree, SVM, KNN, XG boost, random forest and extra tree classifier. In the deep learning models multi-layer perceptron model are used.

Using the label encoder the “Label” feature was converted from categorical data to numerical data i.e. true or false to 1 and 0. There are around 32,000 data available in the data set. The features like source port, destination IP address, destination port number, number of net flows, tcp, udp and total number of packets are considered as the independent feature. Label is separated as the dependent feature.

The bi-directional net flow data are separated as train data and test data using the 'train test split' function from the sklearn package. The data is separated into two parts in the ratio of 70:30 i.e. 70 percent of the data will be training data and 30 percent of data will be test data.

The machine learning algorithms are used as the ensemble technique. The data set is trained on all the machine learning algorithms mentioned above. The accuracy score and test score of the data set are also obtained. After the machine learning algorithm, the Deep learning models like RNN and Multi-layer perceptron are used to train the training data.

Using the keras API the sequential model is build. Three dense layer, with the activation function,'re Lu' was mentioned. The final dense layer has the activation function 'sigmoid' as it is binary classification method. The model is compiled with the loss function of "binary cross entropy", optimizer as "adam" and "accuracy" metrics of the model are compiled. The obtained model is trained with train data of batch size 100, verbose 1 and epoch size of 100, i.e. the model will run 100 epochs times to reach the good accuracy. The live loss plot of the model is obtained to know about the accuracy and loss of the model.

The results to be obtained for model are analyzed based on the accuracy of the model. The model with high accuracy metric is considered as the best model and will be used as the detection system.

4. Results and Discussions

4.1. Results

The model is built using the classification algorithms, the train and test data are fitted and the expected results are obtained from it. The obtained results are to be examined now using the classification metrics. The common metrics used are Accuracy, Confusion matrix, F-Score and Precision, recall. The confusion matrix with normalization and without normalization helped to define the accuracy of the model. Each classification model is built and fitted with test values.

From the classification models Decision tree has obtained the highest accuracy of **0.9221**. So this model is considered as the model to be used in the detection algorithm.

4.1.1. Decision Tree

The Performance metrics of the Decision Tree Model

- Precision: 0.9221303880625214
- Recall: 0.922120520093111
- F Score: 0.922123969238367
- Decision Tree Score: 0.9221242788234181

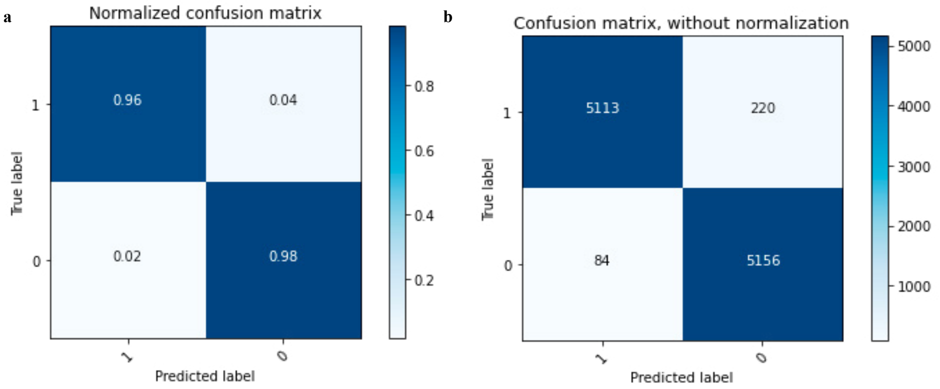


Fig.3. (a) Normalized Confusion matrix of Decision Tree; (b) without normalized confusion matrix of Decision Tree

4.2. Disscussions

The bi – directional net flow data is trained and tested with machine learning and deep learning models. The model with high accuracy and other confusion metrics are considered to conclude as the best model to fit in to the detection model. The below table consists of the accuracy of each model.

Table1. Metrics of the Classification model

Classification Algorithm	Accuracy Score	True Positive	True Negative	False Positive	False Negative
KNN	0.82383713231816	4987	5098	346	142
XG Boost	0.85455783599735	5107	5053	226	187
Decision Tree	0.9221242788234	5113	5156	220	84
Random Forest	0.8803962924430	5101	5056	232	184
Extra Tree Classifier	0.892658658848	4673	5118	660	122
SVM	0.51835997351745	5282	114	52	5126
MLP	0.8382	5087	5098	302	136

Table 4.1. Accuracy Score

From this Table 4.1 it is clear that Decision tree has the highest accuracy rate of **0.92212** and it has very less value of false positive and false negative. From Fig.3, it has true positive i.e. number of correctly predicted values as 5113 and true negative value i.e. number of correctly predicted false values 5156. So this model is to be considered as the best model for the detection system. With the process of all other model Decision tree model performs well classifying the malware packets.

5. Summary and Future Work

Botnet is a malicious infected network that also infects other system or devices connected in the same network. This botnet is hosted in server called CC server from where the command to the infected computers are sent, the attacker who control this botnet is called as bot-herder. This botnets has two types of control system centralized and de-centralized system. There are many types of botnets like DDoS attack botnets, custom built botnets, spam based botnets etc. These botnets can be detected in many ways like DGA based detection, anomaly based detection, network based detection etc. In this research botnet detection system is built based on the de- centralized peer-peer control system server. The detection system is based on the network detection process which has passive monitoring of packets. In this network flows of the packets are collected from the CTU-13 data set which has 13 different scenarios of real time botnet attacks. This packets consists of single directional flows of data which is later converted into bi directional net flows traffic with help of scapy. This data set is separated into train, test data and trained tested using different machine algorithms, deep learning algorithms. After the training and test of models, the decision tree classification algorithm results with highest accuracy of **92%** with less false positive and false negative rate. So this model is considered as the model to be used in the detection system. The slow_http_test a attack simulator provided by open source Kali Linux is used to simulate the prediction. The model is converted into pickle file, then virtual attack is performed using the kali Linux with the slow_http_test in which the model performed well good in indicating the malware packets. The dis advantage of this model is, it takes time in classifying the packets and also lags to find the packets with active monitoring packets. In the perspective of security analysis the detection has a weak security i.e. it can be easily attacked and also it can be crashed. The future work of the project is to build a counter attack system like if a packet is found malware in the network a sql injection or DDoS counter attack will be done against the source IP system.

References

[1] Daniel Ramsbrock, Xinyuan Wang. (2013) "The Botnet Problem", in science direct (eds) *Computer and Information Security Handbook,* 223–238.

[2] S. Haq and Y. Singh. (2018) "Botnet Detection using Machine Learning" *Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC)*: 240-245, doi: 10.1109/PDGC.2018.8745912.

[3] Peng J, Fu Y, Cheng Y, Chen C and Guo Z. (2019). "Botnet Detection Method Based on Artificial Intelligence", *IEEE Fourth International Conference on Data Science in Cyberspace (DSC), IEEE*, 487-494.

[4] Hasan Alkahtani, Theyazn H. H. Aldhyani. (2021). "Botnet Attack Detection by Using CNN-LSTM Model for Internet of Things Applications". *Security and Communication Networks, vol. 2021, Article ID 3806459, 23 pages, 2021. https://doi.org/10.1155/2021/3806459*

[5] A. Alharbi and K. Alsubhi. (2021). "Botnet Detection Approach Using Graph-Based Machine Learning", *IEEE* ,**9** :99166–99180.

[6] Astha Bhargava and Neeraj Parihar. (2021). "Botnet Detection Using Artificial Intelligenc", *Artificial Intelligence and Data Mining Approaches in Security Frameworks ch4. DOI: 10.1002/9781119760429.ch4.*

[7] Manmeet Singh, Maninder Singh, Sanmeet Kaur, (2019). "Issues and challenges in DNS based botnet detection: A survey", *Computers Security,* **86**:28-52.

[8] Nguyen.Q and Hoang.X. (2018). "Botnet Detection Based On Machine Learning Techniques Using DNS Query Data", *Future Internet,* **10(5)**: 43.

[9] Binkley. J, Harley. D, Evron. G, Bradley. T, Willems. C and Cross MQ Schiller C.A. (2007). "Alternative Botnet CCs". *Botnets, Science Direct*,77–95.

[10] Sher. M, Bi. Y and Umer. M.F. (2017). "Flow-based intrusion detection: Techniques and challenges", *Computers Security, Science Direct*, **70** : 238–254.

[11] Bhaowal. M, Lee. T, Nivargi. V.(2019). "Machine Learning Based Botnet Detection", *Standford cs*.

[12] Ahmed.A. I. A, Almogren. A, Raza. M. A, Shah. S. A, Khan.A, Gani. A and Ali. I.(2020). "Systematic Literature Review on IoT-Based Botnet Attack", *IEEE*, **8**: 212220–212232.

[13] Grill. M,Stiborek. J, Zunino. A and Garcia. S. (2014). "An empirical comparison of botnet detection methods", *Computers Security, Science Direct,* **45**:100–123

[14] "What is a Botnet ?". (2018). *Palo Alto Networks*.

[15] Manickam. S. (2020). "Botnet Monitoring Mechanisms on Peer-to-Peer (P2p) Botnet.", *SSRN Electronic Journal*.

[16] S. M and T. S. (2011). "Advanced Methods for Botnet Intrusion Detection Systems", *Intrusion Detection Systems, Science Direct*.

[17] Nesarani A. Gurulakshmi, K(2018). "Analysis of IoT Bots Against DDOS Attack Using Machine Learning Algorithm",*2nd International Conference on Trends in Electronics and Informatics.*

[18] Irwin B Stalmans, E(2011). "A framework for DNS based detection and mitigation of malware infections on a network", *Information Security*.

[19] Khdour T. Freehat R. Manasrah, A. M(2022). "DGA-based botnets detection using DNS traffic mining. Journal of King Saud University - Computer and Information Sciences", *Science Direct*.

[20] Wang J. Zhang X. Li, X(2017). "Botnet Detection Technology Based on DNS", *Future Internet*, **9(4)**: pp.55

[21] Keller J. Spiekermann, D(2021). "Unsupervised packet-based anomaly detection in virtual networks", *Science Direct*, **192**.

[22] T Shin(2022). "All Machine Learning Models Explained in 6 Minutes - Towards Data Science", *Medium*.

[23] V. Worri(2018). "Extracting the payload from a pcap file using Python.", *Medium*.

[24] Sharma A Choudhary, S(2020). "Malware Detection Classification using Machine Learning International", *IEEE*.