©NIJEP Publications

Open Access
ISSN: 2992-5487

NEWPORT INTERNATIONAL JOURNAL OF ENGINEERING AND PHYSICAL SCIENCES (NIJEP)

Volume 3 Issue 3 2023

https://doi.org/10.59298/NIJEP/2023/10.4.1100

Detecting and Preventing of DDoS Attack in Cloud Computing Environment Based on Hybrid Technique (Cloudflare and WAF)

[1]Joshua John,[2]Okonkwo Obikwelu, [3]Godspower Akawuku and [4]Chika Lilian Onyagu

[1]Institute of Computing & ICT, Ahmadu Bello University, Zaria, Nigeria.
[2,3]Department of Computer Science, Faculty of Physical Sciences, Nnamdi Azikiwe University, Awka, Nigeria.
[4]Department of Cyber Security, Margaret Lawrence University, Umunede Delta, Nigeria.

ABSTRACT

The adoption of cloud computing has revolutionized data storage and processing globally. Nevertheless, the need for a close watch on security is paramount. Hence, the aim of this paper, to develop a cloud security model that detects and prevents the risks of Distributed Denial of Service (DDoS) attacks in cloud computing systems which are gravely potent and, on the increase, today. This was done using two approaches: analyzing Transmission Control Protocol/Internet Protocol (TCP/IP) header features of incoming packets in cloud computing environment in order to detect and classify spoofed IP address during DDoS attack via a custom-made Web Application Firewall (WAF); and the integration of the cloud resources with Cloudflare. The result shows that a total of 1,625,192 packets were transmitted in a short period which were captured and analyzed via Wireshark. Several TCP errors were observed over a very short time interval which indicated successful DDoS attack effectively crashing the system. The result varied when the custom-made WAF was put in place, and the attacking lab machine launched a TCP syn flood attack against the web server on port http port 80. A total of 2,353,585 packets were transmitted in a short period which were captured and analyzed using Wireshark and contained less TCP errors indicating successful mitigation of DDoS attacks. When the resources were hosted online and integrated with Cloudflare, integrity checks were successful before the resources were loaded, indicating complete mitigation of attacks.

Keywords: Bandwidth, Botnet, Cloudflare, Wireshark and Zombie

INTRODUCTION

Cloud computing is a way of accessing compute and storage systems without actually owning and doing active management of the resources. Billions of devices are part of this network and tend to make physical objects, devices and many other deployment areas smarter. Cloud technology had a $543 billion market in 2021. Experts estimate that it will become an $864 billion market by 2025 [1]. The Internet is used by the cloud computing

Open Access
ISSN: 2992-5487

method of IT service delivery to distribute computing resources and software tools. Under this service model, the user simply pays for the time spent using the computer as well as the amount of storage and bandwidth. However, besides the advantages that cloud computing are offering, it comes along with numerous security challenges as well. Increased social dependence on the information and communication technology has resulted in enhanced vulnerability to the plethora of critical cyber oriented attacks. One such attack is the cyber-attack infamously called Distributed Denial of Service (DDoS).

The day when first DDoS attack was launched, an increased annual impact, not only in the number but also in the type and rigorousness of DDoS incidents, has been observed. In fact, nowadays, DDoS attacks are considered to be one of the most severe threats to the stability of the entire Internet, particularly cloud computing. The year 2023 marked a significant rise in attacks, sustaining the growth trajectory from the previous quarter with a 68% Year-over-Year (YoY) increase in DDoS [2, 3, 4]. DDoS attacks can be launched for various reasons ranging from activism to state-sponsored disruption, with many attacks being carried out simply for profit. Hiring services online for DDoS attacks is relatively inexpensive, especially in relation to the amount of damage they can cause. The impact of these attacks are the availability of applications or websites and these attacks can last for hours or even days and block your users from normal usage of your applications. The financial impact on business, this can lead to loss of revenue, increased expenses on IT infrastructure provisioning. DDoS attacks can lead to data losses and reputation of an organization. DDoS attacks can be mainly divided by which layer of the OSI model they attack. DDoS attacks can be mainly divided by which layer of the OSI model they attack: Application-Layer Attacks (layer 7) - HTTP floods, DNS query floods: Composed of requests (HTTP GETs and DNS queries are popular) that are designed to consume application resources (memory, CPU, bandwidth). An example is an attacker who continuously uses a website functionality (submitting a contact form or any API requests) where he knows that it causes database and application processing so that the underlying web service is busy with malicious requests and cannot deliver to other users anymore. State-Exhaustion Attacks (layer 4) - SYN Flood: Consume the TCP connection state tables present in many network infrastructure and security devices, including routers, firewalls, and load-balancers, as well as the application servers themselves. The attacker quickly initiates a connection to a server without finalizing the connection. These attacks can block access for legitimate users or make security devices inoperative, sometimes even leaving defenses wide-open to data exfiltration. Volumetric Attacks (layer 3): Also referred to as Network floods, and includes UDP floods (UDP reflection attacks) and ICMP floods. This type of attack occurs when a network is overwhelmed by a large amount of malicious traffic, causing your applications or services to become unavailable to users. DDoS attacks in cloud environments have been a significant concern due to the proliferation of cloud services and the potential for attackers to exploit the shared infrastructure. All this has been fueled by a rise in botnet usage. This trend heightens the risks for organizations lacking DDoS protection, making them prone to severe and extended outages. A DDoS attack is an attempt to disrupt the regular operation of a system by overwhelming it with traffic. In the case of a cloud environment, this usually takes place by sending thousands upon thousands of connections simultaneously. These requests flood the server and prevent it from processing legitimate requests.

Open Access
ISSN: 2992-5487

Figure 1: DDos Attack Scenario

LITERATURE REVIEW

An in-depth literature review is being conducted in line with the scope of the paper which is detecting and preventing of DDoS attack in cloud computing environment. However, specifically, the literature reviewed were also to help find categorized processes for the detection, prevention and hybrid techniques of averting the calamity. Furthermore, we illustrate the various detection and prevention techniques that are widely used to cater for the DDoS attacks in cloud computing using these literature reviewed for justifications purposes.

Detection Techniques of DDoS Attack in Cloud Computing Environment

Some techniques only perform the process of detection of DDoS attacks while some techniques prevent the cloud networks from being attacked by a DDoS source. However, there are some hybrid techniques as well that are capable of detecting and preventing the cloud network from DDoS attacks. Detection of DDoS attacks can be done by exploiting statistical properties of normal requests and patterns of attack. [5], a statistical model is explained in which a statistical model for normal traffic is fitted and then a statistical inference test is applied to determine if a new instance belongs to this model. Instances that do not conform to the learnt model, based on the applied test statistics, are classified as anomalies. [6] distributed change point (DCP) detection architecture is used along with change aggregation trees (CATs) and non-parametric CUSUM (cumulative sum control chart). When a DDoS flooding attack is being launched, the cumulative deviation is noticeably higher than random fluctuations.

A. Intrusion Detection System

The Most standard feature of IDS is that it is reliable for each virtual machine in cloud environment. This is the method is used for detecting the DDoS attacks [7]. In IDS system, the IDS is used at the cluster controller. And it is applied to each virtual machine and in this way cloud computing platform avoids the overloading problem that could be caused by DDoS attack. And further more advantage of this strategy as described by [8] is the benefit of reducing the impact of the possible attacks by the IDS Sensor VMs.

B. IDS Based DempsterShafer Theory

This technique mainly focuses on detecting and analyzing the Distributed Denial of Service (DDoS) attacks in cloud computing environments. The DDoS attacks mainly targets on cloud service disruptions. [9], a solution is imposed to combine the previous work of Intrusion Detection Systems (IDSs) deployed in the virtual machines of the cloud environment along with a data fusion methodology in the front-end. So when the attacker attacks cloud system, the VM-based IDS will get a warning, which will be stored into the Mysql database or any database that is joined to the cloud system placed within the (CFU) *i.e.* Cloud Fusion Unit of the front-end server. A quantitative solution is proposed for analyzing alerts generated by the IDSs, using the Dempster-Shafer theory (DST) operations in 3-valued logic and the fault-tree analysis (FTA) for the mentioned flooding attacks. At the last step, solution uses the Dempsters combination rule to fuse evidence from multiple independent sources.

C. Packet Information Gathering and Pre-processing

DDos attacks have many categories like Zobbie Cloud Client, related to virtual machines a like hypervisor attack of virtual machine. [10], authors concentrated on detecting the hypervisor attack. When this attack took place it causes the resource imbalance and data loss. The detection procedure is done with packet analysis that consist a packet loader and packet collector. Packet loader stores the files about collected packets using packet capture tool in HDFS. Packet collector performs packet information gathering through Libcap and Jpcap module from live interface.

D. Host Based Intrusion Detection Systems

[11] explained a host-based intrusion detection system *i.e.* HIDS which monitors and analyzes the information collected from a host machine. And the detection procedure follows in which the different type of the information such as system files used, calls of system, type of data etc. Then this detection system observes the modification in the host kernel and also checks for program from the default behaviour then the report of the attack is generated.

E. Network Based Detection System

[12] proposed network detection solution by combining supervised learning technique and unsupervised learning

technique. They used K-Means algorithm for unsupervised learning and Naive Bayes algorithm for supervised learning. The first step of algorithm is using K-Means algorithm to group data to normal or attack. Then, use Naïve Bayes algorithm to classify the obtained result into attack type. The KDD99 dataset was used to evaluate the performance of this algorithm. The detection rate was improved to 99.6 percent. However, this solution is not practical for real network because K-Means algorithm requires more time to process huge data in real networks which could lead to bottleneck problem and system clash.

F. Real-Time Detection System

[13] proposed a real-time detection approach. They used packet sniffer to sniff network packets in every 2 seconds and pre-processed it into 12 features and used decision tree algorithm to classify the network data. The output can be categorized into three (3) types which are DDoS, Probe and normal. The result shows that this algorithm has 97.5 percent of detection rate. The technique is fast and able to use in real network. However, it was not designed to detect unknown attacks.

Prevention Techniques against DDoS Attacks in Cloud Computing Environment

Diverse opinions are raised on the best ways to prevent DDoS attacks in cloud computing environment. Literatures that were leveraged upon covered only four areas as follows:

a. Hop-Count Filtering method

This method uses the relationship of source IP address and TTL value to carry out filtering. The inspection algorithm extracts the source IP address and the final TTL value from each IP packet. The algorithm infers the initial TTL value and subtracts the final TTL value from it to obtain the hop-count. The source IP address serves as the index into the table to retrieve the correct hop-count for this IP address. If the calculated hop-count matches the stored hop-count, the packet has been "authenticated" otherwise; the packet is likely spoofed [14].

b. **CBF (Confidence-Based Filtering) method**

This method focuses our probe on transport and network layers. In order to discriminate attack packets from legitimate ones, this method utilizes correlation patterns. CBF utilizes the attribute value pairs in TCP and IP headers to construct correlation patterns. The concept of correlation refers to the situation that some interior characteristics and there are indeed some unique correlation patterns in legitimate packet flows. In user browsing behaviors, when a person logs on a certain website, his/her focuses tend to make up a certain pattern. For example, since the majority of NBA fans who live in Los Angeles love the team Los Angeles Lakers, the website of ESPN will have more packets containing correlations between visits of Lakers webpage and the IP addresses from the area around Los Angels. Considering that there are a large amount of correlation patterns like this or even more complicate ones, it is quite hard for attackers to notice and mimic these patterns when carrying out DoS or DDoS attacks. The correlation patterns in network and transport layers are the co-appearances between attributes in IP header and TCP header. These attribute pair patterns are distinctive because certain characteristics of the operating system, network structure and even hobbies of users can affect the values of these attributes, and thus make some attribute pairs related. This method uses two concepts: the one named confidence for measuring correlation patterns, and the one named CBF score for judging the legitimacy of packets [15].

c. Port Hopping Technique

This approach is an end point based solution to DoS/DDoS protection, in that changes are made to the servers or clients, but not to the Internet routers. The tests are carried out by the end hosts, and can be conducted at the network layer (IP), transport layer (TCP) application layer. The PRNGs are algorithms that use mathematical formulae or simply precalculated list of tables to produce sequences of numbers that appear randomly. A good example of a PRNG is the linear congruential method. In this scheme, different port numbers are used in different time slots for the same communication service. Let Pi represents the port number used by the server in time slot Si. k is a shared cryptographic key between the server and the client communication and is a pseudo-random number generator. When a client needs to communicate with the server, it will identify the server's current port number Pi using the shared secret key k and the time slot number i. When the server receives packets of data that carry "invalid" port numbers, they can be easily detected and filtered off. There is no need for the server to examine the contents of the packets in order to identify if a packet is malicious. As a result, the computational resources needed to detect and filter off the malicious data packets is reduced [16].

d. Ingress/Engress Filtering

Ingress Filtering, proposed by Ferguson et al., is a restrictive mechanism to drop traffic with IP addresses that do not match a domain prefix connected to the ingress router. Egress filtering is an outbound filter, which ensures that only assigned or allocated IP address space leaves the network. A key requirement for ingress or egress filtering is knowledge of the expected IP addresses at a particular port [17].

Hybrid Techniques against DDoS Attacks

The motivation of using hybrid approach is to detect and defend DDoS attacks and other network level malicious activities in cloud computing, the use of only traditional IDS/IPS techniques such as signature based detection,

Open Access
ISSN: 2992-5487

anomaly detection and firewall is not an efficient solution. In this research work, we propose a hybrid network intrusion detection and prevention system on network attacks in the cloud environment by monitoring network traffic, while maintaining performance and service quality (Feature Extraction and Selection Method). This helps to reduce computation cost for detecting intrusions at other servers, and improve detection rate in overall cloud environment. [18] suggested the DDoS attack prediction using a hybrid deep-learning (DL) model, namely, a CNN with BiLSTM (bidirectional long/short-term memory), in order to effectively anticipate DDoS attacks using benchmark data from other models. By ranking and choosing features that scored the highest in the provided data set, only the most pertinent features were picked. Experiment findings demonstrate that the proposed CNN-BI-LSTM attained an accuracy of up to 94.52 percent using the data set CICDDoS2019 during training, testing, and validation. Using hybrid techniques, [19], proposed a hybrid detection method to DDoS attack, using entropy-based methods and custom-tailored methods. Custom tailored methods can only be used for the ones they are specially designed whereas anomaly-based methods can detect a wide range of anomalies and attack. The proposed hybrid method is combination of both feature-based and volume-based detection along with high and low rate attacks which gets a comparison between proposed and previously known methods. Some of the most common DDoS attacks are ICMP flood, SYN flood, DNS amplification attacks and the earlier Smurf Attack and Fraggle Attack. A script file was developed to test the following attacks: ICMP flood attack with a high packet rate attack on the specified target. Using this approach, a real-life botnet is simulated. As in a real-life situation, the attack does not start with all attackers at the same time, but instead with attackers initiated after a random delay period. [20], proposed an ML-based model for the detection of DDoS attacks. The authors applied three different machine learning models: K-Nearest Neighbors (KNN), Random Forest (RF) and Naive Bayes (NB) classifier. The proposed approach can detect any type of DDoS attack in the network. The results of the proposed approach showed that the model can detect attacks with an average accuracy of 98.5%. Earlier, [21] designed a technique for identifying cloud computing DDoS attacks. This technique employs machine learning algorithms such as support vector machine (SVM), naive Bayes (NB), and random forest (RF) for classification. The study was carried out using Tor Hammer as an attacking tool on a cloud environment, and a new dataset for the intrusion detection technique was developed. [22], proposed a hybrid model of intrusion detection system in a cloud computing environment. The model detects violations, using the IP (Internet protocol)/MAC (Media access control) address at the point of entering the network of a cloud based system. The methodology adopted is Object-Oriented Hypermedia Method (OOHDM) and the programming languages used is PHP, JavaScript, CSS and MySQL. The new system brings about a new method of detecting intruders by the combined use of IP/MAC address. [23] used Neural Networks and Data mining technique to detect DDOS attacks. This model needs less memory and claims that they have faster detection. The result shows that most of TCP attacks are detected. This system help in detecting layer 7 (Application Layer) attacks and carry a lot of overhead. TTL value low and slow attacks. Packet monitoring in the cloud for TTL value said to have greater advantages in detecting DDOS attacks in the cloud, but it slows down the system performance which creates a limitation for cloud service providers. [24] proposed a hybrid deep-learning technique that utilizes two deep neural network models for effective feature extraction and accurate DDoS attack detection and classification without human intervention. Moreover [25], developed an amalgam network primarily based interruption to find cloud DDoS incursions. SNORT, associate degree open supply signature primarily based discovery technique which keeps guidelines of notable DDoS incursions styles, and theorem classifier, an applied mathematics categorizer which forecasts the likelihood of a network incident to a category either traditional or malevolent by high precision.. Eucalyptus, associate degree open supply cloud was utilized in this experimental system wherever invasion discovery structure was put in every node controller with each port opened for verification purpose. In obtaining custom packets, Scapy was used whereas performance and feature were assessed exploitation KDD '99 data set.

METHODOLOGY

The paper adopted to use the OSSTMM methodology (Open Source Security Testing Methodology Manual) which provides a scientific methodology for network penetration testing and vulnerability assessments. This methodology allows testers to customize their assessment to fit the specific needs or the technological context of the organization, thereby, obtaining an accurate overview of the organization network's cybersecurity, as well as providing reliable solutions that can be adapted to the organization's technological context to help make the right decisions to secure the network(s).

Justification for the System

Cloud computing has introduced a host of new security threats and challenges. With so much data going into the cloud and into public cloud services in particular, these resources become natural targets for bad actors. When attempts by malicious users become successful, so much loss (data, money, time, resources, etc) could be incurred by the organization. Having a system that will prevent or thwart these bad actors from gaining access into an organization's cloud based systems is not only required but necessary. For sure, so many systems are in the

Open Access
ISSN: 2992-5487

market, and many more are being developed, but these come with some weaknesses as highlighted in this work. While most security systems work on either one or two layers of the OSI, this current study strengthens security checks and surveillance at three layers of the OSI, namely, network, transport and application. Hence, when a threat escapes one layer, it is most likely going to be trapped in the other layers.

Architecture of the Design

Figure 2 gives the architecture of the system. Entry into the system either by a real user or an attacker is by the use of a computing device via the cloud. Access can be granted either by passing through a VPN or through Cloudflare/Web Application Firewall. In a sharp contrast to the existing system, no IP address is blacklisted or whitelisted, but rather action is based on the scrutiny of either Cloudflare or Web Application Firewall. However, a dedicated IP address (or a block of it) is configured for secured VPN access. It is only after Cloudflare/WAF and/or VPN has certified the genuiness of the request(s) that such is passed via the cloud to the location of the intended resource (Webserver, File Server, etc).

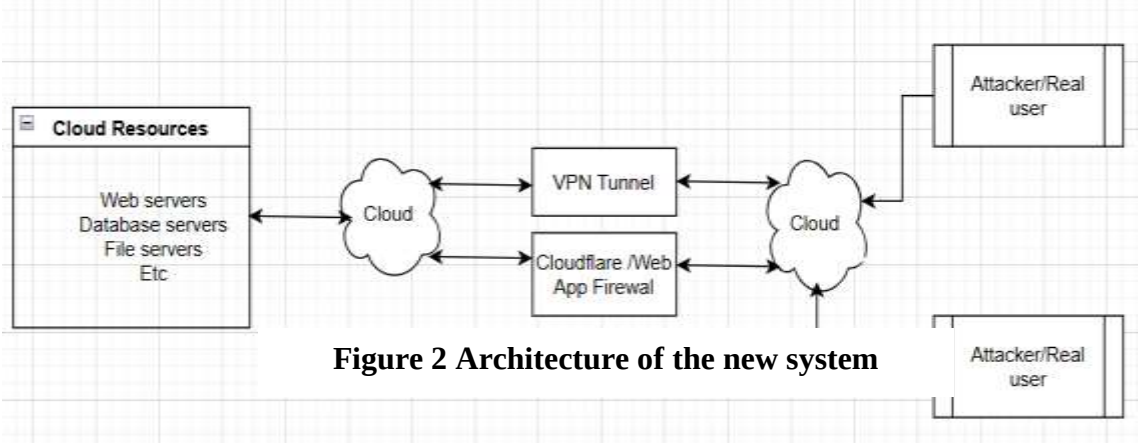

Figure 2 Architecture of the new system

System Testing

Test Plan

To test the model on real-time network traffic, a simulation environment was set-up within Linode. The application captures real-time network traffic and predicts the presence or absence of an attack based on the user-selected model through command line interface. The set-up consisted of a 64-bit, 2GB RAM, Debian 10 Web Server, a VPN Server to facilitate access to the cloud-based lab, A 64-bit, 4GB Kali Linux and Windows 10 Hosts to send attack traffic. The attacking hosts are connected to simulation environment via VPN tunnel and the Debian 10 Web Server and VPN server reside within the same data center. The set-up is shown in Figure 3. This was done to create a Virtual Private Network (VPN) to contain the attack simulations within a cloud environment.

Open Access
ISSN: 2992-5487

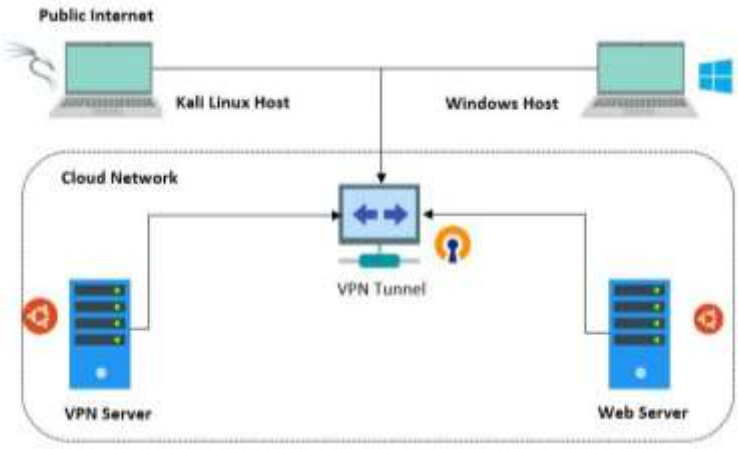

Figure 3: Simulation Testbed Setup Diagram

Test Data

Each of the four machines was assigned an internal IP address by the VPN Server. This is shown in Table 1.

Table 1: Internal IP addresses assigned by the VPN Server to the machines used in the simulation

Host and Servers	Assigned IP
Debian Web Server	192.168.10.2/24
VPN Server	192.168.10.3/24
Kali Linux (Host)	192.168.10.15/24
Windows 10 (Host)	192.168.10.16/24

The Debian Web Server is the host running the DDoS Detection and Prevention tool whereas the Kali Host runs Hping3 and Windows 10 Host ran a software called Low Orbit Ion Cannon (LOIC) to send UDP/TCP/HTTP attack traffic to the Debian Web Server. Low Orbit Ion Cannon (LOIC) is an open-source network stress testing and denial-of-service attack application, written in C#. LOIC can be used to perform both DoS and DDoS attack (depending on the number of people using the software to target the same server) on a target by flooding it with TCP, UDP or HTTP packets with the aim of disrupting the services and also the use of bash programming to simulate the attack.

Open Access
ISSN: 2992-5487

The LOIC application is shown in Figure 4.

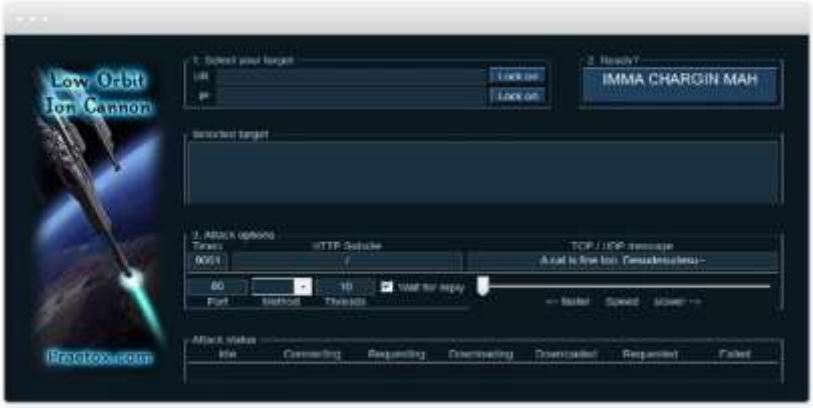

Figure 4: Low Orbit Ion Cannon (LOIC) application.
(Retrieved January 10, 2021 from https://www.imperva.com/learn/ddos/low-orbit-ion-cannon/)

System Flowchart: Figure 5 presents how the hybrid architecture is designed to monitor, detect, analyse and mitigate against multiple DDOS attack techniques targeting critical information system infrastructure.

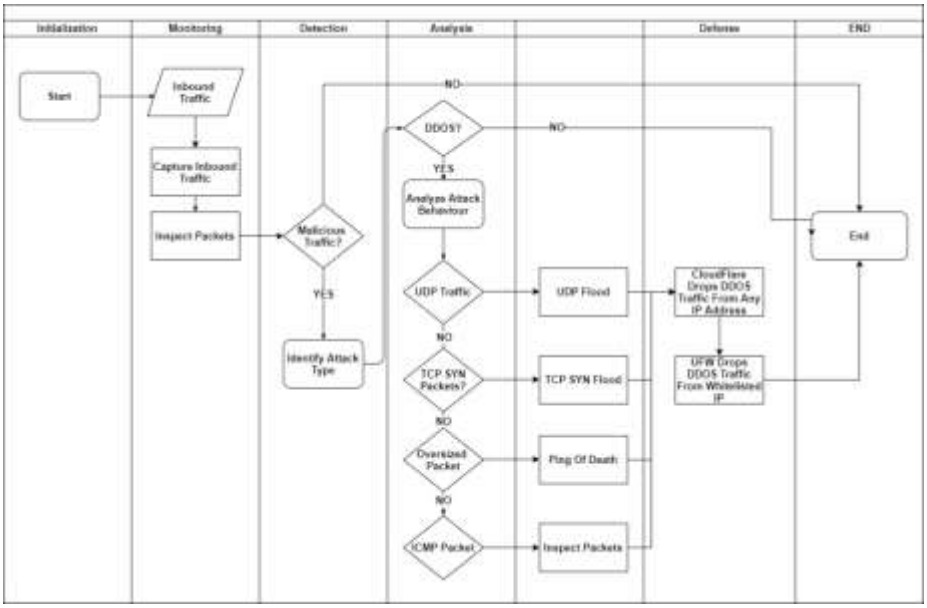

Figure 5: System Flowchart

System Implementation

The Cloudflare firewall setting is 'Managed Challenge' which has been implemented in this design. It manages challenges automatically as they arise unlike the 'Legacy CAPTCHA' rule. On receiving a request from a client browser, it performs necessary security checks to confirm that all is well before redirecting traffic.

Open Access
ISSN: 2992-5487

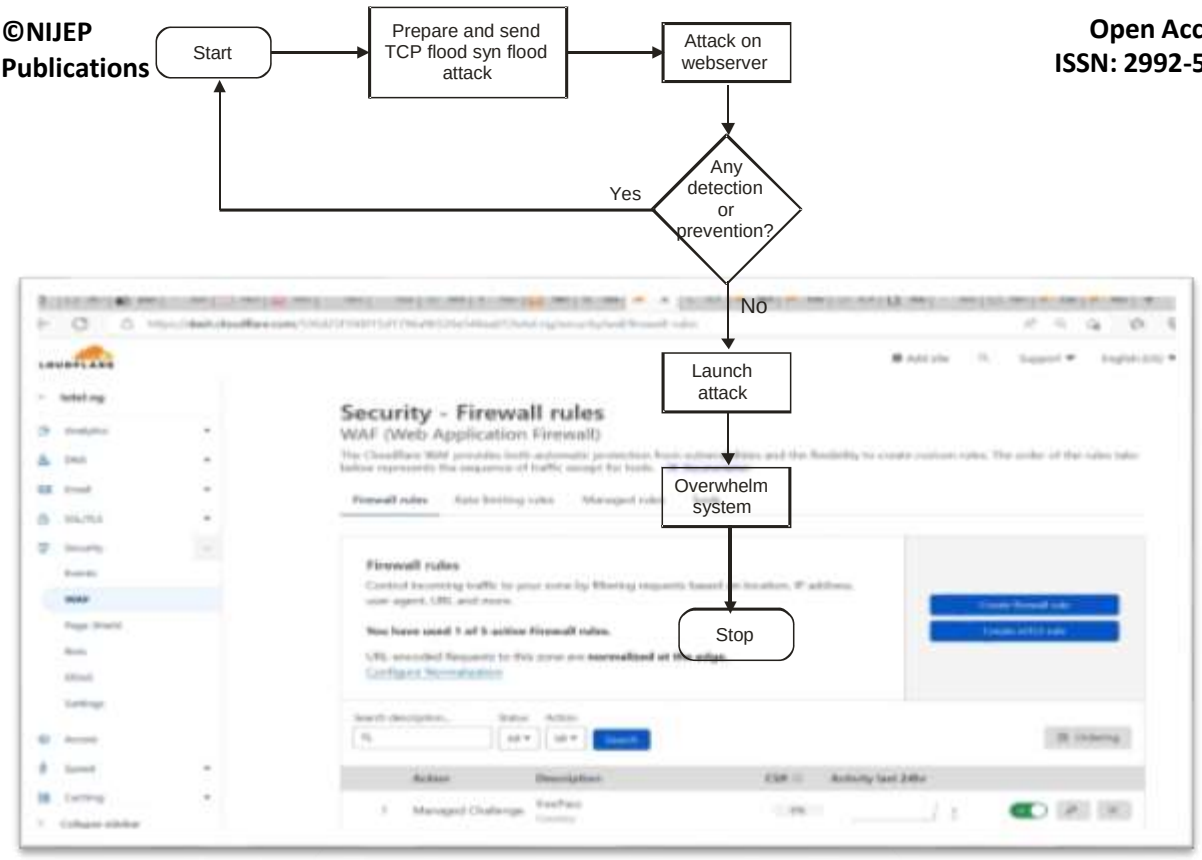

Figure 6: Cloudflare security firewall settings

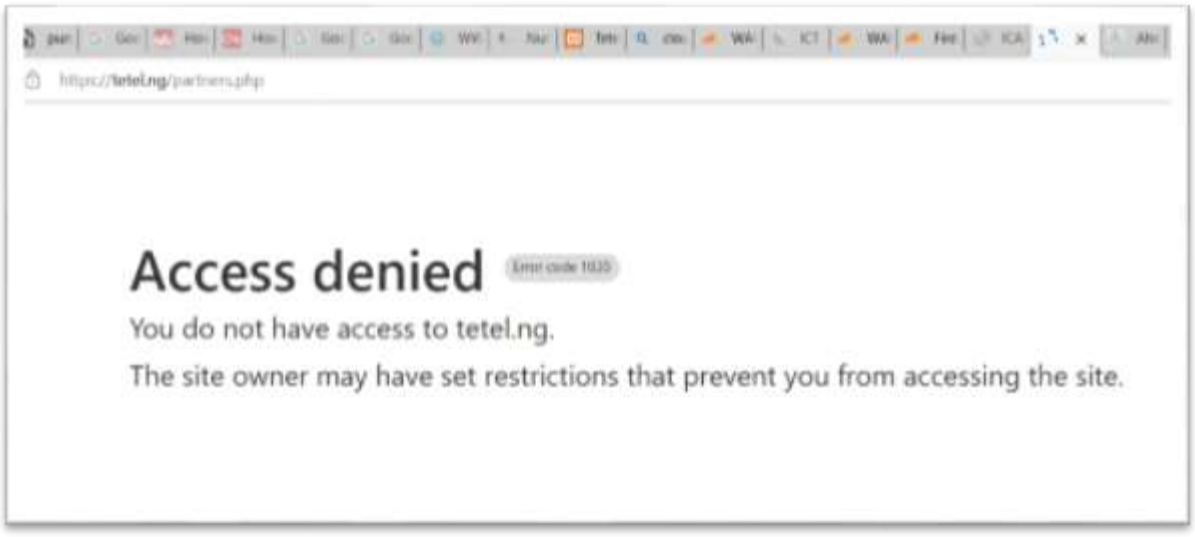

Figure 7: Blocked request to tetel.ng via Clouflare WAF

Attacks through vulnerable host

The model was set up in such a way that TCP syn flood attacks were targeted to a particular webserver (41.33.72.47) on port 80 through an attacking lab machine using hping3. This is depicted in the flowchart shown in Figure 6 while the command line for the attack is shown in Figure 7.

Open Access
ISSN: 2992-5487

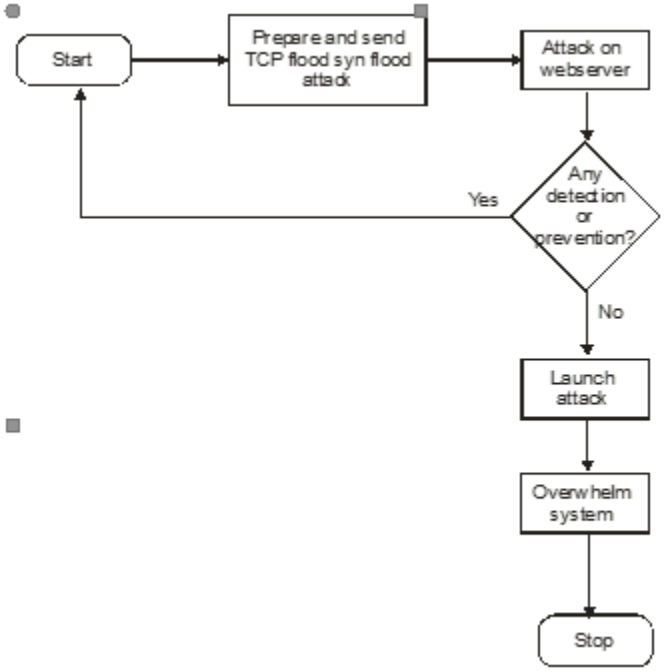

Figure 8: Flow chart of the attack on vulnerable host

```
oluwatobi@kali:~$ sudo hping3 -S --flood -V -p 80 45.33.72.47
[sudo] password for oluwatobi:
using eth0, addr: 45.33.100.49, MTU: 1500
HPING 45.33.72.47 (eth0 45.33.72.47): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 45.33.72.47 hping statistic ---
1625192 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

Figure 9: Hping3 TCP SYN Flood Attack on an Unsecured Targeted Host

This machine did not have any prevention/detection mechanism against these attacks. There was 100% packet loss as no replies were received, effectively overwhelming the system. A total of 1,625,192 packets are transmitted in a short period. Without any firewall protection, on the test victim machine, tcpdump is used to capture packets on the eth0 interface (Figure 10). The result of the capture is saved into a pcap format file named http_ddos.pcap

```
root@localhost:~# sudo tcpdump -w http_ddos.pcap -i eth0
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes

^C2555779 packets captured
2556096 packets received by filter
0 packets dropped by kernel
```

Figure 10: tcpdump of Flood Attack on an Unsecured Targeted Host

Using Wireshark, the file is analyzed to show the distribution of traffic recorded over time, it can be observed that there was a lot of TCP errors within a short time interval (Figure 11); this is an indication of a successful DDoS attack.

Open Access
ISSN: 2992-5487

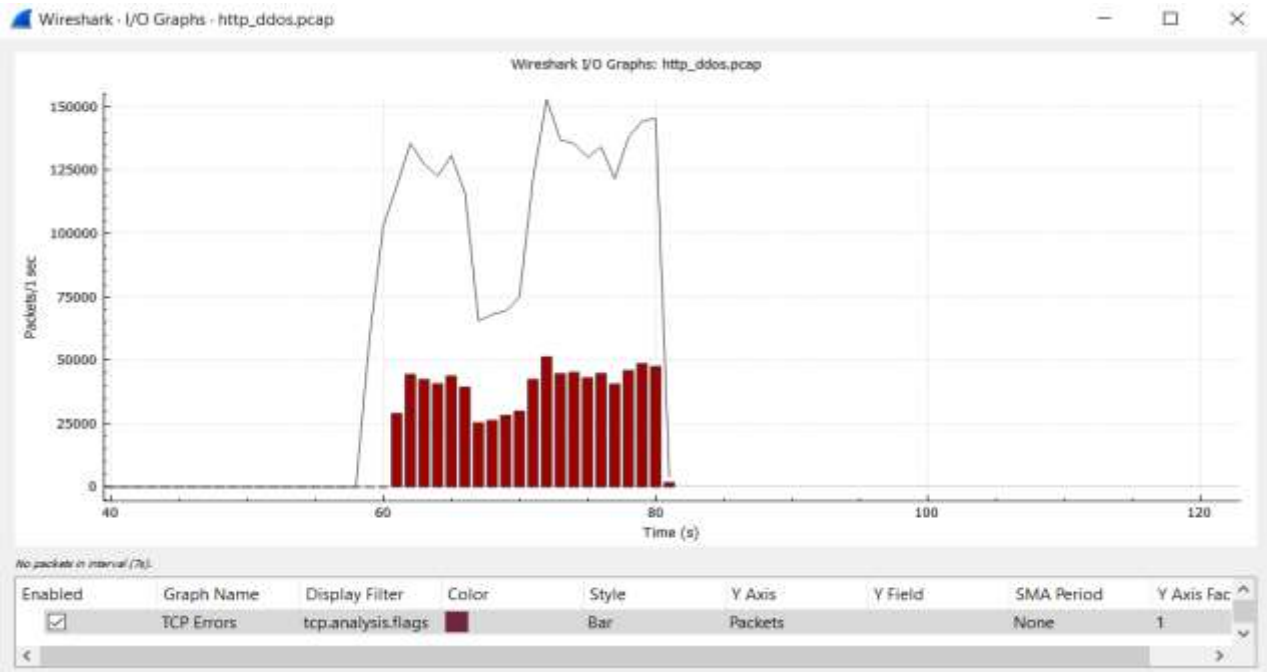

Figure 11: Distribution of traffic over time on an Unsecured Host

The attacking lab machine using **hping3** launches a TCP syn flood attack against the web server on port http port 80. A total of **2,353,585 packets** were transmitted in a short period.

```
oluwatobi@kali:~$ sudo hping3 -S --flood -V -p 80 45.33.72.47
using eth0, addr: 45.33.100.49, MTU: 1500
HPING 45.33.72.47 (eth0 45.33.72.47): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 45.33.72.47 hping statistic ---
2353585 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

Figure 12: Hping3 TCP SYN Flood Attack on a Secured Targeted Host

With a firewall protection, on the test victim machine, tcpdump is used to capture packets on the eth0 interface. The result of the capture is saved into a pcap format file named firewall_http_ddos.pcap

```
oluwatobi@localhost:~$ sudo tcpdump -w firewall_http_ddos.pcap -i eth0
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
^C60 packets captured
62 packets received by filter
0 packets dropped by kernel
```

Figure 13: tcpdump of Flood Attack on Secured Targeted Host

Using Wireshark, the file is analyzed to show the distribution of traffic recorded over time, it can be observed that there was less of TCP errors within a short time interval, this is an indication of a successfully mitigating the DDOS attack with a network and application layer firewall.

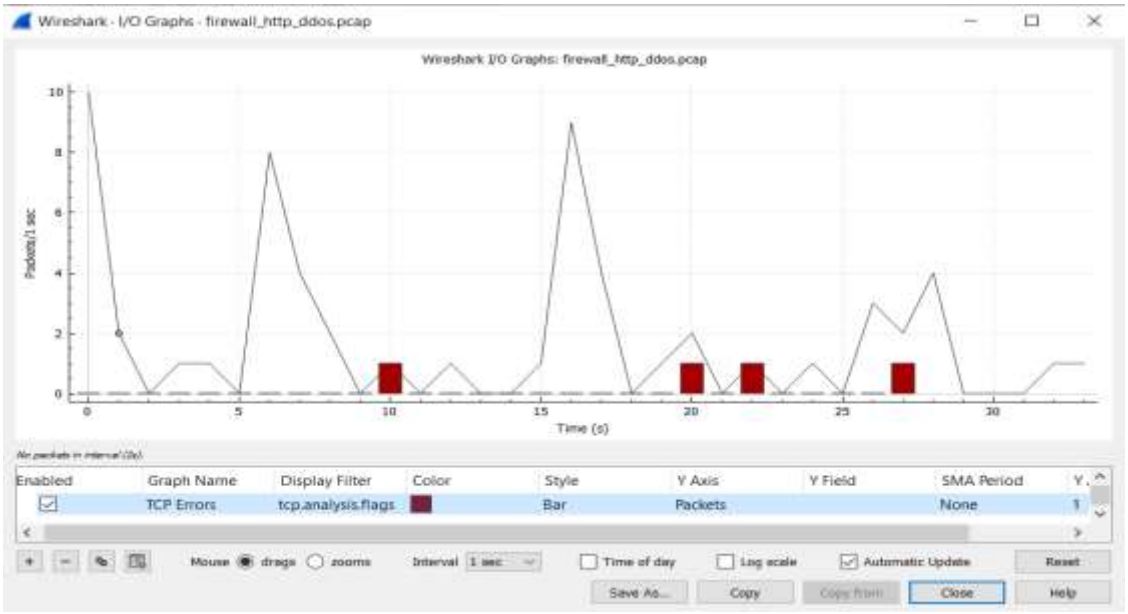

Figure 14 Distribution of traffic over time on a Secured Host

RECOMMENDATION

Cloud computing comes with a lot of security risks. It is recommended that the model developed in this study be used by cloud based companies to safeguard their infrastructure, platform and software. It is further recommended that further studies be conducted in this area due to ever-increasing threats and vulnerabilities evident in computing environment to further mitigate DDoS attacks.

REFERENCES

1. Nancy Pais, (2023). Future of Cloud Computing 2025: 10 Trends and Predictions. Available on https://www.31west.net/blog/future-of-cloud-computing/ Retrieved on 25th September 2023
2. StormWall, (2023). DDoS Attack Statistics by Country in Q2 2023. Available on https://stormwall.network/ddos-report-stormwall-q-2-2023 Retrieved 25th September, 2023.
3. Val Hyginus U. Eze, Chinyere Nneoma Ugwu and Ifeanyi Cornelius Ugwuanyi (2023). A Study of Cyber Security Threats, Challenges in Different Fields and its Prospective Solutions: A Review. INOSR Scientific Research 9(1):13-24. http://www.inosr.net/wp-content/uploads/2023/02/INOSR-SR-9113-24-2023..pdf
4. Eze, M. C., Eze, V. H. U., Chidebelu, N. O., Ugwu, S. A., Odo, J. I., & Odi, J. I. (2017). NOVEL PASSIVE NEGATIVE AND POSITIVE CLAMPER CIRCUITS DESIGN FOR ELECTRONIC SYSTEMS. *International Journal of Scientific & Engineering Research*, *8*(5), 856–867.
5. Bhuyan, M. H., Kashyap, H. J., Bhattacharyy, D. K., & Kalita, J. K. (2013). Detecting Distributed Denial of Service Attacks: Methods, Tools and Future Directions at Colorado Springs.
6. Chen, Y., Hwang, K., & Ku, W. S. (2006). "Distributed change-point detection of DDoS attacks over multiple network domains", *Proceedings of the IEEE International Symposium on Collaborative Technologies and Systems*, Las Vegas, NV, IEEE CS, vol. 14-17, pp. 543–550.
7. Lonea, A. M., & Popescu, D. E. (2013). "Tianfield Detecting DDoS Attacks in, *Cloud Comp Int. Journal Comput Commun*, ISSN 1841-9836 8, no. 1,(2013), pp. 70-78.
8. Roschke, S., Cheng, F., & Meinel, C. (2009). "Intrusion detection in the cloud," in *Dependable, Autonomic and Secure Computing, 2009. DASC'09. Eighth IEEE International Conference on*, pp. 729-734.
9. Lo, C. C., Huang, C. C., & Ku, J. (2010). "A Cooperative Intrusion Detection System Framework for Cloud Computing Networks", *In 39th International Conference on Parallel Processing Workshops*, pp. 280-284.
10. Choi, J., Choi, C., Ko, B., Choi, D., & Kim, P. (2013). "Detecting Web based DDoS Attack using MapReduce operations in Cloud Computing Environment". *Journal of Internet Services and Information Security (JISIS)*, volume: 3, number: 3/4, pp. 28-37.
11. Modi, C., Patel, D., Borisaniya, B., Patel, H., Patel, A., Rajarajan, M., & Gujarat, N. S. (2013). Survey of intrusion detection "A techniques Elsevier in *Journal Cloud of Network and Computer Applications*, vol. 36, (2013), pp. 42– 57.
12. Muda, Z., Yassin, W., Sulaiman, M.N. & Udzir, N. I. (2011). "I and Naïve Bayes classification", *7th*

Open Access
ISSN: 2992-5487

International Conference, on Emerging Convergences and Singularity of Forms (CITA).

13. Komviriyavut, T., Sangkatsanee, P., & Wattanapongsakor, N. (2009). "Detection and classification with decision tree and r on Communications and Information Technology (ISCIT), pp. 1046-1050.
14. Chouhan, V. and Peddoju, S. K. (2012). "Packet Monitoring Approach to Prevent DDoS Attack in Cloud Computing," no. 2315, pp. 38–42.
15. Dou, W., Chen, Q. and Chen, J. (2013). "A confidence-based filtering method for DDoS attack defense in cloud environment," Futur. Gener. Comput. Syst., vol. 29, no. 7, pp. 1838–1850.
16. Siva, T., Krishna, E. S. P., Vidyanikethan, S. and Dist, C. (2013). "Controlling various network based A DoS Attacks in cloud computing environment: International Journal of Engineering Trends and Technology (IJETT), vol. 4,no. 5 pp. 2099–2104.
17. Ankali, S. B. (2011). "Detection Architecture of Application Layer DDoS Attack for Internet," vol. 990, pp. 984–990.
18. Gadze, J. D., Bamfo-Asante, A. A., Agyemang, J. O., Nunoo-Mensah, H. and Opare, K. A. B. (2021). "An investigation into the application of deep learning in the detection and mitigation of DDOS attack on SDN controllers," *Technologies*, vol. 9, no. 1, p. 14.
19. Bojović, P.D., Bašičević, I., Ocovaj, S. & Popović, M. (2019). "A practical approach to detection of distributed denial-of-service attacks using a hybrid detection method". Computers & Electrical Engineering, 73, pp.84-96.
20. Priya, S.S., Sivaram, M., Yuvaraj, D., & Jayanthiladevi, A. (2020). "Machine learning based DDoS detection". In Proceedings of the 2020 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 12–14; pp. 234–237.
21. Wani, A.R., Rana, Q.P., Saxena, U., & Pandey, N. (2019). "Analysis and Detection of DDoS Attacks on Cloud Computing Environment using Machine Learning Techniques". In Proceedings of the Amity International Conference on Artificial Intelligence (AICAI), Dubai, United Arab Emirates, pp.4–6.
22. Ogbomo-Odikayor. I. F., Anigbogu. S.O., Edebeatu Dom & Anigbogu, G.N. (2018). "A hybrid model of intrusion detection system in a cloud computing environment," International Research Journal of Advanced Engineering and Science, Volume 3, Issue 3, pp. 194-200.
23. Khalid, A. F. (2016). "An Overview of DDOS Attacks Detection and Prevention in the Cloud". International Journal of Applied Information Systems (IJAIS) – ISSN: 2249-0868 Foundation of Computer Science FCS, New York, USA Volume, pp.11 – No. 7 – www.ijais.org
24. Wei Y., Jang-Jaccard J., Sabrina F., Singh A., Xu, W. and Camtepe, S. (2021). "AE-MLP: a hybrid deep learning approach for DDoS detection and classification," IEEE Access, vol. 9, pp. 146810–146821.
25. Modi, C. N., Patel, D. R., Patel, A. and Muttukrishnan, R. (2012). "Bayesian Classifier and Snort based network intrusion detection system in cloud computing," 2012 Third Int. Conf. Comput. Commun. Netw. Technol., vol. 39, no. July, pp. 1–7.

CITE AS: Joshua John, Okonkwo Obikwelu, Godspower Akawuku and Chika Lilian Onyagu (2023). Detecting and Preventing of DDoS Attack in Cloud Computing Environment Based on Hybrid Technique (Cloudflare and WAF). *NEWPORT INTERNATIONAL JOURNAL OF ENGINEERING AND PHYSICAL SCIENCES (NIJEP)* 3(3)28-40. https://doi.org/10.59298/NIJEP/2023/10.4.1100