**A REPORT**

**ON**

**Distributed Denial of Service (DDoS) Mitigation Technique**

**BY**

**Ishank Jain 2014A7PS0051U Computer Science**

**AT**



**BITS, Pilani – Dubai Campus**

**Dubai International Academic City (DIAC)**

**Dubai, U.A.E**

**Second Semester, 2017-2018**

**A REPORT**

**ON**

**Distributed Denial of Service (DDoS) Mitigation Technique**

**BY**

**Ishank Jain 2014A7PS0051U computer Science**

**Prepared in Partial Fulfillment of the**

**Project Course: CS F376**

**AT**

****

**BITS, Pilani – Dubai Campus**

**Dubai International Academic City (DIAC)**

**Dubai, UAE**

**Second Semester, 2017-18**

**BITS, Pilani – Dubai Campus**

**Dubai International Academic City (DIAC)**

**Dubai, UAE**

**Course Name:** Design Project

**Course No:** CS F376

**Duration:** 4 months

**Date of Start:** 22nd January 2018

**Date of Submission:** 20th May

**Title of Report: Distributed Denial of Service (DDoS) Mitigation Technique**

**Name:** Ishank Jain

**ID Number:** 2014A7PS0051U

**Discipline:** Computer Science Engineering

**Name of Project Supervisor:** Dr. Santhosh Kumar

**Keywords:** Botnet, DDoS attack, Random forest algorithm, ADAboost algorithm, blockchain, Gradientboost Algorithm, Extra tree algorithm, machine learning

**Project Area:** Computer Security

# **Abstract**

# DDoS attacks is a big threat to networks, to effectively detect DDoS attack is a important topic of information security.My project aims to look at various techniques to reduce possibility of DDoS (‘Distributed Denial of Service’) attack. We will try to implement machine learning techniques (Random forest algorithm, ADAboost algorithm, blockchain, Gradientboost Algorithm, Extra tree algorithm) to filter infected network packet from overloading websites. Project provides theoretical and statistical advantage of implementing such model. We will be using NSL\_KDD dataset to run our algorithm and infer the results. We will first train the data and then run test data on it. We will visualize the results in the form of tables, graph and heat maps. For future studies we will try and introduce blockchain architecture (using smart contracts) to block infected IP’s.

**Signature of the Student Signature of Faculty**

**Date: Date:**

# Acknowledgements

Firstly, I would like to express my heartfelt gratitude to the Director of BITS Pilani, Dubai Campus, Prof. R.N. Saha, who has ushered a new light on our college.

Next, I would like to thank Dr. Santhosh Kumar, my Project Supervisor, for his faith in me even before I started working under him and for his motivation, support, guidance and encouragement throughout the course of this project. He has been a great influence, urging me to learn and to be innovative, pushing me to excel.

Ishank Jain

2014A7PS0051U

Contents

[Abstract ii](#_Toc514542549)

[Acknowledgements iii](#_Toc514542550)

[LIST OF FIGURES vi](#_Toc514542551)

[CHAPTER 1 DDOS ATTACK 7](#_Toc514542552)

[1.1 Introduction 7](#_Toc514542553)

[1.1.1 What is a DDoS attack? 7](#_Toc514542554)

[1.2 Types of DDoS attacks: 8](#_Toc514542555)

[1.2.1 Volume Based Attacks 8](#_Toc514542556)

[1.2.2 Protocol Attacks 8](#_Toc514542557)

[1.2.3 Application Layer Attacks 8](#_Toc514542558)

[1.2.4 Zero-day DDoS Attacks 9](#_Toc514542559)

[1.3 How DDoS attacks work 9](#_Toc514542560)

[1.3.1 What is a Botnet? 10](#_Toc514542561)

[1.4 Challenge 11](#_Toc514542562)

[1.5 Goal 11](#_Toc514542563)

[2 CHAPTER 2 PRIOR RESEARCH 12](#_Toc514542564)

[2.1 Related work 12](#_Toc514542565)

[2.2 DDoS Mitigation in Cloud Services 12](#_Toc514542566)

[2.3 DDoS Mitigation using ML 12](#_Toc514542567)

[2.4 SDN based DDoS mitigation 13](#_Toc514542568)

[3 CHAPTER 3 PROPOSED WORK 14](#_Toc514542569)

[3.1 Random Forest 15](#_Toc514542570)

[3.2 Gradient Boost Classifier 15](#_Toc514542571)

[3.3 AdaBoost Classifier 15](#_Toc514542572)

[3.4 Extra Randomized Tree Classifier 15](#_Toc514542573)

[3.5 NSL-KDD dataset 17](#_Toc514542574)

[3.6 RESULTS 17](#_Toc514542575)

[3.6.1 Model1: Modelling is done with complete feature set 17](#_Toc514542576)

[3.6.2 Model 2: Feature set-> features with small deviation removed 19](#_Toc514542577)

[3.6.3 Model 3: Feature set-> without low deviation and high correlation 20](#_Toc514542578)

[Conclusion 22](#_Toc514542579)

[Future Scope 22](#_Toc514542580)

[REFERENCES 23](#_Toc514542581)

# LIST OF FIGURES

|  |  |
| --- | --- |
| FIGURES | Page Number |
| Fig: Live digital attack map On 15th Feb | 7 |
| Fig: Incapsula mitigates a massive HTTP flood: 690,000,000 DDoS requests from 180,000 botnets IPs | 9 |
| Fig: How DDoS attack works | 10 |
| Fig: Botnets | 10 |
| Fig: SDN mitigation technique | 13 |
| Fig: classes of different attacks | 14 |
| Fig: How model works | 14 |
| Fig: Training machine | 15 |
| Fig: Sample for Machine learning algorithms executing | 16 |
| Fig: Results for ML algorithm according to specific features (Ensemble) | 16 |
| Fig: 1st model | 17 |
| Fig: 2nd model | 18 |
| Fig: Heatmap showing correlation in features | 19 |
| Fig: 3rd model | 20 |
| Fig: Blockchain Architecture | 21 |

# CHAPTER 1 DDOS ATTACK

## Introduction

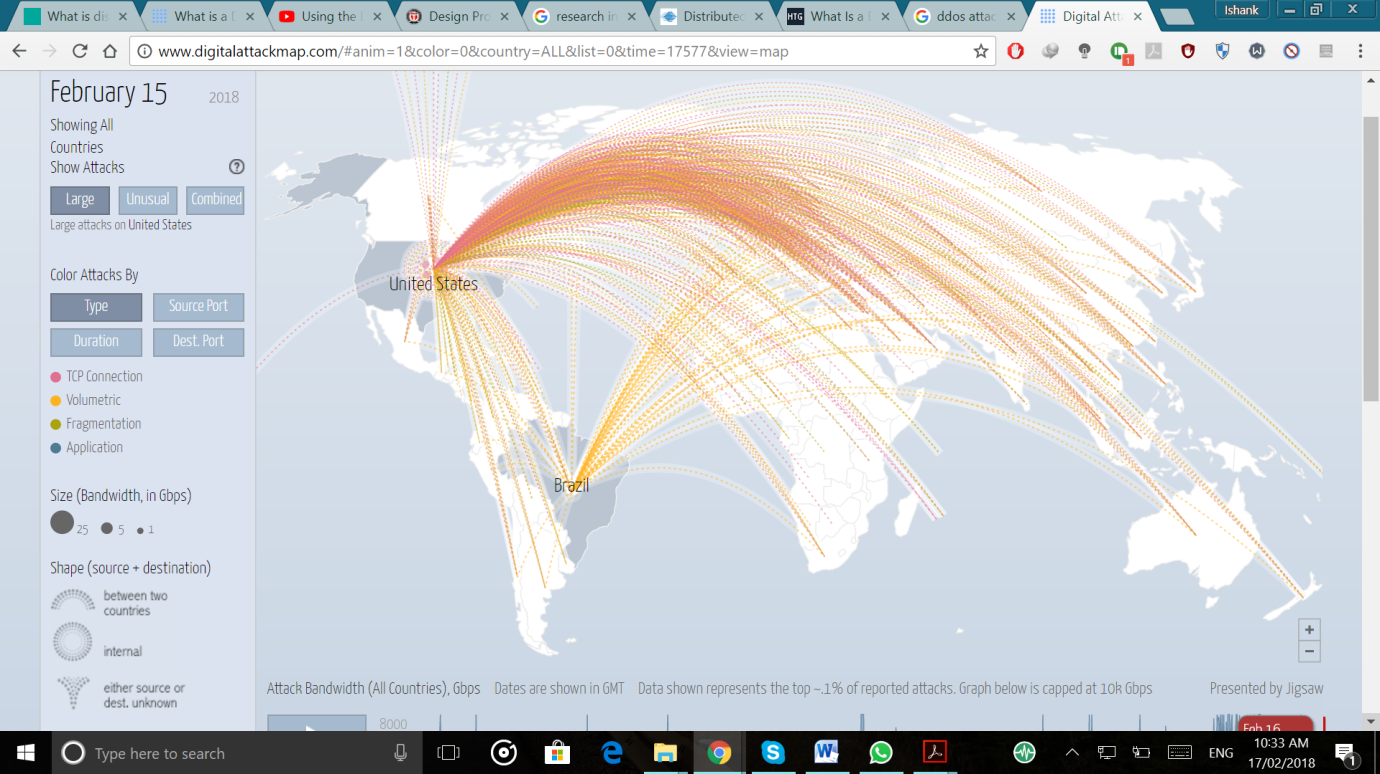
### What is a DDoS attack?

A distributed deniaI-of-service (DDoS) attack is an attack in which multipIe compromised computer systems attack a target, such as a server, website or other network resource, and cause a [denial of service](http://searchsecurity.techtarget.com/definition/denial-of-service) for users of the targeted resource. The flood of incoming messages, [connection](http://searchnetworking.techtarget.com/definition/connection) requests or malformed [packets](http://searchnetworking.techtarget.com/definition/packet) to the target system forces it to slow down or even crash and shut down, thereby denying service to legitimate users or systems.

<https://youtu.be/NogCN78XN2w>

A Distributed Denial of Service (DDoS) attack is an attempt to make an online service unavailable by overloading it with traffic from multiple sources. They target a wide variety of important resources, from banks to government websites, and pose a major challenge to making sure people can circulate and access important information.

From 2002 to 2007, the worldwide DNS root servers have over and over been targeted by DDoS Attacks, bringing about an extensive number of unresponsive servers; In March 2015, GitHub experienced expansive activity DDOS attack; In September 2016, programmers used Mirai to target the Brian Krebs site and the attack traffic reached 665 Gbps.



**Fig: Live digital attack map On 15th Feb**

## Types of DDoS attacks:

 DDoS attacks can be divided into three types:

### Volume Based Attacks

Includes UDP floods, ICMP floods, and other spoofed-packet floods. The attack’s goal is to saturate the bandwidth of the attacked site, and magnitude is measured in bits per second (Bps).

1. [UDP Flood](https://www.incapsula.com/ddos/attack-glossary/udp-flood.html)

A UDP flood, by definition, is any DDoS attack that floods a target with User Datagram Protocol (UDP) packets. The goal of the attack is to flood random ports on a remote host. This causes the host to repeatedly check for the application listening at that port, and (when no application is found) reply with an ICMP ‘Destination Unreachable’ packet. This process saps host resources, which can ultimately lead to inaccessibility.

1. [ICMP (Ping) Flood](https://www.incapsula.com/ddos/attack-glossary/ping-icmp-flood.html)

Similar in principle to the UDP flood attack, an ICMP flood overwhelms the target resource with ICMP Echo Request (ping) packets, generally sending packets as fast as possible without waiting for replies. This type of attack can consume both outgoing and incoming bandwidth, since the victim’s servers will often attempt to respond with ICMP Echo Reply packets, resulting a significant overall system slowdown.

### Protocol Attacks

Includes SYN floods, fragmented packet attacks, Ping of Death, Smurf DDoS and more. This type of attack consumes actual server resources, or those of intermediate communication equipment, such as firewalls and load balancers, and is measured in packets per second (Pps).

1. [Slowloris](https://www.incapsula.com/ddos/attack-glossary/slowloris.html)

Slowloris is an highly focused attack, empowering one web server to bring down another server, without influencing different services or ports on the target system. Slowloris does this by holding whatever number connection with the objective web server open for whatever length of time that could be expected under the circumstances. It achieves this by making connection with the objective server, yet sending just a fractional demand. Slowloris continually sends more HTTP headers, yet never finishes a request. The focused on server keeps every one of these false connections open. This in the end floods the maximum concurrent connection pool, and leads to denial of additional connections from legitimate clients.

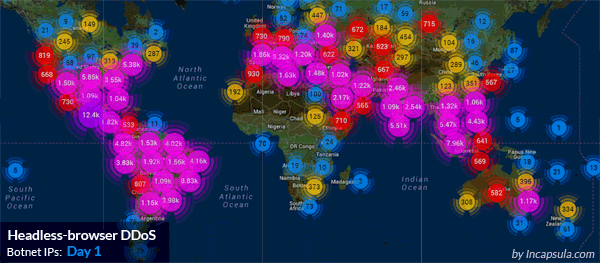
1. [NTP Amplification](https://www.incapsula.com/ddos/attack-glossary/ntp-amplification.html)

In NTP enhancement attack, the adversary misuses publically-available Network Time Protocol (NTP) servers to overpower a targeted server with UDP movement. The attack is characterized as an intensification attack on the grounds that the request to-response proportion in such situations is anyplace in the vicinity of 1:20 and at least 1:200. This implies any attacker that gets a rundown of open NTP servers (e.g., by an utilizing instrument like Metasploit or information from the Open NTP Project) can without much of a stretch produce an overwhelming high-data transmission, high-volume DDoS attack.

### Application Layer Attacks

Includes low-and-slow attacks, GET/POST floods, attacks that target Apache, Windows or OpenBSD vulnerabilities and more. Comprised of seemingly legitimate and innocent requests, the goal of these attacks is to crash the web server, and the magnitude is measured in Requests per second (Rps).

1. [HTTP Flood](https://www.incapsula.com/ddos/attack-glossary/http-flood.html)

In a HTTP surge DDoS attack, the adversary exploit apparently real HTTP GET or POST request to target a web server or application. HTTP floods don't utilize deformed parcels, spoofing or reflection methods, and require less transmission capacity than different attacks to cut down the target in the vicinity or server. The assault is best when it powers the server or application to apportion the most extreme assets conceivable because of each single demand.Fig: Incapsula mitigates a massive HTTP flood: 690,000,000 DDoS requests from 180,000 botnets IPs.

### Zero-day DDoS Attacks

The “Zero-day” definition encompasses all unknown or new attacks, exploiting vulnerabilities for which no patch has yet been released. The term is well-known amongst the members of the hacker community, where the practice of trading zero-day vulnerabilities has become a popular activity.

1. DDoS Reflector attack. It is a kind of attack which is difficult to defend as the victim computer is flooded with traffic from other Internet servers, which were not even compromised. This attack exploits the SYN ACKs in response to the TCP SYN requests and other TCP packets.

## How DDoS attacks work

In an ordinary DDoS assault, the foe starts by abusing a defenselessness in one PC framework and making it the DDoS ace. The assault ace framework recognizes other powerless frameworks and additions control over them by either tainting the frameworks with malware or through bypassing the confirmation controls (i.e., speculating the default watchword on a generally utilized framework or gadget).

A PC or arranged gadget under the control of an interloper is known as a zombie, or bot. The aggressor makes what is known as an order and-control server to charge the system of bots, likewise called a botnet. The individual responsible for a botnet is once in a while alluded to as the botmaster (that term has likewise generally been utilized to allude to the principal framework "enrolled" into a botnet in light of the fact that it is utilized to control the spread and movement of different frameworks in the botnet).

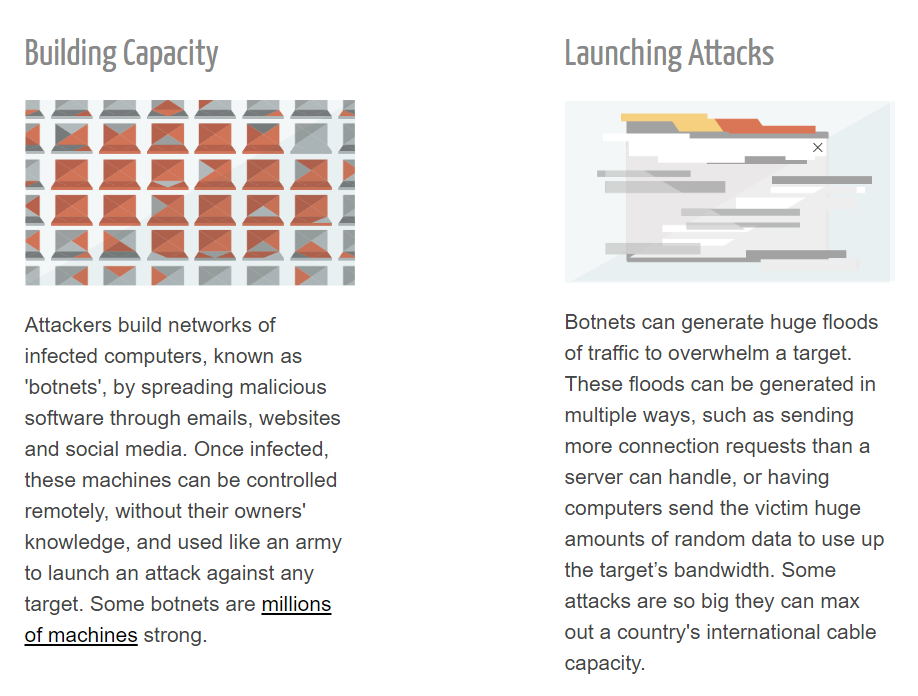


Fig: How DDoS attack works

### **What is a Botnet?**

Botnets are networks made up of remote-controlled computers, or “bots.” These computers have been infected with malware that allows them to be remotely controlled. Some botnets consist of hundreds of thousands — or even millions — of computers.



Fig: Botnets

“Bot” is just a short word for “robot.” Like robots, software bots can be either good or evil. The word “bot” doesn’t always mean a bad piece of software, but most people refer to the type of malware when they use this word.

Botnets can include any number of bots; botnets with tens or countless hubs have turned out to be progressively common, and there may not be a upper limit to their size. Once the botnet is amassed, the adversary can utilize the traffic created by the compromised devices to surge the target domain and force it offline.

## Challenge

It is very challenging for **fake source IP DDoS** to trace back. Also, it is very easy for the detection to be misled **by flash crowd traffic**, which is quite similar to the high-speed Distributed Denial of Service attacks. Accordingly, differentiating between Distributed Denial of Service attacks and Flash Crowd is a hot topic to research. Be that as it may, there are couple of impediments to identify Distributed Denial of Service assaults, including low location rate and high false positive rate. It is dire to propose a way to deal with accurately identify DDoS assaults and channel the malevolent activity in spines before they make drawbacks PCs and servers. It is troublesome for counterfeit source IP Distributed Denial of Service to follow back.

## Goal

To acquire the ideal subset of highlights with a high detection rate. The purpose of this project is following:

* Detect known and unknown attack from the data set.
* To prevent the attacks before it reaches the target ;
* Effectively filter out the random source IP attack;

# CHAPTER 2 PRIOR RESEARCH

## Related work

There are various researches done to prevent Distributed Denial of Service (DDoS) attack as they can cause financial mislays and can also reduce availability of server to the user by overloading host website. Few techniques to mitigate the DDoS attack have been discussed below briefly. We measure accuracy for each model and evaluate which is the best method to prevent DDoS attack.

Accuracy is one metric for evaluating classification models. Informally, **accuracy** is the fraction of predictions our model got right. Formally, accuracy has the following definition:

Accuracy= Number of correct predictions/Total number of predictions

For binary classification, accuracy can also be calculated in terms of positives and negatives as follows:

Accuracy=TP+TNTP+TN+FP+FN

Where *TP* = True Positives, *TN* = True Negatives, *FP* = False Positives, and *FN* = False Negatives.

## DDoS Mitigation in Cloud Services

Dynamic resource allocation procedure is utilized to counter the DDoS attacks. However cloud is as yet helpless against different attacks since cloud stage still runs its servicess in a conventional way. To counteract DDoS attacks, filtration of attacks packets is required. Dynamic resource allocation ought to be done to utilize the set of ideal resources to the cloud clients and guarantee them with the quality of service. The primary issue behind DDoS assaults is the opposition between the assets.

In model proposed in a paper by Preeti Dafu published in WECON 2016 the tool used to beat the DDoS attacks is CloudSim, to prevent on cloud service.

The steps to be followed to mitigate a DDoS attacks or to reduce them are as follows:

1. The start and complete time of the packet is checked by running the ordinary summon.

2. The packets having the expansive information overhead are identified as the assault bundles and are sifted through in the event that they are from the obscure sender.

3. MTTSF (mean time to security failure) is computed and after that elective dynamic designs embraced to moderate those assaults.

4. The assaults bundles are sifted through toward the end.

## DDoS Mitigation using ML

Machine learning based systems use machine learning algorithms or classifiers to learn system normal behaviour and build models that help in classifying new traffic. Machine learning techniques are based on establishing an explicit or implicit model that enables the patterns analysed to be categorized.

As of late, individuals have effectively done heaps of work on recognition. As indicated by the examination of calculations, the discovery can be predominantly separated into three classifications: in light of source end, goal end and center layer. We have arrived at the accompanying conclusions [4]:

* Source end identification: It can adequately recognize the fashioned source IP assaults, and is anything but difficult to follow back the assault have. Be that as it may, it would get a low identification rate, ISP can't profit by it [13, 14].
* Destination end identification: ISP can pick up advantage and it is anything but difficult to send. Be that as it may, a higher false rate, the host has been assaulted before distinguished [15, 16]
* The center layer identification: It can recognize DDoS assaults before the assaults have happened. In any case, numerous examinations neglected to be elevated to the spine organize [17, 18].

At present, calculations of machine learning have been brought into DDoS recognition. Jiang Qi et al. (Jiang Qi, Zhuang Yi, and Xie Dong, "Exploration on age technique for SYN Flood assault identification rules in view of SVM classifier.) Proposed a moment recognition and barrier strategy utilizing SVM, yet the calculation don't accomplish pruning of the choice tree.

## SDN based DDoS mitigation

SDN-Guard, a novel SDN application that secures SDN systems against DoS assaults and relieve their effect on the SDN controller execution, the utilization of the control plane transmission capacity and the switch TCAM use. Not at all like existing works, SDN-Guard is intended to moderate all the while these issues by progressively overseeing stream courses, lead passage timeouts and the total stream administer sections in light of the stream danger likelihood gave by an Intrusion Detection System (IDS).

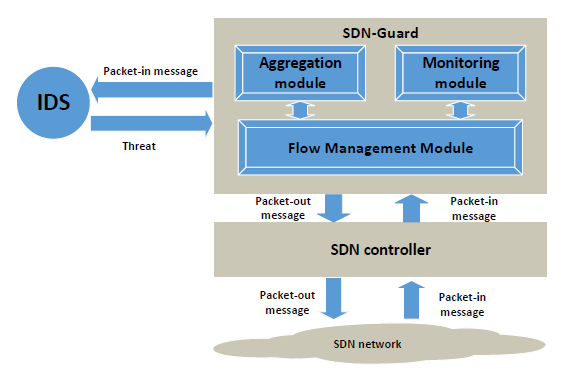
****

Fig: SDN mitigation technique

At the point when a switch gets another stream that can't be coordinated with any administer in its stream table, it asks the controller for a lead keeping in mind the end goal to proficiently forward the stream to its goal. The parcel in messages are forever sent to the IDS to investigate the streams and measure their danger probabilities. The danger likelihood is utilized by the stream administration module to take a choice about the steering of every one of the streams and the timeout for its comparing passages in the switches' TCAMs.

# CHAPTER 3 PROPOSED WORK

There are a few reasons why information mining approach assumes a part in the space of IDS. In the first place, different information should be investigated, which contains verifiable information for characterization of security episodes. It is troublesome for individuals to discover designs in such a lot of information. Nonetheless, information mining appears to be proper to address this issue and along these lines can be utilized to discover this interruption and ordinary examples.

We plan on using Random forest, GradientBoost, Linear Regression, AdaBoost, ExtraTree to train system to get higher (better) rate of detection for DDoS attacks.

The simulated attack comes under following four categories: (check ‘xattack’ column in data file)

* Probing Attack (RPOBE): information gathering attacks
* Users to Root Attack (U2R): unofficial access to local super user or root
* Distributed Denial of Service Attack (DDoS): deny real request to a system
* Remote to Local Attack (R2L): unauthorized local access from a remote machine

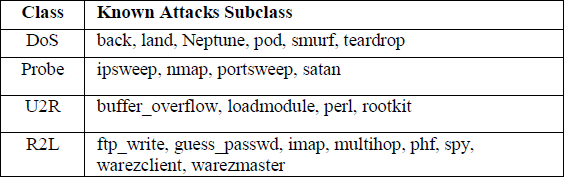


Fig: classes of different attacks

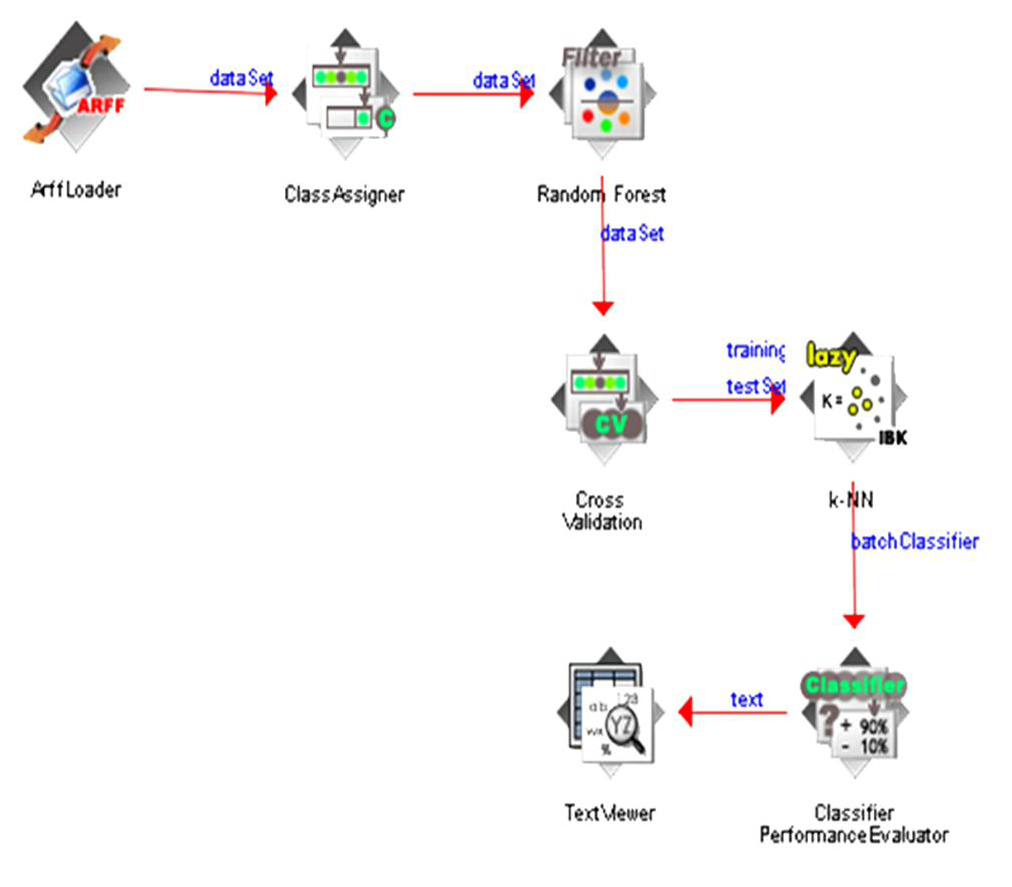


Fig: How model works

The algorithms are implemented in python language, which calls Random forest, GradientBoost, Linear Regression, AdaBoost, Extra Tree library from sci-kit (sklearn) learn library.

## Random Forest

Random Forest Algorithm is truly outstanding among the characterization calculation - can group a lot of information with exactness. Random Forest is a decent device to influence expectations when you to consider that they are not as per the law of vast numbers. The introduction of the right to make their random classifiers appropriate and manageable.

RF uses decision trees, which are very prone to overfitting. In order to achieve higher accuracy, RF decides to create a large number of them based on [bagging](https://en.wikipedia.org/wiki/Bootstrap_aggregating). The basic idea is to resample the data over and over and for each sample train a new classifier. Different classifiers overfit the data in a different way, and through voting those differences are averaged out.

## Gradient Boost Classifier

Gradient boosting is a [machine learning](https://en.wikipedia.org/wiki/Machine_learning) technique for [regression](https://en.wikipedia.org/wiki/Regression_(machine_learning)) and [classification](https://en.wikipedia.org/wiki/Classification_(machine_learning)) problems, which produces a prediction model in the form of an [ensemble](https://en.wikipedia.org/wiki/Ensemble_learning) of weak prediction models, typically [decision trees](https://en.wikipedia.org/wiki/Decision_tree_learning). It builds the model in a stage-wise fashion like other [boosting](https://en.wikipedia.org/wiki/Boosting_(meta-algorithm)) methods do, and it generalizes them by allowing optimization of an arbitrary [differentiable](https://en.wikipedia.org/wiki/Differentiable_function) [loss function](https://en.wikipedia.org/wiki/Loss_function).

GBM is a boosting method, which builds on [weak classifiers](https://stats.stackexchange.com/questions/82049/what-is-meant-by-weak-learner). The idea is to add a classifier at a time, so that the next classifier is trained to improve the already trained ensemble. Notice that for RF each iteration the classifier is trained independently from the rest.

## AdaBoost Classifier

AdaBoost is best used to boost the performance of decision trees on binary classification problems. AdaBoost can be used to boost the performance of any machine learning algorithm. It is best used with weak learners. These are models that achieve accuracy just above random chance on a classification problem. The most suited and therefore most common algorithm used with AdaBoost are decision trees with one level. Because these trees are so short and only contain one decision for classification, they are often called decision stumps.

## Extra Randomized Tree Classifier

The Extra-Trees algorithm builds an ensemble of unpruned decision or regression trees according to the classical top-down procedure. Its two main differences with other tree based ensemble methods are that it splits nodes by choosing cut-points fully at random and that it uses the whole learning sample (rather than a bootstrap replica) to grow the trees.

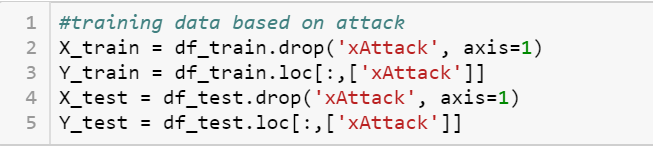
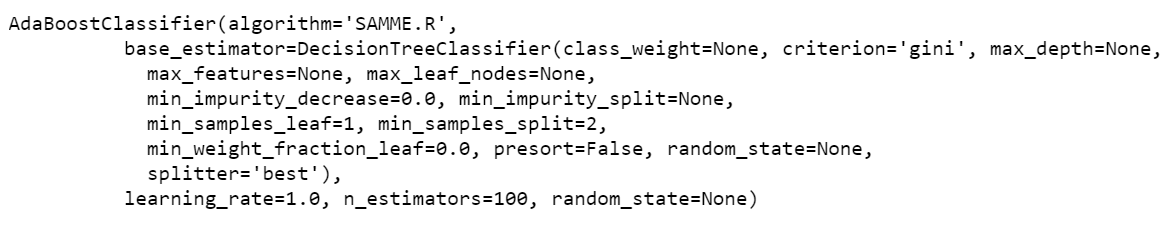
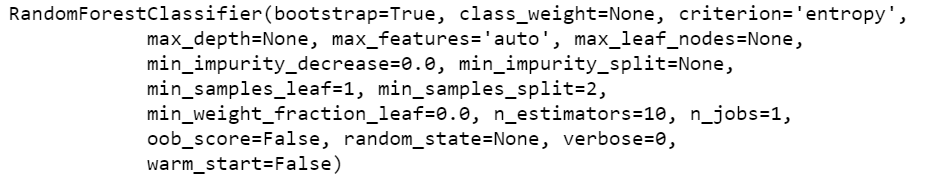
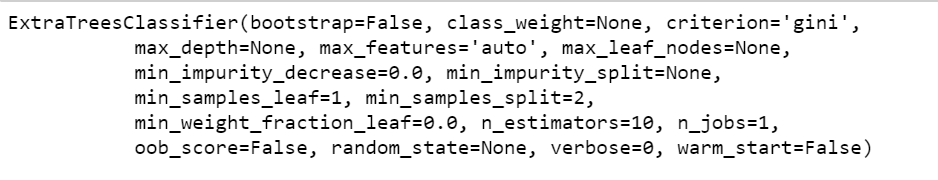


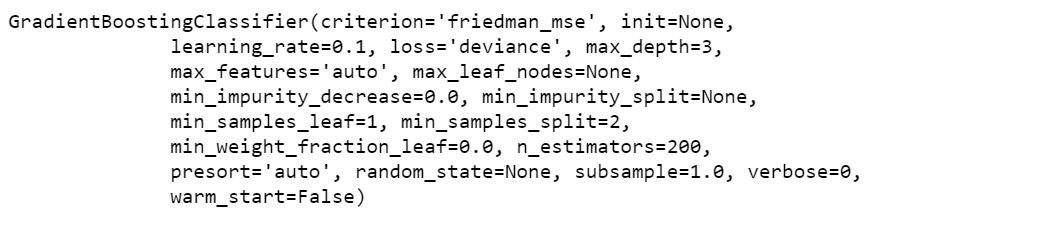
Fig: Training machine

Fig: Sample for Machine learning algorithms executing









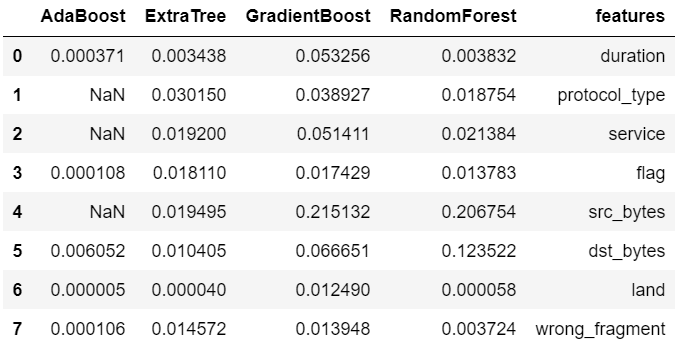


Fig: Results for ML algorithm according to specific features (Ensemble)

The experiment gathered three types of data: real IP DDoS, random IP DDoS and Flash crowd. To investigate the normal for streams, 12 essential stream highlights are extricated to recognize DDoS. (e.g. the likelihood of source IP address, likelihood of goal IP address, length of every bundle, convention compose, add up to parcel byte, the normal parcel byte, fluctuation of bundle byte, standard deviation of bundle byte, normal of data transfer capacity, normal of parcel number, change of parcel number, standard deviation of bundle number, the non-zero part of stream, the greatest piece of stream, and the quantity of bundles in first time section).

Experimental statistics prove that the precision/Accuracy of RDF, GradientBoost, Linear Regression, AdaBoost, ExtraTree algorithm in detecting DDoS attacks in four models and evaluates whose performance is better than other.

Meanwhile, we conclude that it can successfully differentiate random IP address attacks, real IP address attacks and Flash Crowd.

We can extend this experiment by combining those two algorithms; the system may expect to get the more accurate and detection rate to detected intrusion. Example: Random Forest will process in the filtering stage and the SVM will use as a classifier.

(<https://ieeexplore.ieee.org/document/8089926/>).

## NSL-KDD dataset

NSL-KDD is an informational collection recommended to take care of a portion of the inalienable issues of the KDD'99 informational collection which are said in [20]. In spite of the fact that, this new form of the KDD informational collection still experiences a portion of the issues examined by McHugh and may not be an ideal illustrative of existing genuine systems, in view of the absence of open informational indexes for arrange based IDSs, we trust regardless it can be connected as a compelling benchmark informational index to enable specialists to think about various interruption identification techniques.

Moreover, the quantity of records in the NSL-KDD prepare and test sets are sensible. This preferred standpoint makes it moderate to run the trials on the total set without the need to arbitrarily choose a little part. Therefore, assessment consequences of various research work will be reliable and similar. The NSL-KDD data set has the following advantages over the original KDD data set:

1. It does not include redundant records in the train set.
2. The number of records in the train and test sets are reasonable, which makes it affordable to run the experiments on the complete set without the need to randomly select a small portion.
3. There is no duplicate records in the proposed test sets.

## RESULTS

### Model1: Modelling is done with complete feature set

This dataset might be tainted and have errors.

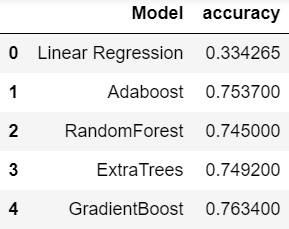
****

Fig: 1st model

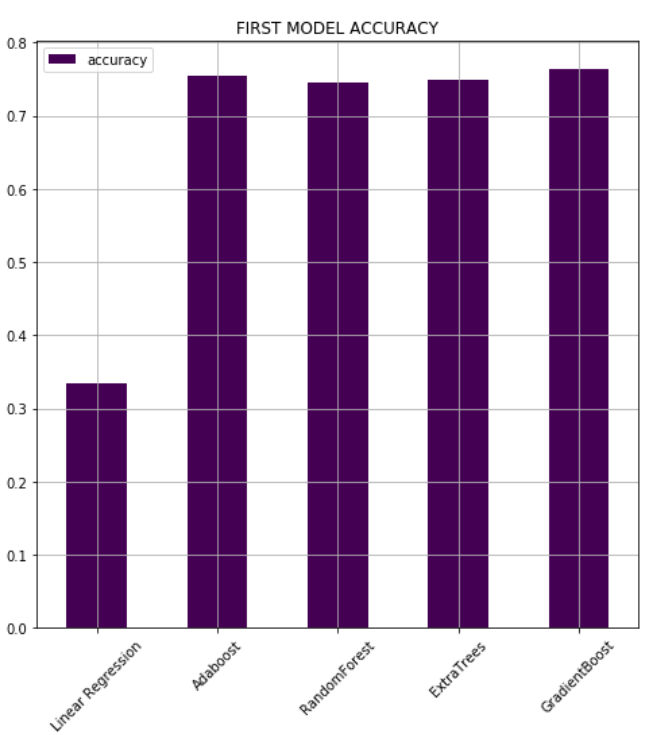
****

Fig: 1st model

### Model 2: Feature set-> features with small deviation removed

Below are the results of training with the exception of the features with small standard deviations.

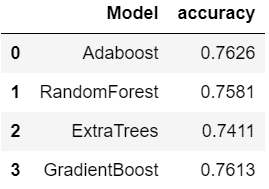
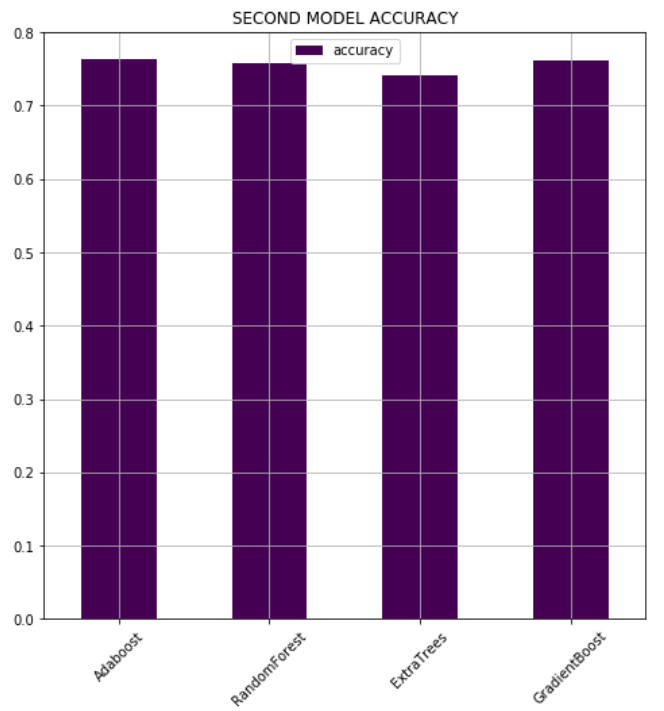
****

Fig: 2nd model

****

### Model 3: Feature set-> without low deviation and high correlation

Heatmap analysis shows that the dependency is high in the following features.

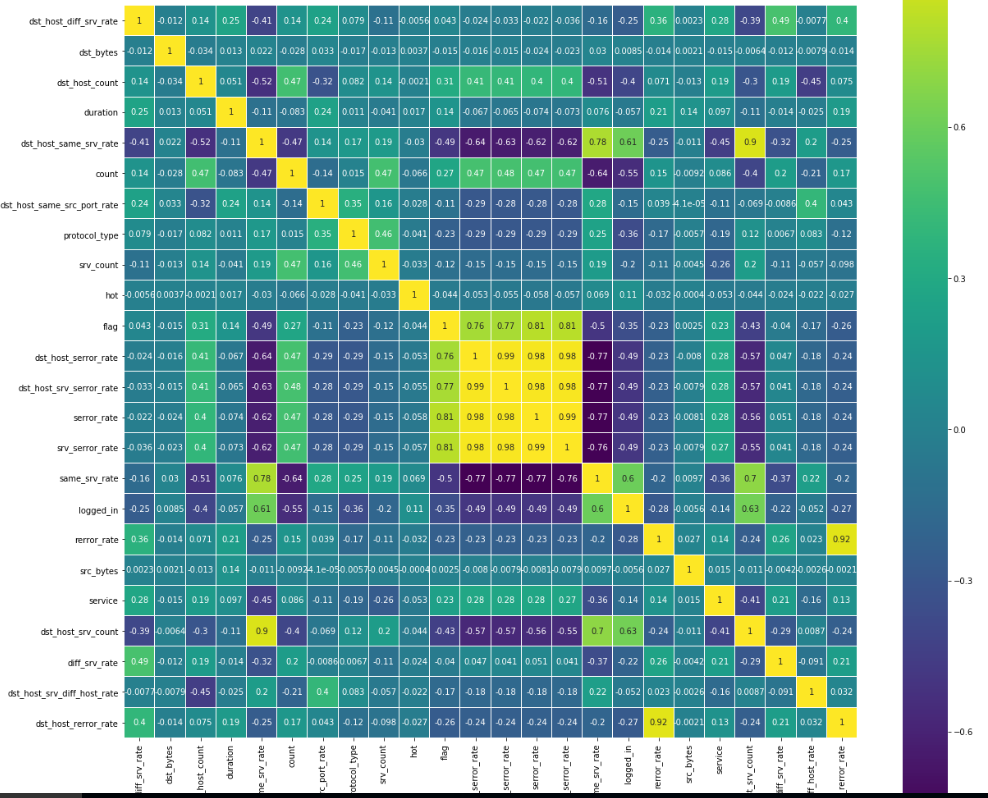
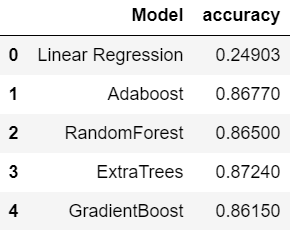


Fig: Heatmap showing correlation in features

Modeling after completing the feature selection process (removing low deviation, high correlation)



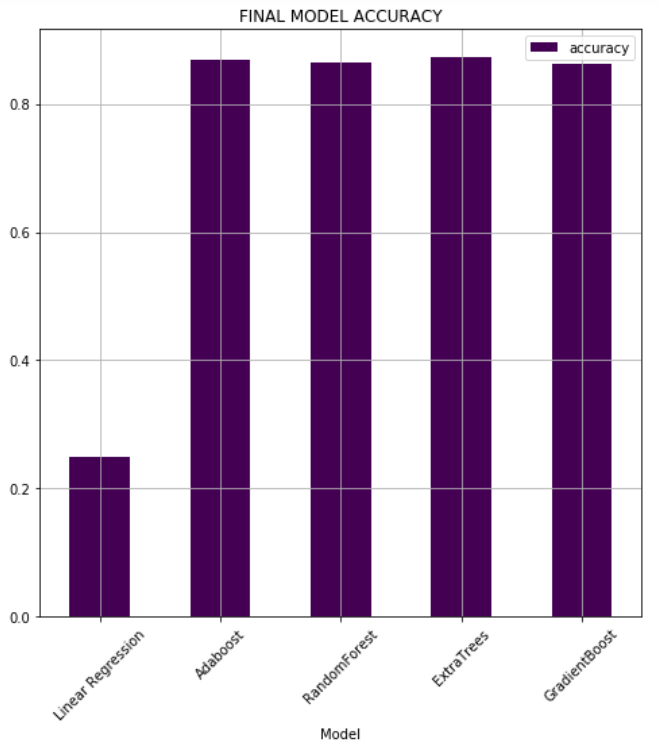
****

Fig: 3rd model

We Observe that Extra Randomized Tree is the most accurate and fastest among the 5 machine algorithms we tested. We achieve accuracy of 87.2 % percent while finding faulty packets. This result can be further boosted by ensembling the dataset more and we can also combine and layer of classifier the result obtained from the first algorithm.

With Extra randomized tree having higher accuracy, we can combine Extra randomized tree algoritm for filtering and further classify the dataset using SVM/Random forest algorithm. The output of the other learning algorithms ('weak learners') is combined into a weighted sum that represents the final output of the boosted classifier.

# Conclusion

This paper, deals with the evaluation of machine learning algorithms for effectively detecting the DDoS attacks. NSL\_KDD data set is used as the attack data and based information gain ranking, relevant features have been selected. Experimental results show that Extra Randomized tree classifier gives better classification.

# Future Scope

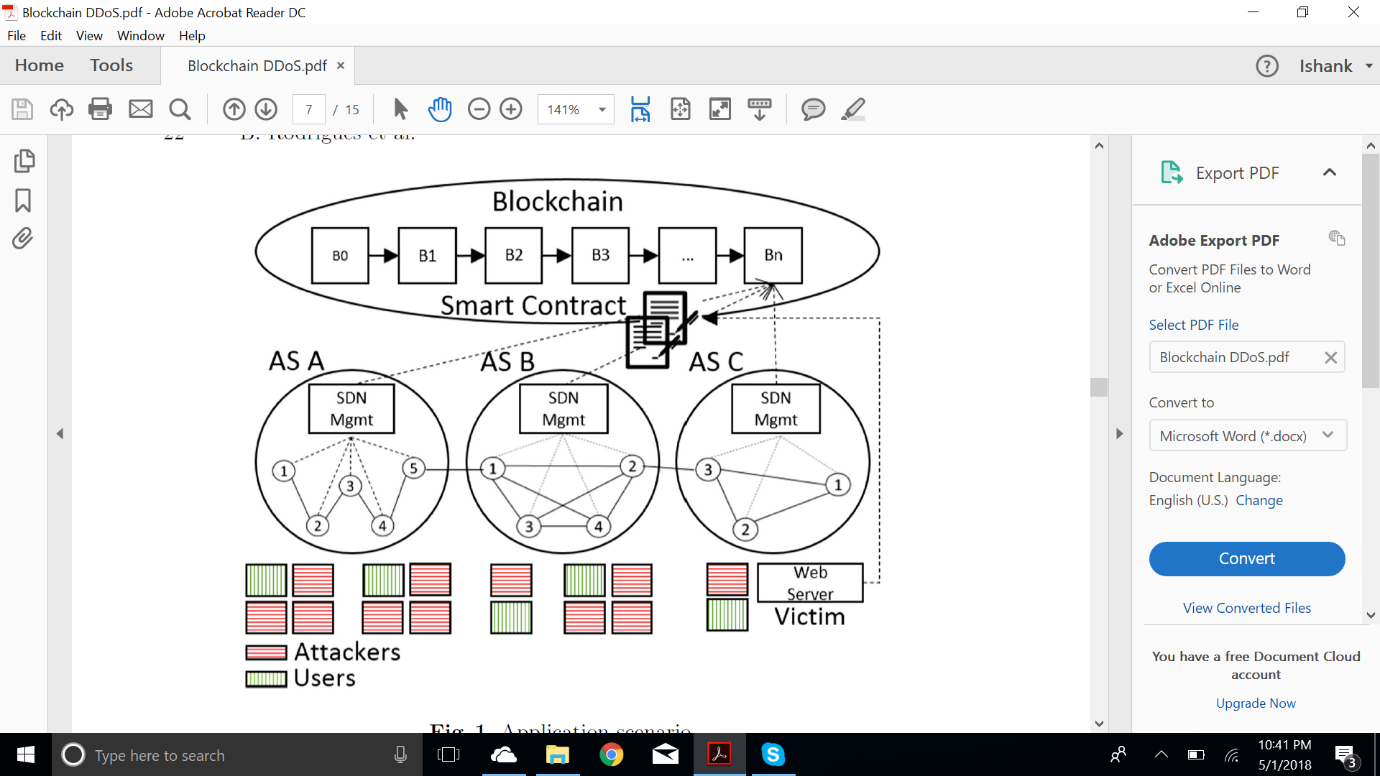
****

Fig: Blockchain Architecture

The objective is to create an automated, and easy-to-manage DDoS mitigation. Three major building blocks are identified to build such a mechanism. Blockchains and Smart Contracts. This approach proposes an engineering and a usage of a way to deal with flagging white or boycotted IP addresses over various areas in light of blockchains and savvy contracts. The upside of utilizing shrewd contracts in a blockchain is: (a) to make utilization of an officially existing framework to circulate rules without the need to fabricate specific registries or other appropriation instruments/conventions, (b) to apply runs over numerous areas, which implies that regardless of whether the AS (Autonomous System) of the casualty isn't mattering these guidelines, some activity can in any case be separated, and (c) the casualty or its AS can control which clients get blocked. The main focal component remaining is to indicate verification of IP proprietorship.

# REFERENCES

1. *Attackers use NTP reflection in huge DDoS attack*, [http://www.computerworld.com/s/article/9246230/Attackers\_use\_NTP\_reflection\_in\_huge\_DDoS\_attack(link is external)](http://www.computerworld.com/s/article/9246230/Attackers_use_NTP_reflection_in_huge_DDoS_attack)
2. *Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing,*[http://tools.ietf.org/html/bcp38(link is external)](http://tools.ietf.org/html/bcp38)
3. *Ingress Filtering for Multihomed Networks,*[https://tools.ietf.org/html/bcp84(link is external)](https://tools.ietf.org/html/bcp84)
4. *Remote Triggered Black Hole Filtering with Unicast Reverse Path Forwarding (uRPF)*, [http://tools.ietf.org/html/rfc5635(link is external)](http://tools.ietf.org/html/rfc5635)
5. [https://www.dhs.gov/trusted-internet-connections(link is external)](https://www.dhs.gov/trusted-internet-connections)
6. https://www.incapsula.com/ddos/ddos-attacks/
7. [www.cs.uml.edu/~hachreka/files/DDOS\_Thesis.pdf](http://www.cs.uml.edu/~hachreka/files/DDOS_Thesis.pdf)
8. <https://pdfs.semanticscholar.org/2f17/87e73b712eb647b463430e46d5fbdab38b58.pdf>
9. McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford,
   1. J., Shenker, S., Turner, J.: Openflow: enabling innovation in campus networks.
   2. ACM SIGCOMM Comput. Communi. Rev. **38**(2), 69–74 (2008)
10. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system (2008)
11. Nishizuka, K., Xia, L., Xia, J., Zhang, D., Fang, L., Gray, C.: Inter-organization
    1. cooperative DDOS protection mechanism. Draft. https://tools.ietf.org/html/
    2. draft-nishizuka-dots-inter-domain-mechanism-02
12. <https://youtu.be/NogCN78XN2w>
13. Petiz, I., P. Salvador, A. Nogueira, and E. Rocha, "Detecting DDoS attacks at the source using multiscaling analysis." Telecommunications Network Strategy and Planning Symposium, 2014, vol.16, pp.1-5.
14. Williamson, M.M, "Throttling viruses: restricting propagation to defeat malicious mobile code." Computer Security Applications Conference, 2002. Proceeding, 2009, pp.61-68.
15. Chen Chen, and Xu Yang, "Research on DDoS based distributed defense system based on SDN." Fujian computer, 2017, vol.33, no.4, PP.13-16.
16. Liu Anli, and Zhao Huaixun, "a fast algorithm for DDoS attack source tracking based on traffic intensity." Modern electronic technology, 2010, vol. 33, no.7, PP.131-134.
17. Yang Jungang, Wang Xintong, and Liu Guqing, "DDoS attack detection methods based on traffic and IP entropy." Computer application research, 2016, vol.33, no.4, PP.1145-1149.
18. Zhao Xin, and Zhang Yu, "Defensive strategies for adaptive DDoS attacks based on border gateways." Intelligent
19. computers and applications, 2011, vol.01, no.3, PP.79-82.
20. M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, “A Detailed Analysis of the KDD CUP 99 Data Set,” *Submitted to Second IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA)*, 2009

