

Application-Layer DDoS Attacks and Defences: A Taxonomy, Comparative Evaluation Framework, and Research Directions

Aditya Arsh

CDAC Bangalore, Bangalore, Karnataka 560100, India

E-mail: aditya.arsh4417@gmail.com

ORCID iD: <https://orcid.org/0009-0009-3234-0927>

Priyanka Biswas*

National Institute of Technology Agartala, Agartala, Tripura - 799046, India

E-mail: priyankacse@ieee.org

ORCID iD: <https://orcid.org/0000-0003-0626-3242>

*Corresponding Author

Nirmalya Kar

National Institute of Technology Agartala, Agartala, Tripura - 799046, India

E-mail: nirmalya@ieee.org

ORCID iD: <https://orcid.org/0000-0002-7371-232X>

Received: February 25, 2026; Revised: March 25, 2026; Accepted: July 15, 2026; Published: August 8, 2026

Abstract: Distributed Denial of Service (DDoS) attacks have gained popularity among cybercriminals as a favoured method of disruption. Application layer DDoS attacks are particularly intricate, as they overload web servers with requests, rendering them inaccessible to legitimate users and causing availability issues. These attacks are challenging to detect through network and transport-level security measures, making them even more concerning. This paper explores various categories of DDoS attacks, encompassing volumetric and protocol-focused attacks, with a particular focus on application-layer attacks, classifying them into Protocol-specific attacks and Generic attacks. It also delves into diverse defence strategies tailored to combat related attacks, such as HTTP Flood, DHCP starvation, SlowLoris, and others. Unlike earlier surveys, which centre on vulnerability-oriented taxonomies through 2017–2020, this work introduces an explicit, criterion-based comparative evaluation framework for attacks and defences, and extends the taxonomy with post-2020 developments containerized and cloud-native low-rate attack surfaces, machine learning-driven detection and adversarial evasion, and zero-trust-based mitigation illustrated with the 2023 HTTP/2 ‘Rapid Reset’ incident. Finding that detection-only mechanisms still dominate current defences, the paper identifies recurring bottlenecks and proposes concrete future-research directions, including detection resistant to adversarial machine learning and low-rate attack detection in containerized and serverless environments.

IndexTerms: DoS attack, DDoS attack, Application Layer, DDoS Defending Techniques, Cloud-Native Security, Zero Trust Architecture

1. Introduction

The Internet was created to accommodate the basic urge for data relocation between systems. Earlier, the focus on security concerns was significantly less, making it vulnerable to various attacks which threaten both service providers and users. The Internet has altered the way people communicate and conduct business. Advances in network technology provide invaders and hackers with numerous avenues to compromise systems. The responsibility to secure crucial data, information, and services on the Internet and computer networks is a key area of inquiry today. Many new hazards are emerging, and defences against them are regularly evolving. Malware, password strikes, denial of service (DoS) attacks [37], and reconnaissance attacks are the four major types of computer menace. Among various types of attacks in cyberspace, one of the most renowned cyber security threats to the internet is the DoS (Denial of Service) attack. A DoS attack is so simple to execute that protecting the network from this type of threat becomes critical. DoS attacks are

the most antiquated due to their destructive nature and ability to drain a victim's computer resources and bandwidth in minutes. As implied from the name, a DoS attack is put in motion to make a service inaccessible to real users. A common way to get to DoS is to flood the target server with tons of irrelevant data packages[25]. To strengthen the attack, an enhanced concept, DDoS (Distributed DoS) came into the picture where a vast number of compromised systems (called "Botnets") from different locations are used to yield a boisterous traffic stream. There has been a growing trend in recent years toward more sophisticated DDoS attacks on Layer 7 of the OSI Model [30].

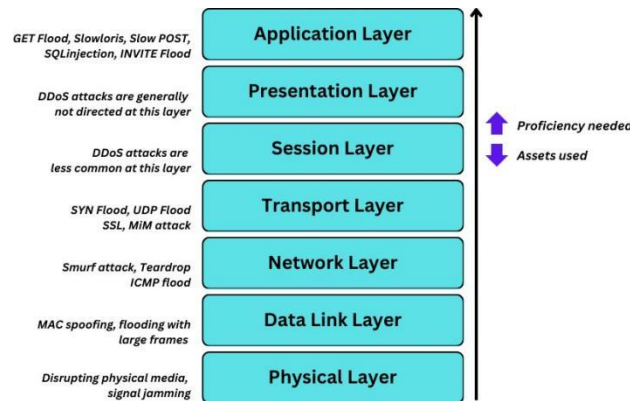


Fig. 1. Different types of DDoS attacks on OSI Layers

The OSI framework, crucial for network system communications, is organized into seven layers, with the Application layer (Layer 7) facilitating user access to network resources. This layer manages data transmission and supports various applications like HTTP, FTP, SMTP, DNS, and Telnet. Handling Layer 7 DDoS attacks is challenging due to their human-like nature and slow rate. Figure 1 illustrates attacks at different OSI layers, emphasizing the increasing sophistication of application-layer DDoS attacks. These attacks focus on specific vulnerable parts rather than directing a big traffic flood at the target. It is obvious that no system in the present day can be regarded as 100 percent secure. Therefore, the possibility of such attacks is rising along with the number of online applications available today. The chart in Figure 2 shows the increasing DDoS events in various sectors. It can be derived that the technological and banking sectors are the highest targets.

The paper provides a comprehensive taxonomy and analysis of application layer DDoS attacks, categorizing them into protocol-focused attacks (e.g. DHCP starvation, Slow HTTP) and generic attacks (e.g. request floods, slow rate attacks). It examines the unique properties of application layer attacks that make them potent, like ambiguity, effectiveness, targeting multiple applications, and limited resources needed. A variety of defense strategies are explored, including cryptographic techniques, traffic profiling, machine learning, traffic inspection, and others. The pros and cons of these techniques are analyzed. Attacks and defenses are compared on dimensions like complexity, stealthiness, ability to spoof, and real-world prevalence, providing a holistic perspective.

1.1. Contribution and Differentiation from Prior Surveys

Tripathi et. al [37] provide a vulnerability-oriented taxonomy current to roughly 2020, while Praseed et. al [26] focus on web-application-specific challenges through 2017. This survey differs in two respects: (i) it introduces an explicit, criterion-based evaluation framework rather than narrative comparison, and (ii) it extends the taxonomy with post-2020 material absent from both prior surveys: containerized/cloud-native low-rate attacks, AI/ML-driven detection and adversarial evasion, and zero-trust mitigation, illustrated with the 2023 HTTP/2 Rapid Reset incident.

Section 2 reviews DoS/DDoS fundamentals and prior surveys. Section 3 characterizes AL-DDoS attacks. Section 4 develops the attack/defence taxonomy, including emerging cloud-native, AI-driven, and zero-trust developments. Section 5 presents the comparative evaluation framework. Section 6 concludes with bottlenecks and future directions.

1.2. Literature Search and Selection Methodology

To ground the taxonomy and comparative framework presented in this survey, literature was identified through IEEE Xplore, the ACM Digital Library, ScienceDirect, SpringerLink, and arXiv, using combinations of the terms "application layer DDoS", "Layer-7 DoS", "slow-rate attack", "HTTP flood", "DHCP starvation", "zero trust DDoS", and "cloud-native DDoS". Foundational attack-taxonomy works from 2011–2018 were retained where they remain the primary source for a named attack technique (e.g., Slowloris, SlowDrop); all retrievable post-2020 work matching the search terms was reviewed for inclusion in Section 4.3. Studies were excluded if they addressed only network or transport-layer (Layer 3/4) DDoS with no application-layer component, or if they were non-peer-reviewed commentary without a verifiable technical claim.

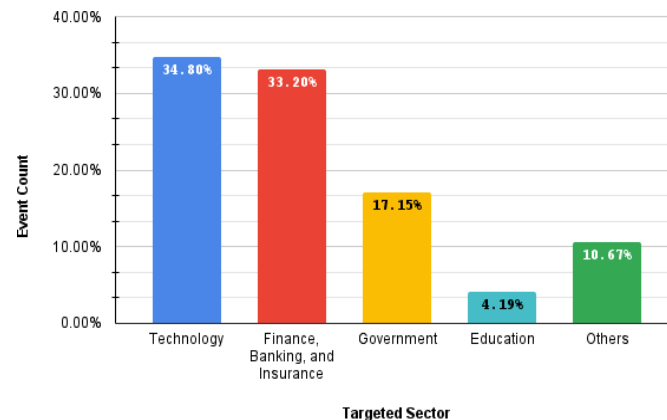


Fig. 2. 2022 DDoS event counts by targeted industry. Technology and banking are the most targeted.

Data Source : <https://www.f5.com/labs/articles/threat-intelligence/2023-ddos-attack-trends>

2. Denial of Service (DoS) Attacks

A purposeful network or computer system overloading with fictitious traffic to stop it from performing its service is known as a denial of service attack. It might stop the target server from responding to legitimate requests, but it could also cause the server to crash. Denial of service attacks often do not compromise systems, cause the loss of information or lead to theft. Because a DoS assault can last anywhere from a few hours to several months, it can cost the targeted service a significant amount of time and money. Distributed denial-of-service (DDoS) attacks, as opposed to denial-of-service attacks, are carried out by multiple computers. A DoS attack's technique is rather simple: it aims to use traffic to overload the target's resources. The targeted system's vulnerability will determine the precise method of carrying out the attack. For instance, sending several requests to a server with fictitious return addresses (i.e. junk) is one method of achieving this. The server is unable to validate their origins as a result. It may cause a server to run out of RAM or CPU power and fail.

DoS assaults are well known; over time, many studies have been conducted to detect and prevent the attacks. However, only a few survey papers categorize different DoS attack types in the literature. Zargar et. al[7] gave a thorough analysis of several protection strategies to stave off assaults, including HTTP flooding attacks as well as Slowloris attacks. Some studies have also concentrated on SIP application layer DoS threats [26]. Additional explanations for DoS attacks on the application layer against HTTP and further web-associated protocols may be found in [25], where the authors also discussed defence strategies to stop specific attacks. Table 1 summarises various works on DoS attacks in brief.

Tripathi et. al. [27] gave an organised and complete evaluation of application layer DDoS threats and defensive strategies in their study. They also addressed the pros and cons of several cutting-edge protection strategies and in-demand DDoS mitigation technologies acknowledged to counter distinct application layer DDoS assaults. Aiming to offer a framework for organising research on application-layer DDoS attacks, a taxonomy of extant AL-Distributed DoS assaults is presented by G. Mantas et al. [14] complemented with typical examples taken from both industry and academic community. Karanpreet Singh et.al. [25] work adds to the disclosure of HTTP-GET flood Distributed DoS assaults in various methods. They created a taxonomy to characterise probable assault techniques by a complete investigation of all exclusive works using a systematic methodology, followed by a thorough assessment of the detection properties. They also classify cutting-edge detection techniques based on methodologies and modelling methods.

A full understanding of current application-layer DDoS assaults backed by ambiguous language is necessary for the advanced design and implementation of protection measures against these attacks. The diversity of research and industry-wide attention to this area quickly revealed gaps in the terminology and understanding of these attacks. Even though most current research in this area attempts to provide some taxonomy, many researchers have defined different types of attacks, used inconsistent terminology, or had vague descriptions of specific attacks, often contradicting each other. Beyond these foundational works, research after 2020 has shifted along three lines this survey synthesizes: a move toward ML/DL-based detection [35], a new low-rate attack surface in containerized/cloud-native deployments [32], and the emergence of zero-trust mitigation [39].

2.1. Distributed Denial of Service (DDoS) Attacks

A DDoS attack originates from multiple distributed sources and typically follows a staged invasion scenario. Multiple bots, known as botnets, are used in this attack to send massive amounts of traffic to the victim's server. DDoS attacks are seen as being more potent than DoS attacks because they increase the denseness of the attack by using multiple computers concurrently. DDoS occurs when many computers overflow a victim's resources, making DoS attacks more effective and difficult to trace.

Table 1. Existing studies of DoS attacks

Ref.	Area of Work	Description
[7]	Distributed DoS Flooding Attacks	Provided study of diverse defence mechanisms to thwart flood attacks against HTTP and SlowLoris attacks
[33]	SIP-oriented VoIP net-works	Focused on denial of service attacks against SIP on Layer7.
[26]	Application Layer DDoS	Application-layer DDoS attacks counter to HTTP and more web-related protocols are explained.
[37]	DoS Attacks on the Appli-cation layer	Gave an organized and complete evaluation of application layer DoS threats and defensive strategies in their study
[14]	Application layer DDoS	Presented a taxonomy of existing application-layer DDoS at-tacks
[25]	DDoS using HTTP-GET Flood	Provides assistance in a number of ways in the identification of HTTP-GET flood DDoS attacks. On the basis of modelling ap-proaches and methodology, they also categorize cutting-edge de-tection techniques.
[31]	DDoS attack mitigation by SDN tech-nologies in IoT	Proposed a taxonomy on DDoS attack mitigation using SDN technologies in an IoT environment.
[29]	DDoS based on DNS in SDN Envi-ronment	This study explores the use of a "Stale Cache" and a slight modi-fication to DNS resolver caching behaviour as a mitigation strategy for DNS Dos attacks.
[34]	DoS and DDoS At-tacks at OSI Layers	In-depth information about DoS and DDoS assaults at various OSI Model layers is provided in this paper, along with a study of such attacks' methodologies, but no countermeasures are of-fered.
[19]	DDoS on Application Layer	This article discusses how DDoS attacks are categorized based on unusual application layer behaviour and offers a summary of the different DDoS tools. Furthermore, it organizes DDoS attack handling methods according to principles for tracking, avoiding, spotting, and neutralizing.

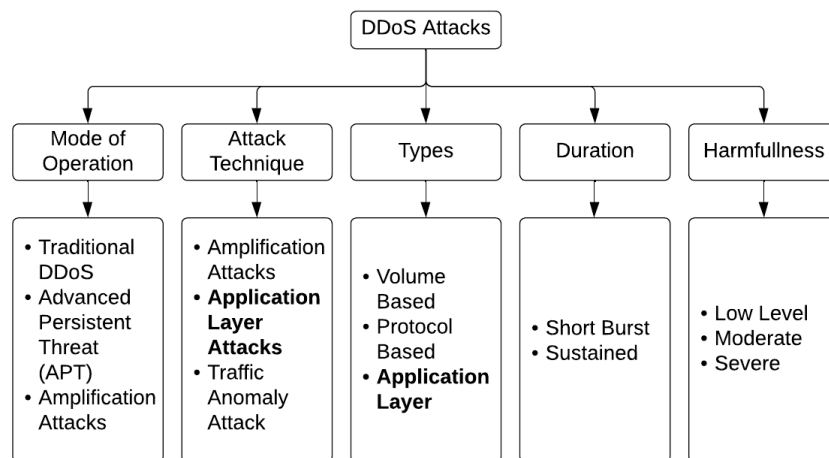


Fig. 3. Classification of DDoS Attacks

A Distributed Denial of Service attack begins with creating a network of botnets, and attackers govern this network. In order to disguise their identity, attackers generally hide behind these multi-staged botnets. The attacker initiates the attack process by broadcasting orders to these malicious, compromised systems. As the botnets acquire orders from the attacker, they launch the attack flow toward the victim, overloading its bandwidth and other resources. Such attacks typically have competitive corporate interests, financial gain, and Hacktivism as motivations.

Volume-based attacks flood the victim's infrastructure bandwidth with an excessive amount of unsolicited data, including methods like ICMP floods and UDP floods. Protocol attacks exploit vulnerabilities at the network layer, with examples such as Smurf DDoS and SYN flooding attacks. Application layer attacks capitalize on the characteristics of application layer protocols and services, including HTTP-GET/POST floods and SlowLoris.

Regarding attack duration, sustained attacks endure for extended periods, often days or weeks, necessitating long-term defense strategies. Conversely, short-burst attacks are brief, usually lasting minutes to hours, intending to cause momentary disruptions. Amplification attacks exploit reflection and amplification vulnerabilities, utilizing third-party servers to magnify attack traffic. In contrast, application layer attacks target specific applications and services, exploiting application vulnerabilities like SQL injection. Traffic anomaly attacks simulate legitimate traffic but exhibit unusual patterns, making them challenging to detect due to their similarity to normal traffic.

Concerning the mode of operation, traditional DDoS attacks stem from botnets, typically using brute force to overwhelm target resources. Advanced Persistent Threat (APT) attacks are more sophisticated, often employing multiple attack vectors and evasion techniques, with a specific target in mind. Amplification attacks rely on reflection and amplification through vulnerable servers, involving unintentional assistance from third-party servers.

In terms of harmfulness, low-level disruption attacks constitute basic volumetric attacks that might briefly slow down services. Moderate disruption attacks are more sophisticated, impacting service availability and potentially

requiring mitigation. Severe disruption attacks are advanced, causing significant downtime and possibly leading to financial losses and reputation damage.

3. Application Layer DDoS Attacks

As discussed in Section 2, network- and transport-layer DDoS techniques have lost much of their effectiveness as infrastructure has scaled. Modern DDoS assaults seldom require surpassing the network's throughput capacity since the target frequently fails before reaching capacity. Even then, flooding continues to be a practical form of disruption. This encouraged the growth of layer-7 (Application layer) DDoS assaults, now one of the most rapidly expanding categories and accounting for more than 20% of all attacks [40]. The user apps affected by this new class of assaults, which target the application layer explicitly, are unaffected by network resources.

In Figure 5, we categorise DDoS attacks perpetrated at the application layer into a taxonomy as indicated and explore several attack categories based on attack-generating approaches, network schema, victim kinds, attack rates, and attack impact. Application layer strikes might be further divided into a number of child classes based on their level of automation, exploited vulnerabilities, varieties of attack networks employed, frequency of attacks produced, victims' profiles, and the effects of the attacks.

Application Layer Distributed DoS attacks are often more complicated since they interfere with essential network-wide functionality or operations. User protocols and Support protocols are the two categories of application layer protocols. Users are directly served through user protocols like HTTP, SMTP/POP, FTP, and others. Support protocols are designed to offer universal system features- TLS/SSL, DNS, NTP, DHCP, and so on. A DDoS attack can be launched by misusing any of these protocols [34]. A DDoS assault targeting application layer protocol focuses on their particular weaknesses. The attackers use various methods to exploit the vulnerabilities present at the application layer. Application Layer Distributed DoS attacks are thought to be sneaky, smart, and undetectable at the network layer. These attacks can do the same level of damage as classical flooding DDoS attacks at a significantly cheaper cost since they concentrate on particular traits and weaknesses of application layer protocols. Table 3 examines the factors that make DDoS assaults at the application layer highly disruptive compared to attacks mounted at other levels [11, 34].

Table 2. Few DDoS attacks in recent past

Year	Target	Scale	Attack type
2017	Swedish Transportation service	Not unveiled	HTTP Flood
2017	Bitcoin Gold website	Not unveiled	HTTP Flood
2018	GitHub	126.9 million per second	Memcached DDoS
2019	An Imperva client	2,92,000 Requests per second	Not unveiled
2020	Amazon Web Services	2.3 terabits per second	CLDAP Reflection/Amplification DDoS
2020	A state voter registration site	2lakh DNS requests	DNS Flood
2021	Colonial Pipeline in US	Not unveiled	Ransomware DDoS
2022	Cloudflare	26M request per second	HTTP Flood
2022	Ukraine and Russia war	Not unveiled	HTTP Flood

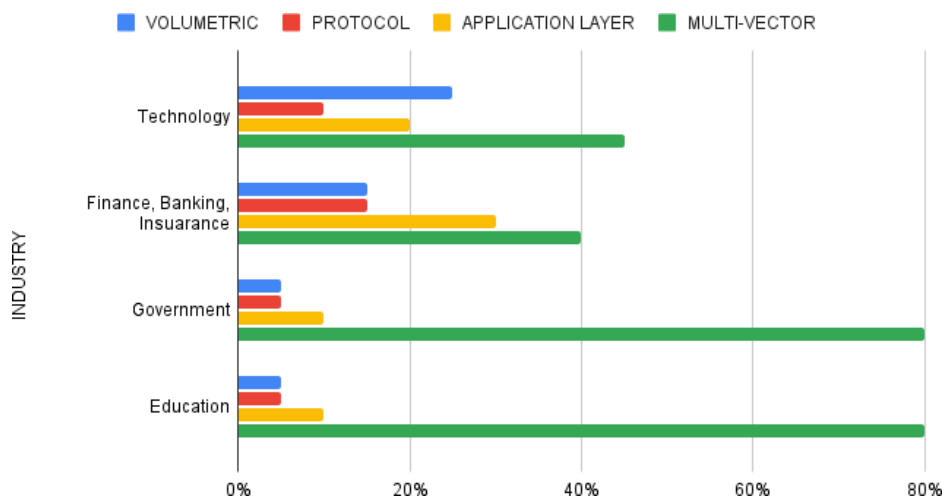


Fig. 4. DDoS Attack types by different industries. Note the frequency of Application layer attacks.

Data Source : <https://www.f5.com/labs/articles/threat-intelligence/2023-ddos-attack-trends>

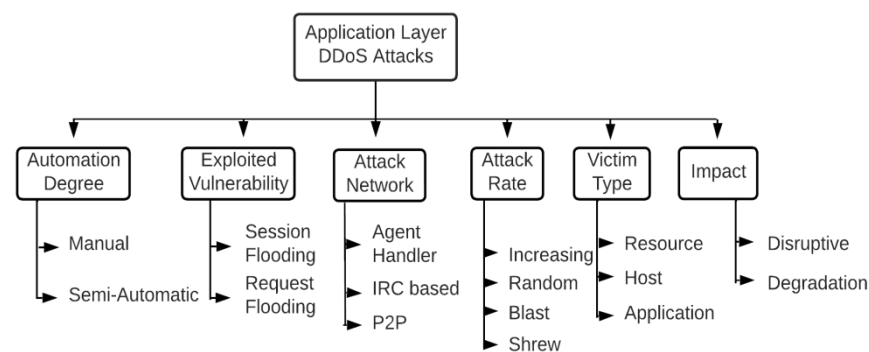


Fig. 5. Taxonomy of AL-DDoS

The variety of application-layer Distributed DoS threats necessitates a thorough investigation of them. The crucial first step in characterising and comprehending this variability is establishing a set of fundamental factors that will enable a complete classification of the present attacks. In this study, we extend the corpus of information concerning application layer attacks and limit our attention to higher-level exclusionary traits.

4. Classification of AL-DDoS Attacks and Corresponding Defending Techniques

Apart from the above taxonomy[5], the DDoS attack on the application layer can also be categorised based on the working principles as shown in figure 7. Initially, AL-DDoS attack is divided into protocol-focused attacks and Generic attacks. Then, these attack types are further divided into DHCP Starvation, SlowHTTP, Request flood and Slow Rate attacks. We discuss each of them in the below sections and provide appropriate defence mechanisms for them. This split is exclusive and exhaustive over the attack set considered here: an attack is classed as protocol-specific if its exploited weakness exists only in one named protocol's state machine (e.g., DHCP's lack of authentication), and generic otherwise, i.e., if the same mechanical weakness (incomplete requests, persistent connections, sheer request volume) generalizes across multiple TCP-based application protocols.

4.1. Protocol-focused DDoS Attacks

Protocol-focused DDoS attacks are only productive against a specific protocol at the application layer. DHCP starvation attack [37] is one type of such attack. Another such attack is Slow HTTP DoS attacks, as shown in figure 7.

4.1.1. DHCP Starvation Attack

Because DHCP lacks a built-in authentication scheme, it is infamously susceptible to DHCP exhaustion attacks [16, 37]. These attacks aim to steal IP addresses from real clients. To deplete the IP address pool, a classic DHCP hunger attack involves generating fictitious requests repeatedly (using fake arbitrary MAC addresses). The DHCP server won't be able to provide new clients with IP addresses once the allocation is full [23, 37]. Another issue that prohibits clients from getting an internet address is an Induced DHCP Starvation attack. This attack takes advantage of IP conflict detection methods on both the server and client sides.

Table 3. Properties that make Application layer DDoS more Disruptive

Property	Description	Disruptiveness
High Ambiguity	Since these attacks employ genuine TCP/UDP connections, distinguishing them from rightful users is difficult.	Severe
Highly Effective DoS	Denial of Service at the application layer is very powerful and effective because it requires limited connections.	Severe
Multiple Effects	These attacks might have direct or indirect repercussions on a large number of victims.	Severe
Normal Traffic Regulation	These assaults follow the norms of valid traffic and complete the TCP handshake, giving the impression that the communication is real.	Moderate
Affect Multiple Applications	These attacks have an impact on many applications since a variety of protocols can be used to execute a Denial of Service attack.	Severe
Ease of Use	These attacks take advantage of layer 7 simplicity.	Low
Limited Resources	The number of resources available for these attacks is limited. Attackers can perform advantageous attacks with little effort.	Severe
Very Targeted	These attacks target individual applications. Destinations are fabricated via HTTP requests. It can conflict with the Web server's hash-ing mechanism by providing ambiguous and duplicate responses.	Severe

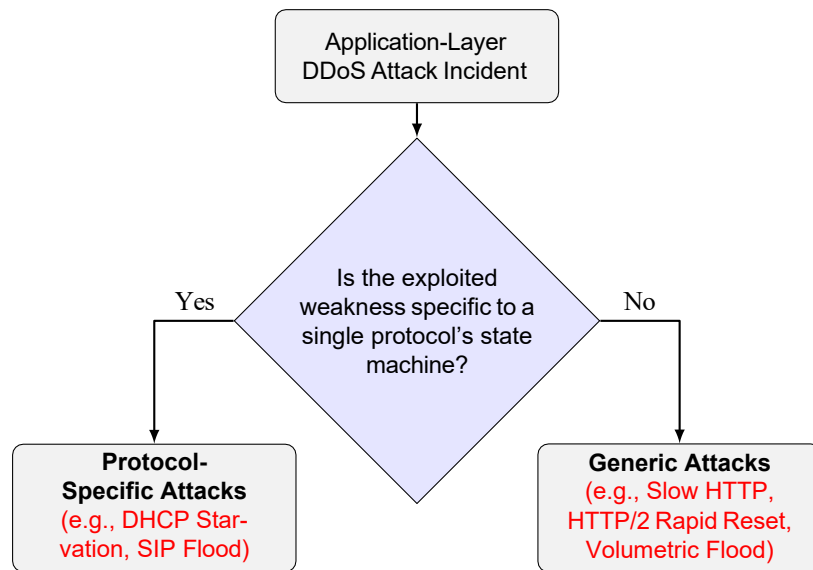


Fig. 6. Taxonomy Classification Logic Flowchart for Layer-7 DDoS Attacks

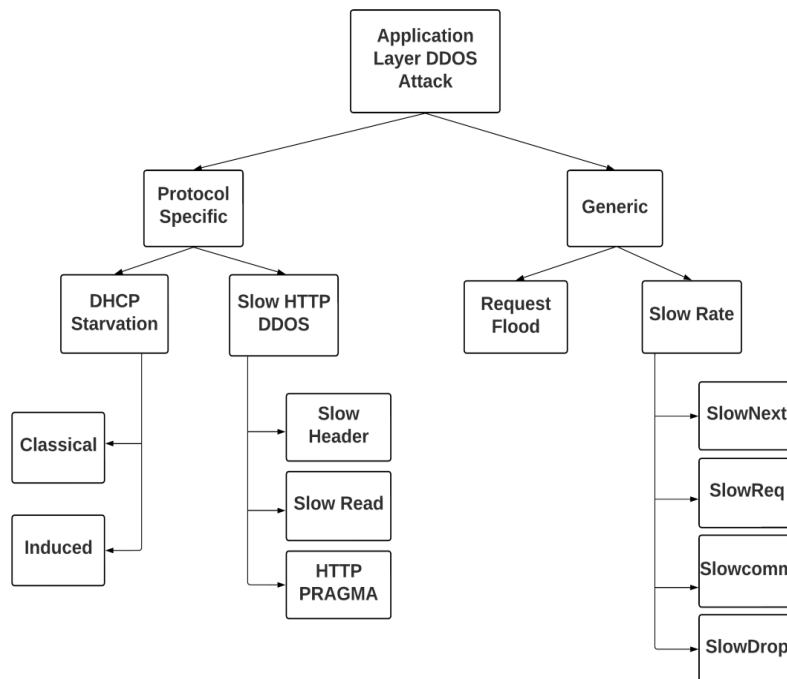


Fig. 7. Categorization of Application Layer DDoS Attacks

Defence Mechanisms against DHCP Starvation: This directly targets the High Ambiguity and Limited Resources properties of Table 3, since DHCP exhaustion requires few forged requests yet is hard to distinguish from genuine client lease requests. Some defence mechanisms against various categories of DHCP Starvation attacks, along with their weaknesses and strengths, are classified in table 4.

4.1.2. Slow HTTP DDoS Attacks

One of the application layer protocols that has been examined the most for protection against DDoS attacks is HTTP. A slow HTTP DDoS attack[7, 20] sends a distinctively designed web request to a server, making it very slow to cooperate with the server. The server queues the request in a connection queue in its limited storage space till it has been completely processed.

- SlowHeader/SlowLoris attack: In order to take advantage of the connection queue space, the attacker launches this attack by uploading incomplete HTTP GET requests to the target server [7, 20].
- Slow Read Attack: An attacker conducts a Slow Read attack by initially sending a web server an innocent

GET request, followed by a TCP packet where the window size is 0 bytes[3, 37]. As a result, the server is unable to contact the attacker.

- **HTTP PRAGMA attack:** The PRAGMA element of the HTTP header is utilized by the HTTP PRAGMA attack [10]. The transmission of a resource that the client has previously requested is solicited by the browser using this element to the web server.

Defence Mechanism against Slow HTTP DDoS: These defences primarily address the Normal Traffic Regulation and Ease of Use properties of Table 3, since Slowloris, Slow Read, and HTTP PRAGMA all complete a valid TCP handshake and exploit the simplicity of the HTTP protocol rather than its volume. Various countermeasures designed to mitigate Slow HTTP DDoS Attacks, detailing both their advantages and limitations are encompassed in Table 5 divided across three categories: (i)Implementation Modules, (ii)Traffic profile comparison, and (iii)Traffic Feature Inspection.

Table 4. Defence techniques against DHCP Starvation attacks

Category	Defending Tech-nique	Strength	Weakness
Cryptographic Tech-niques[37]	Mitigation by imple-menting authentica-tion with DHCP	Any type of iden-tity spoof can be dis-solved	Implementation can be complex
Security Features in switches	Block the switch port if Threshold Exceeds	Easily Deployable	Induced DHCP star-vation attack cannot be mitigated
	Dropping packets as per MAC-IP associa-tion	Easily Deployable	Induced DHCP star-vation attack cannot be mitigated
DHCP Traffic profil-ing[23]	Hellinger distance is used to compare DHCP traffic profiles for detection	Featherweight	Classical DHCP star-vation attack cannot be detected
	Make a track of DHCP messages Over different timescales.	Classical, as well as Induced DHCP star-vation, can be de-tected	A predetermined threshold is difficult to set.
Machine Learning[37]	Using ML algo-rithms	Easy deployment	Classical DHCP can-not be detected

4.2. Generic Attacks

A broad variety of application layer protocols may be successfully attacked using this form of DDoS assault at the application layer. Generic application layer DDoS assaults include Request Flood and Slow Rate generic attacks. They are frequently known as “meta-attacks”[4, 37].

4.2.1. Slow Rate generic DDoS Attacks

Figure 7 displays four standard Slow Rate DDoS assaults that are unaffected by protocols. These attacks take advantage of well-documented security holes in TCP-based layer-7 protocols including HTTP, FTP, and SMTP [12, 28]. These attacks can also change how they behave across a variety of TCP-based protocols.

- **SlowNext Attacks:** The Slow Next DDoS assault is carried out by an attacker using the permanent connection characteristics of protocols like HTTP or SSH or FTP which retain the connections open over an extended period of time [12, 37].
- **SlowReq/SlowDroid Attacks:** To exhaust the connection buffer, the Slowloris attack bombards the web server with countless incomplete HTTP requests. Unlike Slowloris, which only works with HTTP, SlowReq is fruitful against a wide range of TCP-oriented protocols at the application layer.
- **Slowcomm Attacks:** Cambiaso et al. [22] presented Slowcomm, a cyberattack that makes use of the Retransmission Time Out (RTO) mechanism, that TCP utilizes to control overcrowding. Due to its ability to recognize connection shutdown, Slowcomm is more productive than the SlowReq technique.
- **SlowDrop Attacks:** The attacker poses as an unreliable client and requests a large number of resources from the server prior to actually discarding packets. The persistent connection and the deliberate packet rejection lengthen the server’s waiting period. The attacker then initiates a vast amount of such sessions with the host, which results in the DoS problem, in order to prevent genuine clients from interacting with the host.

Defence Mechanism against slow rate generic DDoS: This defence targets the High Ambiguity and Normal Traffic Regulation properties of Table 3, as SlowNext, SlowComm, and SlowDrop all maintain protocol-compliant, persistent connections that are otherwise indistinguishable from genuinely slow legitimate clients.

- **Traffic Feature Inspection:** We have studied protection methods that track the amount of bytes being received by a server over the course of a specific period of time to identify such attacks. If the quantity of packets collected during the period exceeds a specified limit, the period is considered to involve malicious traffic[8, 15].

Table 5. Defence Mechanisms against Slow HTTP DDoS Attacks

Category	Defence Mechanism	Strength	Weakness
Implementation Mod-ules[43]	Incomplete requests should not be added to the connection queue of the server.	It is simple to configure on the Apache web server	HTTP PRAGMA attacks cannot be mitigated
	The amount of incoming requests per IP is confined	Can protect against all sorts of Slow HTTP DoS attacks	If an attack is launched from many IP addresses, it is ineffective
	Assuring a complete request is received in a certain time interval	It is simple to configure on the Apache web server	HTTP PRAGMA attacks cannot be mitigated
Traffic Profile comparison[20, 27]	Detection using HTTP traffic comparison profiles Created during training Hellinger Distance testing phase.	Slow Message body as well as Slow Header assaults can be identified with great veracity.	The attacks cannot be mitigated.
	Using the Chi-square quantitative approach, the training and testing stages' HTTP/2 traffic patterns are contrasted.	Every Known DoS attacks counter to HTTP/2 can be detected	Mitigation cannot be done.
Traffic Features inspection[10]	The connection should terminate if the quantity of transmitted and received packets exceeds a predestined limit	Can help to mitigate attacks	If the web page being browsed includes big items, genuine connections may be dropped

Strength: It is simple to implement in order to detect traffic irregularities.

Weakness: In certain circumstances such as Flash Event, there is a high risk of false positives.

4.2.2. Request Flood Attacks

A further Generic AL-Distributed DDoS threat is Request Flood. In these attacks, an attacker bombards a target server with requests to overburden it and restrict it from handling legitimate ones. As opposed to Slow Rate DDoS assaults, such attacks need the utilization of many computers. They are, therefore, typically regarded as falling under the Distributed DoS kind. Request flood attacks may take many different forms and scales since the attacker has always had the option of sending a stream of data packets from several protocols and the most significant subcategories are HTTP flood followed by SIP flood and DNS Flood as shown in Figure 8.

- **HTTP Flood:** An HTTP flood attack is one of the most common forms of attack that uses request flooding. The above attack can be started by an attacker bombarding a web server with HTTP requests to exhaust its resources and trigger a DoS for genuine customers. HTTP-GET flooding and HTTP-POST flooding assaults are the most prevalent types of attacks in this class [25].
- **SIP Flood:** A SIP flood attack [13] entails flooding a SIP entity with SIP messages in an effort to make it challenging for it to process them and ultimately crash the system. This attack primarily aims to crash the SIP system by consuming its bandwidth, memory, and CPU. Sending innumerable INVITE, REGISTER, or BYE packets to various SIP system parts, such as the SIP Registrar, SIP proxy, etc. enables the initiation of SIP flood assaults.

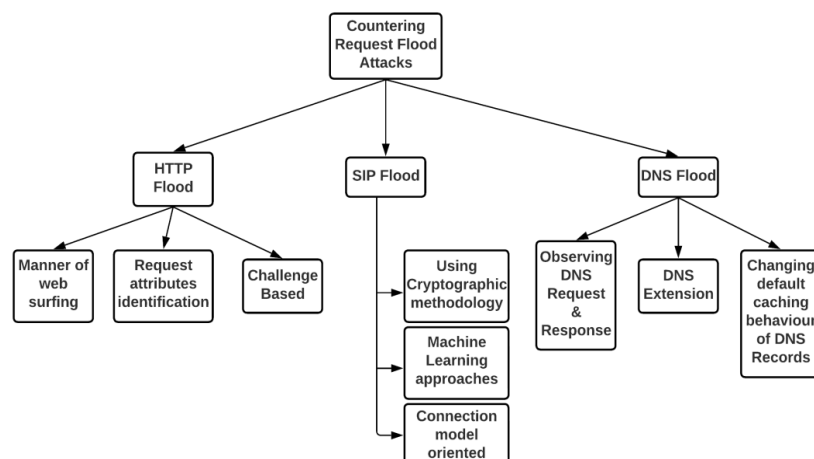


Fig. 8. Classification of Defence techniques against Request flood attacks

- **DNS Flood:** Domain Name System or the DNS protocol is among the most important ones for Internet frameworks. Other crucial services like the web, mail, etc. may be disrupted if the DNS protocol's functionality is compromised. To stop a DNS server from processing and responding to DNS requests provided by good clients, an adversary might execute a DNS flood attack [44].

Defence Mechanism against request flood attacks:

- **Defending techniques against HTTP Flood Attack:** These mechanisms address the Highly Effective DoS and Very Targeted properties of Table 3, since HTTP floods overload a server with comparatively few connections by repeatedly requesting specific resource-intensive pages. Defence strategies entail simulating typical web browsing behaviour using features like request rate, frequency of seeing distinct source IP addresses over time, packet size, order of accessing web pages, etc. Other mechanisms use the number of requests sent for each resource from each connection to distinguish between fraudulent and genuine HTTP connections. Using challenges like "Completely Automated Public Turing Test to Tell Computers and Humans Apart" (CAPTCHA) and "Are You a Human" (AYAH) is another highly popular defence strategy [6].
- **Defending techniques against SIP flood attack:** This defence class addresses the Multiple Effects property of Table 3, since SIP floods directed at the Registrar, Proxy, or other SIP components produce cascading repercussions across the VoIP infrastructure. Employing a range of cryptographic methods and further using machine learning as anomaly detectors, a class of defending strategies can identify SIP flood attacks [7].
- **Defending techniques against DNS flood attack:** This defence targets the Affect Multiple Applications property of Table 3, since degraded DNS resolution cascades into failures of dependent services such as web and mail. Matching DNS requests made by a server towards the responses it sends back is the simplest approach to spot a DNS flood attack. Using DNS Guard, which recognizes attempts using DNS floods to steal someone's identity. Even after a record's TTL has expired, DNS resolvers keep its cached copy. Instead, these outdated records ought to be kept in a backup cache [8].

4.3. Emerging Attack Surfaces and Defence Paradigms

The taxonomy in Figure 7 was developed primarily around host-centric protocol behaviour. The following subsections extend it with three developments published after the foundational works surveyed in Sections 4.1 and 4.2.

4.3.1. Cloud-Native and Protocol-Level Developments

Containerized and serverless deployments introduce a low-rate attack surface distinct from the host-based Slow-Rate attacks of Section 4.2.1: shared kernel resources and orchestration-layer rate limits can be exhausted by request patterns that are individually indistinguishable from legitimate traffic, a property explored for container-based clouds by Li et al.

Table 6. Defence Mechanisms against HTTP Flood Attack

Category	Defence Mechanism	Strength	Weakness
Manner of Web Surfing	Detection by comparison of the profiles created during testing and training procedure.	Effectively categorize various attack types, find them, and mitigate them.	Is evaded by controlling packet-sending rates using a substantial botnet.
	Using browsing order to analyze page access behaviour and a relationship across browsing frequency with page size.	Dynamic Flood can also be recognized.	False alarms as a result of studying as it is a page re-request sequence may differ for various people.
	Comparing a benign's surfing preferences with an adversary.	Improved detecting performance in comparison to others.	The popularity of websites is prone to shift throughout time, so some reports of security breaches may turn out to be incorrect..
Request Attributes identification [5, 24]	Clustering Detection depending on the number of requests per connection per resource.	Ability to recognize invisible bots	problems in detecting Short attack sequences.
	Variation in the ratio of HTTP requests and the network data packets is detected.	Computing complexity is very less.	Asymmetric attack.
Challenge Based [1,6, 21]	Utilizing physical uplink bandwidth on the client side as a tool to detect malicious traffic.	Ability to detect as well as mitigate the attack.	Complex deployment.
	Detection and blocking of illegitimate requests applying the AYAH challenge.	Ability to detect as well as mitigate the attack.	Turnaround time may increase.

	Use of a bound puzzle employing hash-based computing to identify abnormal re-quests.	User experience is not af-fected by using a non-interactive challenge.	Turnaround time may in-crease.
--	--	--	--------------------------------

Table 7. Defence Mechanism against SIP Flood Attack

Category	Defence Mechanism	Strength	Weakness
Using Cryptographic methodology[2]	Mitigation through the negotiation of se-curity protocols like TLS between the UA and proxy	Able to identify and stop the attack	Implementation complexity can be high
	Mitigation by MAC message verification of each exchanged SIP communication	Backward compatible	Requires shared key to be distributed beforehand
Machine Learning Approaches[9]	Utilizing packet-based analysis for detec-tion of a collection of geographical and temporal characteristics	No need to convert network packet streams into traffic flow	Attack initiated using pre-cisely prepared malicious SIP communications cannot be detected
	Using SVM training for detection	Low complexity of imple-mentation	Attack initiated using pre-cisely prepared malicious SIP communications cannot be detected
Connection model oriented[18]	Use of Timed HCPN for Detection	able to identify every attack pertaining to an invite trans-action	flood attack started utilizing several message types can-not be detected
	Detection by the use of Probabilistic Counting Deterministic Timed Automata to simulate SIP communications	Able to recognize flood at-tacks executed at various rates	Computation costs associ-ated with PCDTA genera-tion

Table 8. Defence Mechanisms against DNS Flood

Category	Defence Mechanism	Strength	Weakness
Observing DNS Requests & Re-sponses[36]	Imposing a limit on DNS re-quests that do not have an equivalent response	Easily Deployable	Unscalable for busy nameservers
	Using Bloom filters to con-nect DNS requests and re-sponses	Easily Deployable	Ineffective if equal ISP is used by the adversary and victim nameservers
DNS Extension[17]	In order to identify tampered	DNS protocol does	Attacks conducted
	DNS requests, every request initiator is linked to a dis-tinct cookie	not need modifica-tion	from several com-puters without faking the source IP are failed
	Following DNS above TLS	Implementation needs less memory	Connection oriented DoS attacks
Manipulating the Caching nature of DNS Records	Using a secondary stale cache	Ability to detect as well as mitigate the attack	Implementation is complex
	Extending the TTL for DNS infrastructure entries	Operationally feasi-ble and Immediately available	Updating records is tough task for opera-tors

[32] and addressed architecturally through container-specific zero-trust segmentation by Leahy and Thorpe [38]. A second development is protocol-level rather than volumetric. The 2023 disclosure of the HTTP/2 "Rapid Reset" technique (CVE-2023-44487) demonstrated a new generic Request Flood variant (Section 4.2.2) that abuses HTTP/2 stream multiplexing: a client opens and immediately cancels streams via RST STREAM frames, letting it issue a new request before the server finishes tearing down the previous one, multiplying effective request throughput without exceeding any per-connection concurrency limit. Coordinated disclosures reported peak rates an order of magnitude beyond the HTTP-GET/POST flood rates discussed in Section 4.2.2, generated from comparatively small botnets [41, 42], illustrating that Request Flood attacks continue to evolve through protocol mechanics rather than sheer botnet size alone.

4.3.2. AI/ML-Driven Detection and Adversarial Evasion

Sections 4.1 and 4.2 describe machine-learning-based defences (Tables 4, 7) only at the level of "uses ML algorithms." In practice, ML/DL-based AL-DDoS detectors typically operate on flow-level feature vectors – inter-arrival time between requests, HTTP header-field entropy, request-to-response size ratio, and source-IP request-rate distribution – extracted over a sliding window and used to train supervised classifiers or deep sequence models on labelled traffic. This shift introduces a defence-side risk largely absent from Sections 4.1 and 4.2: adversarial evasion, in which an attacker perturbs request timing or header fields specifically to remain inside the decision boundary learned by the classifier, undermining the detection confidence that motivated the move to ML in the first place. None of the cryptographic, profiling, or rule-based defences catalogued in Tables 4–8 address this failure mode, which we revisit as an open problem in the Conclusion.

4.3.3. Zero-Trust Integration

The defences in Sections 4.1 and 4.2 are predominantly perimeter- or server-side controls. Zero Trust Architecture (ZTA) instead assumes no implicit trust for any request regardless of network origin, continuously verifying identity and context before granting access to application resources [39]. Applied to AL-DDoS mitigation, this reframes challenge-based defences (Table 6) from a one-off CAPTCHA/AYAH gate into continuous, risk-scored request verification, and complements traffic-profiling defences (Tables 5, 6) by adding identity- and context-aware policy enforcement at the point of each request rather than relying solely on aggregate traffic statistics. We treat ZTA integration with AL-DDoS mitigation as a research direction rather than an established, evaluated defence, since no peer-reviewed study has yet benchmarked a ZTA-integrated pipeline against the attack taxonomy of Figure 7.

5. Comparison of AL-DDoS Attacks and Defence Techniques

A comparison of several application layer DDoS attacks opens in this section. Then, based on a number of factors, we evaluate the current attack-prevention mechanisms in place.

5.1. Comparison of Attacks

In order to evaluate diverse application layer DoS attacks, we take into account a number of factors, including complexity, target, stealthiness, attack type, spoofing, and real incidents as described in Table 9.

Table 9. Comparing Different Layer-7 DoS Attacks

Attack Category	Specific Attack	Complex	Target	Stealth	Attack Type	Spoofing	Incidents
<i>Attacks against DHCP</i>	Induced	Low	Server	Medium	DoS	✓	[16]
	Classical	High	Server	Low	DoS	✓	[23]
<i>Slow HTTP DoS</i>	Slow Header	Low	Server	Medium	DoS	✗	[20]
	Slow Read	Low	Server	Medium	DoS	✗	[3]
	HTTP PRAGMA	Low	Server	Medium	DoS	✗	[10]
<i>Slow Rate Generic DoS</i>	SlowReq	Low	Server	Medium	DoS	✗	✗
	SlowComm	Low	Server	Medium	DoS	✗	✗
	SlowNext	Low	Server	High	DoS	✗	✗
	SlowDrop	Low	Server	High	DoS	✗	✗
<i>Request Flood</i>	HTTP Flood	High	Server	Medium	DDoS	✗	[25]
	SIP Flood	High	Server	Medium	DDoS	✗	[13]
	DNS Flood	High	Server	Medium	DDoS	✗	[44]
	HTTP/2 Rapid Reset	Medium	Server	High	DDoS	✗	[41, 42]

Criteria. Complexity – Low: single host, no coordination required; Medium: protocol-state manipulation from a single host; High: requires coordinated multi-host/botnet orchestration. Stealth – Low: produces a volumetric/rate anomaly detectable by simple thresholding; Medium: requires traffic-profile or protocol-state inspection to detect; High: statistically indistinguishable from legitimate sessions at the flow level without deep packet inspection.

Criteria. Proactive (preventive) – the mechanism blocks or limits the attack vector before it can be exploited. Reactive (detective) – the mechanism identifies an attack already in progress via monitoring or traffic analysis and may or may not trigger automatic mitigation.

The traditional DHCP hunger exploit is very sophisticated since it necessitates making several false queries using random MAC addresses. Request flood attacks are sophisticated because they call for flooding the target server with requests. An induced DHCP starvation attack is more stealthy than a traditional DHCP starvation attack since it only sends spoof probe replies. The data also shows that the Slow HTTP Denial of Service attacks and Slow Rate generic Denial of Service attacks incur minimal bandwidth congestion and are hence more covert. Of all the AL-Distributed DoS strategies examined in this study, only DHCP hunger attacks are confined to local networks. Every other assault is usable online and has the potential to have a significant impact on a mass audience.

5.2. Comparison of Defence Techniques

To assess various defence techniques, we mainly take three parameters into consideration viz.

- Proactive or Reactive: To distinguish between defence strategies that actively stop attacks from happening or

those just respond to those that are already underway.

- **Mitigation:** If they are simply able to detect assaults or if they are able to counteract DoS attacks.
- **Deployment Complexity:** It has to be simple and practical to implement defence systems during a conflict.

The detailed comparison is presented in Table 10.

Table 10. Comparing All Defence Techniques

Category	Defence Mechanism	Nature	Mitigation	Deployment Complexity
Cryptographic Techniques	DHCP authentication	Proactive	Yes	High
	SIP TLS negotiation	Proactive	Yes	High
Architectural / Implementation Controls	Built-in Security in Switches	Proactive	Yes	Low
	Implementation Modules (Slow HTTP)	Proactive	Yes	Low
	DNS Extension	Proactive	Yes	Low
	Manipulating the Caching nature of DNS Records	Proactive	Yes	High
Traffic Profiling / Statistical Inspection	Analyzing DHCP Traffic	Reactive	No	Low
	Traffic Profile Comparison (Slow HTTP)	Reactive	No	Low
	Traffic Features inspection (Slow HTTP)	Reactive	Yes	Low
	Traffic Features Inspection (Slow Rate Generic)	Reactive	Yes	Low
	Manner of Web Surfing (HTTP Flood)	Reactive	No	Low
	Request Attributes identification (HTTP Flood)	Reactive	No	Low
	Observing DNS Requests and Responses	Proactive	Yes	Low
Machine Learning	Using concept of Machine Learning (DHCP)	Reactive	No	Low
	Machine Learning Approaches (SIP Flood)	Reactive	No	Low
	Connection model oriented (SIP Flood)	Reactive	No	Low
Challenge-Based	Challenge Based (HTTP Flood)	Proactive	Yes	High

The "Cryptographic Methods" class of defence mechanisms has strongholds with considerable deployment complexity. Since third-party elements like CAPTCHA formation and verification servers must be implemented, challenge-based protection mechanisms against HTTP flood attacks are hard to set up. Mitigation techniques that alter the functionality of the typical DNS cache to thwart DNS flood assaults are highly complicated to deploy since they change the way the protocol operates.

Table 11 extends this comparison by cross-referencing each defence category in Table 10 against the underlying technical principle it relies on, making explicit which principles recur across attack families and which remain attack-specific.

Table 11. Defence-Principle Comparison Matrix

Underlying Principle	DHCP	Slow HTTP	Slow Rate Generic	Request Flood	SIP/DNS
Signature/Threshold	✓	✓	✓	✓	✓
Statistical Profiling	✓	✓	✗	✓	✓
Supervised ML	✓	✗	✗	✗	✓
Cryptographic Challenge	✓	✗	✗	✓	✓
Architectural Isolation	✓	✓	✓	✓	✗

5.3. Analytical Remarks on DDoS Attacks & Mitigation

A Distributed Denial of Service (DDoS) attack is a type of cyber attack in which a large number of compromised systems, known as botnets, are used to flood a target website or network with traffic, rendering it inaccessible to legitimate users. DDoS attacks can be launched at different layers of the network stack, including the application layer.

When a DDoS attack targets the application layer, it can overwhelm the server by sending a large number of incomplete or malicious requests. These requests can exploit vulnerabilities in the application layer protocols, such as HTTP, SMTP, or FTP, causing the server to crash or become unresponsive. DDoS attacks on the application layer are currently a significant security risk because they can be difficult to detect and mitigate.

This survey highlights that there has been a sharp increase in the development of several defence mechanisms to identify and reduce DDoS attacks on the application layer. The research and studies have shown some of the new or updated approaches that have been developed to improve the ability to handle attacks and increase the degree of security against these attacks.

One of the key methods for detecting and mitigating DDoS attacks on the application layer is Deep Packet Inspection (DPI). DPI is a technique that examines the content of network traffic in real-time and identifies potential threats. By using DPI, security systems can detect attacks like Slowloris and SlowReq, which demand delivering unfinished requests, and discard them before they reach the target server.

Another important approach to defend against DDoS attacks is to keep track of the rate of incoming traffic for a specific customer and reroute that traffic to the security provider's data centers. This helps to prevent request flood attacks, where the target server is overwhelmed with a large number of requests. The attack traffic should be first filtered in these data centers, and then the cleaned traffic is routed to the user's Internet address range (IP).

Defence mechanisms must also keep track of the connection properties of the fundamental transport layer protocol, such as TCP, to defend against SlowNext and Slowdrop attacks. This is because malicious clients may carry out these attacks by taking advantage of TCP's persistent connection capability.

The study also highlights that while some defence measures can only identify DDoS attacks, they may also neutralize operations against DHCP and Slow HTTP DDoS attacks. For example, traffic profiling comparison and online browsing analysis can help to identify DDoS attacks and prevent them from causing significant damage.

In a nutshell, DDoS attacks on the application layer are a significant security risk, and there has been a sharp increase in the development of several defence mechanisms to identify and reduce these attacks. The authors have shown some of the new or updated approaches that have been developed to improve the ability to handle attacks and increase the degree of security against these attacks. These defence mechanisms use techniques like DPI, rerouting traffic to data centers, and keeping track of connection properties to defend against various types of DDoS attacks.

6. Conclusion

This survey developed a taxonomy of AL-DDoS attacks and an explicit, criterion-based comparative framework. Three bottlenecks recur: most statistical/ML defences are detection-only and do not mitigate; the highest-mitigation-capability defences (cryptographic, challenge-based) carry the highest deployment complexity; and profiling-based detectors remain vulnerable to Flash-Event false positives. This survey's scope is limited by the absence of original experimental validation and by reliance on secondary incident. Future work should pursue (i) detection robust to adversarial perturbation, (ii) low-rate AL-DDoS detection for containerized/serverless environments, and (iii) empirical evaluation of zero-trust-integrated Layer-7 mitigation.

All the Declarations and Statements

Author Contributions Statement

Aditya Arsh – Conceptualization, literature review, methodology design, framework formalization, original draft.

Priyanka Biswas – Methodology refinement, comparative analysis, review and editing. Nirmalya Kar – Supervision, validation, project administration.

All authors have read and agreed to the published version of the manuscript.

Conflict of Interest Statement

The authors declare no conflicts of interest.

Funding Declaration

This research received no external funding.

Data Availability Statement

No new datasets were generated; Table 2 data are drawn from the cited public sources.

Ethical Declarations

Not applicable.

Acknowledgments

We thank the editor and reviewers for their valuable feedback.

Declaration of Generative AI in Scholarly Writing

Generative AI tools were used to assist with language editing and reference formatting during revision; the authors reviewed and verified all technical content and take full responsibility for it.

Abbreviations

The following abbreviations are used in this manuscript:

AL-DDoS – Application Layer DDoS

DPI – Deep Packet Inspection

ML – Machine Learning

ZTA – Zero Trust Architecture

Appendix A/B/C..., with appendix title

N/A

References

- [1] Suriadi et al. “Defending web services against denial of service attacks using client puzzles”. In: 2011 IEEE International Conference on Web Services. IEEE. 2011, pp. 25–32. DOI: 10.1109/ICWS.2011.22.
- [2] Ryan Farley and Xinyuan Wang. “VoIP shield: A transparent protection of deployed VoIP systems from SIP-based exploits”. In: 2012 IEEE Network Operations and Management Symposium. IEEE. 2012, pp. 486–489. DOI: 10.1109/NOMS.2012.6211937.
- [3] Sergey Shekhan. “Are you ready for slow reading?” In: QUALYS BLOG, Available: <https://community.qualys.com/blogs/securitylabs/2012/01/05/slow-read> (Last access: Dec. 12, 2014) (2012).
- [4] Enrico Cambiaso et al. “Slow DoS attacks: definition and categorisation”. In: International Journal of Trust Management in Computing and Communications 1.3-4 (2013), pp. 300–319. DOI: <https://doi.org/10.1504/IJTMCC.2013.056440>.
- [5] Pawel Chwalinski, Roman Belavkin, and Xiaochun Cheng. “Detection of application layer DDoS attack with clustering and likelihood analysis”. In: 2013 IEEE Globecom Workshops (GC Wkshps). IEEE. 2013, pp. 217–222. DOI: 10.1109/GLOCOMW.2013.6824989.
- [6] Sujatha Sivabalan and PJ Radcliffe. “A novel framework to detect and block DDoS attack at the application layer”. In: IEEE 2013 Tencon-Spring. IEEE. 2013, pp. 578–582. DOI: 10.1109/TENCONSpring.2013.6584511.
- [7] Saman Taghavi Zargar, James Joshi, and David Tipper. “A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks”. In: IEEE communications surveys & tutorials 15.4 (2013), pp. 2046–2069. DOI: 10.1109/SURV.2013.031413.00127.
- [8] Maurizio Aiello et al. “An on-line intrusion detection approach to identify low-rate DoS attacks”. In: 2014 International Carnahan Conference on Security Technology (ICCST). IEEE. 2014, pp. 1–6. DOI: 10.1109/CCST.2014.6987039.
- [9] Muhammad Ali Akbar and Muddassar Farooq. “Securing SIP-based VoIP infrastructure against flooding attacks and Spam Over IP Telephony”. In: Knowledge and information systems 38.2 (2014), pp. 491–510. DOI: <https://doi.org/10.1007/s10115-012-0595-5>.
- [10] Yuri Gil Dantas, Vivek Nigam, and Iguatemi E Fonseca. “A selective defense for application layer DDoS attacks”. In: 2014 IEEE joint intelligence and security informatics conference. IEEE. 2014, pp. 75–82. DOI: 10.1109/JISIC.2014.21.
- [11] Gulshan Kumar et al. “Understanding denial of service (dos) attacks using osi reference model”. In: International Journal of Education and Science Research 1.5 (2014).
- [12] Enrico Cambiaso et al. “Designing and modeling the slow next DoS attack”. In: Computational intelligence in security for information systems conference. Springer. 2015, pp. 249–259. DOI: https://doi.org/10.1007/978-3-319-19713-5_22.
- [13] Intesab Hussain et al. “A comprehensive study of flooding attack consequences and countermeasures in session initiation protocol (sip)”. In: Security and Communication Networks 8.18 (2015), pp. 4436–4451. DOI: <https://doi.org/10.1002/sec.1328>.
- [14] Georgios Mantas et al. “Application-layer denial of service attacks: taxonomy and survey”. In: International Journal of Information and Computer Security 7.2/3/4 (2015), pp. 216–239. DOI: <https://doi.org/10.1504/IJICS.2015.073028>.
- [15] Maurizio Mongelli et al. “Detection of DoS attacks through Fourier transform and mutual information”. In: 2015 IEEE International Conference on Communications (ICC). IEEE. 2015, pp. 7204–7209. DOI: 10.1109/ICC.2015.7249476.
- [16] Nikhil Tripathi and Neminath Hubballi. “Exploiting DHCP server-side IP address conflict detection: A DHCP starvation attack”. In: 2015 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS). IEEE. 2015, pp. 1–3. DOI: 10.1109/ANTS.2015.7413661.
- [17] Liang Zhu et al. “Connection-oriented DNS to improve privacy and security”. In: 2015 IEEE symposium on security and privacy. IEEE. 2015, pp. 171–186. DOI: 10.1109/SP.2015.18.
- [18] Diksha Golait and Neminath Hubballi. “Detecting anomalous behavior in VoIP systems: A discrete event system modeling”. In: IEEE Transactions on Information Forensics and Security 12.3 (2016), pp. 730–745. DOI: 10.1109/TIFS.2016.2632071.
- [19] Vinod Kumar, Krishan Kumar, et al. “Classification of DDoS attack tools and its handling techniques and strategy at application layer”. In: 2016 2nd International Conference on Advances in Computing, Communication, & Automation (ICACCA)(Fall). IEEE. 2016, pp. 1–6. DOI: 10.1109/ICACCAF.2016.7749002.

- [20] Nikhil Tripathi, Neminath Hubballi, and Yogendra Singh. "How secure are web servers? An empirical study of slow HTTP DoS attacks and detection". In: 2016 11th International Conference on Availability, Reliability and Security (ARES). IEEE. 2016, pp. 454–463. DOI: 10.1109/ARES.2016.20.
- [21] Heng Zhang et al. "Sentry: A novel approach for mitigating application layer DDoS threats". In: 2016 IEEE TrustCom/BigDataSE/ISPA. IEEE. 2016, pp. 465–472. DOI: 10.1109/TrustCom.2016.0098.
- [22] Enrico Cambiaso, Gianluca Papaleo, and Maurizio Aiello. "Slowcomm: Design, development and performance evaluation of a new slow DoS attack". In: Journal of Information Security and Applications 35 (2017), pp. 23–31. DOI: <https://doi.org/10.1016/j.jisa.2017.05.005>.
- [23] Neminath Hubballi and Nikhil Tripathi. "A closer look into DHCP starvation attack in wireless networks". In: Computers & Security 65 (2017), pp. 387–404. DOI: <https://doi.org/10.1016/j.cose.2016.10.002>.
- [24] Hossein Hadian Jazi et al. "Detecting HTTP-based application layer DoS attacks on web servers in the presence of sampling". In: Computer Networks 121 (2017), pp. 25–36. DOI: 10.1016/j.comnet.2017.03.018.
- [25] Karanpreet Singh, Paramvir Singh, and Krishan Kumar. "Application layer HTTP-GET flood DDoS attacks: Re-search landscape and challenges". In: Computers & security 65 (2017), pp. 344–372. DOI: <https://doi.org/10.1016/j.cose.2016.10.005>.
- [26] Amit Praseed and P Santhi Thilagam. "DDoS attacks at the application layer: Challenges and research perspectives for safeguarding web applications". In: IEEE Communications Surveys & Tutorials 21.1 (2018), pp. 661–685. DOI: 10.1109/COMST.2018.2870658.
- [27] Nikhil Tripathi and Neminath Hubballi. "Slow rate denial of service attacks against HTTP/2 and detection". In: Computers & security 72 (2018), pp. 255–272. DOI: <https://doi.org/10.1016/j.cose.2017.09.009>.
- [28] Enrico Cambiaso, Giovanni Chiola, and Maurizio Aiello. "Introducing the slowdrop attack". In: Computer Networks 150 (2019), pp. 234–249. DOI: 10.1016/j.comnet.2019.01.007.
- [29] Shail Saharan and Vishal Gupta. "Prevention and Mitigation of DNS based DDoS attacks in SDN Environment". In: 2019 11th international conference on communication systems & networks (COMSNETS). IEEE. 2019, pp. 571–573. DOI: 10.1109/COMSNETS.2019.8711258.
- [30] Amazon Web Services. AWS Shield Threat Landscape Report – Q1 2020. 2020. URL: <https://aws.amazon.com/blogs/security/aws-shield-threat-landscape-report-now-available/> (visited on 06/23/2026).
- [31] Felipe S Dantas Silva et al. "A taxonomy of DDoS attack mitigation approaches featured by SDN technologies in IoT scenarios". In: Sensors 20.11 (2020), p. 3078. DOI: <https://doi.org/10.3390/s20113078>.
- [32] Zhi Li et al. "Exploring New Opportunities to Defeat Low-Rate DDoS Attack in Container-Based Cloud Environment". In: IEEE Transactions on Parallel and Distributed Systems 31.3 (2020), pp. 695–706. DOI: 10.1109/TPDS.2019.2942591.
- [33] Waleed Nazih et al. "Survey of countering DoS/DDoS attacks on SIP based VoIP networks". In: Electronics 9.11 (2020), p. 1827. DOI: [SurveyofcounteringDoS/DDoSattacksSIPbasedVoIPnetworks](https://doi.org/10.3390/e9111827).
- [34] Hadeel S Obaid and Esamaddin H Abeed. "DoS and DDoS attacks at OSI layers". In: International Journal of Multidisciplinary Research and Publications 2.8 (2020), pp. 1–9. ISSN: 2581-6187(Online).
- [35] Modupe Odusami et al. A Survey and Meta-Analysis of Application-Layer Distributed Denial-of-Service Attack. 2020. DOI: <https://doi.org/10.1002/dac.4603>.
- [36] Marcin Nawrocki et al. "The far side of DNS amplification: tracing the DDoS attack ecosystem from the internet core". In: Proceedings of the 21st ACM Internet Measurement Conference. 2021, pp. 419–434. DOI: <https://doi.org/10.1145/3487552.3487835>.
- [37] Nikhil Tripathi and Neminath Hubballi. "Application layer denial-of-service attacks and defense mechanisms: a survey". In: ACM Computing Surveys (CSUR) 54.4 (2021), pp. 1–33. DOI: <https://doi.org/10.1145/3448291>.
- [38] Darragh Leahy and Christina Thorpe. "Zero Trust Container Architecture (ZTCA): A Framework for Applying Zero Trust Principles to Docker Containers". In: International Conference on Cyber Warfare and Security. Vol. 17. 2022, pp. 111–120.
- [39] Naeem Firdous Syed et al. "Zero Trust Architecture (ZTA): A Comprehensive Survey". In: IEEE Access 10 (2022), pp. 57143–57179. DOI: 10.1109/ACCESS.2022.3174679.
- [40] Omer Yoachimik. DDoS attack trends for 2022 Q2. July 6, 2022. URL: <https://blog.cloudflare.com/ddos-attack-trends-for-2022-q2/> (visited on 06/23/2026).
- [41] CISA. HTTP/2 Rapid Reset Vulnerability, CVE-2023-44487. 2023. URL: <https://www.cisa.gov/news-events/alerts/2023/10/10/http2-rapid-reset-vulnerability-cve-2023-44487> (visited on 06/23/2026).
- [42] Lucas Pardue and Will Desgats. HTTP/2 Rapid Reset: Deconstructing the Record-Breaking Attack. 2023. URL: <https://blog.cloudflare.com/technical-breakdown-http2-rapid-reset-ddos-attack/> (visited on 06/23/2026).
- [43] Apache Software Foundation. Core Apache HTTP Server Features. Apache HTTP Server Version 2.4 Documentation. Apache Software Foundation. 2026. URL: <https://httpd.apache.org/docs/2.4/mod/core.html>.
- [44] Beloslava Petrova. DNS flood attack explained in details. June 18, 2026. URL: <https://www.cloudns.net/blog/dns-flood-attack-explained-in-details/> (visited on 06/23/2026).

Authors' Profiles



Aditya Arsh is a Project Engineer at CDAC Bangalore, engaged in cyber security research and development initiatives. His areas of work include DNS security, domain name and infrastructure analysis, malware analysis, vulnerability assessment and penetration testing (VAPT), and threat detection and analysis. He holds a bachelor's degree in computer science and engineering from BIT Mesra and a Master of Technology (M. Tech) degree in Cyber Security from NIT Agartala. His interests lie in applied cyber security research, detection of malicious online activity, and strengthening defensive security mechanisms.



Priyanka Biswas is a Ph.D. Scholar in the Department of CSE, NIT Agartala. She has a B. Tech and an M. Tech degree in Information Technology from Maulana Abul Kalam Azad University of Technology. She is currently working on DNA computing, Cryptography, Steganography, and privacy. She is a student member of IEEE.



Nirmalya Kar is an Assistant Professor in the Department of Computer Science and Engineering at the National Institute of Technology (NIT) Agartala. He also serves as the Chief Information Security Officer at the Institution. Having more than 18 years of teaching and research experience, Dr. Kar specializes in Information Security, Cryptography, Computational Intelligence and the Internet of Things (IoT). He has more than 50+ research contributions in referred journals, book chapters and conference proceedings and has been involved in various academic and administrative roles, including organising chair and general chair of International Conferences, overseeing campus-wide networking and coordinating high-performance computing initiatives at NIT Agartala.

How to cite this paper

Aditya Arsh, Priyanka Biswas, Nirmalya Kar, "Application-Layer DDoS Attacks and Defences: A Taxonomy, Comparative Evaluation Framework, and Research Directions", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.16, No.4, pp. 300-316, 2026. DOI:10.5815/ijwmt.2026.04.17