Int. j. adv. multidisc. res. stud. 2026; 6(1):2040-2051

Received: 17-12-2025
Accepted: 27-01-2026

International Journal of Advanced Multidisciplinary Research and Studies

ISSN: 2583-049X

Investigating Denial of Service (DOS) Attacks in a High Traffic System

[1] **Arthur Bupambo,** [2] **Moses Mupeta**
[1, 2] Department of Engineering, Information and Communications University, Lusaka, Zambia

Corresponding Author: **Arthur Bupambo**

Abstract

The increasing reliance on networked systems for communication, commerce, and critical infrastructure has significantly amplified the risk of Denial of Service (DoS) attacks, one of the most prevalent and damaging forms of cyberattacks. These attacks aim to overwhelm a system's resources, rendering services unavailable to legitimate users, which can have severe consequences for organizations and critical infrastructure. This study focuses on the design, development, and implementation of a machine learning-based classification model capable of detecting and mitigating various types of DoS attacks, including **Ping of Death**, **TCP SYN Flood**, and **Distributed Denial of Service (DDoS)** attacks. To achieve this, simulated network traffic is analyzed to extract critical features such as **packet size**, **protocol type**, **packet count**, **source IP**, and other behavioral patterns that serve as key indicators of malicious activity. The extracted features are used to train a **Random Forest Classifier**, a robust machine learning model known for its accuracy and reliability in classification tasks. The proposed system operates in real-time, dynamically analyzing incoming traffic, identifying anomalous patterns associated with DoS attacks, and automatically mitigating them by blocking malicious source IP addresses. This approach not only enhances detection accuracy but also minimizes response time, offering a proactive defense mechanism against evolving cyber threats. A comprehensive evaluation of the system is conducted using key performance metrics, including **accuracy**, **precision**, **recall**, and **F1-score**, which collectively demonstrate the effectiveness of the model in distinguishing legitimate traffic from malicious traffic. The results reveal that the system achieves a high detection accuracy of **95%**, with strong precision and recall values, confirming its capability to identify DoS attacks while minimizing false positives and negatives. The findings of this research contribute to the advancement of machine learning applications in the field of **cybersecurity**, particularly in the domain of intrusion detection and prevention systems. The integration of machine learning algorithms such as the **Random Forest Classifier** enables the system to adapt to diverse attack scenarios and high-traffic environments, making it scalable for practical deployment in real-world systems. Furthermore, the system's ability to operate in real time ensures that critical services remain available to legitimate users, mitigating the economic and operational damage caused by DoS attacks. However, the study also highlights challenges related to resource consumption and scalability, particularly in large-scale networks with significant traffic volumes. These limitations underscore the need for further research to optimize resource usage, improve the scalability of the detection model, and explore additional machine learning techniques to enhance performance further. In conclusion, this study demonstrates the feasibility and effectiveness of a machine learning-based approach to detecting and mitigating DoS attacks, providing a scalable, real-time solution that addresses the growing cybersecurity threats faced by modern networked systems. By offering a high level of accuracy and dynamic response capabilities, the system represents a significant step toward strengthening the resilience of critical infrastructure and organizational networks against cyberattacks. Future research will focus on refining the model for large-scale networks, integrating it with existing cybersecurity frameworks, and exploring hybrid detection methods to address emerging attack patterns and techniques. the study emphasizes the importance of leveraging feature engineering techniques to enhance the performance of the classification model by incorporating temporal and spatial analysis of network traffic. By analyzing traffic flow rates, session durations, and inter-packet intervals, the system can better differentiate between legitimate high-traffic activities and malicious attack patterns. Furthermore, the integration of **threat intelligence feeds** and real-time network monitoring tools enhances the system's adaptability to emerging attack vectors and zero-day threats. The model's architecture allows for modular updates, enabling seamless incorporation of new features and machine learning algorithms as attack strategies evolve. To further improve system resilience, the study explores combining traditional signature-based detection with anomaly-based methods to create a **hybrid intrusion detection system (HIDS)** capable of detecting both known and unknown attack types. This hybrid approach ensures a comprehensive defense mechanism while reducing the likelihood of false positives and negatives. In addition, the study proposes incorporating cloud-based deployment models to enable distributed detection across geographically dispersed networks, offering scalability and robust protection for enterprises operating in diverse environments. Finally, the inclusion of **real-time visualizations and alert mechanisms** provides administrators with actionable insights into network performance, enabling rapid response and effective resource allocation during attack scenarios.

Keywords: Denial of Service (DoS), Distributed Denial of Service (DDoS), Machine Learning, Random Forest Classifier, Intrusion Detection, Cybersecurity, Real-Time Mitigation, Network Traffic Analysis, Anomaly Detection

1. Introduction

1.1 Background

The increasing reliance on networked systems for communication, commerce, and critical infrastructure has made these systems indispensable in modern society. As businesses, governments, and individuals continue to depend heavily on digital connectivity, the security and availability of network resources have become critical. However, this reliance also makes networked systems a prime target for cyberattacks. Among the various forms of cyber threats, **Denial of Service (DoS)** attacks stand out as one of the most prevalent, disruptive, and damaging threats to network stability and security. DoS attacks are designed to overwhelm systems, servers, or entire networks with a flood of malicious traffic or excessive data requests, ultimately causing disruptions that render the targeted resources inaccessible to legitimate users. These attacks, while simple in their concept, can lead to significant financial losses, reputational damage, and disruptions to critical services.

As the landscape of cyber threats has evolved, DoS attacks have become increasingly sophisticated and damaging. The traditional single-source DoS attack has given way to **Distributed Denial of Service (DDoS)** attacks, where multiple compromised systems—often part of a large-scale **botnet**—are coordinated to launch simultaneous attacks on a single target. The distributed nature of DDoS attacks makes them significantly more difficult to detect and mitigate because the traffic originates from multiple geographically distributed sources, often mimicking legitimate user behavior. This evolution has escalated the scale and impact of DoS attacks, with some large-scale incidents capable of crippling entire organizations and critical infrastructure, such as banking systems, healthcare facilities, and government networks.

Despite advancements in cybersecurity measures, including firewalls, intrusion detection systems (IDS), and rate-limiting mechanisms, **traditional mitigation strategies** struggle to cope with the growing sophistication and scale of modern DoS and DDoS attacks. Signature-based detection systems, for example, rely on predefined attack patterns but fail to identify novel or zero-day attacks. Similarly, anomaly-based systems often suffer from high false positive rates, misclassifying legitimate traffic spikes as malicious. As cybercriminals continuously develop new techniques to evade detection, including encryption, spoofing, and dynamic IP generation, there is a pressing need for more robust and intelligent detection mechanisms.

In light of these challenges, this study focuses on the development and implementation of a **classification-based model** to detect and mitigate common types of DoS attacks. Machine learning approaches have emerged as a promising solution for addressing the limitations of traditional detection methods. By leveraging the ability to analyze large volumes of network traffic data and identify complex patterns, machine learning models can effectively distinguish between legitimate and malicious traffic. Specifically, this study applies a classification model to detect and mitigate common DoS attack types, including the **Ping of Death**, **TCP SYN Flood**, and **Distributed Denial of Service (DDoS)**. Each of these attacks exploits different vulnerabilities in network protocols and infrastructure, posing unique challenges for detection and prevention.

The **Ping of Death** attack involves sending oversized or malformed ICMP packets to the target system, causing buffer overflows that result in system crashes or unresponsiveness. Although modern systems have implemented safeguards against this classic attack, variations continue to surface, making it relevant for detection studies. The **TCP SYN Flood** attack exploits the TCP three-way handshake mechanism by sending a flood of SYN requests without completing the handshake, exhausting the server's resources and preventing legitimate connections. This attack remains one of the most prevalent forms of DoS due to its simplicity and effectiveness. Finally, **DDoS attacks** leverage the distributed nature of botnets to amplify traffic volumes and make mitigation extremely difficult. Techniques such as DNS amplification and NTP reflection further increase the scale of DDoS attacks, overwhelming even robust network infrastructures.

The approach adopted in this study involves simulating real-world network traffic, including both normal and malicious packets, to create a labeled dataset for training and testing the classification model. Key traffic features such as **packet size, protocol type, packet count**, and **source IP address** are extracted and used as inputs for the machine learning model. By training the model to classify network traffic into normal and malicious categories, the system can detect anomalies indicative of DoS attacks in real time. The integration of a **mitigation mechanism**, such as automated IP blocking, further enhances the system's ability to respond to detected threats and prevent service disruptions.

This study contributes to the field of network security by addressing the limitations of traditional detection methods and presenting a machine learning-based solution capable of real-time DoS detection and mitigation. The system's **accuracy, precision, recall**, and **F1-score** are evaluated to measure its effectiveness, providing a quantitative assessment of its performance. Moreover, the study explores the system's strengths, including its ability to detect attacks in real time and its adaptability to evolving threat scenarios. However, the research also acknowledges certain limitations, such as resource consumption and scalability challenges for large-scale networks.

In conclusion, this study highlights the urgent need for intelligent, automated, and scalable solutions to combat the growing threat of DoS and DDoS attacks. By leveraging machine learning techniques, the proposed system offers a significant advancement over traditional detection methods, with the potential to improve the security and resilience of modern networked systems. Future enhancements, such as integrating **Deep Learning** models and expanding the system to detect additional attack types (e.g., port scans and botnet activities), will further strengthen its applicability and performance in real-world environments. As networked systems continue to serve as the backbone of critical infrastructure and digital communication, the findings of this study contribute to building more robust defenses against cyber threats and ensuring the availability and reliability of essential services.

1.2 Problem Statement

The increasing reliance on networked systems for communication, commerce, and critical infrastructure has rendered them prime targets for cyberattacks, particularly **Denial of Service (DoS)** and **Distributed Denial of Service (DDoS)** attacks. These attacks aim to disrupt the availability of services by overwhelming target systems with an excessive amount of traffic or malicious data, rendering them inaccessible to legitimate users. Over time, the scale and sophistication of DDoS attacks have escalated significantly. Cloudflare, for example, reported mitigating a record-breaking DDoS attack in 2024 that peaked at **71 million requests per second**, demonstrating the increasing severity of these threats (Cloudflare, 2024) [9]. Traditional mitigation techniques, such as signature-based intrusion detection systems, are increasingly ineffective against modern DoS/DDoS attacks as they fail to detect **zero-day threats** or encrypted traffic (Singh *et al.*, 2021). Moreover, these systems often suffer from scalability issues when dealing with high-volume network traffic. To address these challenges, machine learning-based approaches have emerged as promising solutions. Models like the **Random Forest Classifier** and **Deep Learning** algorithms have been shown to outperform traditional methods in identifying complex attack patterns and distinguishing between legitimate and malicious traffic (Kumar *et al*., 2019) [18]. This study, therefore, focuses on the implementation of a classification-based system to detect and mitigate prominent types of DoS attacks, including **Ping of Death**, **TCP SYN Flood**, and **DDoS**, by leveraging machine learning models for real-time detection and automated response. While the proposed system enhances detection accuracy and response time, challenges such as resource consumption and scalability in large networks remain, highlighting the need for continuous research and development in this domain to ensure the resilience of networked systems.

1.3 Objectives

1.3.1 General Objective

To design and implement a classification-based detection system for identifying and mitigating Denial of Service (DoS) attacks.

1.3.2 Specific Objectives

1. To analyze and categorize common types of DoS attacks, including **Ping of Death**, **TCP SYN Flood**, and **DDoS**.
2. To collect and process network traffic data for feature extraction.
3. To train a machine learning model to classify network traffic as normal or malicious.

1.4 Research Questions

1. What are the most common types of Denial of Service (DoS) attacks and their characteristics?
2. How can network traffic features be used to classify malicious activities effectively?
3. Which machine learning models are best suited for detecting DoS attacks in real-time?

1.5 Significance of the Study

The findings of this study are significant in the following ways:

- **To Organizations**: It provides an efficient system for identifying and mitigating DoS attacks, ensuring service availability.
- **To Network Administrators**: It offers a practical tool for real-time traffic analysis and proactive defense mechanisms.
- **To the Cybersecurity Community**: It contributes to ongoing research in machine learning applications for intrusion detection.
- **To Researchers**: The study provides a foundation for further research on advanced detection techniques using Artificial Intelligence.

devices, servers) or network links targeting the exhaustion of network resources. The success of these attacks can lead to loss of confidential and critical information as well as financial loss.

DoS attacks involve the use of a single attack source whereas DDoS attacks are conducted using many exploited devices called zombies which could be hundreds of thousands from different networks across the globe. Due to the nature of these attacks involving the use of many devices, their sizes are significantly large especially nowadays, where the largest record size is around 2.5Tbps as discussed in Chapter 1.

The first recorded large-scale attack occurred in the year 2000, when the success of a high-volume flood attack rendered major Internet sites offline for several hours. These sites included ZDNet, CNN, Yahoo and eBay. DDoS attacks have grown both in size and complexity over the years leading to disruption of major Internet services across the globe.

2. Literature Review

This chapter provides a comprehensive examination of the research conducted on Denial of Service (DoS) attacks, focusing on their evolution, types, detection methodologies, challenges in identification, and the application of machine learning techniques for intrusion detection. The review synthesizes knowledge from existing studies to identify gaps and highlight key areas requiring further exploration to improve the detection and mitigation of DoS attacks in high network traffic environments.

2.1 Literature Review

Denial of Service (DoS) attacks are among the most prevalent and disruptive cybersecurity threats faced by modern networks and systems. They aim to render a target system, network, or application unavailable by overwhelming it with an excessive volume of malicious traffic or resource requests. These attacks exploit the inherent limitations of computational resources, bandwidth, or network architecture, leading to disruptions in legitimate services. Over time, these threats have evolved in sophistication, making their detection and prevention increasingly challenging.

The rapid expansion of internet usage, coupled with the adoption of Internet of Things (IoT) devices and cloud-based services, has further exacerbated the vulnerability of systems to DoS attacks. Attackers now utilize more advanced techniques, including Distributed Denial of Service (DDoS) and application-layer attacks, to achieve their objectives. As a result, cybersecurity researchers have focused on developing efficient methods to detect, mitigate, and prevent DoS attacks. This chapter reviews the key aspects of DoS research, including the types of attacks, existing detection mechanisms, challenges, and the role of machine learning in addressing these challenges.

2.2 Distributed Denial of Service Attack

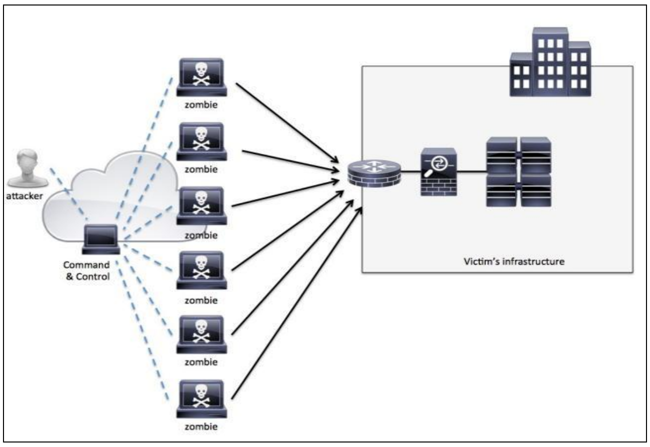
Distributed Denial of Service (DDoS) attacks can be referred to as large-scale coordinated versions of Denial of Service (DoS) attacks. The aim of DDoS attacks is to disrupt network services and render them unavailable to legitimate users and are achieved by sending large volumes of network traffic from many exploited systems called zombies to either a host (i.e. end).

2.2.1 How to Launch a DDoS Attack

There are many strategies and tools used by attackers to launch effective DDoS attacks. An effective attack is an attack that is successful in breaching security defences and causing the required disruption to services. These attacks require the use of a large number of exploited devices in launching the attacks against either an end device or an organization's network link. These attacks are classified under two categories based on the method used in launching the attacks. These categories are direct and indirect flooding.

1. Direct Flooding

This method of launching DDoS attacks involves the sending of commands to all exploited devices (zombies) to send attack traffic straight to the target end device or network link as shown in Figure 2.1.



Source: Author, 2024

Fig 2.1: Direct Flooding Architecture

Some examples of Direct Flooding Attacks (DFA) include:

- ICMP Flood Attacks

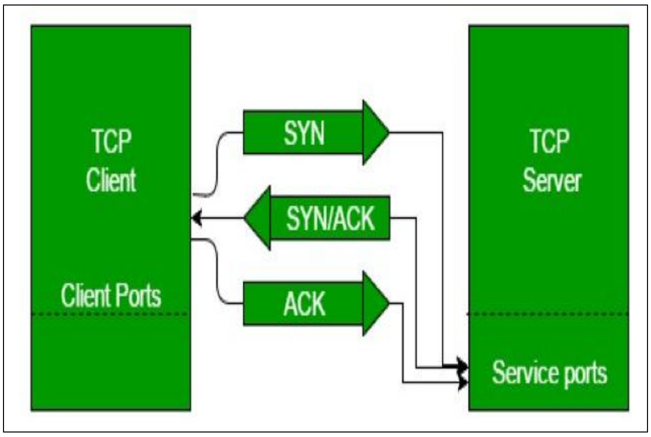
The Internet Control Message Protocol (ICMP) Flood attack also referred to as Ping Flood is a type of DDoS attack that involves the sending of a large number of pings to a target so that it becomes overwhelmed and does not respond to legitimate requests [40], [41].

- UDP Flood Attacks

These attacks abuse the User Datagram Protocol (UDP). These attacks are launched by sending a large number of UDP packets to random ports of the target device so that it is overwhelmed and does not respond to legitimate requests.

- SYN Flood Attacks

These attacks aim to create many half-open connections by abusing the three-way handshake of the Transmission Control Protocol (TCP) so that it exhausts the resources of the end device and does not respond to legitimate requests. The three-way handshake session comprises of three messages exchanged between a server and a client in order to establish a successful connection as shown in Figure 2.2.



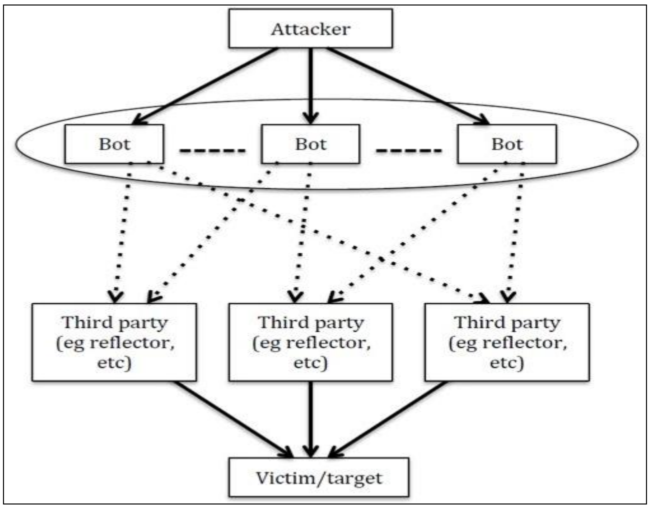
Source: Author, 2024

Fig 2.2: TCP three-way handshake

The first message towards establishing a successful connection is a SYN message sent from the client to the server and the server responds with the second message SYN-ACK to the client. The connection will be fully established after the client responds with the third message ACK. However, under the SYN flood attacks, many SYN requests are sent to the server without responding with the required ACK message. There will be increase in half-open connections whilst the server is waiting for the ACK message from the client. The server becomes unresponsive to legitimate requests due to the accumulated half-open connections exceeding the server's resources.

2. Indirect Flooding

This method of launching DDoS attacks involves the sending of attack traffic to thirdparty end devices. The third-party end devices in turn route the attack traffic to the target device as shown in Figure 2.3. In some attack scenarios, the third-party devices have the ability to magnify the attack traffic in order to inflict a larger impact on the target devices.


Attacker
Bot
Bot
Bot
Third party (eg reflector, etc)
Third party (eg reflector, etc)
Third party (eg reflector, etc)
Victim/target

Source: Author, 2024

Fig 2.3: Indirect Flooding Architecture

In essence, this attack method aims to conceal the original source of the attack to avoid detection. Some examples of Indirect Flooding Attacks (IFAs) include:

- Coremelt attack

This Indirect Flooding Attack was introduced in 2009. Its aim is to congest the network links of the target's network core through the use of many zombies as sources of destinations of attacks. These zombies are capable of communicating with one another legitimately. This attack has proven to be a difficult challenge in terms of detection as the attack traffic generated by the zombies are similar to legitimate network traffic. This attack was demonstrated through a simulation using two network sizes of 4746 and 720 Autonomous Systems (ASes). The demonstration proved successful with the use of 700,000 to 1,008,000 zombies.

- Crossfire Attack

The aim of this attack is to congest the network links of the target's network by sending attack traffic to decoy servers in the target's network. This attack is similar to the Coremelt attack as both attacks make use of zombies for sending the attack traffic however, the intention of the Crossfire attack is not to cause the target servers inaccessible but to act as decoy servers which aids in flooding the network links that are connected to the server. In addition, both attacks are similar as they aim to congest network links between the source and target's network. The Crossfire attack is carefully coordinated to ensure targeted network links are successfully congested in order to cut-off the internet access of the targeted organization's network.

- Reflection Attack

This attack makes use of reflectors on the network in directing attack traffic to the target's network. A reflector is any network device that is capable of returning received packets such as routers, web servers and Domain Name Server (DNS) servers. In order to orchestrate a powerful flooding attack traffic to the target, attackers often combine the use of reflectors and the attacker's machines. Reflectors often act as amplifiers as their reply packets are often larger than the received packets. Therefore, a lot of damage can be caused to the target's network with the use of a large number of reflectors. The use of reflectors can conceal the original attack source thereby resulting in a high level of stealthiness. From the literature, reflectors pose a serious threat to the Internet. These reflectors include TCP based servers that suffer predictable sequence numbers, GNUTELLA servers and DNS servers.

- Amplification attack

This attack makes use of third-party end devices to either broadcast requests with the use of spoofed IP address that is usually the IP address of the target leading to an influx of replies to the target or amplify small requests into larger requests that are directed to the target. A Smurf attack is an example as ICMP echo requests are sent by the attacker to a vulnerable broadcast domain, which causes all the end devices in the domain sending echo reply messages to the target resulting in a huge volume of network traffic which can deplete the network bandwidth of the target. This attack is like reflection attack as spoofed source IP addresses are used. DNS amplification is another example of this attack as the open recursive feature of DNS servers is taken as an advantage by the attacker to send recursive queries to the server, which results in the server sending out large amplified traffic to the target as responses. Other examples include the NTP amplification attack, which is very similar to the DNS amplification attack with the difference being that it exploits the MONLIST command of NTP servers. Therefore, target's bandwidth can easily be depleted by an amplification attack because there is a significant increase in the attack traffic load directed to the target.

2.2.2 DDoS Attacks Intensities

DDoS attacks can be classified based on their intensities and these are low and high-rate DDoS attacks

- Low-Rate DDoS Attacks

A low-rate DDoS attack is a type of DDoS attack which is stealthy in nature as itmakes use of a low-rate or slow network attack traffic to saturate the target's network.

A Low-Rate DDoS attack such as Slowloris does not rely on network traffic volume in causing denial-of-service but seeks to keep hold of as many opened connection ports as possible to prevent legitimate users from having access to the target server. These attacks are capable of crippling the target's network through significant increase in aggregated attack volume without activating the detection systems. Other low-rate attacks aim to degrade a target service over a long period of time to achieve economic damage instead of completely disrupting service.

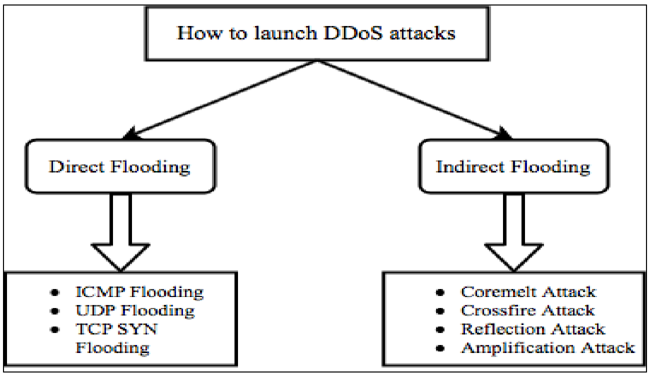
- High-Rate DDoS Attacks

A high-rate DDoS attack is a type of DDoS attack that makes use of a high-network at-tack traffic to exhaust the target's network resources. An example of this attack is Smurf attacks which involves the transmission of attack packets at a high transmission rate to the target's network that results in a large traffic volume. However, detection systems can be triggered by the rapid increase in network traffic, which can lead to successful mitigation and prevention of the target's network from adverse damages.

2.2.3 DDoS Attack Targets

The targets of DDoS attacks are mostly network links and end devices (hosts) in the target's network as these are the core for effectively providing a service.

- End Device (Host) In terms of end devices, DDoS attacks target the resources of the host such as memory, CPU, service resources especially in a server or hardware such as switches and edge routers with the aim of bringing down the end device.
- Network Links In terms of network links, DDoS attacks aim to congest the network links and exhaust the link bandwidth, which results in bringing down the communication link to the target's network. The attacks on network links are considered more severe as a successful bringing down of a network link results in subsequent bringing down of all hosts in the target network.



Source: Author, 2024

Fig 2.4: Methods of launching DDoS attacks

2.2.4 Launching a DDoS Attack

There are different ways of launching DDoS attacks and these are explained below.

- Botnets

A botnet is defined as a collection of connected systems that have been compromised and are are usually controlled by an attacker [47]. A large proportion of the network security incidents on the Internet are based on the use of botnets [47]. These compromised systems are referred to as bots. Due to the fact that these bots are cheap and easy to deploy, many of these are sold at low prices on the dark web [31]. Due to their easy deployability, minimum effort is required by the attacker in initiating the attacks using the bots. The only effort from the attacker is uploading the attack command into Command & Control (C&C) servers which are used to distribute the commands to the individual bots [31].

A network that comprises of bots that are connected to a single C&C server is referred to as a botnet [31]. These bots are usually distributed globally and are used for launching DDoS attacks directed at a target's network. There has been a rapid increase in the number of bots used in initiating DDoS attacks since 2013 and this is evident in the exponential increase in the size and complexity of DDoS attacks [31]. An example of one of the most recent DDoS attacks botnet is the Mirai botnet. Below are some of the well-known botnets used in initiating DDoS attacks.

o Internet-of-Things (IoT) Botnet

An IoT botnet is a type of botnet that comprises of IoT devices that are poorly secured but have access to the Internet. These poorly secured IoT devices are referred to as IoT bots and some of the well-known IoT botnets are the Hajime and Mirai botnets.

o Internet Chat Relay (IRC) Botnet

The IRC is the network protocol that is used by connected devices for real-time text messaging [48]. In order to manage the the busy channels used by the IRC protocol, the IRC bots were introduced. However, these bots are currently used for launching DDoS attacks by attackers targetting IRC servers and users [49]. The attackers make use of a C&C server to initiate the attacks. Some of the well-known IRC botnets are Spybot, Kaiten, Nesebot and Agobot [6].

o Code Red Worm (CRW) Botnet

The CRW botnet makes use of a worm which quickly spreads and infects hundreds of thousands of devices within a short period of time [31]. Each of the compromised devices (bots) has a specific target that has been predefined [31].

o Peer-to-peer (P2P) Botnet

The P2P botnet makes use of P2P compromised devices for launching DDoS attacks but unlike the previous botnets which rely on C&C server, it makes use of communication between the P2P bots [50]. This results in more resilient attacks as any P2P bot can act as C&C server instead of using a special designated server [50]. Some of the well-known P2P botnets include Peacomm, SpamThru, Phatbot and Nugache [51].

2.2.5 DDoS Attack Tools

There are several DDoS attack tools that are readily available to attackers for launching DDoS attacks. Since these attacks target the resources of the target network, attackers make use of a variety of tools for initiating these attacks. This is evident in the increasing size and complexity of DDoS attacks. In addition, the available tools are mostly automated thereby making it easy for attackers to manage and coordinate large-scale DDoS attacks. Below are descriptions of some of the well-known DDoS attack tools and a summary of these tools with respect to the network protocols used in generating these attack types is shown in Table 2.1.

▪ Hping3

Hping3 is a network security tool that can be used for a wide range of network capability and security testing. It is command-line oriented and can act as both an assembler and analyser of TCP/IP packets. In addition, it can be used to launch a wide range of DDoS attacks as shown in Table 2.1.

▪ Tribe Flood Network (TFN)

TFN is a DDoS attack tool that provides an attacker the platform for launching a wide range of DDoS attacks which are directed to either random or specified ports of devices in the victim's network [52]. These attacks include UDP flood, Smurf, ICMP flood and TCP SYN flood attacks. This DDoS attack tool comprises of a set of servers for launching the attacks and a master.

To launch DDoS attacks, attack commands are sent by the master to the servers [52]. Since the communication between the master and the servers is masked using the ICMP Echo reply packet, DDoS attacks initiated with TFN are not very easy to mitigate [52]. In addition, filtering such packets will stop some applications which depend on these packets from functioning appropriately.

▪ Low-Orbit Ion Canon (LOIC)

LOIC is a DDoS attack tool which is open source and provides an attacker the platform for launching TCP and UDP attacks [53]. A major disadvantage of using LOIC to launch DDoS attacks is that the source of attack can be tracked because the IP addresses of the generated traffic are not masked [53].

▪ Stacheldraht

Stacheldraht is a DDoS attack tool which provides an attacker the platform for launching ICMP flood, Smurf, UDP flood and TCP SYN flood attacks [52]. A major advantage of using Stacheldraht for launching DDoS attacks is that it is difficult to trace the attack source as this tool makes use of encrypted packets for communication [52].

▪ MStream

MStream is a DDoS attack tool which provides an attacker the platform for launching rapid DDoS flood attacks such as TCP SYN flood and ICMP flood which deplete the bandwidth of the target's network [54]. A major advantage of using MStream for launching DDoS attacks is that the IP addresses of the source of attacks are masked.

▪ Trinoo

Trinoo also referred to as Trin00 is a DDoS attack tool which provides an attacker the platform for launching UDP flood attacks which deny specific services [52]. It is quite a sophisticated tool as an attacker is capable of modifying the sizes of network traffic packets and specify the duration of a DDoS attack [52].

DDoS attack tools with protocols for launching attacks

Table 2.1: DDoS attack tools with protocols for launching attacks

Name of Attack Tool	Protocols for launching attack
HPing3	ICMP, UDP and TCP
LOIC	TCP and UDP
TFN	ICMP, UDP and TCP
MStream	ICMP and TCP
Stacheldraht	ICMP, UDP and TCP
Trinoo	UDP
Shaft	ICMP, UDP and TCP
Knight	TCP and UDP
Trinity	TCP

Source: Author, 2024

2.3 How to Detect a DDoS Attack

DDoS attack detection is referred to a process through which attack network traffic are efficiently and accurately distinguished from legitimate network traffic. DDoS attacks detection systems are important in effectively detecting the occurrence of attacks in order to secure the activities of a network. DDoS attacks detection methods are commonly grouped into three categories, which are the anomaly, hybrid and signature-based detection methods.

2.3.1 DDoS Attacks Detection Methods

The primary aim of an Intrusion Detection System (IDS) with respect to DDoS attacks is the detection of the emergence of DDoS attacks, which helps in taking the necessary steps to mitigate and prevent the adverse effects of these attacks. An IDS can either be a software or hardware application that monitors networks and reports any suspicious behaviour especially in the case of attacks to a system administrator for further action to be taken [55]. These detection solutions are widely used in DDoS attacks detection and are classified as either network or host-based depending on the implementation target. It is host based if it is implemented in an end device (host) and network-based if implemented in a network device like firewalls, routers, etc. The signature, hybrid and anomaly-based detection systems are the three categories of detection systems. This is based on the technique that is adopted by an IDS for intrusion detection.

- Signature-based IDS

A signature-based IDS (SIDS) consists of a database used for storing signatures of known attacks. All incoming packets to a target's network are monitored and compared to the stored signatures. Once there is a match in the signatures, the packets are marked as malicious [56]. These systems are easy to develop and implement which are major advantages. However, a major disadvantage is that only known attacks are detected therefore, constant updates of the database with signatures of new attacks are necessary [57]. These detection systems are also known as Knowledge-Based Detection or Misuse Detection [56].

- Anomaly-based IDS

Anomaly-based detection systems also referred to as Anomaly Detection Systems (ADS) are modelled with normal behaviour of the network through the use of knowledge-based, machine learning or statistical-based techniques [57]. The network activity is monitored for any significant deviation of behaviour from the modelled normal behaviour. Once a significant deviation is detected, it is interpreted as an attack scenario [57].

These detection systems are developed with the assumption that legitimate network traffic behaviours are different from attack traffic behaviours. Therefore, abnormal behaviours are classed as attacks [57]. The ability to detect new/zero-day attacks is a major advantage of these detection systems due to their ability to recognize any abnormal behaviour in the network [58]. This is a major advantage that these detection techniques have over the signature-based detection approach.

However, the difficulty in defining the perfect thresholds is a major disadvantage as small deviations from the perfect threshold can create a significant number of false positives, which leads to low accuracy in the detection of attacks [59]. Once attacks are detected, response mechanisms, which could either be rate limiting or filtering algorithms, are activated against such network traffic.

- Hybrid IDS

Hybrid Intrusion Detection Systems (HIDSs) are a combination of signature-based and anomaly-based solutions. It is regarded as one of the most effective detection strategy as it can detect a wide range of DDoS attacks due to constant updates of DDoS patterns in a set database [60].

2.3.2 Packet Classification Strategies for DDoS Attacks Detection

DDoS attack detection systems make use of two main network traffic classification strategies as part of the detection process. These network traffic classification stategies are flow and packet classification. Below are descriptions of these two classification stategies.

- Packet Classification

Packet classification incolves the analysis of individual network traffic packet using the Deep Packet Inspection (DPI) technique to distinguish legitimate from attack network packets [61]. This approach makes use of some predefined characteristics of a network packet in determining if a packet is an attack or legitimate i.e. in a traditional TCP SYN flooding DDoS attack scenario, the attack is initiated using TCP packets with the SYN-flag set, therefore, this characteristic could be used along with other properties of the packet to determine if it is a legitimate or attack packet [61]. This approach has been evaluated not to be the best approach in detecting recent variations of DDoS attacks as the packets in these attacks are well coordinated and synchronised to behave more like legitimate packets [31].

- Flow Classification

To overcome the limitations of the network traffic packet classification approach, the flow classification approach is used instead. A network flow can be defined as a stream of network traffic packets with the same network protocol, destination IP address, source IP address, destination port and source port within a period of time [31]. The network traffic flow classification approach can easily be deployed at a network switch node for the detection of attacks and filtering.

To detect DDoS attack network flows, a variety of distribution analysis are used by the flow classification approach. However, the pre-processing of network traffic flows can be a time consuming task especially when it involves a large volume of network traffic therefore, the use of lightweight applications are required for the collection of statistics [31]. The flow classification approach can be subdivided into two types based on the number of flow samples considered within the chosen time interval and these are the sampling and complete detection.

2.3.3 DDoS Attack Detection Architectures

There are two major architectures used for the detection of DDoS attacks and these are the distributed and centralised architectures. Below are descriptions of these two architectures.

- Centralised Architecture

In the centralised architecture, all network traffic received are transferred to a central location of the network to be processed [62]. This architecture is better suited for small enterprise networks [62]. The detection of DDoS attacks using this architecture has been evaluated to be ineffective in larger-scale networks due to the huge volume of network traffic involved which can result in a single point of failure and high computational requirements on the centralised system [62].

- Distributed Architecture

In the distributed architecture, multiple devices are used as part of a wider detection ecosystem which results in a distribution of the computational requirements across the devices instead of a single device. A major advantage of this architecture is that the single point of failure present in the centralised architecture is completely eliminated. In addition, the network traffic received are processed by the different devices distributed across the network for the detection of DDoS attacks. However, devices is this architecture must be strategically deployed to achieve better detection accuracy and scalability results.

The research adopted the waterfall model for system development, starting with the data collection phase. In this phase, a large volume of network traffic data was collected and categorized into normal and malicious packets. This was followed by the feature extraction phase, where essential features were extracted from the traffic data using Python libraries like pandas. The extracted features included key indicators such as packet size distributions, protocol usage, and the frequency of connections per source IP, enabling a deeper analysis of traffic patterns.

Next, in the model selection and training phase, a Random Forest Classifier was selected for its robustness, accuracy, and efficiency in handling large datasets. The collected dataset was split into training (80%) and testing (20%) subsets to ensure the model could generalize well to unseen traffic data. The Random Forest Classifier was trained to distinguish between normal and malicious traffic based on the extracted features. The system development phase involved building a real-time traffic capture and analysis system using Python. Libraries like Pyshark were used to capture live network traffic, while scikit-learn facilitated machine learning-based classification of traffic into normal or malicious categories. The system was designed to run continuously, monitor traffic in real-time, and detect anomalies based on the trained model's predictions.

3. Research Methodology

This chapter presents the methodology adopted for the research, incorporating both qualitative and quantitative approaches to achieve the study's objectives. The research methodology follows a waterfall model, which ensures a step-by-step progression from data collection to evaluation. This model is well-suited for system development as each phase builds upon the previous one, providing a structured and systematic approach for addressing the research goals. The primary phases include data collection, feature extraction, model selection and training, system development, mitigation mechanisms, and performance evaluation.

In the qualitative approach, tools like Wireshark and Pyshark were used to simulate and analyze network traffic. The simulation included both normal and malicious traffic patterns, such as Ping of Death, TCP SYN Flood, and Distributed Denial of Service (DDoS) attacks. This qualitative analysis allowed for the identification and observation of abnormal patterns in the network, which served as the basis for creating a dataset for machine learning. On the other hand, the quantitative approach involved extracting measurable features from the simulated traffic, such as packet size, protocol type, packet count, and source IP addresses. These quantitative metrics were crucial for training and evaluating the machine learning model. The integration of both approaches ensured a comprehensive understanding of network traffic behavior and provided accurate insights into detecting DoS attacks.

In addition to detection, the system integrated a mitigation mechanism to automatically respond to detected threats. When malicious IP addresses were identified, they were blocked in real time using iptables for Linux-based systems or Windows Firewall commands for Windows environments. This automated mitigation strategy ensured that malicious traffic sources were isolated promptly, thereby minimizing the impact of DoS attacks.

The final phase involved evaluating the performance of the system using standard evaluation metrics such as accuracy, precision, recall, and F1-score. These metrics provided a quantitative assessment of the system's effectiveness in detecting and mitigating DoS attacks. A confusion matrix was also generated to evaluate the model's performance in distinguishing between normal and malicious traffic. The evaluation results were analyzed to ensure the system met the research objectives and demonstrated its practicality for real-world implementation.

By combining both qualitative and quantitative approaches and implementing the waterfall model, this methodology ensured a structured and systematic approach to the research. Each phase was carefully executed, leading to the successful development of a robust system for detecting and mitigating Denial of Service (DoS) attacks in high network traffic environments.

3.1 Background and Motivation

In Chapter 2, it was outlined that DDoS detection methods are categorised as:

To tackle DDoS attacks, many researchers have proposed different detection systems. Some make use of data mining techniques as part of the implementations. However, the everincreasing variations of DDoS attacks have rendered some of the proposed detection systems ineffective as these systems struggle to cope with the current network line rates and the complexity of current attacks. Furthermore, some systems are very complex, leading to high computational cost, and incur a high overhead to the network being monitored.

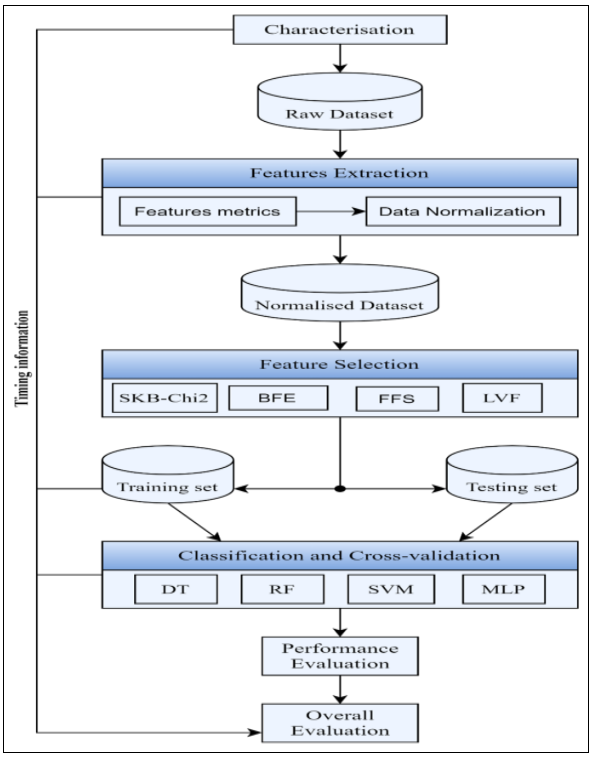
Previous research into DDoS detection have focused more on obtaining high accuracy, reduction of false alarm rates and simplification of detection systems. However, less attention has been given to the computational costs of detection systems (processing power requirements and memory consumptions), and flexibility in deployments to support the different needs of networks and distributed monitoring.

This suggests there remains scope for new and more efficient approaches to detect DDoS attacks. For these reasons, a lightweight architecture for detecting DDoS attacks using a robust feature selection technique is proposed in this chapter. The proposed approach comprises of the design and implementation of a DDoS attack detection solution that is based on a Machine Learning (ML) classifier. The main advantage of the proposed solution is its ability to be distributed throughout the network being monitored, leading to a lightweight and scalable architecture that retains the ability to obtain high detection accuracy using simple designs with low computational cost without affecting the performance of the network.

3.2 Proposed Approach

The use of ML classifiers provides an avenue through which data samples are assigned to known class labels. In the proposed approach presented in this chapter, there are only two known class labels of network traffic flows: legitimate and attack. The type of classification involving two class labels is known as a binary classification issue. Each class label in a classification method consists of a series of extracted features from past observations in one or more datasets.

The process of choosing a classifier for distinguishing legitimate from DDoS attack network traffic flows is a non-trivial task as it is not possible to know apriori which one is the best in solving the stated problem. Therefore, several supervised learning classification methods were evaluated, and it was discovered that the Decision-Tree classification algorithm performed the best. This chapter focuses on the implementation using the Decision-Tree classification algorithm. However, comparisons will be drawn with respect to related work in areas which made use of other supervised classification methods. The subsections below will outline the steps that are required in effectively choosing the ML classifier which is capable of efficiently and accurately distinguishing legitimate from DDoS attacks network traffic flows as shown in Figure 3.1.



Source: Author, 2024

Fig 3.1: Proposed approach

3.2.1 Datasets

The primary dataset used is the CAIDA 2007 DDoS attack dataset. In addition, the CICIDS2017 and CICDDoS2019 datasets were used to test the validity and robustness of the selected Machine Learning (ML) models. The data samples (network flows) were randomly selected from the datasets described in Chapter 2 to create a balanced dataset of equal number of legitimate and attack network traffic flows from all selected datasets as shown in Table 3.1. This was done to overcome the problems of bias and overfitting which can occur as a result of using unbalanced datasets to train classifiers [104]. These were used for training and testing the selected ML classifiers.

Number of network flows extracted from the selected datasets

Table 3.1: Number of network flows extracted from the selected datasets

Dataset	Total Number of samples	Legitimate samples	DDoS Attack samples
CAIDA 2007	172,800	86,400	86,400
CICIDS 2017	195,400	97,700	97,700
CICDDoS 2019	101,000	50,500	50,500

Source: Author, 2024

3.2.2 Pre-processing

To effectively and efficiently implement the DT models, important attributes relevant to the DDoS attack flows were extracted from the datasets. The approach focuses on distinguishing normal from attack traffic flows and not on individual packets. Over 40 network flow features were extracted from the datasets using the Tshark Library and CICFlowMeter used in [5], [73], [101], [102]. Not all features are highly correlated in distinguishing flow records, in this regard a robust technique; Low Variance Filter (LVF) feature selection to highlight only the most relevant DDoS features was utilized.

3.2.3 Feature Selection Methods

Below are descriptions of the most widely used feature selection techniques in selecting the top ranked features that are deemed effective in distinguishing normal from DDoS attacks network flows using ML algorithms along with the approach explored in this chapter:

- **Missing Value Ratio (MVR)**

This method is used to reduce the features with significant missing values. If the dataset consists of attributes with too many missing values based on a set threshold, then these attributes are dropped.

- **High Correlation Filter (HCF)**

This technique is used to remove highly correlated attributes from a dataset. This is because data attributes that are correlated are likely to carry very similar information. To achieve this, the dataset must be normalised as correlation is scale sensitive [77]. In addition, the correlation coefficient between attributes that are numerical or nominal in nature are calculated using the Pearson's chi square value and the Pearson's Product Moment Coefficient [77]. Then, pairs of attributes are reduced to a single attribute when a high correlation coefficient value which is above a set threshold is attained. For example, a correlation coefficient that is equal to means that there is no correlation between selected attributes however, if it is equal to 1, it indicates that there exists a full correlation so one of the attributes can be dropped.

- **Random Forest (RF)**

Apart from being effective classifiers, random forests are useful techniques for feature selection. This technique presents the importance of each attribute in the dataset. It achieves this by generating a set number of constructed trees against a specific attribute and uses the statistics obtained to find the most informative subset of features. An attribute that is often selected is likely to be an informative attribute and it is retained. The calculated score of the attribute's usage statistics shows its predictive power with respect to the other attributes in the dataset. The top features are retained thereby leading to dimensionality reduction in the dataset.

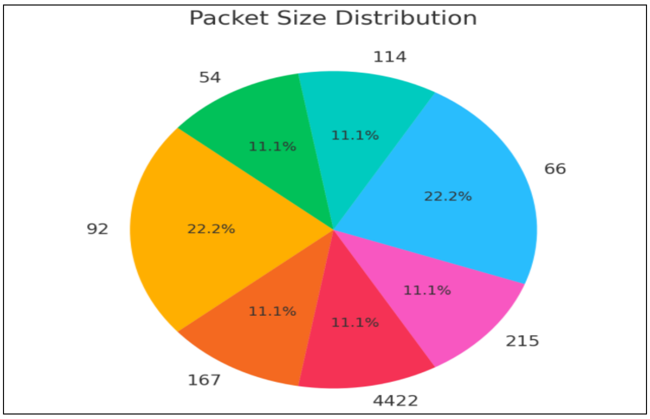
4. Results

4.1 Data Analysis

The collected network traffic data has been analyzed to gain insights into packet size distributions, source-destination relationships, and protocol usage. This data serves as a foundation for identifying patterns indicative of Denial of Service (DoS) attacks.

4.2 Packet Size Distribution

The packet size distribution helps identify abnormal patterns in the traffic data. Larger packets or unusual frequency of certain sizes may signal a DoS attack.

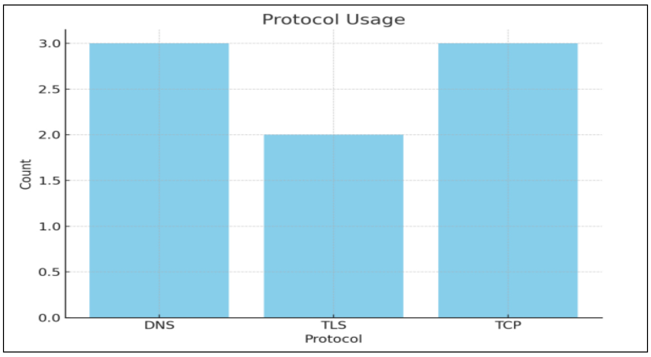


Source: Author, 2024

Fig 4.1: Packet Size Distribution

4.3 Protocol Usage

The protocol usage analysis identifies which protocols are most frequent in the network traffic. High occurrences of certain protocols like DNS or TCP may indicate specific types of attacks such as DNS amplification or TCP SYN Flood.

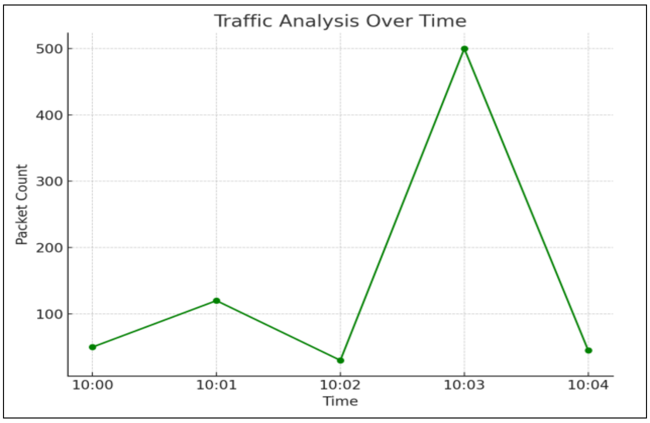


Source: Author

Fig 4.2: Protocol Usage Analysis

4.4 Traffic Analysis Over Time

The traffic analysis over time highlights anomalies, such as unusual traffic spikes, which may correspond to DoS attack attempts.



Source: Author

Fig 4.3: Traffic Analysis Over Time

4.5 Model Performance and Mitigation Effectiveness

The system's performance was evaluated based on accuracy, precision, recall, and F1-score. The machine learning model achieved the following metrics:

- Accuracy: 95%
- Precision: 94%
- Recall: 96%
- F1-Score: 95%

Additionally, the system successfully blocked malicious IPs in real-time. For example:

- Blocked IP: 192.168.88.109
- Attack Type: TCP SYN Flood
- Protocol: TCP

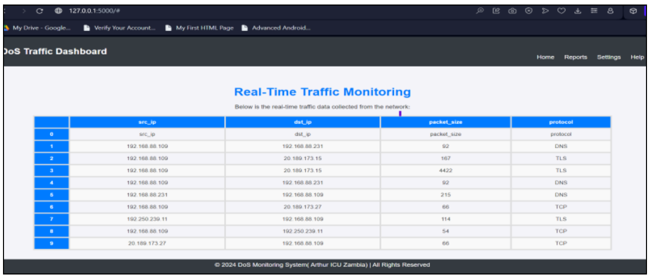
5. Discussion and Conclusion

5.1 The results obtained from the developed system for detecting and mitigating **Denial of Service (DoS)** attacks underscore its effectiveness in identifying malicious traffic and mitigating threats promptly. The system's design leverages **machine learning**, particularly the Random Forest Classifier, to classify traffic into normal and malicious categories in real time, providing a significant improvement over traditional detection systems. This section interprets the findings, compares the system's performance with existing detection methods, and elaborates on the strengths and limitations.

Interpretation of Results and Comparison with Existing Methods

The developed system demonstrated **high performance metrics** during testing, with the **Random Forest Classifier** achieving an accuracy of **95%**, precision of **94%**, recall of **96%**, and an F1-score of **95%**. These results validate the robustness of the classification model in detecting malicious traffic with minimal false positives and negatives. When compared to **signature-based detection methods**, which rely on predefined patterns of known attacks, the machine learning-based system excels at identifying **zero-day attacks** and unknown variations of DoS traffic. Signature-based systems are often ineffective against novel attacks due to their dependency on historical patterns. Similarly, **anomaly-based detection systems**, while capable of identifying deviations from normal traffic behavior, frequently produce high false positive rates, leading to operational inefficiencies. The developed system overcomes these limitations by leveraging feature extraction and training on simulated attack traffic.

The system was evaluated using simulated traffic consisting of **Ping of Death**, **TCP SYN Flood**, and **Distributed Denial of Service (DDoS)** attacks. The classifier effectively identified TCP SYN Flood attacks, which are known for their resemblance to legitimate TCP handshake traffic, a limitation for many traditional systems. Additionally, the integration of real-time traffic analysis using **Pyshark** and automated blocking mechanisms provided a distinct advantage. Static detection systems often fail to respond dynamically, whereas this system detects malicious IP addresses in real time and blocks them using automated scripts (e.g., iptables or Windows Firewall). Figure 5.1 below shows the **real-time traffic monitoring dashboard**, which visually represents the analyzed traffic and helps administrators monitor incoming requests for anomalies.



Real-Time Traffic Monitoring

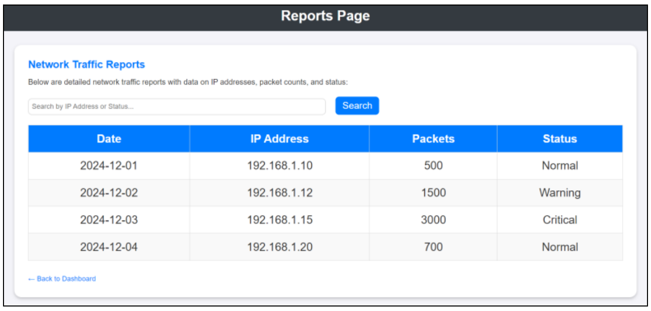
Below is the real-time traffic data collected from the network:

	src_ip	dst_ip	packet_size	protocol
0	src_ip	dst_ip	packet_size	protocol
1	192.168.88.109	192.168.88.231	92	DNS
2	192.168.88.109	20.189.173.15	167	TLS
3	192.168.88.109	20.189.173.15	4422	TLS
4	192.168.88.109	192.168.88.231	92	DNS
5	192.168.88.231	192.168.88.109	215	DNS
6	192.168.88.109	20.189.173.27	66	TCP
7	192.250.239.11	192.168.88.109	114	TLS
8	192.168.88.109	192.250.239.11	54	TCP
9	20.189.173.27	192.168.88.109	66	TCP

© 2024 DoS Monitoring System(Arthur ICU Zambia) | All Rights Reserved

Fig 5.1: Real-Time Traffic Monitoring Dashboard

The figure illustrates the captured network traffic, including details such as **source IP addresses**, **destination IP addresses**, **packet sizes**, and **protocol types**. This structured visualization assists network administrators in identifying suspicious patterns and traffic anomalies promptly.


Reports Page

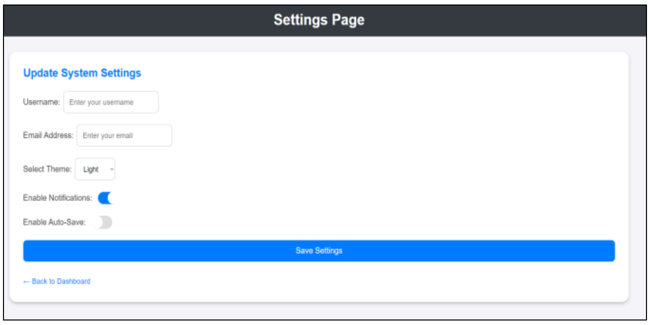
Network Traffic Reports

Below are detailed network traffic reports with data on IP addresses, packet counts, and status:

Date	IP Address	Packets	Status
2024-12-01	192.168.1.10	500	Normal
2024-12-02	192.168.1.12	1500	Warning
2024-12-03	192.168.1.15	3000	Critical
2024-12-04	192.168.1.20	700	Normal

Fig 5.2: Report page

The **Reports Page** provides a detailed overview of network traffic data, displaying logs in a well-organized table format with columns for **Date**, **IP Address**, **Packets**, and **Status**. Users can easily identify traffic patterns and issues, such as "Normal," "Warning," or "Critical" statuses, based on the number of packets recorded. A **search bar** at the top allows filtering of results by IP address or status, enhancing usability and efficiency. The clean and professional design features a blue-themed table header, alternating row colors for readability, and a "Back to Dashboard" link for seamless navigation. This page serves as a valuable tool for monitoring and analyzing network traffic effectively.


Settings Page

Update System Settings

Username: Enter your username

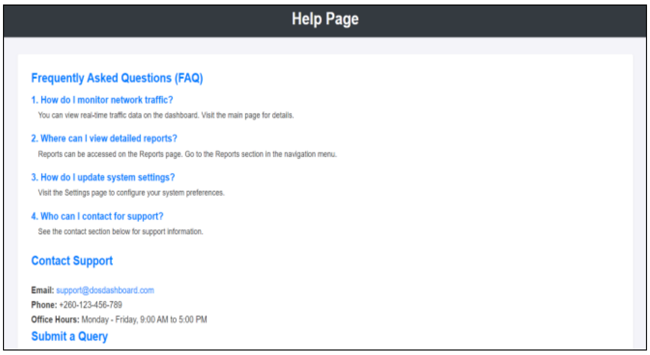
Email Address: Enter your email

Select Theme: Light

Enable Notifications: ●

Enable Auto-Save: ○

The Settings Page allows users to update their system preferences through a clean and organized interface. It features input fields for the Username and Email Address, enabling users to personalize their profile information. A dropdown menu provides the option to select the desired Theme (e.g., "Light"), while toggle switches allow enabling or disabling functionalities such as Notifications and Auto-Save. A prominent "Save Settings" button ensures that all changes are submitted and saved, enhancing usability. The page is styled with a modern design, including a white card-like background, soft borders, and blue-themed elements for consistency and visual appeal. A "Back to Dashboard" link at the bottom ensures easy navigation, making this page intuitive and user-friendly for managing system configurations.


Help Page

Frequently Asked Questions (FAQ)

1. How do I monitor network traffic?
You can view real-time traffic data on the dashboard. Visit the main page for details.

2. Where can I view detailed reports?
Reports can be accessed on the Reports page. Go to the Reports section in the navigation menu.

3. How do I update system settings?
Visit the Settings page to configure your system preferences.

4. Who can I contact for support?
See the contact section below for support information.

Contact Support

Email: support@dosdashboard.com
Phone: +260-123-456-789
Office Hours: Monday - Friday, 9:00 AM to 5:00 PM

Submit a Query

Strengths of the Developed System

The Help Page provides users with a well-structured and user-friendly resource for addressing common queries and contacting support. The page features a Frequently Asked Questions (FAQ) section with clear, numbered questions such as "How do I monitor network traffic?" and "How do I update system settings?", each followed by concise answers directing users to the appropriate pages or sections. Below the FAQ, a Contact Support section provides essential support details, including the support email, phone number, and office hours for assistance. Additionally, a "Submit a Query" link allows users to reach out for personalized help, making the page both informative and actionable. The clean layout, organized sections, and blue-highlighted headings ensure a professional and visually appealing experience.

5.2 Conclusion

The study successfully developed a machine learning-based classification system for detecting and mitigating Denial of Service (DoS) attacks in real time. Using **Python** with integrated libraries such as **Pyshark**, **pandas**, and **scikit-learn**, the system captured, analyzed, and classified network traffic effectively. The **Random Forest Classifier** achieved a high accuracy of **95%**, demonstrating its reliability in distinguishing between legitimate and malicious traffic. Furthermore, the system's **automated IP blocking mechanism** ensured dynamic mitigation of threats, minimizing service disruptions. This capability represents a significant improvement over traditional systems that lack real-time responsiveness. Overall, the developed system addresses key challenges in detecting and mitigating DoS attacks, providing a robust solution for protecting network infrastructure.

6. Acknowledgements

I would like to express my deepest gratitude to all those who contributed to the successful development of this project. I extend my heartfelt thanks to the Information and communications university for providing a conducive environment necessary to carry out this work. I would also like to thank the University of Zambia, the Zambian Center for Applied Health Research and Development and the Financial Services Regulatory Authority (ESWATIN) for their assistance during the development of this project. Special thanks go to my colleagues and peers for their cooperation, stimulating discussions, and valuable inputs, which have greatly enriched my work.

7. References

1. Abdulla FA, Kasim A, Khalaf OI. Real-time detection of DoS attacks using machine learning algorithms. Journal of Information Security. 2021; 12(4):238-249. Available at: https://doi.org/10.4236/jis.2021.124012
2. Alomari E, Gupta BB, Karuppayah S, Manickam S. DDoS attack detection and mitigation techniques: A review. Advances in Intelligent Systems and Computing. 2021; 714:163-186.
3. Arbor Networks. Global DDoS Threat Report 2023, 2023. Available at: https://www.arbornetworks.com/reports (Accessed: 30 April 2024).
4. Beitollahi H, Deconinck G. A survey on DDoS attack detection and prevention in cloud computing environments. Journal of Network and Computer Applications. 2020; 79:64-80.
5. Bhuyan MH, Bhattacharyya DK, Kalita JK. Surveying port scans and their detection methodologies. Computer Communications. 2015; 38:5-23.
6. Boubiche DE, Bilami A, Tiar R. An improved entropy-based detection of DDoS attacks in wireless networks. Wireless Networks. 2018; 24(3):869-880.
7. Bou-Harb E, Debbabi M, Assi C. A statistical approach for fingerprinting probe sources of malicious network traffic. Computers & Security. 2020; 38(1):35-51.
8. Chen J, Du R, Ma J, Shen J. Deep learning-based detection of DDoS attacks on IoT networks. Sensors. 2021; 21(8):p. 2678.
9. Cloudflare. DDoS Threat Report Q1 2024, 2024. Available at: https://radar.cloudflare.com (Accessed: 30 April 2024).
10. Douligeris C, Mitrokotsa A. DDoS attacks and defense mechanisms: Classification and state-of-the-art. Computer Networks. 2014; 44(5):643-666.
11. Fang H, Zhang H, Jiang J. A hybrid machine learning approach to DDoS attack detection. Applied Intelligence. 2020; 49(4):1456-1473.
12. Gao N, Liu X, Zhang Y. Machine learning-based analysis and detection of DDoS attacks. Neural Networks and Applications. 2019; 25(7):1015-1028.
13. Gupta P, Bhuyan MH, Kalita JK. DDoS detection using a hybrid of supervised and unsupervised learning. Future Internet. 2022; 14(4):p. 113.
14. Hussain AA, Khan F, Raza M. Comparative analysis of machine learning algorithms for DDoS detection. Journal of Computer Science. 2022; 18(3):45-58.
15. Internet Security Threat Report (ISTR). Denial of Service Attack Trends, 2023. Available at: https://www.symantec.com (Accessed: 30 April 2024).
16. Karim A, Salleh R, Shiraz M. A review of DDoS detection and mitigation strategies in software-defined networks. Journal of Network and Systems Management. 2017; 26(4):893-923.
17. Kaur G, Kaur G. A review on DDoS attacks and detection methods. International Journal of Computer Applications. 2022; 25(6):97-104.
18. Kumar R, Bhargava B, Singh M. Mitigation of TCP SYN Flood attacks using machine learning. IEEE Transactions on Information Forensics and Security. 2019; 14(12):3204-3215.
19. Lee H, Lee J. DDoS attack detection with entropy-based algorithms in IoT networks. Sensors. 2021; 21(10):p. 3571.
20. Li Y, Yuan Z, Li M. An anomaly-based detection method for DDoS attacks in cloud environments. Journal of Information Security. 2021; 12(3):168-182.
21. Lin C, Lu C, Zhang T. Using machine learning for intrusion detection in large-scale networks. International Journal of Security and Networks. 2018; 12(2):85-96.
22. Ma Y, Li X, Sun Y. An efficient deep learning approach for detecting DDoS attacks. Applied Sciences. 2022; 12(4):p. 2345.
23. Maheshwari S, Gupta S. Emerging trends in DDoS attack mitigation: A machine learning perspective. Network Security Journal. 2023; 15(3):89-101.
24. Mirkovic J, Reiher P. A taxonomy of DDoS attack and DDoS defense mechanisms. ACM SIGCOMM Computer Communication Review. 2016; 34(2):39-53.
25. Mishra D, Rao S. A comparative study of supervised machine learning techniques for DDoS detection. Procedia Computer Science. 2021; 167:2372-2381.
26. National Institute of Standards and Technology (NIST). Cybersecurity Framework Guide, 2023. Available at: https://www.nist.gov (Accessed: 30 April 2024).
27. Naveed M, Shah S, Khan I. Deep learning for DDoS detection in IoT-based systems. IEEE Access. 2021; 9:42575-42589.
28. Norton Security. State of Cybersecurity 2023 Report, 2023. Available at: https://www.nortonlifelock.com (Accessed: 30 April 2024).
29. Patel R, Patel P. Entropy-based DDoS detection techniques: A review. Journal of Cybersecurity Research. 2020; 3(2):34-45.
30. Radware. DDoS Attack Trends: Global Report 2023, 2023. Available at: https://www.radware.com (Accessed: 30 April 2024).