Resource Isolation to Mitigate DoS/DDoS Attacks in Cloud Computing

MSc Research Project

Cybersecurity

Eldhose Shaji

Student ID: x2119598

School of Computing

National College of Ireland

Supervisor: Noel Cosgrave

National College of Ireland
Project Submission Sheet – 2022/2023

Student Name:	**Eldhose Shaji**
Student ID:	**X21195986**
Programme:	**MSc Cybersecurity**
Year:	**2023**
Module:	**MSc Research Project**
Supervisor:	**Noel Cosgrave**
Submission Date:	**14/08/2023**
Project Title:	**Resource Isolation to Mitigate Denial-of-Service and DDoS Attacks in Cloud Computing**
Word Count:	**8738**
Page Count:	**21**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the references section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature: **Eldhose Shaji**

Date: **14/08/2023**

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. **Please attach a completed copy of this sheet to each project (including multiple copies).**
2. **Projects should be submitted to your Programme Coordinator.**
3. You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.**
4. **You must ensure that all projects are submitted to your Programme Coordinator on or before the required submission date.** Late submissions will incur penalties.
5. **All projects must be submitted and passed in order to successfully complete the year.** Any project/assignment not submitted will be marked as a fail.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Resource Isolation to Mitigate Denial-of-Service and DDoS Attacks in Cloud Computing

Eldhose Shaji

Student ID: x21195986

Abstract

Cloud computing systems use shared resources that are naturally susceptible to exploitation, “denial-of-service (DoS)” and “distributed denial-of-service (DDoS)” attacks continue to represent serious dangers. Therefore, to protect cloud infrastructures from these risks, resource isolation becomes an essential tool. This study gives a thorough analysis of resource isolation methods for preventing “DoS” and “DDoS” assaults in cloud computing. The research examines the resource isolation tactics, including virtualization, containerization, RSA token-based access and software-defined networking, which helps in establishing strong boundaries between the resources used by the multiple cloud tenants and also aims to stop those traffics that are meant to cause the DoS/DDoS attacks and thereby preventing vital cloud resources. Considering the dynamic nature of the cloud environment, I have also evaluated each technique's impact in terms of security, performance, and implementation overhead. This study intends to assist cloud service providers in making choices to strengthen their defensive mechanisms against “DoS” and “DDoS” attacks using resource isolation and thereby assuring the continuous and safe operation of cloud-based services.

Keywords**:** Resource Isolation, RSA Tokens, Cloud Computing, DoS/DDoS Attack, Virtualization and Kernel Level Isolation, Hping3, TCP Flooding, IP Tables, I/O graph

1. Introduction

1.1. Research Background

The use of cloud services in information technology is growing exponentially these days as the targeted attacks on the cloud as well. The service is prone to many attacks that have been used in stealing and breaching the network for various needs. Of the different attacks targeting the cloud environment, the DoS/DDoS are the most prominent ones. Both the DoS and the Distributed-DoS are serious attacks that work by flooding the victim system hosted in the cloud with a tremendous number of connection requests and thereby forcing it to stop working and making the whole system inaccessible to legitimate users. These attacks could be launched at multiple target systems sharing the same cloud resources and the DoS/DDoS attacks are aiming at forcing the cloud resources to reach the threshold and they will be no longer available to serve the requests from legitimate users.

To a further extent, if an attacker has managed to gain access to any of the systems hosted in the cloud, then he can implant malware as this would be used for controlling other systems from this system. Such an infected system can be termed a bot, as the group of similar bot systems that could be used remotely is known as a botnet. These bots can be used to launch further attacks on the targeted cloud platform and then floods the system with random requests using these bots. Nevertheless, different methods could be applied to the system that would mitigate the attacker from attacking the system and resource isolation is the key one among them.

The report focuses on suggesting a proper mitigation strategy using resource isolation as the key mitigating technique. The main aim of the report is to identify the type of attack that is caused on the system. This study would be used for the understanding of attacks and the potential misuse an attacker can cause on the victim's system. Among all the different mitigation strategies, the report suggests the best one out of all and that can be applied on the cloud platforms to safeguard them from the attacks.

1.2. Aim

The main aim of the project is to safeguard the cloud resources from DoS/DDoS attacks and to properly isolate the resources with the help of the implementation of the resource isolation methodology. The resource isolation methodology can be applied on the client side and at the cloud service provider level to have a strong defence against DoS/DDoS attacks.

1.3. Objectives

The primary objectives of this research project are:

- To identify the resource isolation process and understand the fundamentals of the process.
- To analyse and examine DoS and DDoS attacks in cloud infrastructure, including their attack strategy and execution.
- To implement the resource isolation methodology as the mitigation strategy for DoS/DDoS attacks.
- To understand and analyse the results and find out the best isolation method that can be used in the cloud platform for mitigating these attacks.

1.4. Research Question

- How well does the resource isolation mechanism mitigate the effects of DoS/DDoS attacks in the cloud computing environment?

1.5. Report Structure

The first section of the report gives the abstract and will provide a brief idea about the overall research. After this section introduction has been discussed. Cloud computing along with the requirement for resource isolation is clearly explained along with the current approaches and the shortcomings or practical gaps in those approaches are explained. In-depth explanations of the problem background of this research and problem definition are provided, leaving the research inquiry out. The literature review of the foundational studies that concentrate on Resource Isolation in cloud computing is then presented with the implementation strategy. The next part of the report serves the details of the implementation of the proposed methodology and details on the results obtained. The report concludes by summarising the key findings, recommendations, limitations of the work and the future work that can be done on resource isolation.

2. Literature Review

2.1. Resource Isolation for Mitigation of DoS/DDoS Attacks using cloud computing.

The isolation of cloud resources has been described in this section of the project, where the mitigation strategy for DoS/DDoS attacks using resource isolation is clearly described. The process of implementation for isolation at the kernel level has been critically discussed in the following part of the report by comparing the results of the existing work and the pros and cons of it. The Kernel-level isolation consists of configuring the cgroups, namespaces, and secomp filters for maintaining proper isolation between the cloud resources. Verma *et al.* (2021) state that this process helps to implement the containers, which help to distribute the overall impact of the environment. This method is used to build containers, which are separated environments that give a collection of processes operating inside them using their own set of resources like CPU, memory, network, disk specification and many more. The authors propose a new technique for resource allocation among containerized programs called "core scheduling." The approach assigns specific CPU cores to containers, minimising context-switching costs and improving performance.

The process consists of controlling groups or cgroups, which are the main characteristics of the kernel, which help to distribute several kinds of resources, which might consist of several types of CPU processes, the memory, and the bandwidth of the input and output devices. Containers can easily be avoided based on the resources of the system along with utilization of the several resources. Namespaces are having a specific set of configurations for separating system resources like the network, filesystem, and processspace. Every cloud container has a unique set of namespace configurations that allows tailored configurations to be applied to it and so as separate the system resources.

Each individual container has different types of namespace collection where several types of perspectives regarding the resources can easily be designed. Additionally, at the end of the step, Seccomp filters must be configured and implemented. This is a kind of Linux kernel's Secure Computing mode (Seccomp) which mainly helps to implement a particular number of system calls that help to make faster processes. Sharma *et al.* (2021) have mentioned that this process has the potential to aid in analysing system calls by analysing the seccomp filter constraints and which in turn helps to stop accessing utilising several resources. This process is combined with several factors where the host system along with other containers can easily be identified.

2.2. Description of isolation steps

There are three different steps for implementing CSP level-using isolation consists of:

Initially, multi-level tenancy provides huge support to cloud computing. This stage consists of the maintenance of Virtualization along with containerization of the isolated environment where different types of clients can easily access the cloud through the internet. Bhushan and Gupta, (2019) refer to the fact that this can easily be implemented with the help of implementing several types of methods, allocation of different resources, and processing of data, which helps to maintain an isolated environment.

Secondly, resource allocation can easily be done with the help of certain policies of resource allocation consisting of storage, bandwidth for the network, memory allocation, CPU space and many more parameters. Finally, isolation for storage must be allocated based on storage level. In addition, to implement the storage the new concept of fog computing must take palace which automatically enhances the quality of cloud computing. This is the concept where isolation can easily be done with the help of the storage level. Cryptographic encryption can easily be done for this implementation where Fog computing plays a huge role in the encryption of the data and thereby mitigating the targeted attacks by using the shared keys. After the data encryption, the generated outcome must be saved into particular storage of glaciers. Radain *et al.* (2021) examined that the CSPs use cryptographic encryption mechanisms mainly used for sharing the keys either as a private or public keys in a secured manner.

2.3. Description of Architecture for Resource Isolation in cloud computing

In this section mainly the discussion has been done based on the architecture of Resource Isolation in cloud computing where a reference diagram for isolation has been implemented and as shown below.

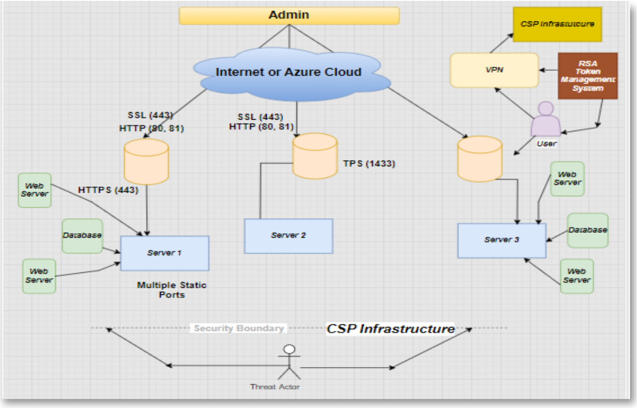


Figure 1: (Mahajan *et al.* 2022) shows the reference architecture diagram.

In this case, the protection of the cloud systems from "DoS/DDoS" attacks and assurance of the availability of the systems is the main concern. For the sole purpose of avoiding resource sharing, the Cloud Service Provider provides Virtual Machine Isolation by way of distributing VMs among the Real Servers and Data Centres in particular. In order to prevent over-allocation as well as potential DoS attacks, resource allocation along with scheduling make sure that the VMs have the resources they require based upon their respective workloads. Mahajan *et al.* (2022) showcase that user authentication is done by means of "role/token-based" access, which restricts access based on roles. Furthermore, by allowing access to the system across a "VPN" network, an "External Token Management system" improves overall security. The ultimate objective of this combination is real as well as the virtual machine isolation, access restrictions, and also enhancing the cloud storage security as a whole.

The reference architecture diagram of the project is shown above. There are five different layers, which are shown in the above figure. The layers are the “user layer”, “cloud computing layer", "virtual layer", "physical layer" and "cloud service providers". Each individual layer has been clearly described in the below section. Data centres are clearly described along with resources. Four different stages have been visualized which consist of "Deployment", "Databases", "Running", "Isolation" and "Determining".

Sattar and Matrawy (2019), point out the specific concerns that are to be addressed when implementing virtualization in the cloud infrastructure using a detailed graphical representation of the resource isolation method. The paper has also postulated a means of securing the processes like single sign-on, malicious connection requests, and authorized requests made by users which are all submitted directly to the IDS cum load balancer system. But the paper lacks in presenting the details on the specific attack patterns and the specific cloud setup. Additionally, there is no information available on how the postulated mitigation strategy works when the system handles legitimate and identified DDoS traffic.

Below shown is the reference network diagram taken for the implementation of the project. Akanji *et al.* (2021) show that the links amongst all of these gadgets get displayed inside the diagram's clear organization, which in turn suggests the existence of potential communication channels. This particular network diagram also renders a look at the underlying framework concerning this network by way of properly highlighting the presence of several devices along with their respective IP addresses. In the real-world scenario, such type of network architecture can be employed as a beginning point for the objective of additional analysis, optimization, and security analysis of the entire network infrastructure on the whole.

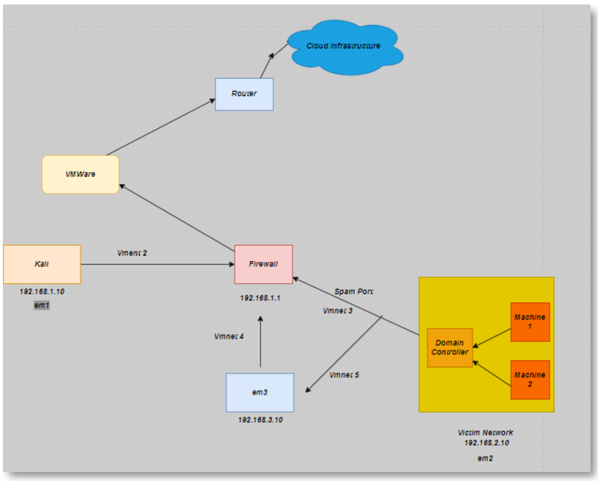


Figure 2: (Akanji *et al.* 2021) shows the reference network diagram.

2.4. How well does the RSA token implementation help in mitigating DoS/DDoS attacks.

As a form of two-factor authentication, RSA tokens help ensure that only authorised users have access to cloud resources. This step is required before any resource isolation measures may be used. Cloud service providers can authenticate users' identities and provide them access to their specified resources by requesting RSA tokens during the login process. Once authenticated, users are provided access only to the resources they are authorised to use, depending on their assigned roles, permissions, and the services they have subscribed to. This separation prevents unauthorised access and supports the isolation of resources. It helps in adding an additional security layer by entertaining only the authorised users and thereby limiting the chances for the occurrence of DDoS/DoS attacks. The below table depicts the advantages of integrating the RSA token in reference to resource isolation in the cloud.

Advantages	Description
Enhanced Authentication Security	RSA tokens provide 2FA. This makes unauthorised access to cloud accounts more difficult for attackers.
Protection of administrative access	The effect of unauthorised access to important cloud services can be considerably decreased by using an extra authentication factor.
Securing Multi-tenant Environments	RSA tokens improve security in multi-tenant cloud settings by making sure that each tenant's resources are secured and only available to authorised users, preventing unauthorised access or resource misuse.
Additional Layer of Defense	In addition to existing DDoS mitigation techniques and security measures used by cloud service providers, integrating RSA tokens adds another line of defence against DoS/DDoS assaults.

2.5. VM interaction in the real cloud infrastructure

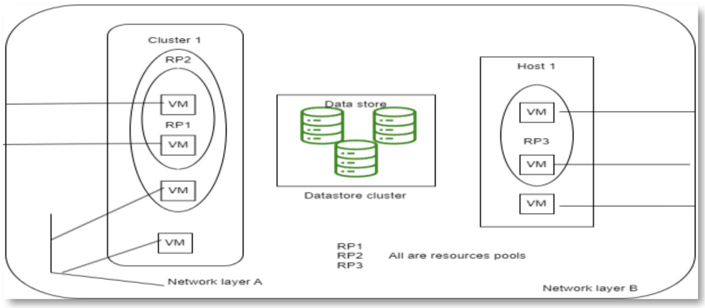


Figure 3: Displaying interaction of the Virtual machine in cloud computing.

The VM interaction in the real cloud infrastructure is shown in the above image. All customer service VMs and VMs designated for individual clients are operating in a single shared cloud platform. This opens the door for co-process interference, which is a factor within the system that can jeopardise the system's availability. Additionally, because all of the assets have been shared, a bigger attack's outer layer remains, which might be used by somebody else to launch a DoS/DDoS under assault.

Kumar and Somani, (2023) state that, fortunately, must put into operation protection measures in addition to every single one of these strategies to thwart cybercriminals attempting to bring about a DoS/DDoS assault. Security monitoring equipment must be installed for the purpose to detect any strange conduct or unauthorised entrance attempts. “Intrusion detection systems (IDS)” and “security information and event management (SIEM) systems” are a couple of instances of these tactics. IDS are capable of tracking communication over the network and detecting any unusual patterns of activity, whereas SIEM may accumulate and analyse vulnerability-related data from a variety of sources to identify possible security problems.

2.6. Impact of DDOS Attack on a cloud computing system

The DDoS attack directly affects the different layers of the cloud system, and it occurs both externally and internally. Among the different types of attacks, DoS/DDoS affects the cloud system in different SaaS layers. Zhijun *et al.* (2020) mentioned that this internal cloud-based DDoS attack in a cloud system provides infected "virtual machine images" that mainly carry internal DDoS attacks as per the same cloud computing system. The main reason for cloud-based DDoS attacks is that they become easier and cheapest to execute in IoT devices. On the other hand, this IoT attack makes default access to attackers for controlling them. For example, in the year 2020, AWS implementation directly mitigated 2.2 terabytes of DDoS attacks that caused thousands of client services unavailable for a certain period. In this case, the attackers have sent fake requests that have generated different types of resource utilization to the target server.

La et al. (2021) highlight the direct effects of DDoS attacks on cloud computing systems, causing issues such as service downtime, economic losses, smoke screening effects, and energy consumption costs. Cloud computing enables businesses to scale their IT infrastructure rapidly and effectively, addressing production demands. Three main services offered by cloud computing are Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). These services provide virtualization of hardware resources, including storage, servers, and networking parameters. Cloud computing also offers flexibility in implementing cost-effective processes based on scalability and other factors and these are prone to serious DoS/DDoS attacks and are listed as follows:

i. ***IP spoofing attack***

IP spoofing attacks generate unreachable IP addresses, which impact server replies and transactions. This makes tracing and detecting attacks utilising bogus IP addresses and packets challenging. PaaS layers are used to detect IP spoofing by using resources from the IaaS layer. To detect faked packets, IP-to-hop-count mapping is utilised, with the HCF approach addressing 90% of spoofed addresses. To defend against this threat, enhanced defence mechanisms and detection methods are required.

ii. ***SYN flooding attack***

Another type of DDoS assault is SYN flooding, which leverages a TCP connection in a three-way process to generate message synchronisation. The attack acknowledges the server and sends an ACK request, establishing a connection and transmitting packets that fail the three-way handshake. According to Li et al. (2019), the attack completes the packet process and demands a faked IP address. It also contains a sniffer attack, which makes it harder for the server to reply and lowers cloud system performance. This kind of attack has also consisted of a sniffing attack, which makes the server unable to reply to the connection requests and reduces the performance of the cloud system. This will eventually pave the pathway for DoS/DDoS attacks.

2.7. Methods that can be integrated with Resource isolation for improved security.

The DDoS attack on a system could be done at different levels, as this would be used in order to provide a complete method that would be difficult for the victim system to detect. The following methods can be used in accordance with the resource isolation mechanism so as to provide increased security, they are listed as follows:

- ***"Detecting":*** The primary or the basic step in mitigating the DDoS attack is by isolating the cloud resources safely and securely and also monitoring all the inbound traffic to the server, that can launch an attack. When getting a connection request, the system should check for the legitimacy of the connection request by comparing data from the logs. This type of method could be used in the mitigation of the DDoS attack by prohibiting the request from a particular source, that was previously identified and once found the server could automatically terminate the connection request from that source and can get rid of the attack.

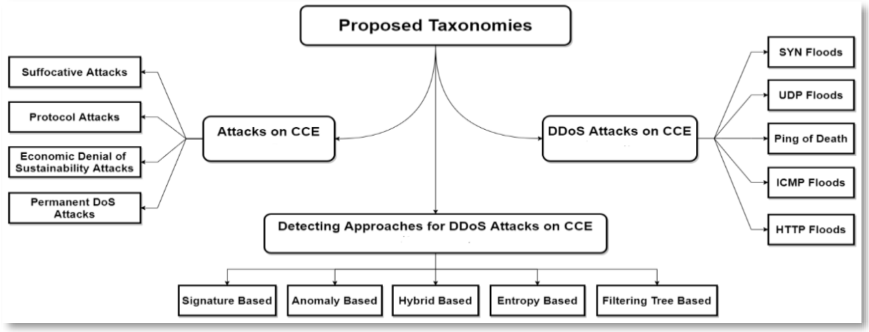


Figure 4. DDoS attack and attack mitigation takeaway (Source: Alashhabet *al*. 2022)

- ***"Routing"*** This method can be used in the cloud system, when identified a large number of connection requests to the cloud server, it can be redirected to or routed to another system employed with a honey pot to safeguard the server. Xu *et al. (*2021) highlight that this service can be used in the cloud system and exhibits the results as this method subsequently routes the malicious traffic and saves the system. With such a configuration in place, the attacker could jam the server with multiple requests and will be impossible for making the target system to get halted.

- ***"Response":*** By using the firewall configured in the server, the system would be able to block the malicious requests. Configuring ***"WAF (Web Application Firewall)"*** on the system could be able to identify and mitigate the attack by applying the rules like Traffic Filtering and Rate Limiting, bot protection, IP reputation and geo-location blocking. This would be used in the system as the attacker has been able to exploit the "***network time protocol services NTP".***

- ***"Adaptation":*** This type of protection process includes the bulk number of data that has been saved from the previous session. As this session has been able to find the pattern of the attack and the understanding of the system that would be under attack. Thus, making the system much more intelligent in determining the DDoS attack. The server will make use of the data logs to learn of the attack logs and this could be helpful in creating a protection much stronger for a similar attack; it would be able to stop the attack from even getting access to the system.

2.8. Present research limitations

The advantages and the limitations of each of the papers taken in the literature review of the report are tabulated in a table and are shown below.

Articles	Pros	Limitations	Other Metrics
Verma *et al*. (2021)	Outlines a cloud-based mitigation solution for "DoS/DDoS" attacks that focuses on resource isolation through implementation at the kernel level, control groupings (cgroups), namespaces, as well as "Seccomp" filters.	The assessment uses a small number of simulated assaults; may not accurately represent real-world conditions.	The success rate of attacks both with and with no mitigation. The resource overhead that the mitigating measures cause.
Sharma *et al*. (2021)	Explores the use of "Seccomp" filters for monitoring system calls and limiting access to resources, potentially detecting, and reducing DoS/DDoS attacks.	Extensive testing is needed to evaluate efficiency against various DDoS attacks.	Seccomp filters effectively detect assaults, but the long filtering process degrades performance.
Radain *et al*. (2021)	This emphasizes the usage of "cryptographic encryption" for safeguarding information and introduces the idea of "fog computing" for storage isolation in cloud computing.	Fog computing, essentially lacks empirical examination and real-world implementation along with its usefulness and efficacy in practice require proof.	Fog computing encryption and decryption speeds vary for various data sizes and it’s a long process.
Mahajan *et al*. (2022)	Provides reference Resource isolation diagram for cloud computing, enhancing security and limiting access.	Failed to provide potential security flaws within the authentication as well as access control procedures.	The success rate of authentication under normal as well as attack settings.

Sattar and Matrawy, (2019)	This includes a detailed graphic of the resource isolation procedure as well as a description of the architecture used in cloud computing. It emphasizes the utilization of different layers of cloud computing.	Performance evaluation under significant traffic loads is lacking and failed to get the system performance during the attack.	The impact of resource isolation strategies on system performance has not been mentioned.
Akanji *et al.* (2021)	This demonstrates a network topology that can be employed to ensure proper resource isolation.	The diagram's simplified network architecture may not accurately depict the intricate nature and variety of real-world network systems.	Architecture is pretty light to be considered in the cloud migration scenario.
Kumar and Somani, (2023)	Proposes service separation as the mitigation strategy for resource isolation.	There are no precise implementation details or evaluations of the suggested initiatives, therefore their usefulness is uncertain also the success of the attack as well.	The rate of adoption of recommended measures within real-world cloud systems is not exhibited.
Zhijun *et al.* (2020)	Demonstrates real-world instances of "attacks using DDoS and the impact that they have upon cloud services.	There has been no quantitative assessment of the monetary effects of "DDoS" assaults on cloud services.	Financial damages caused by DDoS assaults are quantified.
La *et al.* (2021)	The implications of "DDoS" assaults on cloud computing systems are highlighted, also provides useful details about the effects of resource isolation on cloud infrastructure.	More research is needed to determine the extent of the suitable mitigating measures.	Research is more confronted with financial losses.

Li *et al.* (2019)	This explains how the "SYN" flooding attack affected a cloud system. It emphasizes the importance that security monitoring tools like "IDS" and "SIEM" are for identifying and preventing "DDoS" assaults.	There has been no examination of the usefulness of the suggested security monitoring technologies in real-world circumstances.	IDS and SIEM tools aid to strengthen mitigation.
Xu *et al.* (2021)	Postulated the DDoS attack mitigation strategies by resource isolation, including s detection, routing, reaction, and adaptability. Failed to get the success rate of the proposed method.	There has been insufficient testing of the versatility and efficacy of the recommended mitigation techniques in real-world "DDoS" assault scenarios.	The mitigation process's efficiency under large traffic volumes needs to be validated.

3. Research Methodology

3.1. Research design

Besides this descriptive research design, here it has mainly discarded the exploratory research design because this research is conducted to integrate the existing ideas in order to get a proper solution for cloud resource isolation and provide generalized information through an explanatory research design (Shaaban *et al.* 2019). Thus, with the help of this descriptive research design, it helps to collect vast information from different secondary sources (qualitative data analysis) about resource isolation as a mechanism for mitigating DDoS attacks in cloud computing systems.

3.2. Data Collection

The data collection process is used for gathering data as well as process parameters, information for the implementation and completion of the report. The type of data collection method used for the report is the “Secondary data collection method”. Various types of data that would fit the category of qualitative, quantitative, and mixed data will also be collected for the tests and deducting the conclusions out of it. This raises the quality of the study by pushing researchers to investigate further resource isolation. This data would be provided in the roadmap for the application of DDoS attacks that could be determined in the analysis. The investigation includes a collection of diverse literary concepts from existing research work in various "journals, and articles" by considering some important terms linked with the use of resource isolation in securing the cloud resources.

3.3. Implementation Strategy

i. Isolation of Virtual Machines

At the level of the ***“Cloud Service Provider (CSP)”***, machine isolation must be established initially. The VMware software's virtual machine deployment and isolation techniques are being used for the implementation of the project. These techniques spread virtual machines among actual servers as well as data centres, guaranteeing workload distribution and avoiding DoS/DDoS attacks that overload an individual server (Kaur Chahal *et al.* 2019). To provide a safe and isolated setting for each of the client's virtual machines, the CSP sets up and maintains the virtual computer's isolation settings, such as segmentation of the network, rules for firewalls, and access restrictions.

ii. Resource Scheduling and Allocation

The CSP makes use of the allocation of resources and scheduling methodologies to handle resource allocation issues and prevent excessive allocation to a particular virtual machine. The VM management capabilities provided by VMware software allow the placement of suitable resources in accordance with workload needs and thereby able to meet the requirements for the project. In the real cloud, CSPs can use resource pools to allocate virtual machines with CPU, storage space, memory, and internet connections in accordance with specified regulations. This guarantees effective resource use, improves performance, and lowers the danger of DoS attacks brought on by running out of resources.

iii. Token-based or role-based access control

The solution uses a role-based or token-based approach to access management to improve it and add another layer of protection. To do this, external credential management needs to be integrated, including the use of RSA tokens for VPN-based access to the online storage facility. CSPs sets up ***“Role Based Access Control (RBAC)”*** regulations that assign rights and actions to different user roles. The solution makes use of VPN technology and encryption standards to create an encrypted link between the user as well as the online storage system, protecting the privacy and integrity of all information while in transit. Kali-Linux virtual machine loaded in the VMware can be used to simulate and explain the RSA token implementation, which exactly matches the real-world implementation.

iv. Testing and Evaluation of the Proposed Method

The testing goes through a thorough assessment and testing process after the implementation of the abovementioned security measures. This entails carrying out vulnerability analyses, penetration tests, and load tests to confirm the viability of the proposed security mechanisms. In order to find any possible flaws or vulnerabilities within the network's infrastructure in the real world, CSPs use vivid security testing methods and devices will be used to ensure the system is out of all vulnerabilities and safe. The necessary modifications and changes are done in accordance with the evaluation of the obtained results to guarantee the cloud resources are being properly safeguarded and secured from attacks.

3.4. Data Analysis

The data that would be collected as the result of running the DDoS attack would be analysed in order to find the impact of the attack that would be provided. This analysis would be able to provide the file as the data would be able to identify the attacker's IP and request type. This analysis would be helpful in identifying as well as the mitigation process that could be applied. The application of VMware would be used in the representation of a DDoS attack and the analysis of the attack would be done by using Wireshark. Wireshark would be helpful in the visualization of the type of attack by monitoring the inbound and outbound data traffic on the system and the analysis of the attack done in the victim’s system (Shah *et al.* 2022).

3.5. Ethical Consideration

The output that would be expected in the report would follow the error-free and ethical consideration of the type of data collection taken into consideration for the report. This research mainly consisted of secondary data collection that could be used in the understanding and well as the support for the processing of the DDoS attack. The information regarding the DDoS attack has been taken from valid sources. The protection of the data is effective in order to protect the data which could not be manipulated after collecting and analysing for ***“resource isolation to mitigate denial-of-service and DDoS attacks in cloud computing”***.

4. Design and Implementation

4.1. Design of the Proposed Resource Isolation Methodology

The design of the lab setup for the project is done so as to mimic the exact same arrangement and configurations in the real could system. The lab configuration is basically a simulation done in VMware. VMware has been used for virtualization and to run the Kali-Linux and the Ubuntu VMs simultaneously in a controlled environment to launch and study the attacks, and also to analyse how well the resource isolation can help in mitigating the DoS/DDoS attack in the cloud.

The components used in the network diagram are two VMs: namely Kali Linux as the attacker VM and Ubuntu VM as the victim VM. Kali Linux is selected as the atter VM because of the wide range of ethical hacking and pen-testing tools that are built-in in Kali Linux. The Kali VM is pretty lighter on the host machine, and it would require minimum hardware requirements to function and be easily customisable. Another peculiarity of Kali Linux is that we can easily get support resources, like the different application packages that are required for the lab environment.

Ubuntu VM, which basically works on the Linux Foundation (based on Debian) is chosen as the victim VM or it is the same as the system hosted in the cloud. Ubuntu OSs are the most commonly used operating system in servers because of their easy user interface and stability. It is a reliable option for a variety of cloud builds because of its enormous resources, optimisation for cloud settings, and dependability. The main reason for choosing Ubuntu as the victim VM is because if it is being attacked the attack would be much similar to the one that happens in the real cloud itself. Then the proposed mitigation steps can be applied to the VM, and the results will be much alike to the real cloud environment. The proposed resource isolation can be implemented at the CPS level and at the client side for improved mitigation of the attack. The networking diagram of the lab setup used for the proposed implementation is shown below.

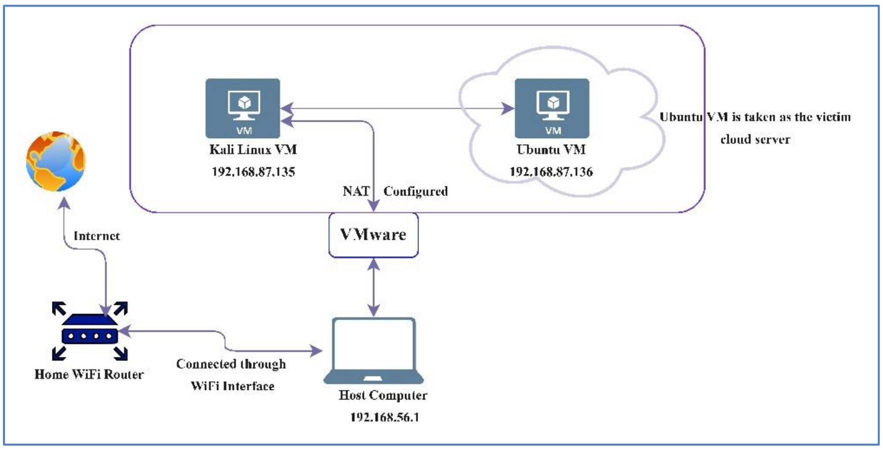


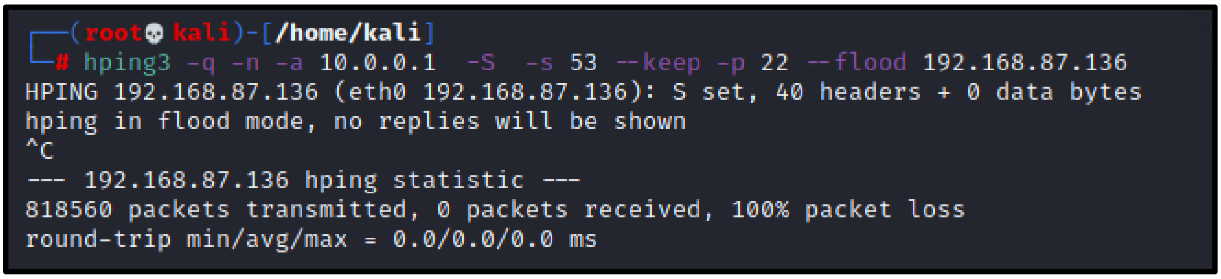
Figure 5. Network Diagram of the Lab used in the implementation.

Both the VMs are loaded in the VMware and NAT is configured to enable the communication between the two VMs and to the host system. The host system used here is having the specifications of core i7 with 16GB RAM and 512 GB SSD running on Windows 11. The host machine is directly connected to the home router which provides internet access. The VMs are first connected to the internet for updating all the built-in packages and then the connection is terminated after use, and this is depicted as the dotted lines in the diagram.

4.2. Implementation of proposed Resource Isolation

The project is accountable to deal with the circumstances linked to the growth of a certain system that can function as the sole answer for DDoS attack mitigation inside the cloud by implementing the process of resource isolation. A number of details are associated with the variables approach to display the overall process of a DDoS attack and further illustrate the possible solutions. The all-around task is exemplified and experimented with the employment of Kali Linux. The Kali Linux platform is admiringly efficient to perpetrate penetration testing.

The DOS incursion commenced with the identification of the target system. The target system taken here is the Ubuntu operating system and the IP address has been identified as 192.168.87.136 using the ‘ifconfig’ command’. With the IP of the target machine, the attacker's IP address also has been identified and for the identification the “ifconfig” command has been utilized. The IP address is 192.168.87.135. Prior to starting to attack, the ping command can be used to ensure the availability of the victim system. The availability of the victim system is confirmed by the reply for the ping and the detailed image of the reply is shown in the configuration manual.



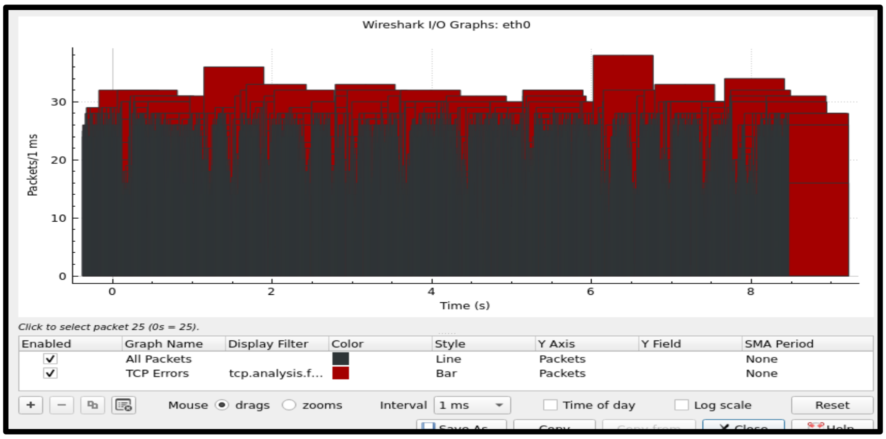
```
┌──(root💀 kali)-[/home/kali]
└─# hping3 -q -n -a 10.0.0.1  -S  -s 53 --keep -p 22 --flood 192.168.87.136
HPING 192.168.87.136 (eth0 192.168.87.136): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 192.168.87.136 hping statistic ---
818560 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

Figure 6: Performing DDoS attack.

The Hping3 command is reliable for performing DDoS attacks. SYN flood is nothing but the most manipulated scanning procedure, and the basis for this is that it is the most perilous. SYN flood is responsible for transmitting a massive quantity of TCP packets with solely the SYN flag on. The reason indicates an SYN packet is typically operated for opening a TCP link, the target’s box intends to attempt to open all these links. These linkages, kept inside the association tables, will further remain open for a particular part of the time at the time while the assailant persists to attack with SYN packets. Once the victims’ connection table is filled up, it will not additionally receive any further connections and thereby the attacker forces the system to get halted and making unavailable for users, resulting in the DoS/DDoS attack. The above picture is acting as evidence to direct the SYN flood attack to the respective target IP.

This is mentionable as a number of total 15000 packets (-c 15000) whose size is 120 bytes (-d 120) individually. In that case, the SYN Flag (-S) has been allowed, using a “TCP window size of 64 (-w 64)”. For the identification of the incursion to the respective target’s “HTTP web server”, port 80 (-p 80) has been specified and operates the “--flood flag” for transmitting packets instantly. The “--rand-source flag” is responsible for generating IP addresses with spoofed anomalies to hide the actual commencement and bypass identification but at the identical period stopping the target’s SYN-ACK response packets from surpassing the attacker is essential. Once the attack is launched, the need is to implement specific methods for inspecting the transferring packets. Henceforth, the Wireshark software can be utilized to track TCP packets.

Once the attack is begun, it is competent to observe all functional data packets. Additionally, the packets are liable for displaying the data packet for enacting pen-testing and influencing exposures. This is notable that the filter for SYN packets is utilized without a divulgence operating the subsequent filter: “tcp.flags.syn == 1 and tcp.flags.ack == 0”. The packet transfer statistics from Wireshark are attached in the configuration manual.



Click to select packet 25 (0s = 25).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	All Packets		■	Line	Packets		None
☑	TCP Errors	tcp.analysis.f…	🟥	Bar	Packets		None

Mouse ● drags ○ zooms Interval 1 ms □ Time of day □ Log scale

Figure 7 Packet Statistics Showing the detected DDoS Packets

Further, the Wireshark I/O graph has been illustrated to display visual articulation of the uptick present in traffic. It can be identified that there are many unusual TRRC P traffics, which is actually a sign of a TCP flood attack, and the same is interpreted in the I/O graph as TCP Errors.

By separating user requests as well as isolating server resources, RSA token incorporation improves safety in a distributed system. Users needed to enter their login information, containing the RSA token, in order to get into the application in the cloud. Before providing access, the RSA tokens implementation verifies these credentials. Only those with the proper authorization can access certain cloud services thanks to this two-factor authentication approach. The system can distinguish between trusted users and prospective dangers by grouping user requests, effectively protecting cloud resources from unwanted access. This connection helps create strong security architecture, protecting private information and reducing risks in a real-world cloud environment.

4.3. Data Analysis

The typical data packets seen at the same period of time had a range of sizes and lacked the SYN flood assault characteristic. These packets presumably reflected valid data transmission and did not constitute a part of the assault. The study shows the effects and features of a distributed denial of service (DDoS) assault on the target system by clearly differentiating between the maliciously sent packets and normal data packets. Security techniques, such as resource separation and RSA token integration, may be used to effectively secure cloud resources against such attacks and guarantee the system's capacity and dependability by comprehending and isolating the assault packets.

5. Evaluation of the Proposed Mitigation Technique

The massive effect of a DDoS attack is very difficult to stop while for minimal attacks it is possible to detect the same and further perform the mitigation. A very much possible strategy to stop DDoS syn attacks within the cloud is the SYN cookies protection guideline in the "PaaS layer". As per the need of the project, it is mentionable that there are certain strategies that can be implemented in order to provide mitigation based on the current aspect. In such a scenario, the user can block those ports for a certain time for protection with the use of iptables.

In the first step, for the utilisation of the iptable, there is a need of creating a syn-flood chain and further, jump into the syn-flood chain while Wireshark has detected a syn packet. Henceforth, the packet rate is able to be restricted to two per second utilising a specific six-per-second burst. “Iptables -N thyl-syn-flood” command can be used to create a syn flood chain. Further, “iptables -A thyl-syn-flood -m limit -- 2/s” command is responsible for limiting the packet transfer.

As the next step, “iptables -A thyl-syn-flood -m recent --name blacklist_180 --set -m comment --comment "Blacklist source IP" -j DROP” command can be utilised for the banning the port for a few minutes as a mitigation strategy. These banning can help by making the vulnerable port unavailable for the attacker and thus can mitigate the attack. But in the real cloud system, all these processes should be automated, and it should leave a message to the legitimate user that the system has been attacked and the service will be resumed only after a couple of mins. In the worst case of setting up this method, the attacker can keep on flooding the port though it is not available and will be having the possibility to get attacked again. Rate limiting can be applied in this situation to limit the amount of inbound traffic to the system. Rate-limiting configurations can be applied in the NIC.

Rate limiting is a process to control the amount of all-around traffic unrestrained to an unusual “Network Interface Controller (NIC)”. It is capable of accomplishing to mitigate the opportunity of the tumbling target connected to a DDoS attack. Iptables can be effective in blocking unusual packets. Additionally, this can be competent to block packets from confidential subnets as well. Another try for the detection after dropping the data from the Ip table will result in a null outcome. As it has already blocked all TCP, ports and it will not further detect the TCP protocols for a few durations.

Server-side intervention associated with SYN cookies is nothing but a widespread strategy to combat SYN flood invasions. In the current scenario, upon sending SYN-ACK packets back to the client, the server discards the actual SYN entry from the queue. In case, the ACK message ultimately anchors, the server reconstructs the SYN packet operating a cryptographic practice. After the creation of DDoS attacks through VMware, it directly evaluated different issues faced by users in several outages of memory loss that bring the resources back online.

Among the different types of DDoS attacks, it has particularly acted as an SYN flood that mainly provides a short burst of “SYN messages” in different ports, which directly create insecure connections and as a result complete crash of the server. Further, it is also evaluated that after sending a huge amount of TCP packets it directly creates of victim box by selecting the open connection (Muakhori *et al.* 2020). This collection has been stored in connection with a certain number of tables and continues to flood by using SYN packets. Further, the use of “limited syn flooding” makes creating of a greater firewall system than a probable syn flood. Besides this, receiving unaware of attacks through this server that make create legitimate requests further establishes the communication system.

Moreover, after identifying this DDoS attack it has mainly created of direct attack of SYN flood which mainly does not spoof IP address that mainly attacks a single source of the device in the selection of IP address providing a highly vulnerable mitigation process. In order to mitigate this direct attack, it has mainly achieved firewall rules that mainly stop sending outgoing packets that mainly create malicious functions in the user's machine. Besides this, also blocks of each IP address are carrying malicious packets to the target systems.

Moreover, it is also created of spoofed attack that directly spoofs IP addresses in different types of SYN packets that mainly create mitigation efforts after identifying difficulties in the system. On the other hand, those packets that are spoofed are only traced back to resources and used ISP in order to mitigate this. Further, in the case of this research to mitigate this particular DDoS attack that mainly uses iptables for providing proper filtering of packets sent by SYN and creating block sources of ports. It mainly creates a block of IP addresses and forwards these packets through NAT . Using this system is more reliable for blocking "destination ports" and identifying source addresses.

5.1. Confusion Matrix

The confusion matrix has been created for evaluating the efficiency and accuracy of the proposed methodology. 15000 IP packets or SYN flooding requests were used to test the proposed model. The total packets have been segregated as packets successful in attack and those seem like normal traffic, the same has been used for constructing the confusion matrix. By using Jupiter notebook, the accuracy, precision, Misclassification Rate, True Positive Rate, False Positive Rate, True Negative Rate and Prevalence of the model has been evaluated and the outcomes are as follows. Examining our model's DDoS attack packet detection performance in depth reveals significant findings.

The prevalence, at around 73.3%, emphasises the significant presence of attack scenarios in the dataset. A precision of around 85.4% shows a high prediction accuracy for recognised attacks. The model efficiently detects non-attacks with a true negative rate of around 55.0%. A false positive rate of roughly 44.9%, on the other hand, implies that there is potential for progress in lowering false alarms. The model excels at successfully recognising assaults, with a true positive rate of roughly 95.5%. Nonetheless, a 15.3% misclassification rate necessitates improving overall classification ability. The model's overall accuracy is acceptable, with an accuracy of roughly 84.7%. *(Refer to the configuration manual for more details and codes of the confusion matrix)*

6. Conclusion and Future Work

6.1. Conclusion

The experiments and analysis that were done have given us useful information on how to implement resource isolation and DoS/DDoS attack prevention in a cloud environment. The experiment, which made use of Kali Linux, successfully illustrated the effectiveness of an SYN flooding attack as a typical DDoS technique. The safety of the system that is distributed proved to be significantly improved by the incorporation of RSA token verification and resource isolation by the implementation and configuration of the Iptables and also the tailored configurations made at the NIC will tremendously reduce the effects of the attack on the system. It is evident from the confusion matrix that the proposed system is having an accuracy of roughly 84.7 and a precision of around 85.4%, which shows the stability of the system. The effective deployment of the RSA token successfully separated cloud resources against malicious activities by validating user credentials and separating incoming requests as legitimate traffic will be allowed to hit the target. The various features of the DDoS assault were highlighted by comparing attack packets to regular data packets, enabling quick identification and mitigation. Incorporating RSA tokens as well as resource isolation acts as a strong security mechanism to protect cloud resources from potential attacks, delivering durable and secure cloud architecture, as is shown in this research.

6.2. Limitations

As there are, other attacks that could be done and multiple attacks done on the server have been much more robust and would require a different protection protocol. The method used for the report has been a straightforward analysis, which is having the limitation to detect and mitigate the spoofed request; that is the attackers are capable of camouflaging the malicious traffic inside a legitimate-looking request. The designed system fails to find the camouflaged packets and the same can be done in the future scope. Thus, the server needed a much more powerful system that would be able to handle the request as well as the implementation of more powerful mitigation strategies that can be more flexible and robust to mitigate all forms of attacks.

6.3. Future Work

The research is conducted mainly focusing on the TCP-SYN flood DDoS attacks. But in the real cloud environment, there are a lot more different dynamic attack methods being used by the attackers. The system should be set up to examine the performance of the system. Those results can be used to find out the details like the reliability of the system, scalability, and performance. Real-time testing is an essential factor since the proposed system has not been built in such a way that it is not able to find out and mitigate the camouflaged malicious requests. Also, the techniques like machine learning can be applied to the suggested method to train the model to learn about the blocked ports and make the checking in a continuous loop to strongly mitigate the attack. The implementation of a powerful mitigation strategy could be used to handle the huge volume of traffic in the real-time environment as well as for different types of attack processes. The isolation of resources could be done in a much wider range addressing the different flavours of the DoS/DDoS attack.

7. References

Akanji, O.S., Abisoye, O.A. and Iliyasu, M.A., 2021. Mitigating slow hypertext transfer protocol distributed denial of service attacks in software defined networks. *Journal of Information and Communication Technology*, *20*(3), pp.277-304.

Alashhab, Z.R., Anbar, M., Singh, M.M., Hasbullah, I.H., Jain, P. and Al-Amiedy, T.A., 2022. Distributed Denial of Service Attacks against Cloud Computing Environment: Survey, Issues, Challenges and Coherent Taxonomy. *Applied Sciences*, *12*(23), p.12441.

Kaur Chahal, J., Bhandari, A. and Behal, S., 2019. Distributed denial of service attacks: A threat or challenge. *New Review of Information Networking*, *24*(1), pp.31-103.

Kumar, A. and Somani, G., 2023. Service separation assisted DDoS attack mitigation in cloud targets. Journal of Information Security and Applications, 73, p.103435.

La, T., Pham, K., Powell, J. and Koch, D., 2021. Denial-of-service on FPGA-based cloud infrastructures—attack and defense. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp.441-464.

Li, Z., Jin, H., Zou, D. and Yuan, B., 2019. Exploring new opportunities to defeat low-rate DDoS attack in container-based cloud environment. *IEEE Transactions on Parallel and Distributed Systems*, *31*(3), pp.695-706.

Mahajan, N., Chauhan, A., Kumar, H., Kaushal, S. and Sangaiah, A.K., 2022. A deep learning approach to detection and mitigation of distributed denial of service attacks in high availability intelligent transport systems. *Mobile Networks and Applications*, *27*(4), pp.1423-1443.

Muakhori, I., Sunardi, S. and Fadlil, A., 2020. Security Of Dynamic Domain Name System Servers Against DDOS Attacks Using IPTABLE And FAIL2BA: Security Of Dynamic Domain Name System Servers Against DDOS Attacks Using IPTABLE And FAIL2BA. *JurnalMantik*, *4*(1), pp.41-49.

Radain, D., Almalki, S., Alsaadi, H. and Salama, S., 2021, March. A review on defense mechanisms against distributed denial of service (ddos) attacks on cloud computing. In *2021 International Conference of Women in Data Science at Taif University (WiDSTaif)* (pp. 1-6). IEEE.

Sattar, D. and Matrawy, A., 2019, June. Towards secure slicing: Using slice isolation to mitigate DDoS attacks on 5G core network slices. In 2019 IEEE Conference on Communications and Network Security (CNS) (pp. 82-90). IEEE.

Shaaban, A.R., Abdelwaness, E. and Hussein, M., 2019, September. TCP and HTTP Flood DDOS Attack Analysis and Detection for space ground Network. In *2019 IEEE International Conference on Vehicular Electronics and Safety (ICVES)* (pp. 1-6). IEEE.

Shah, Z., Ullah, I., Li, H., Levula, A. and Khurshid, K., 2022. Blockchain based solutions to mitigate distributed denial of service (DDoS) attacks in the Internet of Things (IoT): A survey. *Sensors*, *22*(3), p.1094.

Sharma, V., Verma, V. and Sharma, A., 2019. Detection of DDoS attacks using machine learning in cloud computing. In Advanced Informatics for Computing Research: Third International Conference, ICAICR 2019, Shimla, India, June 15–16, 2019, Revised Selected Papers, Part II 3 (pp. 260-273). Springer Singapore.

Verma, P., Tapaswi, S. and Godfrey, W.W., 2021. A service governance and isolation based approach to mitigate internal collateral damages in cloud caused by DDoS attack. Wireless Networks, 27, pp.2529-2548.

Xu, Y., Deng, G., Zhang, T., Qiu, H. and Bao, Y., 2021. Novel denial-of-service attacks against cloud-based multi-robot systems. *Information Sciences*, *576*, pp.329-344.

Zhijun, W., Wenjing, L., Liang, L. and Meng, Y., 2020. Low-rate DoS attacks, detection, defense, and challenges: A survey. *IEEE access*, *8*, pp.43920-43943.