REINFORCING NETWORK RESILIENCE FROM DDOS: A REVIEW OF ADVANCED DISTRIBUTED DENIAL OF SERVICE (DDOS) ATTACKS AND ITS MITIGATION TECHNIQUES.

Honey Gocher[1], Swapnesh Taterh[2], Pankaj Dadheech[3]

Article History: Received: 11.05.2023 **Revised:** 26.06.2023 **Accepted:** 25.07.2023

Abstract

The spread of new technologies like the Internet of Things (IoT) and Software-Defined Networking (SDN) in recent years has extended the distributed denial of service (DDoS) attack vector and created new possibilities for more advanced DDoS attacks on the intended targets. The new attack vector comprises internet-enabled IoT devices that are unprotected and vulnerable. Given the high volume and widespread nature of these attacks, it is very difficult to detect and analyze the frequency of DDoS attacks. There are existing techniques available to mitigate the attacks, including intelligent machine learning models. In this paper we will be discussing emerging techniques and how to implement those new techniques with existing machine learning models to make the detection and mitigation model robust to handle the attacks. We will be discussing advanced anomaly detection and traffic analysis, traffic scrubbing, and Content delivery network (CDN). We will also be discussing ensemble learning to implement an efficient and strong model for the detection and mitigation of Distributed Denial of Service (DDoS) attacks on the internet connected devices.

[1,2]Amity University Rajasthan, India
[3]SKIT, Jaipur, Rajasthan, India
Email: [1]honeygocher@outlook.com, [2]staterh@jpr.amity.edu,
[3]pankajdadheech777@gmail.com

DOI: 10.31838/ecb/2023.12.si5a.0648

1. Introduction:

In recent years, Technologies such as the Internet of Things (IoT) have experienced significant growth, representing a vast network that encompasses a wide range of digital devices. These devices, varying in size from small sensors to large-scale equipment such as mobile phones, televisions, and clinical devices, have become integral components of the physical world. The concept of IoT involves the continuous expansion of physical devices interconnected with the Internet, a trend that has been observed at an exponential rate. The primary objective of IoT is to connect previously unconnected objects, thereby creating a network of smart devices [1].

In the realm of IoT, these embedded smart devices possess the ability to monitor their surroundings, perform common tasks, communicate directly, and make decisions autonomously without human intervention. With its seamless connectivity and convenient communication capabilities, IoT has attracted a growing number of organizations that are embracing this technology. Presently, IoT has evolved into a prominent network encompassing an extensive array of devices, all interconnected to simplify various human tasks.

Based on the research findings, the number of internet-connected devices has surged to over 9 trillion [2][3]. The market revenue for IoT technology exceeded $400 billion in 2022, marking a significant milestone in its growth. Furthermore, the global market for IoT end node products reached a value of 412 billion USD by the end of 2023. Predictions suggest that this figure is expected to soar to approximately 1.8 trillion by the year 2025. These statistics highlight the tremendous growth and potential of the IoT market, indicating a substantial increase in the adoption and utilization of IoT devices in various industries and sectors [3].

The rapid expansion of IoT applications has increased the vulnerability and susceptibility of the technology to attacks. Despite the continuous emergence of IoT service domains, security concerns remain a significant challenge IoT operates across diverse networks that incorporate both large and small devices. However, the small devices possess limited computational power and storage capacity, making it challenging to implement robust protection mechanisms and cryptographic algorithms for security purposes. Additionally, the absence of privacy-preserving algorithms in small IoT devices creates opportunities for attackers to exploit their vulnerabilities and utilize them as bots to carry out malicious attacks [2].

IoT networks are vulnerable to various types of attacks, including the Distributed Denial of Service (DDoS) attacks. In a DDoS attack, the network services and resources are disrupted by legitimate users who are attempting to access them. This disruption is caused by overwhelming the server or website with a multitude of simultaneous requests originating from different locations, thereby diminishing the available bandwidth for genuine users. Detecting a DDoS attack can be challenging since it involves the utilization of diverse locations and devices to carry out the attack. In the context of IoT, attackers employ IoT devices as bots to execute the attack, further complicating detection and prevention efforts. The use of IoT devices as bots presents difficulties in identifying and mitigating the attack since these bots are legitimate IoT devices with low computing power, limited storage, and inadequate security measures, making them susceptible to compromise [3].

The emergence of technologies such as Cloud Computing, IoT, and SDN has brought about significant changes in the architecture of internet networks, presenting new opportunities for attackers to exploit vulnerabilities and carry out Denial of Service (DoS) attacks. Mitigating large-scale Distributed Denial of Service (DDoS) attacks poses a challenge as it is crucial to swiftly respond and prevent potential business and reputational damage for the targeted enterprise organizations. To effectively address this challenge, prompt coordination and collaboration among various stakeholders are essential. These stakeholders include network operators, edge protection providers, Internet service providers, affected organizations, and third-party DDoS mitigation services. Establishing trust and authenticating the involved parties are crucial to enable legitimate actions that can halt the attacks [4]. A blockchain, which serves as a decentralized ledger, offers an efficient and permanent method for recording transactions, making it a potential solution for these kinds of attacks. Fig. 1 shows the high occurrence and future prediction of the DDoS attacks [4].

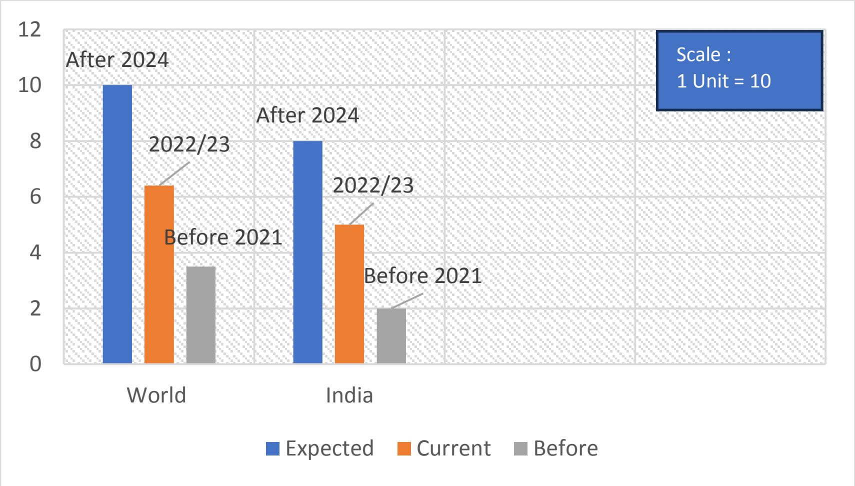

Fig. 1. Graph showing the occurrence and expectation of DDoS attacks [4]

Background:

Confidentiality, integrity, and availability represent fundamental principles in computer security. Both offensive and defensive mechanisms are designed to either compromise or safeguard these principles within digital systems. A Distributed Denial of Service (DDoS) attack refers to an incident where the normal operation of a network or service is disrupted or restricted, rendering them inaccessible to authorized users. Such events often involve attackers exploiting vulnerabilities in the network to undermine, impede, or suppress its functionality. Typically, these attacks aim to achieve their objectives by overwhelming the service infrastructure with an excessive volume of packets [5]. Apart from the quantity of packets involved, these incidents can also arise from software or hardware failures. Insufficient resources, environmental conditions, or a combination of these factors can also contribute to such incidents. Initially, in the early stages of DoS attacks, it was common for attackers to initiate attacks from a single source. This made it relatively straightforward to mitigate the attack by identifying and blocking the source of the attack. However, attackers have adapted their strategies for greater effectiveness, leading to the emergence of variations such as Distributed Denial of Service (DDoS) attacks. The first tool for conducting DDoS attacks was introduced in 1998. Since then, DDoS attacks have evolved in terms of frequency, volume, and complexity. The objective of a DDoS attack is to impede or disrupt legitimate access to services. Attackers achieve this by exploiting vulnerabilities in the infrastructure or overwhelming resources using multiple compromised agents. This section provides an overview of the operational mode of DDoS attacks, different attack types, and the protective measures employed to mitigate DDoS attacks [6].

2. Advancement in DDoS Attacks:

The proliferation of high-capacity internet connectivity and the increasing number of devices connected to the global web has led to the evolution of DoS attacks. Attackers have moved away from relying on a single launch point and instead began hijacking internet-connected machines, allowing them to overcome the limitations of a single source. Consequently, DDoS attacks emerged, which involve coordinating multiple connected devices to target a victim [3][5].

In the traditional model, DDoS attacks exploit the victim's infrastructure by overwhelming its computing resources through the creation of numerous connections from various sources. Another approach is the Link Flooding Attack (LFA), which aims to degrade or disrupt the victim's service by congesting critical links and isolating the victim's network from the internet [7].

The Economic Denial of Sustainability (EDoS) is another method used to inflict harm on victims. In EDoS attacks, the goal is to exhaust the victim's resources, compelling them to allocate additional computing resources. This, in turn, escalates the costs required to maintain the service. Cloud-internal Denial of Service (CIDoS) involves consuming server resources by utilizing multiple virtual machines hosted on the victim's physical host. Attackers increase the workload on these virtual machines to deplete the host's resources, thereby disrupting the service. Ransom DDoS (RDDoS) is yet another variant where attackers demand payment of ransoms to suspend or refrain from launching DDoS attacks against the victim [8].

The key players involved in a DDoS attack are the attackers, infected devices (often referred to as zombies, web robots, or bots), and the victim. Bots are malware-infected devices connected to the internet that carry out pre-programmed tasks. These tasks include activities like sending spam emails, traffic sniffing, capturing sensitive information, phishing, click fraud, keylogging, distributing software for cryptocurrency mining, and launching DDoS attacks. A "botnet" refers to a network of bots controlled remotely by attackers or botmasters. The victim refers to the server or computer network that possesses resources necessary for the proper operation of a service. The botmaster sends commands to the botnet, initiating connections to the victim in order to execute the DDoS attack. The duration of DDoS attacks can vary from minutes to days, reaching terabits per second or millions of requests per second [8][9].

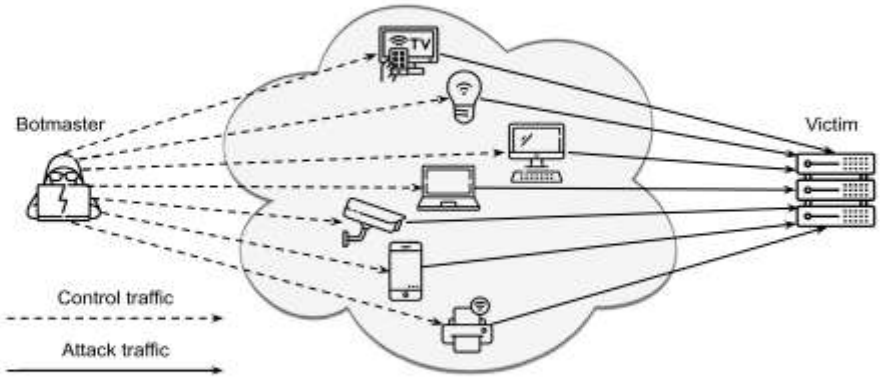

Fig.2 Basic Assembly of DDoS Attack [1]

Fig. 2 This scenario depicts the functioning of a DDoS attack, wherein a botmaster oversees a network of bots through control traffic, prompting the bots to unleash attack traffic on the victim's infrastructure. The botmaster exploits various vulnerabilities in different internet-connected devices to propagate malicious code. The targets of these attacks can include devices with ample resources, such as desktop computers, laptops, servers, tablets, and smartphones, as well as devices with limited resources, like those found in the Internet of Things (IoT) ecosystem, such as security cameras or smart TVs. Once infected, the attackers issue commands for their bots to execute. A DDoS attack takes place when the attacker directs the bots to establish connections with the victim's infrastructure, resulting in the consumption of all available resources [1][8].

The existing literature describes three botnet architectures that attackers employ for control: centralized, peer-to-peer (P2P), and hybrid architecture. In a centralized architecture, attackers communicate with the entire botnet through a central point known as a command and control (C&C) server. While this architecture facilitates botnet management, it suffers from a limitation of requiring high bandwidth to handle communication with the botnet. Additionally, the central communication point becomes a single point of failure, as losing access to the C&C server results in losing control over the botnet [9].

To avoid such a single point of failure, attackers may opt for a P2P architecture for C&C operations. In this architecture, bots act as both clients and servers. Each bot receives commands as a client and distributes them to other bots as servers. If a group of bots leaves the botnet, the remaining bots can continue to operate independently. Although this architecture presents a more complex management process for attackers, it provides increased resilience as the botnet can function without relying on a specific subset of bots. However, the dissemination time for commands in the P2P architecture may be longer [10].

Alternatively, the hybrid architecture aims to reduce the management complexity associated with the P2P architecture. In this approach, only one group of bots acts as both server and client. Consequently, all bots within the botnet search for these specific servers to receive updates and instructions, streamlining the management process for the attackers [11].

To implement the different botnet architectures, it is necessary to utilize either an existing protocol developed for another purpose or an exclusive protocol. One of the primary protocols used for botnet communication is the Internet Relay Chat (IRC) protocol. Originally designed for Internet chat systems, attackers have adopted IRC due to its ease of implementation, widespread usage on the Internet, and its capability to facilitate communication among various entities. However, this protocol has drawbacks, such as limited utilization in corporate networks and the ease with which firewalls can block its traffic [12].

Another commonly employed protocol, both on the internet and by attackers, is the Hypertext Transfer Protocol (HTTP). Operating on the client-server model, HTTP involves the client sending a request and the server responding to it. Like IRC, HTTP communications are relatively straightforward to implement, but they have higher latency compared to IRC. Additionally, HTTP does not support communication between groups since the client (i.e., the bot) must initiate the request [13].

Attackers also utilize the Server Message Block (SMB) protocol for communication within botnets. Originally designed for resource sharing, such as printers, files, and serial ports between computers, SMB allows attackers to communicate over local networks. However, this protocol can be easily blocked directly at the internet gateway, limiting its effectiveness beyond local networks. Many attackers opt for protocols based on a peer-to-peer (P2P) architecture to establish communication with their bots. P2P communication serves as an alternative to relying on centralized servers, resulting in reduced download times. In a P2P network, devices actively participate in sharing various types of files, including text, audio, or video files. As a result, these files become available on multiple servers within the network [14].
When a client needs to download a file in a P2P architecture, it retrieves pieces of the file from nearby servers rather than obtaining the entire file from a single source. As the client receives different parts of the file, it also becomes a file server, thereby increasing the number of available servers. Attackers leverage this communication method to distribute their commands among bots. Consequently, bots promptly disseminate the attackers' commands to other bots once they have downloaded them. In some cases, botmasters create custom protocols exclusively for communication with bots, often based on widely used protocols like Transmission Control Protocol (TCP) or User Datagram Protocol (UDP). However, there are instances where attackers utilize the Internet Control Message Protocol (ICMP) for command and control (C&C) communication. The purpose of developing exclusive protocols for botnets is to make detection more challenging. However, if these protocols exhibit uncommon patterns on the network, security systems can easily identify the communication [13][14].

DDoS attacks typically involve four main steps: reconnaissance, recruitment, command and control, and launching the attack. During reconnaissance, attackers gather information about potential victims and identify suitable devices to form the botnet. This information helps attackers determine which devices can function as C&C servers or clients for carrying out the attack. It also assists them in devising strategies for recruiting these devices. To recruit bots, attackers exploit vulnerabilities to infect devices or conduct campaigns to spread malicious software through email attachments or web downloads. Once the botnet comprises thousands of controlled devices, the attack progresses to the command-and-control phase. At this stage, attackers can conduct timely maintenance on the botnet's code to update the bots' programming and synchronize their actions according to the attackers' intentions. In the command-and-control phase, attackers also conduct brief tests lasting seconds or minutes to evaluate the effectiveness of the attack and make any necessary adjustments. The final step is launching the attack, wherein all active bots in the botnet begin sending malicious traffic to the victim. Attackers can bypass the preparation process of a DDoS attack by utilizing online DDoS attack services, where other attackers have already completed the initial steps of the attack and have botnets ready to launch DDoS attacks in exchange for a ransom [15].

3. Types of DDoS attacks:

The literature describes three categories of DDoS attacks: volume-based attacks, protocol-based attacks, and application layer-based attacks. Volume-based attacks aim to overwhelm the victim's network bandwidth by inundating it with a substantial amount of malicious data. Protocol-based attacks exploit vulnerabilities in the network and transport layers of the TCP/IP model, with the intention of overloading the victim's computing resources. On the other hand, application layer-based attacks target web application services, specifically focusing on denying access and disrupting the functionality of applications, which occurs at the fifth layer of the TCP/IP protocol stack [16].

The literature offers various classifications of DDoS attacks based on different perspectives, considering factors such as the level of attack automation, the vulnerability exploited, and the attack rate. In this particular survey, the classification of DDoS attacks is based on their impacts, following a similar approach to previous studies. The authors of this survey review DDoS attacks across four categories: bandwidth depletion, resource depletion, infrastructure attack, and zero-day attack. This classification was selected to ensure comprehensive coverage of the state-of-the-art in DDoS attacks. Moreover, presenting the classification based on the impact of the attacks aims to facilitate understanding for specialists in various fields, extending beyond DDoS attack specialists alone. It should be noted that the literature has the potential to tailor this classification for specific networks, such as the Internet of Things (IoT) [17].

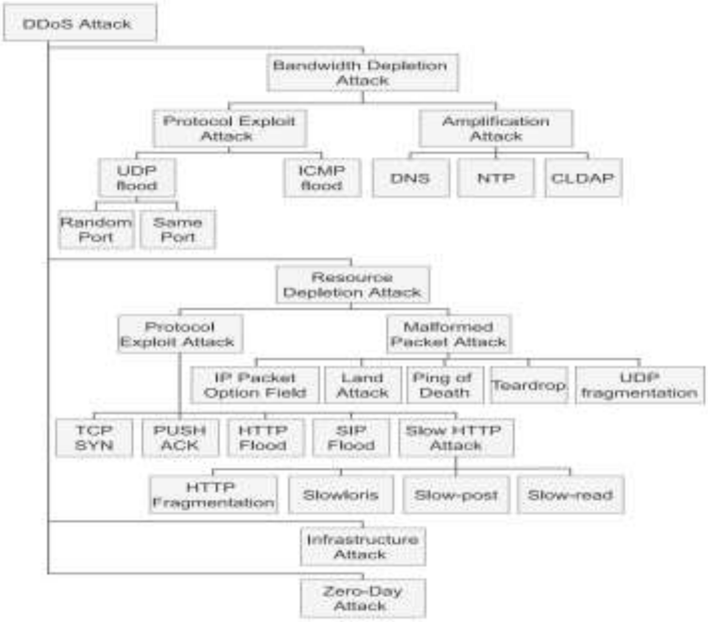

Fig. 3. Classification of Different Mechanisms of DDoS Attack [17]

There are different mechanisms of DDoS attacks are available as shown in Fig. 3, Some of the major attacks are also described and explained with their attack types and vectors below:

3.1 Bandwidth depletion attack:

A bandwidth depletion attack aims to exhaust the victim's network bandwidth by utilizing compromised bots to hinder access to services. This type of attack can persist for extended periods before effective mitigation is achieved. The literature classifies bandwidth depletion attacks into two main types: protocol exploit attacks and amplification attacks. Protocol exploit attacks are characterized by the exploitation of various protocols across different network layers to drive the victim into bandwidth starvation. Within this category, two sub-types of attacks exist UDP flood and ICMP flood [18].

3.2 UDP flood-based attacks:

These attacks are widely encountered and accounted for approximately 43% of attacks in the first quarter of 2023. In this type of attack, the attacker directly or indirectly instructs the botnet to inundate the victim's network bandwidth with a large volume of UDP packets, often with falsified source addresses. This attack exhibits two variations, with the attacker deciding whether the UDP packet destination port remains the same for the entire botnet or is randomized for each packet [19]. The attack operates by overwhelming the server, which receives a continuous stream of UDP packets, and checks if there is any program on the specified port capable of responding to the request. If there is no such program, the server still responds to the request, indicating that the destination address is unreachable. As the server continues to receive, process, and respond to an overwhelming number of UDP packets, it eventually becomes overwhelmed and experiences crashes [20][21].

3.3 ICMP flood attacks:

These attacks deplete network resources by overwhelming the network with a high volume of ICMP requests. ICMP, which is a protocol used for error reporting to clients and is commonly employed by tools like ping and traceroute, becomes a means for attackers to exploit the victim's network bandwidth. By sending an excessive number of ICMP requests, the attacker consumes the server's processing capacity as it responds to these requests and sends ICMP responses back to the source, thus occupying a significant portion of the available bandwidth. To mitigate ICMP flood attacks, a straightforward approach is to disable or restrict the reception of ICMP packets originating from external networks. However, this measure has the consequence of preventing the server from responding to ping or traceroute requests [20]. Amplification attacks, on the other hand, seek to disrupt services by directing substantial responses to the victim from relatively small initial requests. There are three primary types of amplification attacks: DNS amplification, NTP amplification, and CLDAP amplification [22].

3.4 Resource depletion attack:

Another method used to disrupt service for legitimate users is by depleting resources other than network bandwidth. Common examples of resources targeted by attackers include the Central Processing Unit (CPU), memory, and sockets. Resource depletion attacks can be categorized as Protocol Exploit Attacks and Malformed Packet Attacks. Protocol Exploit Attacks leverage various protocols to exhaust critical resources required for service delivery. There are five types of protocol exploit attacks: TCP SYN attack, TCP

PUSH+ACK attack, HTTP flood attack, SIP Flood attack, and Slow HTTP attack [22][23].

3.5 TCP Syn-Attack:

The TCP SYN attack exploits a vulnerability in the TCP protocol to consume memory resources, resulting in a denial of service. To initiate a TCP connection, the handshake process must be completed. The client, intending to establish the connection, sends an SYN-type packet to the server. In response, the server acknowledges the communication by sending an SYN/ACK packet [24].

Script of Syn-Attack to analyze the flags for mitigation:

```
def sourceIPgen():
   not_valid = [10,127,254,255,1,2,169,172,192]
   first = randrange(1, 256)

   while first in not_valid:
      first = randrange(1, 256)
   print(first)
   ip = ".".join([str(first), str(randrange(1, 256)), str(randrange(1, 256)), str(randrange(1, 256))])
   print(ip)
   return ip

def main():
   dstIPs = sys.argv[1:]
   print(dstIPs)
   interface = popen('ifconfig | awk \'/eth0/ {print $1}\'').read()
   print(repr(interface))
   for i in range(10000):
      packets = Ether()/IP(dst=dstIPs, src=sourceIPgen())/TCP(dport=int(RandShort()), sport=int(RandShort()), flags="S")
      print(repr(packets))
      sendp(packets, iface=interface.rstrip(), inter=0.05)

if __name__ == "__main__":
   main()
```

3.6 Malformed packet attacks:

These attacks are designed to disrupt the operation of the victim by utilizing packets that are intentionally crafted in a malformed manner. These attacks can manifest in five different types: IP Packet option field attack, Land attack, Ping of Death, Teardrop attack, and UDP fragmentation attack [25].

Script of Land Attack to analyze the flags for mitigation:

```
def main():
   dstIP = sys.argv[1]
   dst_port = sys.argv[2]
   print(dstIP, dst_port)
   interface = popen('ifconfig | awk \'/eth0/ {print $1}\'').read()
```

```
  print(repr(interface))
  for i in range(0, 1000):
    payload = "LAND packet"
    packets = Ether() / IP(dst=dstIP, src=dstIP) / TCP(dport=int(dst_port),
sport=int(dst_port),
                                       flags="S") / payload

    print(repr(packets))
    sendp(packets, iface=interface.rstrip(), inter=0.05)


# main
if __name__ == "__main__":
  main()
```

3.7 Infrastructure attacks:

These attacks are aimed at disrupting access to services by overwhelming the bandwidth and computing resources of critical infrastructure that supports the functioning of the Internet. A notable example of such an attack is the targeting of DNS servers. When DNS fails to resolve a request, users may be unable to access the desired service [25][26]. In October 2016, DynDNS, a company providing DNS services, became the target of one of the largest infrastructure attacks. In this instance, thousands of IoT devices were employed to flood the company's servers, resulting in the denial of access to essential services like GitHub, Twitter, and Netflix [26].

3.8 Zero-day attacks

These attacks in the context of DDoS refer to attacks that exploit previously unknown vulnerabilities or security breaches, resulting in unprecedented attacks. These attacks utilize vectors that have not yet been documented or catalogued, making it difficult to predict their impact. The term "zero-day" signifies that the attack vector and appropriate defence or response measures can only be recognized after the attack has occurred. To mitigate the risk of zero-day attacks, it is crucial to keep systems updated and properly configured, as this reduces the likelihood of unknown vulnerabilities [27]. Additionally, companies often offer rewards to incentivize researchers to report any vulnerabilities or security breaches they discover. In one instance, experts identified a zero-day vulnerability in the phone system sold by Mitel MiCollab, which allowed attackers to amplify DDoS attacks by an astonishing 300 billion per cent [28].

4. DDoS Defense mechanisms:

Defense mechanisms against DDoS attacks encompass various strategies aimed at preventing or mitigating the damage caused by such attacks. The literature provides different approaches for classifying these defense mechanisms based on criteria related to their application methods, deployment locations, cooperation levels, response strategies, and activity levels. In this survey, the classification of DDoS defense mechanisms primarily focuses on when these mechanisms are implemented, distinguishing between those deployed before the attack and those implemented during the attack [29].

The primary objective in combating a DDoS attack is to prevent its occurrence. Dealing with a DDoS attack after it has started can be challenging. Various preventive measures can be taken, including disabling unnecessary services, maintaining up-to-date and properly configured software protocols and firewalls, addressing software bugs, and implementing service replication in multiple locations. These preventive mechanisms should be implemented before the attack is initiated to minimize or mitigate the adverse impact of the attack [30].

5. Advanced DDoS attack detection techniques:

In this study, we will be discussing the latest attack detection and mitigation techniques which can countermeasure the DDoS attack. We will be discussing Anomaly detection and Traffic Analysis, Traffic Scrubbing, Content delivery network (CDN) and advanced practices using ensemble learning [31].

5.1 Anomaly Detection and Traffic Analysis:

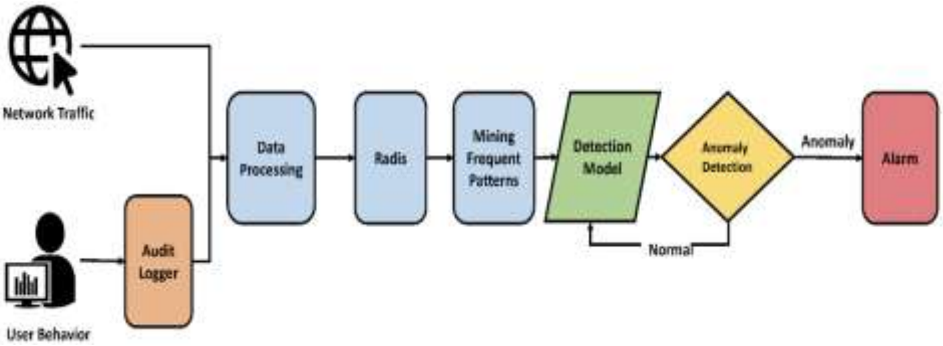

Fig. 4. Anomaly and Traffic Analysis Model [3131]

The system described in this study highlights the importance of deploying an effective intrusion detection system to enhance security and detect denial of service (DoS) attacks before they impact the victim. However, traditional intrusion detection systems may not function optimally in the unique environment of interest. The focus of this study is to explore an anomaly detection system specifically designed to detect distributed denial of service (DDoS) attacks. The goal is to achieve high attack detection rates while minimizing false alarms, thereby achieving optimal performance as shown in Fig. 4 [31][32].

The security management of large high-speed networks, such as IoT devices, poses significant challenges, particularly in detecting suspicious anomalies in network traffic patterns caused by distributed denial of service (DDoS) attacks. To protect IoT devices from denial of service (DoS) attacks, it is crucial to detect them before they affect the user. This requires a high detection rate and a low false alarm rate, ensuring that attack traffic is discarded while legitimate traffic remains unaffected. Traditional DoS defense systems that monitor the volume of packets from a single address or network fail to address DDoS attacks originating from multiple sources.

Intrusion detection systems (IDS) are commonly used to detect DDoS attacks. IDSs inspect network and system activity to identify potential security threats. They can be classified as either misuse detectors or anomaly detectors. Anomaly detection has an advantage over signature-based detection as it can detect new attacks that deviate from normal traffic patterns. However, computational IoT devices present unique challenges for detecting such differences in traffic patterns that do not exist in traditional IDSs [33].

Anomaly-based detection is valuable because it can identify various types of new or unusual traffic behavior, providing early warnings for potential intrusions. This approach covers silent attempts, backdoor activities, and certain network failures. Several techniques and challenges are involved in developing an anomaly detection system. Many articles suggest using traffic volume (flow, packet, byte count) as a metric for anomaly detection [34]. This system implements an IDS that combines both anomaly and entropy-based intrusion detection systems. Entropy is utilized to analyze changes in traffic distribution, offering two advantages: increased detection capability compared to volume-based methods and additional information for categorizing different types of anomalies (e.g., worms, DDoS attack scanning) [35].

5.2 Traffic Scrubbing:

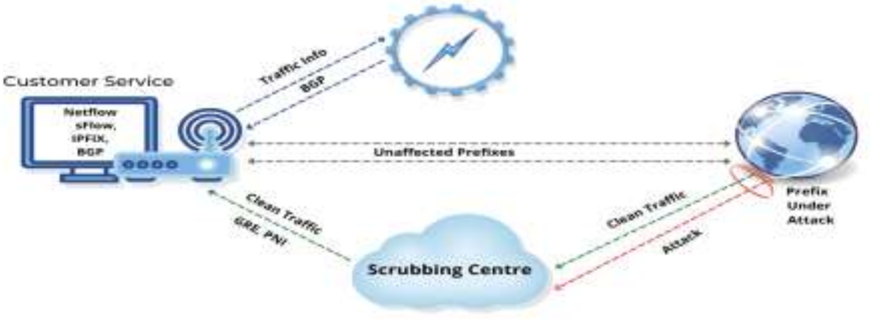

Fig. 5. Traffic Scrubbing for DDoS attack [36]

Traffic Scrubbing is a typical DDoS mitigation method. The traffic intended for a specific IP address range is diverted to datacenters using this technique as shown in Fig. 5. [36], where the attack traffic is "scrubbed" or cleaned. Then, only clean traffic is forwarded to the desired location. The majority of DDoS scrubbing companies have three to seven scrubbing centres, often dispersed throughout the world. Each DDoS mitigation centre is equipped with massive quantities of bandwidth—up to 350Gbps—that feeds traffic to it. When customers are being attacked, they "push the button" to divert all traffic to the nearest cleaning facility. Scrubbing centres are used by enterprise customers in two ways: some use them to route traffic continuously, while others want to route it only when it is needed [37]. Organizations are increasingly using hybrid models of protection due to the complexity of security assaults and IT infrastructures, in order to guard against the most possible attack vectors. As a first line of defence, they frequently use an on-premise solution, with the scrubbing centre stepping in when the on-premise technology is overloaded. Organizations must have seamless integration between cloud and on-premise solutions implemented in front of an infrastructure's network to help mitigate an attack before it reaches the core network assets and data if bad traffic is to be seamlessly diverted to a scrubbing centre to minimise any downtime [39].

However, organisations are also turning to content distribution network (CDN)-based DDoS mitigation services to protect web and mobile applications, as well as application programming interface (API) traffic of many internet of things (IoT) applications. Scrubbing centres are primarily used to protect infrastructure sitting in the customer's environment, such as DNS servers, mail relays, and other IP-based applications [40].

5.3 Content Delivery Network (CDN)

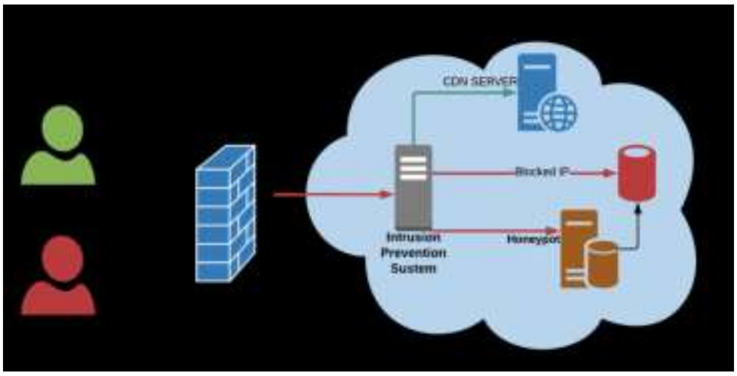

Fig. 6. Basic Content Delivery network architecture for DDoS attacks [41]

However, organisations are also turning to content delivery network (CDN)-based DDoS mitigation services to protect web and mobile applications, as well as application programming interface (API) traffic of many internet of things (IoT) applications. Scrubbing centres are primarily used to protect infrastructure sitting in the customer's environment, such as DNS servers, mail relays, and other IP-based applications [41][47][48].

As a result, you have numerous versions of your website available in various locations rather than having your primary web server serve users from all over the world from a single, centralised location as shown in Fig. 6.

Along with speeding up your site's load times, which is many services' main purpose, CDN services also assist you in relieving the strain that heavy traffic volumes place on your network should you come under attack [42]. The stability of CDNs is increased, and clients from various geographical areas can benefit from a quick, seamless experience. But they also offer mitigation against DDoS attacks with the security measures they employ.

Because CDNs are built expressly to manage high volumes of traffic, if a business encounters the kind of sudden spike in demand that characterises a DDoS assault, it can respond by redistributing this traffic, preventing it from reaching your origin servers and taking down your website [43].

Customers will therefore be able to access your website as usual and won't even be aware that you are being attacked [43].

A CDN offers the advantage of distributing a load of incoming traffic across multiple servers located in different parts of the world. By storing copies of your website's content closer to the end users, the CDN functions by caching data from your web server. As a result, instead of relying on a single, centralized location to serve visitors worldwide, your site becomes available through numerous replicas dispersed across various locations.

In addition to enhancing the speed at which your site loads, which is the primary objective of most CDN services, these services also assist in alleviating the strain placed on your network by large volumes of traffic, especially during an attack [44].

These tools are there to handle the DDoS attack and to mitigate the attack with intelligent practice we need the machine learning for efficient detection and mitigation on the basis of pattern analysis. These tools can be implemented with random forest classifiers and extra tree classifiers for detecting the strong attack vectors [45].

```
“>>>from sklearn.ensemble import RandomForestClassifier”
-Forests of randomized trees
      >>>self.clf = RandomForestClassifier(n_estimators=10);

“>>>from sklearn.ensemble import ExtraTreesClassifier”
-Extra Tree Classifier

>>>self.clf=ExtraTreesClassifier(n_estimators=10,max_depth=None,min_samples_split=2,
        random_state=0);
```

Above python codes can be used for the SDN models to handle the DDoS attack vectors. With the inclusion of these machine learning algorithms the models can be retrained and can be efficiently used for handling the DDoS attack over server or any IoT device [46][49].

6. Conclusion and Future Directions:

This study provides a holistic review of the different kinds of DDoS attacks and its advanced mitigation practices. A proactive defense method getting attention in the literature is attack prediction, detection and mitigation. Techniques for DDoS prediction highlight indicators of an impending attack. It seeks to provide administrators enough time to stop the attack. The most recent research on foreseeing attacks with DDoS was given in this survey. It narrowed down the recognised studies that make suggestions for predicting DDoS attacks. This study is based on the theory that prediction can detect DDoS using ensemble learning attack signs before the attacker launches one. Although there are numerous research for DDoS attack detection in the literature, there are none for DDoS attack prediction. In the upcoming years, DDoS attack prediction will become more important and useful, which will advance the field of cybersecurity. The first factor contributing to this potential rise is network managers' advantage against attackers due to attack prediction. One of the only methods to prevent network managers from being caught off guard by attackers is to predict attacks. In order to lessen the harm inflicted by the attackers, network managers will have more time to deal with the attack. Although forecasting DDoS attacks is not simple, the research shows that it is possible to do so. Despite the challenges in creating methods to anticipate DDoS attacks, research in this field must be prioritised.

Numerous research opportunities are the second factor driving the evolution of DDoS attack prediction.

Despite the fact that the literature has already examined the traditional approaches to attack prediction, there are still plenty of chances for fresh ideas. DDoS attacks are continually changing, necessitating the collaboration of current research with existing literature to propose new and improved methods of attack prediction. Future research would adhere to this idea, investigating unresolved problems like the adoption of distributed systems and the use of deep learning to evaluate a solution that gives network administrators more time to cope with DDoS attacks.

7. Declaration of competing interest:

The authors affirm that they have no known financial or interpersonal conflicts that would have seemed to have an impact on the research presented in this study.

8. Data availability:

No data was used for the research described in the article.

9. Refrences:

1. de Neira, A. B., Kantarci, B., & Nogueira, M. (2023). Distributed denial of service attack prediction: Challenges, open issues and opportunities. *Computer Networks*, *222*, 109553.
2. Gutnikov, A., Kupreev, O., & Badovskaya, E. (2021). DDoS attacks in Q1 2021.
3. Santanna, J. J. (2013). DDoS as a Service. In *nMRG workshop is co-located with Conference on Network and Service Management*.
4. Wilson, C. (2008). *Botnets, cybercrime, and cyberterrorism: Vulnerabilities and policy issues for congress* (Vol. 29). Washington, DC: Congressional Research Service.
5. Mahjabin, T., Xiao, Y., Sun, G., & Jiang, W. (2017). A survey of distributed denial-of-service attack, prevention, and mitigation techniques. *International Journal of Distributed Sensor Networks*, *13*(12), 1550147717741463.
6. Fischer, B., Meissner, D., Nyuur, R., & Sarpong, D. (2022). Guest Editorial: Cyber-Attacks, Strategic Cyber-Foresight, and Security. *IEEE Transactions on Engineering Management*, *69*(6), 3660-3663.
7. Gupta, B. B., Joshi, R. C., & Misra, M. (2012). Distributed denial of service prevention techniques. *arXiv preprint arXiv:1208.3557*.
8. Zargar, S. T., Joshi, J., & Tipper, D. (2013). A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks. *IEEE communications surveys & tutorials*, *15*(4), 2046-2069.
9. Menscher, D. (2021). Exponential growth in DDoS attack volumes (2020).
10. Keshariya, A., & Foukia, N. (2009, September). DDoS defense mechanisms: A new taxonomy. In *International Workshop on Data Privacy Management* (pp. 222-236). Berlin, Heidelberg: Springer Berlin Heidelberg.
11. Xing, J., Wu, W., & Chen, A. (2021). Ripple: A Programmable, Decentralized {Link-Flooding} Defense Against Adaptive Adversaries. In *30th USENIX Security Symposium (USENIX Security 21)* (pp. 3865-3881).
12. Sotelo Monge, M. A., Maestre Vidal, J., & García Villalba, L. J. (2017). Entropy-based economic denial of sustainability detection. *Entropy*, *19*(12), 649.
13. Alarifi, S., & Wolthusen, S. D. (2013, September). Robust coordination of cloud-internal denial of service attacks. In *2013 International Conference on Cloud and Green Computing* (pp. 135-142). IEEE.
14. Sasaki, T., Gañán, C. H., Yoshioka, K., Van Eeten, M., & Matsumoto, T. (2020). Pay the piper: DDoS mitigation technique to deter financially-motivated attackers. *IEICE Transactions on Communications*, *103*(4), 389-404.
15. Ngo, F. T., Agarwal, A., Govindu, R., & MacDonald, C. (2020). Malicious software threats. *The Palgrave Handbook of International Cybercrime and Cyberdeviance*, 793-813.
16. Marzano, A., Alexander, D., Fonseca, O., Fazzion, E., Hoepers, C., Steding-

Jessen, K., ... & Meira, W. (2018, June). The evolution of bashlite and mirai iot botnets. In *2018 IEEE Symposium on Computers and Communications (ISCC)* (pp. 00813-00818). IEEE.

17. Zeidanloo, H. R., & Manaf, A. B. A. (2010). Botnet detection by monitoring similar communication patterns. *arXiv preprint arXiv:1004.1232*.
18. Zeidanloo, H. R., & Manaf, A. B. A. (2010). Botnet detection by monitoring similar communication patterns. *arXiv preprint arXiv:1004.1232*.
19. Musik, P., & Jaroensutasinee, K. (2007). Large-scale simulation using parallel computing toolkit and server message block. *WSEAS TRANSACTIONS ON MATHEMATICS*, *6*(2), 369.
20. Sahu, S. K., & Khare, R. K. (2020). DDoS attacks & mitigation techniques in cloud computing environments. *Gedrag Organ. Rev*, *33*(2), 2426-2435.
21. Santanna, J. J. (2013). DDoS as a Service. In *nMRG workshop is co-located with Conference on Network and Service Management*.
22. El-Sofany, H. F. (2020). A New Cybersecurity Approach for Protecting Cloud Services against DDoS Attacks. *International Journal of Intelligent Engineering & Systems*, *13*(2).
23. Alam, M. M., Arafat, M. Y., & Ahmed, F. (2015). Study on Auto Detecting Defence Mechanisms against Application Layer Ddos Attacks in SIP Server. *J. Networks*, *10*(6), 344-352.
24. Mirkovic, J., & Reiher, P. (2004). A taxonomy of DDoS attack and DDoS defense mechanisms. *ACM SIGCOMM Computer Communication Review*, *34*(2), 39-53.
25. Yusof, A. R. A., Udzir, N. I., & Selamat, A. (2019). Systematic literature review and taxonomy for DDoS attack detection and prediction. *International Journal of Digital Enterprise Technology*, *1*(3), 292-315.
26. Yadav, V. K., Trivedi, M. C., & Mehtre, B. M. (2016). DDA: an approach to handle DDoS (Ping Flood) attack. In *Proceedings of International Conference on ICT for Sustainable Development: ICT4SD 2015 Volume 1* (pp. 11-23). Springer Singapore.
27. Haris, S. H. C., Ahmad, R. B., & Ghani, M. A. H. A. (2010, September). Detecting TCP SYN flood attack based on anomaly detection. In *2010 Second international conference on network applications, protocols and services* (pp. 240-244). IEEE.
28. Bjørk, M., Stovner, L. J., Hagen, K., & Sand, T. (2011). What initiates a migraine attack? Conclusions from four longitudinal studies of quantitative EEG and steady-state visual-evoked potentials in migraineurs. *Acta neurologica scandinavica*, *124*, 56-63.
29. Samta, R., & Sood, M. (2020). Analysis and mitigation of DDoS flooding attacks in software defined networks. In *International Conference on Innovative Computing and Communications: Proceedings of ICICC 2019, Volume 2* (pp. 337-355). Springer Singapore.
30. Hoggan, D. (1994). Teardrop attack. *The Internet Book: Introduction and Reference*.
31. Sonar, K., & Upadhyay, H. (2014). A survey: DDOS attack on Internet of Things. *International Journal of*

Engineering Research and Development, *10*(11), 58-63.

32. Kaur Chahal, J., Bhandari, A., & Behal, S. (2019). Distributed denial of service attacks: A threat or challenge. *New Review of Information Networking*, *24*(1), 31-103.
33. Dalmazo, B. L., Marques, J. A., Costa, L. R., Bonfim, M. S., Carvalho, R. N., da Silva, A. S., ... & Cordeiro, W. (2021). A systematic review on distributed denial of service attack defense mechanisms in programmable networks. *International Journal of Network Management*, *31*(6), e2163.
34. Belenky, A., & Ansari, N. (2003). On IP traceback. *IEEE Communications magazine*, *41*(7), 142-153.
35. Syaifuddin, S., Azis, M. F., & Sumadi, F. D. S. (2021). Comparison analysis of multipath routing implementation in software defined network. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*.
36. Bhardwaj, A., Mangat, V., Vig, R., Halder, S., & Conti, M. (2021). Distributed denial of service attacks in cloud: State-of-the-art of scientific and commercial solutions. *Computer Science Review*, *39*, 100332.
37. Salam, R. A., Das, J. K., Lassi, Z. S., & Bhutta, Z. A. (2016). Adolescent health and well-being: Background and methodology for review of potential interventions. *Journal of adolescent health*, *59*(4), S4-S10.
38. Rajab, M. A., Zarfoss, J., Monrose, F., & Terzis, A. (2006). A Multifaceted Approach to Understanding the Botnet Phenomenon." proceedings of the 6th ACM SIGCOMM on Internet measurement, Rio de Janeriro.
39. Kavlakoglu, E. AI vs. machine learning vs. deep learning vs. neural networks: what’s the difference?: IBM; 2020.
40. Fadlullah, Z. M., Fouda, M. M., Kato, N., Shen, X., & Nozaki, Y. (2011). An early warning system against malicious activities for smart grid communications. *IEEE Network*, *25*(5), 50-55.
41. Tse, A., & Carley, K. M. (2017). Event-based model simulating the change in DDoS attack trends after P/DIME events. In *Social, Cultural, and Behavioral Modeling: 10th International Conference, SBP-BRiMS 2017, Washington, DC, USA, July 5-8, 2017, Proceedings 10* (pp. 120-126). Springer International Publishing.
42. Mahfouz, A., Abuhussein, A., Venugopal, D., & Shiva, S. (2020). Ensemble classifiers for network intrusion detection using a novel network attack dataset. *Future Internet*, *12*(11), 180.
43. Mahmoud, N., Essam, Y., Elshawi, R., & Sakr, S. (2019, July). DLBench: an experimental evaluation of deep learning frameworks. In *2019 IEEE International Congress on Big Data (BigDataCongress)* (pp. 149-156). IEEE.
44. Došilović, F. K., Brčić, M., & Hlupić, N. (2018, May). Explainable artificial intelligence: A survey. In *2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO)* (pp. 0210-0215). IEEE.
45. Rauf, U. (2018). A taxonomy of bio-inspired cyber security approaches: existing techniques and future directions. *Arabian Journal for Science and Engineering*, *43*(12), 6693-6708.
46. Feng, Y., Akiyama, H., Lu, L., & Sakurai, K. (2018, August). Feature selection for machine learning-based early detection of distributed cyber attacks. In *2018 IEEE 16th Intl Conf*

on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech) (pp. 173-180). IEEE.

47. Yu, X., & Guo, H. (2019, August). A survey on IIoT security. In *2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)* (pp. 1-5). IEEE.
48. Sachdeva, M., Singh, G., Kumar, K., & Singh, K. (2009). A comprehensive survey of distributed defense techniques against DDoS attacks. *International Journal of Computer Science and Network Security*, *9*(12), 7-15.
49. Stallings, W. (2020). Handling of personal information and deidentified, aggregated, and pseudonymized information under the California consumer privacy act. *IEEE Security & Privacy*, *18*(1), 61-64.