e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:03/Issue:09/September-2021 Impact Factor- 6.752 www.irjmets.com

DISTRIBUTED DENIAL OF SERVICE (DDOS) ATTACKS DETECTION MECHANISM

Aaron Aby[*1], **Ms. Shoby Sunny**[*2]

[*1]Student, Department Of Computer Applications , SCMS School Of Technology And Management Muttom, Aluva , 683106.

[*2]Professor, Department Of Computer Applications , SCMS School Of Technology And Management, Muttom, Aluva, 683106.

ABSTRACT

A distributed denial of service (DDoS) attack is one within which an oversized number of compromised systems connect with a single target (like a website), thereby causing a denial of service for genuine users of the targeted system. The flood of incoming messages to the target system essentially forces it to clean up, rendering it unable to service legitimate users. In DDOS several machines together become a {part of} a series of computers which start the attack these systems become part of an outsized collection of computers said as a botnet. The paper basically focuses on showing the way within which the attack is performed and discuss the assorted classifying algorithms which might be accustomed classify whether the incoming packet may be a malicious packet or not and perform a comparative analysis to bring out the foremost efficient one and also the paper brings out an algorithm which may be proposed together with these classifying algorithms to boost the general performance.

Keywords: Dos (Denial Of Service), Ddos (Distributed Denial Of Service Attack), Internet Protocol, Transmission Control Protocol And Ping Of Death (POD), Classifiers, Mitigation.

I. INTRODUCTION

DDoS attacks are applied with networks of connected machines. These carries with it computers and other devices which are infected with malware, allowing them to be controlled remotely by an attacker. These individual devices are spoken as bots, and a gaggle of bots is named a botnet. Once a botnet has been established, the attacker is ready to direct an attack by sending remote instructions to every bot. When a victim's server or network is targeted by the botnet, each bot sends requests to the target's IP address, potentially causing the server or network to become overwhelmed, leading to a denial-of-service to normal traffic. Because each bot could be a legitimate Internet device, separating the attack traffic from normal traffic are often difficult.

How to identify a DDoS attack:

The most obvious symptom of a DDoS attack may be a site or service suddenly becoming slow or unavailable. But since variety of causes such a spike in traffic can create similar performance issues, further investigation is usually required. Traffic analytics tools can facilitate your spot a number of the signs of a DDoS attack:

1)Suspicious amounts of traffic originating from one IP address or IP range.

2)A flood of traffic from users who share one behavioural profile, like device type, geolocation, or web browser version.

3)An unexplained surge in requests to one page or endpoint

4)Odd traffic patterns like spikes at odd hours of the day or patterns that appear to be unnatural (e.g. a spike every 10 minutes)

II. METHODOLOGY

Implementing a DDOS attack

In this paper I will be trying to implement a ping of death attack. While performing ping of death attack, the network information needs to be gathered, and to achieve this, ipconfig command can be used. In Fig. 1, the detailed information of the network is gathered after giving ipconfig command. As soon as the network information is gathered, we can start performing the ping of death attack on the IP address. Enter the following command to start the attack.

ping-t –l 65500 XX.XX.XX.XX

"ping" command transfer the data packets to the target

1) "XX.XX.XX.XX" is the IP address of the target

2) "–t" mean s sending packets repeatedly

3)"–l" specifies the data packet load to be sent to the target

Implementing The Ping Of Death Attack

As you can see from the above figure the ping of death is commencing from the source ip 192.168.1.12 with packets of size 65000 in an endless loop

Now this same attack can be converted to a file which can then be sent over the network to millions of other computers over the network and all of them becomes a part of the large army of botnets. Now these botnets can work together and bring down a whole network on its own and that's the reason why its considered to be one of the lethal forms of cyber attack

Experimental Setup

I performed this attack on my own personal laptop network and performed the ping of death attack for several hours before it started to use much of the system resources.

Laptop Configuration is as follows

Device name LAPTOP-NC6DKL7G

Processor AMD Ryzen 5 4600H with Radeon Graphics 3.00 GHz

Installed RAM 8.00 GB (7.42 GB usable)

Device ID 660F8E8A-D0B7-4BFF-8D7D-74B8457E62D7

Product ID 00327-35874-52368-AAOEM

System type 64-bit operating system, x64-based processor

Pen and touch No pen or touch input is available for this display

Displaying CPU usage before performing the ping of death attack is given below

DDOS Detection Using Machine Learning Algorithms

Due to the above-mentioned problems, it becomes a necessity to detect malicious DDOS patterns and thereby develop some mechanisms to mitigate it. For Detecting malicious patterns, i've got selected 4 different classifiers to perform a comparative analysis and convey out which classifier is that the best and which may be used for mitigation purposes. This experiment was disbursed using Weka 3.8 tool

Dataset Used

NSL-KDD dataset for intrusion detection

Random Forest Classifier

Random Forest is an ensemble method that mixes multiple decision trees to classify, that the results of random forest is sometimes better than decision trees.

Random forests have a spread of applications, like recommendation engines, image classification and have selection. It is accustomed classify loyal loan applicants, identify fraudulent activity and predict diseases. It lies at the bottom of the Boruta algorithm, which selects important features in an exceedingly dataset. We can train a dataset as follows from sklearn. ensemble import Random Forest Classifier

classifier = Random Forest Classifier (n_estimators = 50)

classifier.fit(X_train, y_train)

Naïve Bayes Classifier

Naïve Bayes algorithm could be a supervised learning algorithm, which is predicated on Bayes theorem and used for solving classification problems. it's mainly utilized in text classification that features a high-dimensional training dataset. Naïve Bayes Classifier is one in all the straightforward and best Classification algorithms which helps in building the fast machine learning models that may make quick predictions. it's a probabilistic classifier, which suggests it predicts on the premise of the probability of an object. Some popular samples of Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

J48 Classifier

It is an algorithm to come up with a call tree that's generated by C4.5 (an extension of ID3). it's also called a statistical classifier.

DDOS Mitigation

DDoS mitigation may be a set of network management techniques and/or tools for resisting or mitigating the impact of distributed denial-of-service (DDoS) attacks on networks attached to the web by protecting the target and relay networks. DDoS attacks are a relentless threat to businesses and organizations by threatening service performance or to clean up a web site entirely, even for a brief time. Now once the instances are correctly classified as an attack or not the following step within the process involves to mitigate the malicious packets and make sure that these packets don't seem to be received in future.

Proposed Algorithm For DDOS Mitigation This phase are done at the monitor agent. the most purpose of the Proposed DDoS Mitigation Algorithm (PDMA) is to scale back the impact of the DDoS attack. The algorithm maintains two tables: Grey List and Backlist tables as described below:

Grey List: The Algorithm uses the Grey List for temporary blocking of the suspected addresses. Grey List is an important resource to test whether the incoming traffic is an attack traffic or normal traffic. Backlist: this list is employed to dam the attacker address permanently. Components in Backlist are always considered as an attack within the highest priority.

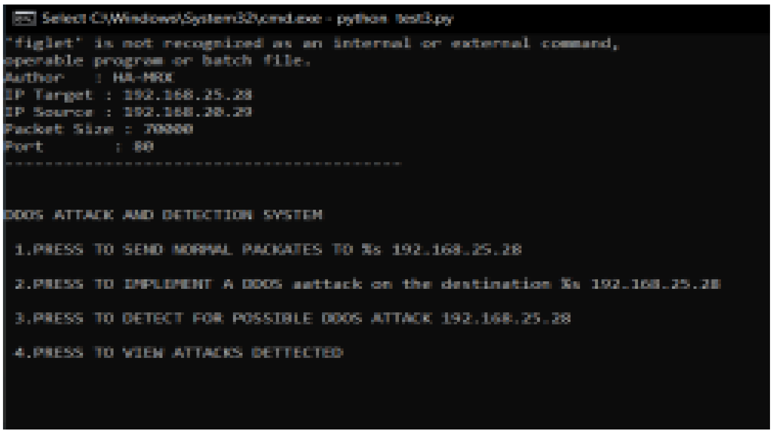
Algorithm:

```
If warning packet then
If in GreyList then
boost Blacklist;
else if high-rate attack warning then
Add to grey list;
end if
Set expiration time to delete record (false Positive);
end if
else // for Data Packets if in white list then forward packet;
else if in grey list and (dest_addr=victim_addr) then
drop packet;
else if in black list then drop packet;
else forward Packet (new flow);
end if
end if
```

The above proposed algorithm can work together with one in all the classifier algorithms and may overall work together as a ddos detection system where the classifier first detects if the incoming packets are malicious or not. After this detection/classification those data set are often missed out through this proposed algorithm which on the opposite hand supported packet size and arrival rate denotes if it's a warning packet and if it's one it moves it to the grey list table and if the identical instance occurs with the identical source they get added to the blacklist and thereby not receiving packets from it anymore.

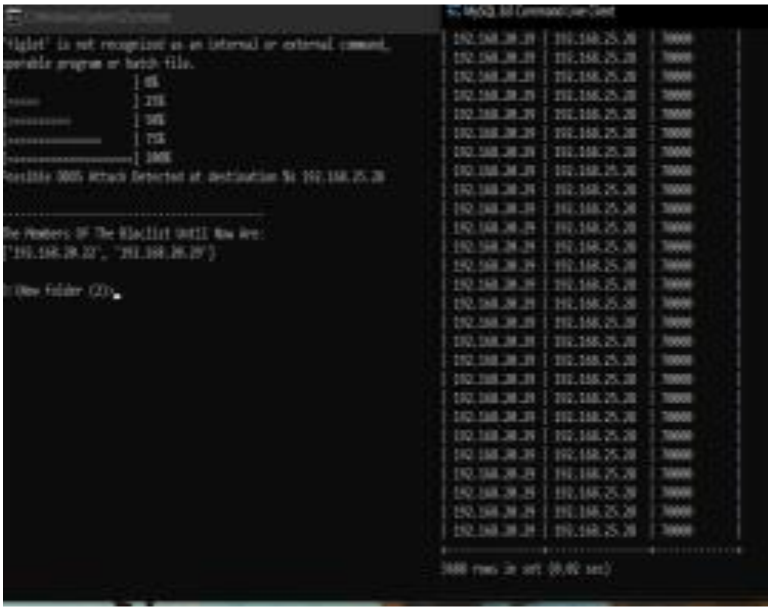
III. MODELING AND ANALYSIS

Model and Material which are used is presented in this section. Table and model should be in prescribed format.



```
Select C:\Windows\System32\cmd.exe - python test3.py
'figlet' is not recognized as an internal or external command,
operable program or batch file.
Author    : HA-MRX
IP Target : 192.168.25.28
IP Source : 192.168.20.29
Packet Size : 70000
Port        : 80
-----------------------------------------


DDOS ATTACK AND DETECTION SYSTEM

 1.PRESS TO SEND NORMAL PACKATES TO %s 192.168.25.28

 2.PRESS TO IMPLEMENT A DDOS aattack on the destination %s 192.168.25.28

 3.PRESS TO DETECT FOR POSSIBLE DDOS ATTACK 192.168.25.28

 4.PRESS TO VIEW ATTACKS DETTECTED
```

Figure 1: Project Implementation and user entry section



```
'figlet' is not recognized as an internal or external command,
operable program or batch file.
[                        ] 0%
[=====                   ] 25%
[===========             ] 50%
[=================       ] 75%
[========================] 100%
Possible DDOS Attack Detected at destination %s 192.168.25.28

-----------------------------------------
The Members Of The Blaclist Until Now Are:
['192.168.20.22', '192.168.20.29']

E:\New folder (2)>
```

```
MySQL 8.0 Command Line Client
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
| 192.168.20.29 | 192.168.25.28 | 70000 |
+---------------+---------------+-------+
[illegible] rows in set (0.02 sec)
```

Figure 2: Project Implementation of prescribed model

IV. RESULTS AND DISCUSSION

A) Random Forest Classifier

Time taken to build model: 7.12 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances	22252	98.7048 %
Incorrectly Classified Instances	292	1.2952 %

Kappa statistic: 0.9736

Mean absolute error: 0.0201

Root mean squared error: 0.0986

Relative absolute error: 4.0972 %

Root relative squared error: 19.9075 %

Total Number of Instances: 22544

=== Confusion Matrix ===

a b <-- classified as

9560 151 | a = normal

141 12692 | b = anomaly

B) Random Forest Classifier

Time taken to build model: 0.3 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances	18201	80.7355 %
Incorrectly Classified Instances	4343	19.2645 %

Kappa statistic: 0.6231

Mean absolute error: 0.1924

Root mean squared error: 0.4371

Relative absolute error: 39.2235 %

Root relative squared error: 88.2663 %

Total Number of Instances: 22544

=== Confusion Matrix ===

a b <-- classified as

9225 486 | a = normal

3857 6 | b = anomaly

C) J48 Classifier

Time taken to build model: 1.99 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances	22228	98.5983 %
Incorrectly Classified Instances	316	1.4017 %

Kappa statistic: 0.9714

Mean absolute error: 0.017

Root mean squared error: 0.1056

Relative absolute error: 3.4631 %

Root relative squared error: 21.316 %

Total Number of Instances" 22544

=== Confusion Matrix ===

```
 a b <-- classified as
 9574 137 | a = normal
 179 12654 | b = anomaly
```

Thus, from the above performed experiments on different classifier algorithms we can clearly conclude that the Random Forest classifiers is the most efficient and correct classifier as it has correctly identified attack instances with a 98.7048 % accuracy whereas the j48 classifier came behind with a 98.5983% and the naïve bayes classifier performed the least good with only an 80.7355% accuracy

The error statics which is the probability of error and also the confusion matrix is also provided for the different classifiers and the random forest classifier comes out on top of all the other remaining classifiers.

The time required to build the random forest classifier was found to be more as it took almost 7 seconds to build the model itself whereas the naïve bayes classifier model could be built in just 0.3 seconds

V. CONCLUSION

As discussed in the paper earlier DDOS is a high threat to many organizations and therefore its mitigation is of high importance. The paper basically discusses the ways in which a penetration attack can be performed over a network and the paper also looks at the different classifier algorithms that can be used to classify whether a set of instances is an attack or not. Once the detection phase is done the trained data can be modeled or passed on to the proposed DDOS mitigation algorithm.

The proposed DDOS mitigation technique basically looks at these classified instances and based on whether it's a low intensity or high intensity attack, moves it to a Grey List which is a table which consists of temporary address which are blocked. If the attack continues from the same source the algorithm pushes it to the Black List where it will be permanently blocked and thereby won't receive packets from that particular source.

At present the proposed algorithm is executed on a small set of data with less attributes specified and in future I will look into the ways of implementing it on a larger scale and also trying to develop a combined DDOS detection and prevention system.

The main purpose of this study is to bring out the different mechanisms by which we can detect a DDOS attack and explain the ease at which any person can implement a DDOS attack and also provide a mitigation solution which is as mentioned above performed on a small scale over a small set of data

VI. REFERENCES

[1] Amiri, I. S. (2015). Theoretical And Experimental Methods For Defending Against DDoS Attacks.

[2] B. B. Gupta, M. M. (2018). An ISP Level Solution to Combat DDoS Attacks using Combined Statistical Based Approach. Journal of Information Assurance and Security .

[3] Behal, S. (2020). Detection Of DDOS Attacks Using Machine Learning Algorithms. 6.

[4] Bhange, A., Syad, A., & Thakur, S. S. (2019). DDoS Attacks Impact on Network Traffic and its Detection Approach. International Journal of Computer Applications, 5.

[5] Devi, S. R., & Yogesh, P. (2014). DETECTION OF APPLICATION LAYER DDOS ATTACKS USING INFORMATION THEORY BASED METRICS. 7.

[6] Dhruba Kumar Bhattacharyya, J. K. (2016). DDoS attacks : evolution, detection, prevention, reaction, and tolerance. Boca Raton, Florida : CRC Press, [2016].

[7] kumarasamy, S., & Dr.R.Asokan. (December 2011). DISTRIBUTED DENIAL OF SERVICE (DDOS) ATTACKS DETECTION MECHANISM. International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), 11.

[8] Nsaif, M. R., & Abbood, M. F. (2020). Detection and Prevention Algorithm of DDoS Attack Over the IOT Networks. 8.

[9] Sagar Pande, A. K. (2020). DDOS Detection Using Machine Learning Technique.

[10] V.Priyadharshini, & Dr.K.Kuppusamy. (2012). Prevention of DDOS Attacks using New Cracking Algorithm. International Journal of Engineering Research and Applications , 5.

[11] GitHub - masoudslipknot/DDoS_detection: A simple DDoS detection code.

[12] GitHub - steviegoneevil/ANN-for-DDoS-detection: Final Year Project.