Nigerian Journal of Technology (NIJOTECH)
Vol. 44, No. 1, March, 2025, pp.105 - 113
www.nijotech.com

Print ISSN: 0331-8443
Electronic ISSN: 2467-8821
https://doi.org/10.4314/njt.v44i1.12

DEPLOYMENT OF SMART CONTRACTS ON BLOCKCHAIN TECHNOLOGY IN EARLY MITIGATION OF DISTRIBUTED DENIAL OF SERVICE IN SOFTWARE DEFINED NETWORKS

AUTHORS:
C. P. Okide[1], M. A. Ahaneku[2,*], U. N. Nwawelu[3], V. C. Chijindu[4], O. M. Ezeja[5], and B. O. Ekengwu[6]

AFFILIATIONS:
[1,2,3,4,5,6]Department of Electronic and Computer Engineering, University of Nigeria, Nsukka, Enugu State, Nigeria

***CORRESPONDING AUTHOR:**
Email: mamilus.ahaneku@unn.edu.ng

ARTICLE HISTORY:
Received: 10 May, 2024.
Revised: 21 January, 2025.
Accepted: 03 February, 2025.
Published: 14 April, 2025.

KEYWORDS:
Distributed Denial of Service, SDN, Blockchain, Mitigation, Smart Contracts.

ARTICLE INCLUDES:
Peer review

DATA AVAILABILITY:
On request from author(s)

EDITORS:
Chidozie Charles Nnaji
Patrick Akpan

FUNDING:
None

Abstract

The rapid growth of smart and mobile devices that have resulted in content proliferation, server virtualization, and the emergence of cloud services have compelled the communication network industry to re-examine its network topologies. In this work, Distributed Denial of Service (DDoS) attacks are conducted by flooding target systems with traffic, designed to disrupt or suspend internet services for use by legitimate users. These attacks deplete network resources, thereby disabling those services and in turn decreasing the availability of the network. Blockchain technology has emerged as a viable option for DDoS mitigation. This technology has blockchain's core and promising inherent features to combat fatal cyber threats. These features include but are not limited to decentralization, immutability, integrity, anonymity, and verifiability. This work models a Software Defined Network (SDN) that is capable of mitigating DDoS attacks by integrating smart contracts. This offers a security technique that combines SDN and Blockchain to mitigate DDoS attacks at early stage. Intrusion detection and prevention are essential components of a comprehensive network protection strategy and are employed to detect and mitigate DDoS attacks in the SDN. In this paper, a secure network architecture capable of mitigating DDoS attacks in less than a second through the use of Blockchain technology has been presented after deployment. With a maximum of 25 requests per second for a single user and 8,000 compromised nodes, the software defined network successfully mitigated the DDoS at 0.68 seconds.

1.0 INTRODUCTION

The major setback of most communication networks lies in the maintenance, troubleshooting, and expansion. In recent times, demands for Software Defined Networks (SDN) and other virtualized systems are on a rapid increase. This is because most traditional networks can no longer handle the ever-increasing demands of network users and recent network technologies. The demands which include but not limited to minimum downtime, scalability, network techniques reviews, updates, flexibility, connectivity and reachability have prompted a paradigm shift to SDN. Because SDN is becoming increasingly popular. Information technology (IT) support teams are better equipped now to respond more swiftly to users' needs and application requirements. SDN has network control plane and forwarding planes. The separation of these two planes makes the network control plane programmable. Thus, allows users and network administrators to manage resources almost seamlessly [1].

HOW TO CITE:
Okide, C. P., Ahaneku, M. A., Nwawelu, U. N., Chijindu, V. C., Ezeja, O. M., and Ekengwu, B. O. "Deployment of Smart Contracts on Blockchain Technology in Early Mitigation of Distributed Denial of Service in Software Defined Networks", *Nigerian Journal of Technology*, 2025; 44(1), pp. 105 – 113; *https://doi.org/10.4314/njt.v44i1.12*

Despite recent advancements in SDNs, power and network connectivity remain susceptible to Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. The DDoS attacks are typically launched against the main server that is responsible for supplying critical network services [2]. The controller of the software defined network (SDN) can detect and counter such attacks only with the right security measures put in place. Such security measures include gathering data and taking necessary steps to mitigate the number of requests per second transmitted to the primary server. The attack by any intruder will be successful if the controller and primary server are attacked. However, from the study, several systems will have to be compromised in order to launch an attack on the controller [3]. Intruders often make DDoS attacks on these servers so that actual customers are denied access to the needed services. When the system is successfully attacked by hackers, loss of data is certain. For this reason, the use of blockchain technology is significant in countering both the DDoS attacks and its consequent effects.

Blockchain technology can help protect privacy and availability of resources against illegitimate users. A blockchain is a decentralized digital ledger that consists of a list of data records (information). The information is structured into blocks that are sorted chronologically and encrypted. Numerous factors play important role in blockchain security but this study will focus on consensus and immutability. While consensus involves the agreement of nodes within the blockchain to decide on the actual state of the network, immutability comes into play when trying to modify or stop transactions that have been established.

The proposed SDN security solution employs blockchain smart contracts to mitigate DDoS attacks. The first stage involves the development of smart contracts and algorithm. The second stage involves running unit tests and setting up the network on a Google Cloud. The system is deployed and tested in real-time under three critical phases: normal system operation phases, attack phase, and mitigation phase. The flow rules stored on the blockchain are applied in all scenarios to ensure the system's security. Security is one of the major concerns in software defined networks (SDNs). Although using an SDN controller makes the network easier to manage, it is equally vulnerable to hackers. Security is a serious challenge SDN needs to overcome since an attacker only needs to target the SDN controller to be able to take control of the entire network security**.** Protecting the SDN controller is important and requires a robust security architecture. This research used a real-world scenario for DDoS attack mitigation by deploying blockchain smart contracts. The work also provides a real-time attack defensive and early mitigation technique against DDoS in software defined networks.

The rest of the paper is presented as follows: section two presents the literature review. Section three describes the methodology adopted for this work. Section four discusses the results obtained through the experiment. Also, in section four, the obtained result is validated with the existing works. Section five presents the conclusion of the work.

2.0 LITERATURE REVIEW

The varieties of ways through which SDN can be attacked include but are not limited to DoS attack and authorization attack. If the SDN application layer for instance, receives an excessive number of requests, it may be forced to slow down the processing of requests or even stop serving other customers. Denial of service attacks can be carried out intentionally by some malicious users, who will flood the system with numerous requests. DDoS attack requests lack evident characteristics that can differentiate a malicious request from a valid one. These illegitimate users can easily gain access to the attacking tools, which raises the frequency of these attacks [4]. The implication is that a hacker in question may see, copy, edit, or even disrupt communication on the network. It therefore becomes very imperative that any security solution should be scalable and have the capacity to instantly accept dozens of legitimate elements while a single rogue element from a hacker is rejected [5]. To combat these risks in SDNs, it is necessary to provide a holistic security solution that can operate in such a way that its performance or scalability is not affected and an alarm generated when an actual user makes multiple legitimate requests. The SDN framework guarantees such operations [6].

2.1 DDOS Statistics

According to Arbor network, every day, over one thousand (1000) different DDoS attacks are tracked by them around the world [7]. Anyone may be a target of these attacks. The targets of these attacks might include e-commerce sites, banks, and companies of which the internet companies are not excluded. Earning profit serves as one of the primary drivers for targeting these network users. Also, DDoS attacks may frequently target governments and political groups, stock markets, or gaming websites. DDoS attacks have been in existence and attacking techniques keep advancing as the networking systems evolve.

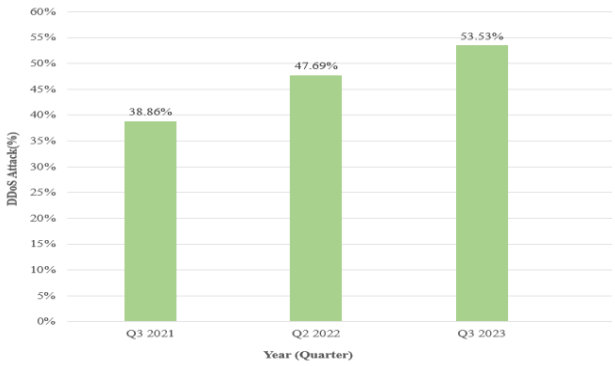

Figure 1: Comparative number of smart attacks, Q3 2021, Q2 /Q3 2022 [8]

Figure 1 shows the quarterly trends of DDoS attacks for the third quarter of 2021, second and third quarter of 2022 as reported by Kaspersky DDoS protection team. The quarterly report shows the highest number of DDoS attack occurrence in the third quarter of 2022.

2.2 DDOS Attack Mitigation

Authors in [9] focus on the development of a system, which effeciently defends against both DDoS and port scan attacks. In addition to a "detection module", the paper also presented and evaluated the performance of a "mitigation module", which, in the case of DDoS attacks, has its basis in a game-theoretic approach, directly associated with a central controller: whilst in the case of port scan attacks, mitigation derived from the use of a 'directed policy', described as "a single source IP drop. Authors in [10] utilized analysis of the frequency of a network flow's traffic in the detection of DDoS attacks. The authors assumed that a denial-of-service attack occurs whenever traffic flow increases beyond the set threshold. When this occurs, the packets associated with this high traffic are dropped to help mitigate the effects on the system.

Authors in [11] proposed an approach to the detection of DDoS attacks, bases on the interrogation of traffic flow features, through the use of the NOX platform. Among a few others, one contribution claimed by the publication is that the utilization of self-organizing maps enabled the simultaneously high detection and very low false alarm rates, recorded by the system. The authors in [12] introduced FortNOX as a software extension that provides role-based authorization and security constraint enforcement for the NOX OpenFlow controller. By introducing FortNOX, the challenge of efficient detection and reconciliation of potentially conflicting flow rules imposed by dynamic OpenFlow (OF) applications is addressed. Authors in [13] discussed extensively the associated DDoS attacks based on intrusion detection strategies. In [14], a low-rate DDoS attack detector based on two novel information metrics (generalized entropy and information distance) is proposed. In [15], the detection of a low-rate DDoS attack is achieved with the Hurst coefficient. The results from the experiment demonstrated that the proposed approach can detect in real-time the unseen DDoS intrusion in the normal data traffic. The authors in [16] evaluated the efficiency and effectiveness of various network information metrics, namely, Renyi's entropy, Shannon entropy, Generalized entropy, and Hartley entropy in detecting low-rate DDoS attacks.

A chaotic model was proposed and used to determine DDoS flooding attack traffic using the property of network self similarity in [17]. In [18], a DDoS intrusion detection algorithm based on network traffic pre-processing and chaos theory is proposed. The method is capable of detecting anomalies caused by busty legitimate traffic or DDoS flooding attacks. In [19], the authors introduced a novel anomaly detector based on a hidden semi-Markov model for detecting DDoS attacks at the application layer. The efficacy of this method is demonstrated through experiments involving real-world web traffic data. The authors concluded that separating identifiers and locations can help prevent DDoS attacks after examining numerical results derived from real-world data [20].

2.3 Related Works

Blockchain-based smart contracts is used to secure SDN controllers [21]. The authors were able to demonstrate various scenarios that use smart contracts to secure the distributed SDN and prevent unauthorized access and DoS attacks. The first scenario achieved a mitigation time of 60 seconds and the authors stated this can further be reduced to 5-10 seconds. The second scenario demonstrated that smart contracts can resist a DoS attack and protect the controller from failure. However, the authors did not address the issue of distributed denial of service (DDoS) involving attacks from different IP addresses. How to identify and mitigate DDoS attacks in SDN data plane is presented in [22]. This is achieved by setting a predefined packet response rate on the blockchain such that any packet transmission above the predefined rules is likely to get blocked due to the nature of implementation. The outcome of the evaluation shows that detection and mitigation take between 100 and 150 seconds on the distributed peer to peer network setup. The SDN application worked with the OpenDaylight controller to collect network information and create new rules. However, since these rules are stored in a centralized manner, any

attack on the on the controller can get the entire network compromised.

An architecture in which a smart contract is put on a private blockchain to enable collaboration across several network domains for DDoS mitigation is proposed [23]. The blockchain application is utilized to provide an additional layer of security so that the solution could run with or without the blockchain since it is an additional feature. However, for an attack that started at 09:35:25 (mm:ss:ms), the security script started at 09:50:00 (mm:ss:ms) resulting in about 15 seconds mitigation time. A number of requests dropped after the mitigation script was executed, but the timing was still not minimal hence the results were poorly presented. Meanwhile, three unique network topologies for minimizing DDoS attacks through the use of Blockchain technology and SDN is proposed [24].

Authors in [25], proposed a strategy for detecting DDoS attacks, analyzed it using Mininet emulation, and implemented it in a controller. According to authors, the method improved attack detection time, but the mitigation time was 6.38 seconds which is high enough and was not executed in real-time network environment, neither was it built with a secure technology such as blockchain.

In [26], the authors proposed a solution that identifies a number of cyber-attacks based on DDoS in real time, mitigation times, detrimental effects on network performance, and ensures that normal traffic is delivered accurately based on the flow rules. The mitigation took about 5 seconds, 2 seconds, and 3 seconds for the different experiments.

3.0 METHODOLOGY

3.1 Smart Contracts Deployment on Blockchain

Smart contracts act as a repository that prevents loss of information should the other machines used as storage be attacked. The beauty of storing with blockchain lies in the fact that time of storage is always on the chain and an audit trail is always available for tracking. The blockchain part of this network provides security through the creation and management of the IP blacklist mapping. Each request that comes into the system is being recorded as transaction on the block starting from the genesis block. These transactions are being mined and every new block generated is connected to the previous one through the use of cryptographic techniques. The first part of the blockchain security procedure typically involves the development and deployment of the smart contracts after a unit test on Genache. The contract address is then generated and blockchain connected to the web3 using remote procedural call (RPC). Finally, authentication and verification are performed with respect to time and storage on the block. Figure 2 describes the network security flow process.

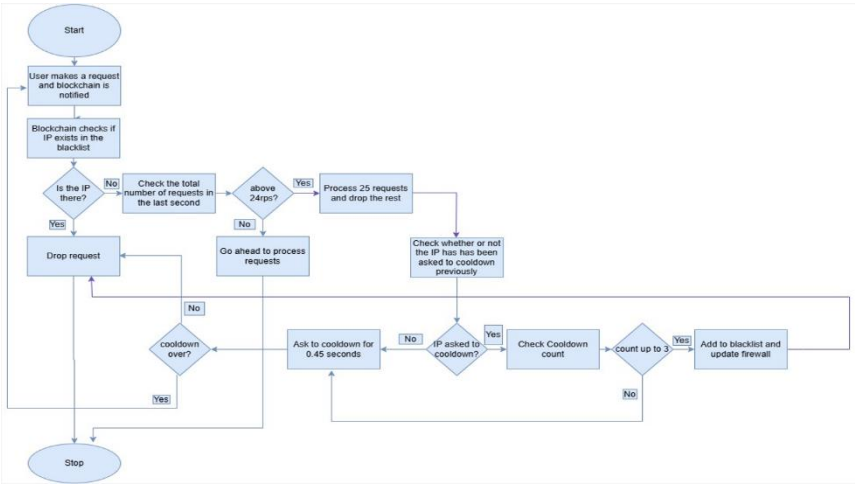

Figure 2: Process flowchart for blockchain security

As shown in Figure 2, requests are automatically dropped once the IP is found on the blacklist stored on the blockchain. This step makes it quicker to filter already known offenders before they try overuse the network resources. On the other hand, if it is not found in the dictionary, the same IP will be searched for on the cache and if it does not exist too, the IP address and time stamp are added. Furthermore, if it does exist, the data is extracted from the cache and filtered out. Filtering is done by calculating the difference between the time stamp and the current time. The result is then converted into seconds and the request will drop if the request count is more than 25 requests per second (rps). However, after filtering and count is

within 25 rps, user receives a response 200 OK. This means that the request has succeeded and user granted access to the application.

3.2 Network Setup

Figure 3 shows a network diagram for the proposed system setup. It shows how the intrusion prevention system is set up such that it is installed directly behind the firewall for anomaly detection in network traffic.

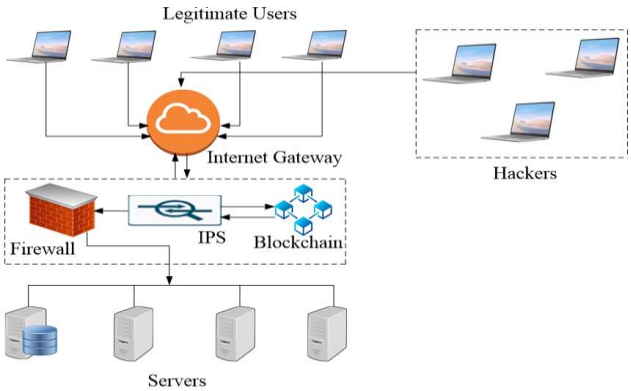

Figure 3: Network setup for early mitigation of DDoS attacks

In general, DDoS attack packets lack any discernible features that would distinguish a malicious stream from a genuine one. Additionally, the hackers can readily gain access to tools used in these attacks, which in turn, makes cyber threats and attacks more frequent [4]. If a hacker should gain access, the implication is that hacker in question may see, copy, edit or even disrupt communication on the network. In order to combat these risks in the proposed network setup in Figure 3, a holistic security solution is necessary and that is why copies of the blacklist, flow rules and IPS controls exist on the blockchain.

3.3 Network Setup Tools

The network installations and setup were done on the console, using a group of cloud computing services provided by the Google Cloud Platform (GCP). They are hosted on the same setup as Google's core products, including Google Search, YouTube, Google Drive, and Gmail. Along with a set of administration tools, GCP provides a modular cloud services portfolio that includes computing, analytics, data storage, and machine learning. Platform-as-a-service (PaaS), Infrastructure-as-a-service (IaaS), and serverless computing environments are all offered on Google Cloud Platform [26]. For the purpose of the DDoS attack, Google Cloud servers were setup. The Google Cloud servers used here are Virtual Machines (VM) - e2 Micro Ubuntu 18.04 1GB RAM, 20GB HDD for the application, load balancer, DDoS servers.

The application itself is being served on Port 3000 of the server (many channels where traffic can go and return on the IP address) and DDoS agent servers have an executable written in them such that once it springs up, it will start trying to make requests to the application. Solidity programming language was used to write the smart contract firewall service that runs on the Ethereum blockchain. This smart contract is a software on the Ethereum blockchain that can't be altered once it is deployed even by the programmer. This ensures information authenticity and transparency.

The development environment based on Ethereum BC-Truffle was used for the application development in a distributed form. For the unit testing, it was done locally before deploying to production environment using Genache. Remix-Ethereum IDE was used to deploy the smart contracts and store user information such as the IP addresses and timestamps. GitHub was used to track changes from anywhere and manage source codes. MetaMask, a cryptocurrency wallet was needed as a gateway to blockchain and launching of the smart contracts. After deployment, crypto test tokens were obtained from BSC Faucet. Terraform was used to setup VPC and public subnets. It was equally used to provision the load balancer and application servers. Prometheus was used to collect information like incoming requests and CPU utilization from servers. Ansible was used to configure Redis. The Javacript and Python are programing languages used to write the load balancer scripts and DDoS scripts, respectively.

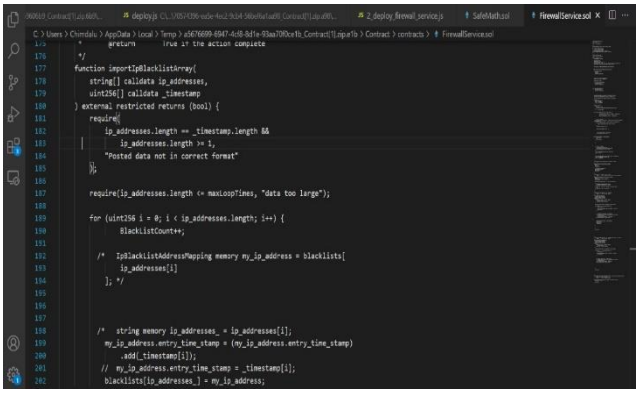

Figure 4: Smart contract firewall service

4.0 RESULTS AND DISCUSSION

The system was deployed and tested in real-time under three critical phases: the normal system operating phase, which involves regular usage by legitimate users; the attack phase, involving request floods by attackers; and the mitigation phase, which immediately restores availability by employing the

proposed solution to track and block defaulting IPs to prevent exhaustion of network resources. Figure 4 shows the smart contract firewall service. It works in tandem with the application level of security to check illegal attempts to sign in. Figure 5 shows the smart contract deployment using Remix-Ethereum IDE. The deployment exercise has several steps that was guided by the smart contract deployment rules.

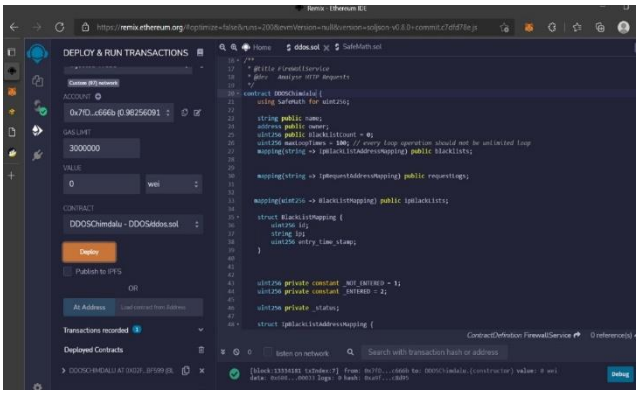

Figure 5: Smart contract deployment using Remix-Ethereum IDE

4.1 Normal System Operation Phase

Figure 6 is a display of successful user requests. It is a report of the logs that read "I'm available" when total requests per second (rps) coming in have not reached the maximum capacity.

Figure 6: Sending normal traffic

The availability of the system to each user or IP is dependent on the total number of requests that go through per second provided it is not an already blacklisted IP. For the proposed SDN security system, the maximum number of requests per user is set at 25 rps and this just gives enough room to accommodate legitimate users' request even though most users would not normally make as much as 25 rps. Figures 7 and 8 show the respective plots of the maximum total request during the normal system operation for a single user and all users.

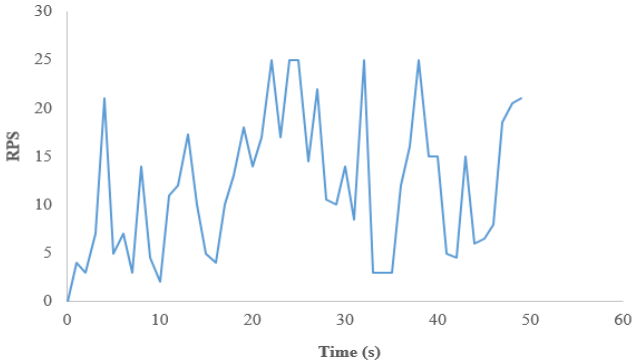

Figure 7: Graph for normal system operation for a single user

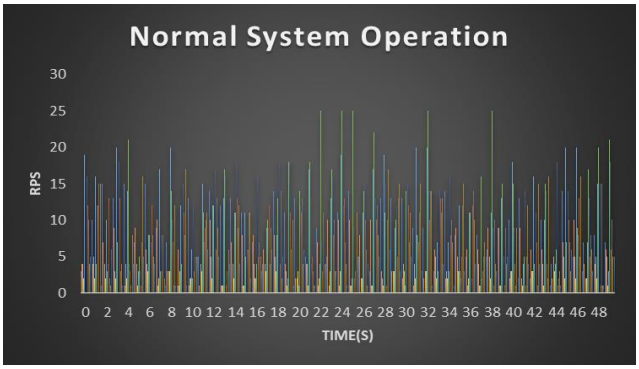

Figure 8: Graph for normal system operation for all users

The graph in Figure 7 shows the total number of requests per second made by a single user for a period of 49 seconds while Figure 8 shows the total number of requests for all the users within the 49 seconds time frame. From both figures, it can be seen that the maximum total request during the normal system operation is 25 rps. The capability of the system to identify a DDoS attack is the focus of the detection phase. A DDoS attack is detected when an anomaly occurs in the request flow of the network. In this work, when a user makes over 25 requests in a second, the users' requests are dropped according to the time stamp to prevent processing not more than 25 requests per second. This increase and overflood in number of requests occurred at the 50th second operation.

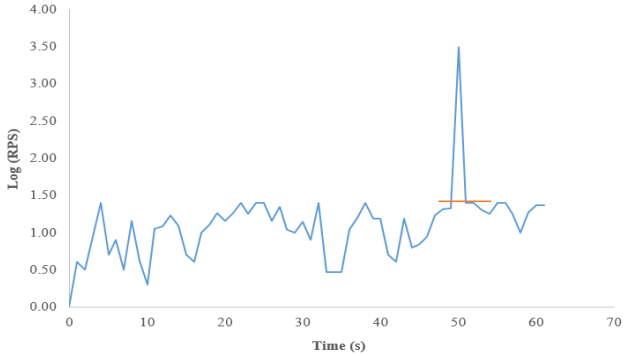

Figure 9: DDoS attack mitigation

4.2 Mitigation Phase

The flow rules for this stage are permanently stored on the blockchain. The limit was exceeded at the 50th second with the attackers sending at least 3,000 rps each as presented in Figure 9.

In the graph of Figure 9, it is clearly seen that the DDoS attack was approximately detected and mitigated in less than a second. This causes the network to stabilize around its "normal" operating status after some requests were dropped by the proposed network security measure between 50-51 seconds. These attacks were successfully mitigated and the total number of requests dropped back to 25 (that is at the 1.40 mark after taking the logarithm) for the defaulting IP. The results presented therefore validate the premise that smart contracts can assure the security of distributed software defined networking by ensuring that each rule on the network is applied in all of its conditions.

4.3 Results Comparison

The proposed SDN security solution is compared with the proposal of Sumantra and Indira in [24]. The comparison is made based on attack traffic (rps) and mitigation time (s) and is presented in Table 1.

Table 1: Comparative results of the proposed and exiting SDN security solutions

SDN Security Solution	Attack Traffic (rps)	Mitigation Time (s)
Proposed Solution	8000	0.68
Sumantra and Indira [25]	300	6.38

From Table 1, it is evident that the proposed SDN security solution outperformed the proposal of Sumantra and Indira in DDoS mitigation time and attack traffic. The result of the proposed solution shows an improvement in the time it takes to mitigate DDoS attacks. Descriptively, it can be seen that the proposed solution was able to mitigate the DDoS attack and restore access to all users at 0.68 seconds. This is less than the mitigation time of 6.38 seconds recorded by the work of [25]. The proposed solution experienced 8,000 rps attack which is even above 300 rps considered by the work of [25] and still able to mitigate the DDoS attacks. This shows that blockchain smart contracts ensured all conditions in the flow rules were applied to secure the SDN from being compromised.

5.0 CONCLUSION

Distributed Denial of Service (DDoS) attacks is a serious security treat to software defined networks (SDN). The integration of blockchain smart contracts to address this security concern in SDN is promising. Some of the related works carried out real-time experiments to test their suggested methods to mitigate DDoS attacks in SDN whereas, others simulated theirs without evidence of the smart contract deployment. Furthermore, there was neither provision for update nor decentralization of existing blacklist and flow rules should the controller be compromised. Motivated by this shortcoming, this work has implemented and experimented a blockchain solution with smart contracts integration that recorded a minimal mitigation time for the SDN on real-time. The mitigation time for DDoS using smart contracts was improved to less than a second and SDN security solution improved because of the security flow rules on blockchain. By comparing the proposed blockchain solution for SDN with related work, this work has demonstrated that the suggested solution can defend and provide early mitigation of DDoS attacks in SDN. With a maximum of 25 requests per second for a single user and 8,000 compromised nodes flooding the system, the DDoS attacks were successfully mitigated at 0.68 seconds. Future work will investigate the integration of IoT and AI with blockchain decentralized systems to address other security threats targeted at SDN.

REFERENCES

[1] Jammal, M., Singh, T., Shami, A., Asal, R. and Li, Y. N "Software Defined Networking: State of the art and Research Challenges", *Computer Networks*, vol.72, pp.74 – 98, 2014. doi.org/10.1016/j.comnet.2014.07.004

[2] Eliyan, L. F. and Pietro, R. D. "DoS and DDoS Attacks in Software Defined Networks: A Survey of Existing Solutions and Research Challenges", *Future Generation Computer Systems*, vol. 122, pp. 149 – 171, 2021. doi.org/10.1016/j.future.2021.03.011

[3] Sanjeetha R., Srivastava, S., Kanavalli, A., Pattanaik, A. and Gupta, A. "Mitigation of Combined DDoS Attack on SDN Controller and Primary Server in Software Defined Networks Using a Priority on Traffic Variation," *2020 International Conference for Emerging Technology (INCET)*, Belgaum, India, June 5 – 7, 2020, pp. 1 – 5. doi.org/10.1109/INCET49848.2020.9153998

[4] Mahjabin, T., Xiao, Y., Sun, G., and Jiang, W. "A Survey of Distributed Denial of Service Attack, Prevention, and Mitigation Techniques", *International Journal of Distributed Sensor Networks*, vol. 13, no. 12, pp. 1 - 33, 2017. doi.org/10.1177/1550147717741463

[5] Sukhdeve, N. M., Sakhare, A. and Gangwar, S. "Overview of SDN with Blockchain over Cloud Environment", *International Journal of Engineering Research & Technology*, vol. 08, no. 12, pp. 279-281, 2019. doi.org/10.17577/IJERTV8IS120166

[6] Braun, W., and Menth, M. "Software Defined Networking Using OpenFlow: Protocols, Applications and Architectural Design Choices", *Future Internet*, vol. 11, pp. 302 – 336, 2014. doi.org/10.3390/fi6020302

[7] Nazario, J. "DDoS Attack Evolution," *Network Security*, vol. 2008, pp.7 – 10, 2008. doi.org/10.1016/S1353-4858(08)70086-2

[8] https://media.kasperskycontenthub.com/wp-content/uploads/sites/43/2022/11/03142348/02-en-ddos-q3-2022.png ddos-attacks-in-q3 2022 access 13 January 2025.

[9] Assis, M., Novaes, M., Zerbini, C., Carvalho, L., Abrao, T. and Proenca, M. "Fast Defense System Against Attacks in Software Defined Networks", *IEEE Access*, vol. 6, pp. 69620 – 69639, 2018. doi.org/10.1109/ACCESS.2018.2878576

[10] YuHunag, C., MinChi, T., YaoTing, C., YuChieh, C., and YanRen, C. "A Novel Design for Future On-demand Service and Security," *International Conference on Communication Technology Proceedings, ICCT*, Nanjing, China, November 11-14, 2010, pp. 385-388. doi.org/10.1109/ICCT.2010.5689156

[11] Braga, R., Mota, E., and Passito, A. "Lightweight DDoS Flooding Attack Detection Using NOX/OpenFlow," *IEEE Local Computer Network Conference*, Denver, CO, USA, 2010, pp. 408-415. doi.org/10.1109/LCN.2010.5735752

[12] Porras, P., Seungwon, S., Yegneswaran, V., Fong, M., Tyson, M. and Gu, G. "A Security Enforcement Kernel for OpenFlow Networks", *Proceedings of the first workshop on Hot topics in software defined networks*, pp. 121–126, 2012. doi.org/10.1145/2342441.2342466

[13] Behal, S. and Kumor, K. "Detection of DDoS Attacks and Flash Events Using Novel Information Theory Metrics", *Computer Networks*, vol. 116, pp.96 – 110, 2017. doi.org/10.1016/j.comnet.2017.02.015

[14] Xiang, Y., Li, K., and Zhou, W. "Low-Rate DDoS Attacks Detection and Traceback by Using New Information Metrics", Transactions on Information Forensics and Security, vol. 6, no. 2, pp. 426-437, 2011. doi.org/10.1109/TIFS.2011.2107320

[15] Xiang, Y., Lin, Y., Lei, W. L. and Haug, S. J. "Detecting DDOS Attack Based on Network Self-similarity", *Communications, IEE Proceedings*, vol. 151, pp. 292-295, 2004. doi.org/10.1049/ip-com:20040526

[16] Bhuyan, M. H., Bhattacharyya, D. K., and Kalita, J. K. "Information Metrics for Low-rate DDoS Attack Detection: A comparative evaluation", *2014 Seventh International Conference on Contemporary Computing (IC3)*, Noida, India, 2014, pp. 80-84. doi.org/10.1109/IC3.2014.6897151

[17] Chonka, A., Singh, J., and Zhou, W. "Chaos Theory-based Detection Against Network Mimicking DDoS Attacks", *IEEE Communications Letters*, vol. 13, pp. 717-719, 2009. DOI:doi.org/10.1109/LCOMM.2009.090615

[18] Chen, Y., Ma, X., and Wu, X. "DDoS Detection Algorithm Based on Preprocessing Network Traffic Predicted Method and Chaos Theory", *IEEE Communications Letters*, vol. 17, pp. 1052-1054, 2013. doi.org/10.1109/LCOMM.2013.031913.130066

[19] Xie, Y., and Yu, S. Z. "Monitoring the Application-Layer DDoS Attacks for Popular Websites", in *IEEE/ACM Transactions on Networking*, vol. 17, no. 1, pp. 15-25, 2009. doi.org/10.1109/TNET.2008.925628

[20] Luo, H., Lin, Y., Zhang H., and Zukerman, M. "Preventing DDoS Attacks by Identifier/Locator Separation", *IEEE Network*, vol. 27, no. 6, pp. 60-65, 2013. doi.org/10.1109/MNET.2013.6678928

[21] Almakhour, M., Wehby, A., Sliman, L., Samhat, A. E., and Mellouk, A. "Smart Contract Based Solution for Secure Distributed SDN", *11th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*, Paris, France, pp. 1-6, 2021. doi.org/10.1109/NTMS49979.2021.9432647

[22] Abdulkarem, H., and Dawod, A. "DDoS Attack Detection and Mitigation at SDN Data Plane Layer," *2020 2nd Global Power, Energy and Communication Conference (GPECOM)*, Izmir, Turkey, pp. 322-326, 2020. doi.org/10.1109/GPECOM49333.2020.9247850

[23] Giri, N., Jaisinghani, R., Kriplani, R., Ramrakhyani, T. and Bhatia, V. "Distributed Denial of Service (DDoS) Mitigation in Software Defined Network Using Blockchain", 3rd *International conference on IoT in Social, Mobile, Analytics and Cloud (I-SMAC)*, Palladam, India, pp. 673-678, 2019. doi.org/10.1109/I-SMAC47947.2019.9032690

[24] Kumar, S., and Amin, R. "Mitigating Distributed Denial of Service Attack: Blockchain and Software Defined Networking Based Approach, Network Model with Future Research Challenges," *Security and Privacy*, vol. 4, no. 4, pp. e163, 2021. doi.org/10.1002/spy2.163

[25] Sumantra, I., and Gandhi, S. I. "DDoS attack Detection and Mitigation in Software Defined Networks", *International Conference on System, Computation, Automation and Networking (ICSCAN)*, Pondicherry, India, pp. 1-5, 2020. doi.org/10.1109/ICSCAN49426.2020.9262408

[26] Manso, P., Moura, J., and Serrao, C. "SDN – Based Intrusion Detection System for Early Detection and Mitigation of DDoS Attacks," *Information, vol. 10, no.* 106, pp. 1 – 17, 2019. doi.org/10.3390/info10030106

[27] GPC, https://cloud.google.com/gcp?utm_source=google&utm_medium=cpc&utm_campaign=emeange-all-en-bkws-all-all-trial-e-gcp-1010042&utm_content=text-ad-none-any-DEV_c-CRE_501794636587-ADGP_Hybrid%20%7C%20BKWS%20-%20EXA%20%Txt%20~%20General%32v2-KWID_43700061569959221-Kwd-26415313501-userloc_1010297&utm_term=KW_google%20cloud%20platform-NET_g-PLAC_&gelid=CjwKCAiA7dKMBhBCEiwAO_erFFh_LY7v_4kYzma6nXWOvmioXG_CIEZ41BzDdelT-bbntj4Vs8iv9xoC5foQAvD_BwE&gclsrc=aw.ds