DDoS ATTACK DETECTION ON BOTNET DEVICES

Shilpa M, Rosebell Paul

Department of Computer Science and Engineering
SCMS School of Engineering and Technology
Karukutty, India
shilpasreedharan23@gmail.com

Assistant Professor
Department of Computer Science and Engineering
SCMS School of Engineering and Technology
Karukutty, India
rosebell@scmsgroup.org

Abstract. The Internet of Things (IoT) [1] is being connected to by an increasing number of devices, however many of these IoT devices lack basic security, leaving the Internet open to various threats. Insecure consumer IoT devices have been leveraged by botnets like Mirai [2]to launch distributed denial of service (DDoS) assaults [3] against vital Internet infrastructure. A distributed denial-of-service (DDoS) attack is a malicious attempt to delay a server, service, or network's regular traffic by overloading the target or its surrounding infrastructure with an excessive amount of Internet traffic. By using numerous compromised computer systems as sources of attack traffic, DDoS attacks are made effective. Computers and other networked resources, like as IoT devices, can be exploited machines. This promotes the development of novel methods to immediately identify consumer IoT attack traffic. In this study, we use a variety of machine learning classifiers to identify DDoS attacks coming from botnet-infected IoT devices.

Keywords: **:** DDoS attack, IoT devices ,Machine learning classifiers

1.Introduction

The Internet of Things (IoT)[3] is the network of physical objects that can communicate with one another and make use of simple network protocols to sense, absorb, and respond to their environment. It is the result of advancements in embedded technologies, wireless sensor networks (WSNs), common networking protocols, and interconnected smart things. The most common uses of IoT devices are in fields where human interaction is difficult, such as manufacturing, transportation, healthcare, smart disaster management systems, smart homes, smart cities, and smart grid systems.

IoT networks face a number of challenges that call for the evolution of traditional internet topology. Network security has recently become more important due to the significant damage that DDoS poses to it. DDoS assaults [4]are now frequent as cyber threats because of the expansion of IoT devices, their complexity, and the use of attack services. A DDoS attack prevents actual internet users from using the suspect's services. IoT device failures and data theft are being caused more frequently by DDoS attacks on IoT devices. In response to this growing threat, new techniques are being developed to identify and halt attack traffic from IoT botnets. Recent anomaly detection[5] experiments using machine learning (ML) have demonstrated its potential to identify malicious Internet traffic.

IoT traffic frequently varies from traffic from other internet-connected devices (e.g. laptops and smart phones). For instance, IoT devices usually communicate with a small, restricted range of endpoints rather than a huge variety of distinct web servers. IoT devices are also more likely to have repeating network traffic patterns, such as frequently pinging the network with small packets at regular intervals for logging purposes. Here, we use a range of machine learning classifiers to identify DDoS attacks coming from botnet-infected IoT devices[6]. Then, that will appropriately differentiate between legitimate traffic and traffic used in DDoS attacks.

1.1 DDoS Attack

By overwhelming the target or its surrounding infrastructure with an insufficient amount of Internet data, a distributed denial-of-service (DDoS) attack[7] aims to delay the regular traffic to a server, service, or network. DDoS attacks are made successful by utilizing several compromised computer systems as sources of attack traffic. It is possible to abuse computers and other networked resources, such as IoT devices.

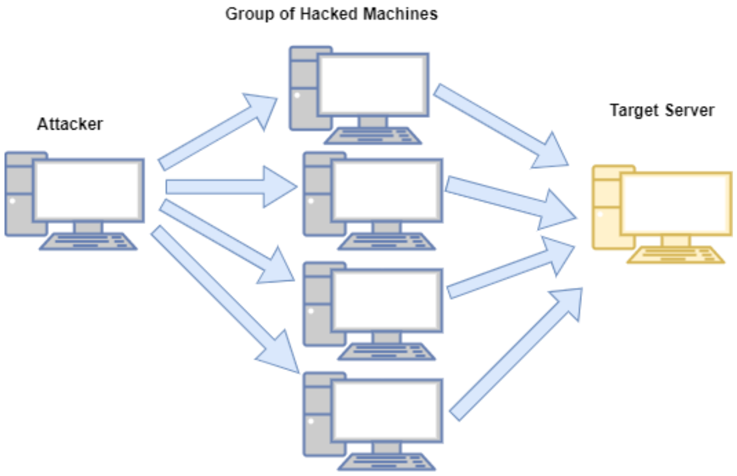

Fig 1 DDoS Attack

DDoS assaults(Fig.1) are undertaken using networks of computers linked to the Internet. These networks are made up of computers and other devices, such as Internet of Things (IoT) devices, that have been infected with malware, enabling an attacker to remotely manage them. These discrete machines are known as bots (or zombies), and a botnet[8] is a collection of bots. Once the botnet is configured, the attacker can command each bot remotely to direct the attack. When a server or network is being attacked by the botnet, each bot in the network sends queries to the victim's IP address. This might cause a server or network overflow, which would disrupt normal traffic. Because each bot is a real Internet device, it may be challenging to discern attack traffic from normal traffic.

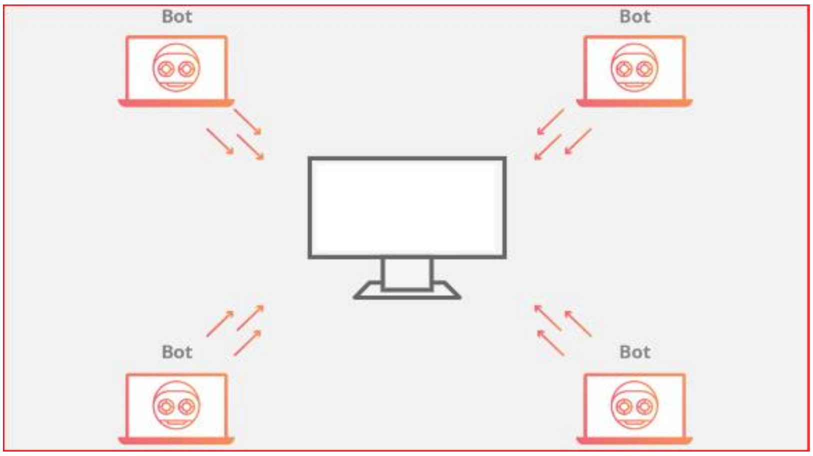

Fig 2 Botnet

The cybercriminal will usually create a "zombie network" of infected machines in order to send an extraordinarily high volume of requests to the victim resource. The sheer size of the attack can be overwhelming for the victim's web resources because the criminal has complete control over the behavior of every infected computer in the zombie network.(Fig. 2)

1.2 Dataset

The BoT-IoT[9] dataset's raw network packets (Pcap files) were made using the tshark program in the Cyber Range Lab of the Australian Center for Cyber Security (ACCS), and they include both regular and unusual traffic. Ostinato tool and Node-red were used to create simulated network traffic (for non-IoT and IoT respectively). The source files for the dataset are offered in a variety of formats, including the original pcap files, created argus files, and finally csv format. To help with labelling, the files were divided based on attack category and subcategory.

IoT systems have become a prominent target for those with malicious intentions because they play a significant role in the majority of IT Technology areas. It is necessary to build efficient defensive measures, such as intrusion detection systems, network forensic systems, etc., in light of such vulnerabilities and difficulties in using such systems. Utilizing security solutions based on machine learning is an effective technique to handle such difficulties. The project's objectives include the use of the Bot-IoT dataset to analyse various attack types as well as applying and contrasting various classification techniques.

- Here using the csv format of the dataset ,Which is DDoSdata.csv it consist of the information about DDoS attack on IoT devices
- Dataset consist of initially 47 features

The dataset is divided into two feature set

1. Basic Features
2. Flow based Features

Basic Features

Table 1.Basic features

Features and descriptions.

Feature	Description
pkSeqID	Row Identifier
Stime	Record start time
flgs	Flow state flags seen in transactions
flgs_number	Numerical representation of feature flags
Proto	Textual representation of transaction protocols present in network flow
proto_number	Numerical representation of feature proto
saddr	Source IP address
sport	Source port number
daddr	Destination IP address
dport	Destination port number
pkts	Total count of packets in transaction
bytes	Totan number of bytes in transaction
state	Transaction state
state_number	Numerical representation of feature state
ltime	Record last time
seq	Argus sequence number
dur	Record total duration
mean	Average duration of aggregated records
stddev	Standard deviation of aggregated records
sum	Total duration of aggregated records
min	Minimum duration of aggregated records
max	Maximum duration of aggregated records
spkts	Source-to-destination packet count
dpkts	Destination-to-source packet count
sbytes	Source-to-destination byte count
dbytes	Destination-to-source byte count
rate	Total packets per second in transaction
srate	Source-to-destination packets per second
drate	Destination-to-source packets per second
attack	Class label: 0 for Normal traffic, 1 for Attack Traffic
category	Traffic category
subcategory	Traffic subcategory

At first, there are 14 flow-based features and 32 basic features(Table 1). We remove some of the key features that are not needed for further analysis .Then there will be 15 basic features and 14 flow based features. Flow based features are derived from basic features and so no cleaning process required. The project aims to analyse different types of attacks using the Bot-IoT dataset and also apply & compare different classification algorithms.

Flow based Features

Table .2 Flow based features

	Feature	Description
1	TnBPSrcIP	Total Number of bytes per source IP
2	TnBPDstIP	Total Number of bytes per Destination IP.
3	TnP_PSrcIP	Total Number of packets per source IP.
4	TnP_PDstIP	Total Number of packets per Destination IP.
5	TnP_PerProto	Total Number of packets per protocol.
6	TnP_Per_Dport	Total Number of packets per dport
7	AR_P_Proto_P_SrcIP	Average rate per protocol per Source IP. (calculated by pkts/dur)
8	AR_P_Proto_P_DstIP	Average rate per protocol per Destination IP.
9	N_IN_Conn_P_SrcIP	Number of inbound connections per source IP.
10	N_IN_Conn_P_DstIP	Number of inbound connections per destination IP.
11	AR_P_Proto_P_Sport	Average rate per protocol per sport
12	AR_P_Proto_P_Dport	Average rate per protocol per dport
13	Pkts_P_State_P_Protocol_P_DestIP	Number of packets grouped by state of flows and protocols per destination IP.
14	Pkts_P_State_P_Protocol_P_SrcIP	Number of packets grouped by state of flows and protocols per source IP.

There are mainly 14 flow based features(Table 2). Here we use all these flow based features for the analysis and detection of DDoS attack on botnet devices.

2 High level Architecture

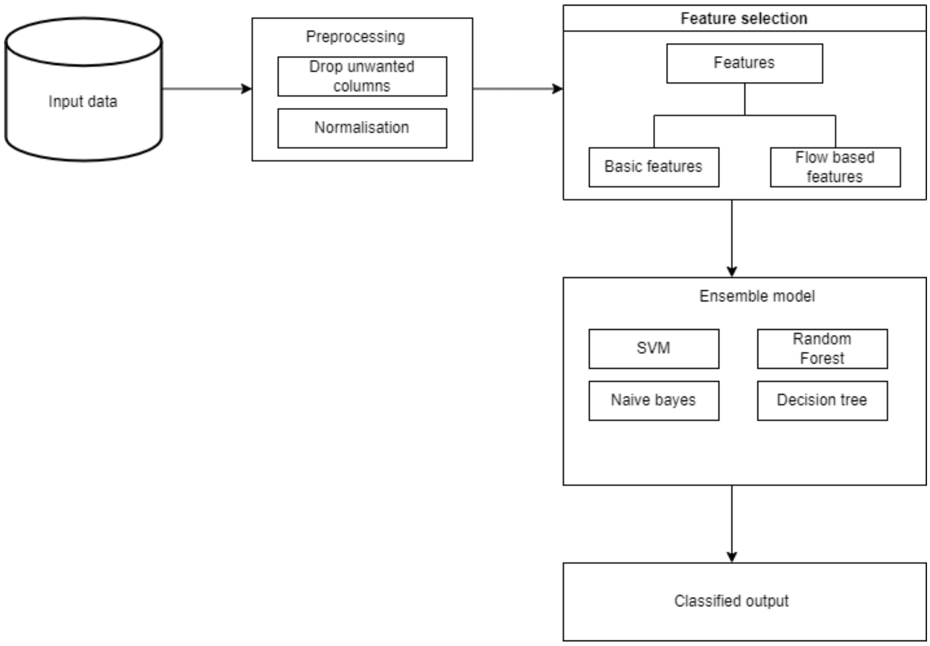

Fig 3. High level architecture

The above Fig 3 depicts the high level architecture of the system. The first step is input data; the dataset used for this system is (BoT-IoT)DDoSdata.csv.The second step is data Pre-processing here we drop the unwanted columns that are not used for further analysis and the normalize the values using Standard scalar. The dataset consist of 2 set of features basic features and flow based features .Initially there are 47 features and after Pre-processing we use 15 basic features and 14 flow based features. After the Pre-processing step .The dataset will be divided into training and testing data. The data that the model will learn from is the training data. We will utilise the testing data to determine how well the model performs on unobserved data. Then build models using SVM, Decision tree ,Naive Bayes, and random forest. Finally build the voting classifier . put each of our four models in the estimators array. We will then develop our voting classifier. Two inputs are required. Our estimator array of our three models comes first. The voting parameter will be set to hard, instructing our classifier to make predictions based on a majority vote. Now that our ensemble model has been fitted to our training data, we can evaluate it using our testing data.

3 Machine Learning Techniques for DDoS Detection

There are numerous methods for detecting DDoS. However, because of the new, intricate attack kinds, conventional ones are becoming outdated. The most effective method for identifying DDoS attacks is to use machine learning algorithms. Here we are using Support Vector Machine (SVM), Decision Tree Classification, Random Forest Classifier, Naive Bayes Classifier[10] for detecting DDoS attack on IoT devices

SVM

Support vector machines display training data as a set of points in space divided into groups by a distinct gap that is as wide as possible. Then, based on which side of the gap they fall, new samples are projected into that same area and predicted to belong to a category. Effective in high-dimensional spaces and memory-efficient due to the decision function's usage of a subset of training points.
Using python we can import the sklearn library as:

from sklearn import svm

Decision Tree

A decision tree uses a tree structure to develop classification or regression models. It incrementally develops an associated decision tree while segmenting a data set into smaller and smaller parts. The outcome is a tree containing leaf nodes and decision nodes. A decision node is represented by a leaf node, which has two or more branches and denotes a categorization or judgement. The root node, which corresponds to the best predictor, is the uppermost decision node in a tree. Decision trees can be used to process both categorical and numerical data.
Using python we can import the sklearn library as:

from sklearn.tree import DecisionTreeClassifier

Random forest

The ensemble learning method known as random forests, also referred to as random choice forests, is used for classification, regression, and other tasks. Many decision trees are constructed during the training phase, and the output class (for classification) or mean prediction (for regression) of the individual trees is represented by the output class. Using python we can import the sklearn library as:

from sklearn.ensemble import RandomForestClassifier

Naive Bayes

The Naive Bayes Classifier is a classification method based on the Bayes Theorem that makes the assumption that predictors are independent. A Naive Bayes classifier, to put it simply, believes that the presence of one feature in a class has nothing to do with the presence of any other feature. All of these traits individually add to the probability, even if they depend on one another or on the existence of other features. Simple to construct and especially helpful for very big data sets is the naive Bayes model. Along with being straightforward, Naive Bayes is known to perform better than even the most complex classification techniques.
Using python we can import the sklearn library as:

from sklearn.naive_bayes import GaussianNB

3.1 Ensemble learning model

Multiple machine learning models are used in ensemble learning[11] in an effort to improve predictions on a dataset. A dataset is used to train a variety of models, and the individual predictions made by each model form the basis of an ensemble model. The ensemble model then combines the outcomes of different model's predictions to produce the final outcome.

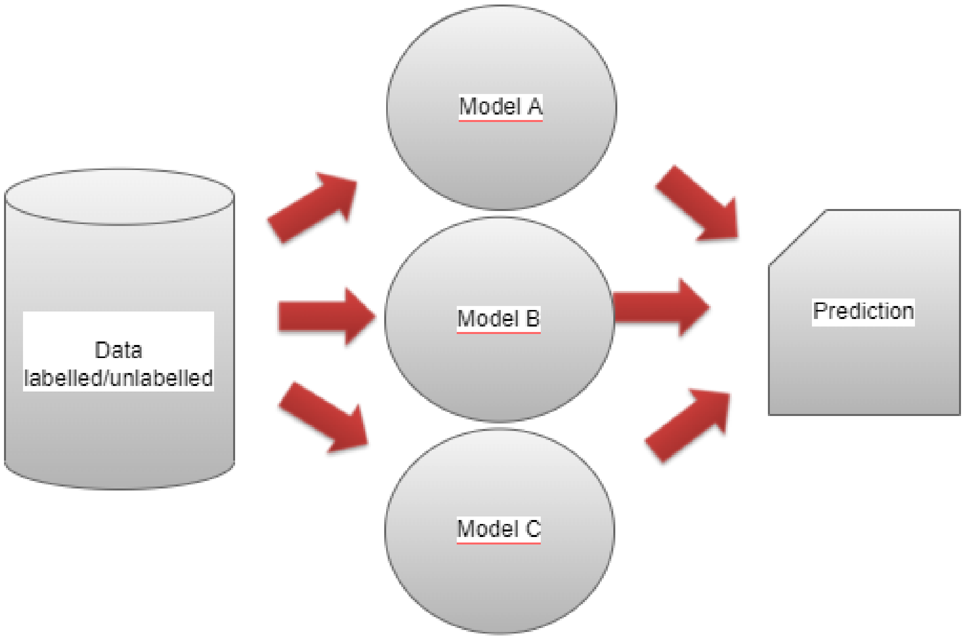

Fig .4 Ensemble model

Each model has advantages and disadvantages. By integrating different individual models, ensemble models (Fig 4)can help mask an individual model's flaws. In this project, we're going to use a voting classifier, where the ensemble model predicts by a vote of the majority. Our voting classifier[12] will be built using four different models: SVM, Random Forest, Decision Tree, and Naive Bayes. To execute these strategies and make use of the DDoSdata, we'll use the Python Scikit-learn module. dataset in csv

4 Results

This section presents and analyses the findings from the comparison of particular algorithms on our experimental model and the DDosdata.csv Dataset. SVM is the most accurate algorithm with an accuracy of 99.99% and Random Forest, Decision tree and Naive Bayes also had acceptable accuracy of 95.24%, 99.92%, and 99.94%. and then the final ensemble model based on majority voting also gives better accuracy 99.99%.

Table 3 .Ensemble result

DDoS ATTACK	
Algorithm	Accuracy
Decision Tree	99.92
Naïve Bayes	99.94
Random Forest	95.24
SVM	99.99

Ensemble=99.99

5 Conclusion and Future work

The primary goal of this study is to develop a detection model for separating DDoS attack traffic from other types of assault using the DDoSdata.csv (BoT-IoT) Dataset. In next studies, this model will be enhanced so that we can classify various assault types. We will also experiment with different algorithms and hybrid tactics in an effort to improve the effectiveness and efficiency of our model. We plan to test this model on more recent datasets as one of our upcoming initiatives..

It was suggested in this investigation that botnet or malicious traffic activity on IoT be detected using machine learning methods. Four classifiers were utilised in this study: Naive Bayes, Random Forest, Support Vector Machine, and Decision Trees. The experimental data showed that the SVM model performed better than the other classifier models. Theoretically, this approach might be used to detect different botnet attacks and other sorts of unwanted network behaviour. The UNSW-NB15 da-taset [13] and the CTU-13 [14], which are more recent datasets, could potentially be added to this study in order to evaluate how well the algorithms work when dealing with different types of botnet traffic. It is also possible to test additional classifiers like logistic regression and neural networks. Furthermore, the supervised learning methods used. Further refining of these results can be done by looking into alternative feature selection techniques. Last but not least, the machine learning model may be evaluated in a controlled real-time environment to determine how well it performs and responds to various attacks, including zero-day threats.

References

[1] Sonar, Krushang, and Hardik Upadhyay. "A survey: DDOS attack on Internet of Things." International Journal of Engineering Research and Development 10.11 (2014): 58-63.
[2] Antonakakis, Manos, et al. "Understanding the mirai botnet." 26th USENIX security symposium (USENIX Security 17). 2017.
[3] Vishwakarma, Ruchi, and Ankit Kumar Jain. "A survey of DDoS attacking techniques and defence mechanisms in the IoT network." Telecommunication systems 73.1 (2020): 3-25.
[4] Aysa, Mahdi Hassan, Abdullahi Abdu Ibrahim, and Alaa Hamid Mohammed. "Iot ddos attack detection using machine learning." 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT). IEEE, 2020.
[5] Chan, Philip K., Matthew V. Mahoney, and Muhammad H. Arshad. A machine learning approach to anomaly detection. 2003.
[6] Tuan, Tong Anh, et al. "Performance evaluation of Botnet DDoS attack detection using machine learning." Evolutionary Intelligence 13.2 (2020): 283-294.
[7] https://www.cloudflare.com/en-in/learning/ddos/what-is-a-ddos-attack/
[8] Choo, Kim-Kwang Raymond. "Zombies and botnets." Trends & Issues in Crime & Criminal Justice 333 (2007).
[9] Koroniotis, Nickolaos, et al. "Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-iot dataset." Future Generation Computer Systems 100 (2019): 779-796.
[10] Doshi, Rohan, Noah Apthorpe, and Nick Feamster. "Machine learning ddos detection for consumer internet of things devices." 2018 IEEE Security and Privacy Workshops (SPW). IEEE, 2018.
[11]https://towardsdatascience.com/ensemble-learning-using-scikit-learn-85c4531ff86a
[12] Polikar, Robi. "Ensemble learning." Ensemble machine learning. Springer, Boston, MA, 2012. 1-34.
[13] Moustafa, Nour, and Jill Slay. "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)." 2015 military communications and information systems conference (MilCIS). IEEE, 2015.
[14] Yavanoglu, Ozlem, and Murat Aydos. "A review on cyber security datasets for machine learning algorithms." 2017 IEEE international conference on big data (big data). IEEE, 2017.
[15] McGregor, Anthony, et al. "Flow clustering using machine learning techniques." International workshop on passive and active network measurement. Springer, Berlin, Heidelberg, 2004.