Discover Internet of Things

REVIEW Open Access

Enhancing public cloud resilience: an analytical review of detection and mitigation strategies against economic denial of sustainability attacks

Zubaidi Maytham Sahar Saeed [1*], Anazida Binti Zainal[1], Fuad A. Ghaleb[2] and Bander Ali Saleh Al-rimy[3]

*Correspondence:
Zubaidi Maytham Sahar Saeed
sahar20@graduate.utm.my
[1]Faculty of Computing, University of Technology Malaysia, Johor Baharu, Malaysia
[2]College of Computing, Faculty of Computing, Engineering and the Built Environment, Birmingham City University, Birmingham B4 7XG, UK
[3]School of Computing, University of Portsmouth, Buckingham Building, Lion Terrace, Portsmouth PO1 3HE, UK

Abstract

Cloud computing (CC) delivers computing resources as utilities, akin to services like electricity or water. However, security concerns—particularly Distributed Denial of Service (DDoS) and its economically targeted variant, Economic Denial of Sustainability (EDoS)—pose significant threats to its adoption. EDoS attacks exploit the pay-per-use and auto-scaling features of CC platforms to incur financial damage by triggering unnecessary resource consumption. While existing studies have proposed various countermeasures, comprehensive, comparative analysis remains limited. This review systematically examines 69 key articles addressing EDoS-specific or joint DDoS–EDoS threats. Beyond merely cataloguing these methods, this review provides a novel analytical synthesis by categorizing defense strategies into detection, prevention, mitigation, and hybrid models, and critically evaluating them against factors such as scalability, computational overhead, and false-positive rates. Importantly, this study introduces a service-model-aware framework, distinguishing which solutions are most effective for Infrastructure as a Service (IaaS) versus Software as a Service (SaaS) environments. By mapping techniques to operational contexts, the review reveals methodological gaps, highlights practical deployment challenges, and proposes priorities for future research and cloud-specific security design. Articles Highlights: • Provides a systematic review of EDoS attacks in cloud computing to understand current issues and limitations. • Classifies EDoS defences into four strategic categories to guide future research on key gaps. • Suggests future EDoS research focusing on AI, blockchain, and economic impacts to enhance defence effectiveness.

Article Highlights

- Provides a systematic review of EDoS attacks in cloud computing to understand current issues and limitations
- Classifies EDoS defences into four strategic categories to guide future research on key gaps
- Suggests future EDoS research focusing on AI, blockchain, and economic impacts to enhance defence effectiveness

Keywords Cloud computing, Distributed denial of service, Economic denial of sustainability, Graphical turing test, Machine learning

1 Introduction

Over the past few years, the computational field has changed tremendously due to the rising demand for advanced computing technology. This technological advancement has facilitated various innovative computing models, such as cloud, grid, and cluster computing. Hence, cloud computing has emerged as a prevalent infrastructure that enables customers to access shared resources with minimal assistance from cloud service providers (CSPs). Given that this infrastructure can increase effectiveness, dependability, and performance by combining scattered resources, cloud computing can permit service sharing [1].

Cloud computing models have emerged as the most viable approach, offering consumers a suitable and economical means to access computing utilities. The models usually distribute infrastructure, platform, and software services across shared networks, which are accessible on a pay-per-use basis. This process is similar to essential utilities, such as electricity and telecommunications [2].

The pay-per-use model of cloud computing allows payment solely for utilised resources. This model is also scalable and elastic, enabling usage adjustments based on the necessity to reduce organisational capital expenses considerably. Small businesses can then substantially benefit from this outcome, granting them access to ICT infrastructure that is traditionally unaffordable. Consequently, cloud computing remains an appealing choice for organisations aiming to minimise upfront costs [3].

Numerous services offered by cloud providers (vendors) to their clients can be examined using various frameworks, including Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). The SaaS framework represents a software distribution model that allows clients to access applications over a network, with the hosting managed by a vendor. This process is recognised as the most sophisticated manifestation of cloud computing.

The PaaS framework involves developing and executing applications based on platforms equipped by cloud providers. This architectural framework demonstrates rapid and cost-effective application design with deployment. Notable examples of PaaS are Microsoft with its corresponding Windows Azure platform, Salesforce.com with its corresponding Force.com, and Google with its corresponding Google App Engine. Numerous services are also observed in the PaaS framework, such as application serving, database (DB) management, security, and workflow management [4].

The IaaS framework delivers computational power and storage capacity on a demand-driven basis and contains various resources. One leading example of IaaS is Amazon.com, which allows users to execute cloud application programmes on its Elastic Compute Cloud (EC2) web service. Other notable examples employing IaaS solutions are HP, IBM, and VMware [4].

Significant correlations are observed between cloud computing and several industries (e-commerce, e-learning, and healthcare). This feature substantially enhances economic value as cloud computing provides inexpensive internet services. The phenomenon catalyses the next major shift in the internet and commercial sectors. Therefore,

e-commerce businesses, institutions and enterprises are transitioning to the cloud to enhance utility [5].

Big businesses frequently express apprehensions over their extensive data security during internet transmission. This issue indicates that practical cloud computing in an enterprise environment necessitates careful planning and a thorough comprehension of changing risks, threats, vulnerabilities, and workable solutions. Hence, an essential aspect of safeguarding cloud computing environments is developing secure applications from inherently unreliable components. Despite significant security risks in cloud computing, this technology can also reduce enterprise costs [6].

Several well-known threats can compromise network and cloud security, such as Denial of Service (DoS), Distributed Denial of Service (DDoS), port scanning, phishing, and Man-in-the-Middle attacks. Nonetheless, a newer threat [Economic Denial of Sustainability (EDoS)] has recently been observed in exploiting the pay-per-use and elasticity features of cloud computing to inflate users' costs. This threat is derived from DDoS attacks, which can induce financial strain by covertly increasing resource utilisation expenses.

Generally, EDoS attacks in cloud computing constitute a novel category of financial and security threats. Compared to traditional DDoS attacks that overload server resources to interrupt services, EDoS attacks exploit the flexibility of cloud services. These attacks can impose resources to grow to meet demand, escalating consumer costs dynamically [7].

An EDoS attack presents a distinct variant from the traditional DDoS attack, primarily in its operational influence and outcome. In contrast, a DDoS attack debilitates the capacity of a system to assist its client until recovery is achieved. Therefore, an EDoS attack perniciously exploits excessive resources without triggering the intrusion detection system (IDS) alarms. This consumption can continue undetected for a period ranging from several hours to multiple weeks. Finally, this unauthorised use of resources results in unexpectedly high financial charges for the consumer, creating economic strain without the immediate visibility associated with traditional cyberattacks.

Al-Haidari et al. documented that cloud computing costs were 15 times higher when standard analytic modelling was simulated with EDoS attacks (6000 requests per second) [8]. These attacks could also increase energy costs alongside financial costs. One notable example involved an EDoS attack on Amazon EC2. This threat incurred an additional energy charge of 1440 kWh daily or $5184 monthly, highlighting the more significant expenses associated with the EDoS attacks [9].

Another practical example was the cloud-based GreatFire.org (Amazon-hosted website), which experienced an EDoS attack on March 17, 2015. This attack revealed a substantial economic cost on bandwidth (up to $30,000 per day) and a sudden increase in requests (up to 2.6 billion per hour). The request was also approximately 2500 times the normal range. Considering the overwhelming surge of demands directed to the website administrator beyond the operational planning and capacity, external help was requested [10, 11]. This outcome implied that EDoS attacks unequivocally resulted in resource depletion and financial losses. Consequently, EDoS countermeasures should be examined to safeguard the future and safety of the cloud computing industry.

Previous articles focusing on comprehending and alleviating the impact of EDoS attacks in cloud environments yielded various defence proposals. Nonetheless, this

review identified four significant research gaps: (i) inadequate detection and prevention, (ii) insufficient performance evaluation, (iii) comprehensive focus involving general DoS or DDoS defence, and (iv) outdated research lacking updates since 2019 (neglected latest developments in the field). Thus, this review addressed these gaps regarding EDoS countermeasures by comprehensively reviewing each publicly available defence strategy, performance metrics, and cloud environment limitation.

This review is structured into six sections for clarity. Section 1 offers an overview of the introduction. Section 2 discusses various related articles. Section 3 details the inclusion and exclusion criteria for research articles. Section 4 presents the correlation analysis between DDoS and EDoS attacks. This section also explores the existing solutions, categorisation process, findings from previous articles, and future research directions. Section 5 highlights the importance of addressing EDoS attacks. Finally, Sect. 6 concludes this review by summarising key lessons.

2 Related article

Several pertinent survey articles examined the solutions for mitigating EDoS attacks within cloud computing environments. Therefore, this section delineates these articles, highlighting the methodologies utilised, focal points, and principal shortcomings. Table 1 presents a concise summary of these articles.

Singh and Rehman conducted a literature review on various aspects [12]. The article assessed the approaches, methodologies, scalability, and learnability of existing solutions by exploring EDoS attacks, their motivations, and defence techniques. Consequently, the article concluded that current mechanisms were inadequate for safeguarding cloud architecture from EDoS attacks, emphasising the necessity for more effective strategies. Conversely, limitations were observed, in which mitigation was only discussed while neglecting detection with protection strategies and performance metric analysis (obstructing a comprehensive evaluation of the techniques).

Chowdhury et al. explained the methods for mitigating EDoS attacks based on evaluation criteria and flexibility in cloud environments [13]. The article provided a classification of defensive strategies, critically assessing the drawbacks of each approach. Adequate cloud protection was then concluded to be significantly impacted by EDoS attacker behaviour understanding, strategy identification, and proactive defensive mechanism designs. The article also suggested that mathematical models, honeypots, or dynamic firewalls could improve response times, enhancing the existing solutions. Nevertheless, the article only discussed mitigation methods without including detection and protection techniques, limiting its comprehensiveness.

Singh et al. compared numerous advanced EDoS mitigation techniques within cloud environments [14]. The article highlighted the transition to cloud computing and its related security challenges concerning the increasing threat of EDoS attacks. These attacks (a subset of DDoS) considerably influenced the financial dynamics of cloud-based hosting services. The article also indicated that robust solutions were negatively affected by current mitigation methods, which were inaccurate, ineffective, or could exacerbate the situation. Like other articles, the article did not present prevention and detection strategies and only described mitigation approaches. Insufficient performance metric evaluation was also observed.

Table 1 Summary of EDoS detection, prevention and mitigation techniques identified from previous articles

Ref.	Year	Defence technique	Point of focus	Key result	Key limitation
[14]	2014	Mitigation	The effectiveness of the mitigation techniques, usability, and interoperability of the methods.	The current approaches were insufficient in effectively mitigating EDoS attacks. Therefore, a more thorough and robust strategy should be adopted.	Only mitigation techniques were covered, and prevention and detection techniques were excluded. Performance evaluation metrics were also not discussed.
[15]	2017	Mitigation	The evaluation metrics and adaptability of the models in the cloud environment.	Mathematical models could overcome the existing solution limitations. Thus, honeypots or dynamic firewalls could improve the response time.	Only mitigation techniques were covered, and prevention and detection techniques were excluded.
[16]	2019	Mitigation	The performance evaluation metrics mitigated numerous EDoS attacks within the cloud environment.	The existing methods to avoid EDoS attacks were inaccurate or ineffectual. Occasionally, specific scenarios could exacerbate the attacks.	Only mitigation techniques were covered, and prevention and detection techniques were excluded. Performance evaluation metrics were also not discussed.
[17] [18] [19] [20]	2019	Mitigation	The benefits of each solution and their corresponding limitations.	The existing solutions were insufficient and substandard. Therefore, an intelligent security mechanism could secure the cloud environment against EDoS attacks.	Only mitigation and prevention techniques were covered, and detection techniques were excluded. Performance evaluation metrics were also not discussed. Moreover, the solutions only demonstrated general DoS and DDoS solutions and omitted EDoS-specific solutions.
[21]	2014	Prevention	Detection and mitigation of DDoS and EDoS attacks in cloud environments	Proposed a secure framework integrating authentication, traffic analysis, and filtering mechanisms to reduce attack impact	Lacked real-time performance evaluation and scalability analysis
[22]	2015	Prevention	Identifying and mitigating EDoS attacks through pattern recognition and behavioral analysis	APART adaptively recognized EDoS attack patterns and improved response effectiveness	Limited empirical validation and tested only in controlled environments
[23]	2020	Detection	Signature-based and anomaly-based methods using machine learning for early identification of EDoS activity.	Anomaly-based models showed high accuracy in detecting low-rate EDoS attacks and reduced false positives.	High computational overhead; limited scalability to real-time environments.
[24]	2021	Prevention	Resource provisioning and request filtering policies to prevent the impact of EDoS on billing and service disruption.	Elastic resource allocation combined with threshold-based filtering helped prevent unnecessary resource consumption.	Threshold tuning is environment-dependent and may lead to false negatives.
[3]	2022	Detection & Prevention	Hybrid AI-enabled framework for both identifying and halting EDoS traffic using behavioral analytics.	Provided better adaptability and response to evolving attack patterns.	Requires constant model updates; training data scarcity can affect performance.

Nautiyal and Wadhwa published a review on mitigating EDoS attacks in cloud environments, reviewing the approaches, benefits, and limitations of each defence strategy [15]. The article validated the inadequacy of current solutions in effectively protecting against EDoS attacks. Even though more competent security mechanisms were proposed in the article, limitations were observed. The article only described prevention and mitigation strategies, neglecting detection techniques and performance evaluation metrics. Furthermore, the article only demonstrated general DoS and DDoS solutions and omitted EDoS-specific solutions.

This review tabulated a succinct overview of the methodologies adopted, principal findings, and significant limitations identified from previous articles (see Table 1). The outcome of the literature analysis suggested that no surveys or review articles on EDoS attacks were conducted in the past five years (2019–2024). This analysis indicated a clear gap in the current research on this topic.

3 Methodology

Specific existing literature on EDoS attack defence mechanisms was classified, identified, and examined in this review. Figure 1 below, shows the methodology employed to select relevant articles for this review study. The processes were accomplished by performing an in-depth analysis regarding the features and characteristics of the techniques, including the main issues and challenges encountered in safeguarding the cloud environment against this attack.

3.1 Research objectives

This article identified five primary research objectives as follows:

i. To establish a systematic methodology for identifying and selecting articles that discuss techniques for addressing EDoS attacks in cloud computing.
ii. To conduct a comprehensive review of related articles (surveys and review articles) that explore existing solutions against EDoS attacks in cloud computing.
iii. To analyse the correlation between DDoS and EDoS attacks by understanding their similarities, differences, and impacts on cloud computing.
iv. To identify and evaluate the primary strategies and critical proposed solutions for mitigating EDoS attacks in cloud computing, including an assessment of their benefits and drawbacks.

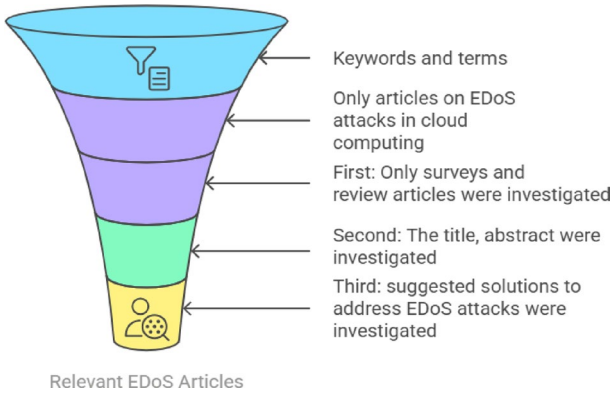

Fig. 1 Article selection process for EDoS attacks review

v. To summarise the main findings of this review and outline future research directions for addressing EDoS attacks in cloud computing.

3.2 Research questions

This review systematically addressed EDoS attacks within the cloud computing environment based on several research questions (RQs). The RQs are as follows:

i. RQ1: What systematic methodologies are applied for the identification, selection, and evaluation of.
ii. literature focused on EDoS attacks in cloud computing?
iii. RQ2: Which review and survey articles critically examine techniques for mitigating EDoS attacks, and how do they categorize or evaluate these solutions?
iv. RQ3: What are the technical and behavioral correlations between DDoS and EDoS attacks, including their points of convergence and divergence in terms of impact, intent, and mitigation?
v. RQ4: What are the most effective strategies and solutions proposed to mitigate EDoS attacks, and what are their comparative strengths and limitations?
vi. RQ5: What are the key gaps identified in current literature, and what future directions can guide research in EDoS attack prevention and resilience in cloud environments?

While previous literature has extensively focused on mitigation techniques for EDoS attacks, there has been a gradual emergence of research aimed at improving detection and prevention mechanisms. Detection strategies often leverage anomaly-based or machine learning techniques to identify deviations in traffic behavior, whereas prevention approaches focus on adaptive resource management and filtering policies. Table 1 has been extended to include a few such representative studies, highlighting recent efforts that contribute toward a more proactive stance against EDoS attacks. These studies show promising results, although challenges remain in terms of real-time performance and generalizability across cloud environments.

3.3 Selection process of relevant articles

Table 2 lists the keywords used to extract the appropriate articles. Table 3 lists the search strings utilised to extract articles for this review.

After filtering, several inclusion criteria (IC) and exclusion criteria (EC) were applied to the retrieved articles. The ICs and EC are as follows:

i. IC-1: The article targeted EDoS attacks and mitigation techniques.

Table 2 Summary of the keywords used to extract the relevant articles

EDoS	Economic attacks in cloud environments	Cloud service abuse	Cloud security challenges
EDoS attack detection	Cloud resource exhaustion	EDoS impact on cloud services	Smart pricing in cloud computing
EDoS mitigation strategies	DDoS vs. EDoS attacks	Resource consumption attacks	Billing attacks in cloud services
Cloud computing security	Cost-based attacks in cloud computing	EDoS protection frameworks	Cloud infrastructure resilience
EDoS defense mechanisms	EDoS attack prevention	Economic sustainability in cloud computing	Cybersecurity in cloud computing
EDoS	Economic attacks in cloud environments	Cloud service abuse	Cloud security challenges

Table 3 Summary of the search strings utilised to extract the relevant articles

Repository	Query	Number of articles
Springer Link	("EDoS") AND ("attach detection") AND ("mitigation techniques") AND ("defense mechanism") AND ("resource consumption attacks") AND ("Privacy Mitigation")	8,564
IEEE Xplore	("Economic attacks in cloud environments") OR ("Cloud Service Abuse") OR ("billing attacks in cloud services") OR ("Cloud infrastructural resilience") OR ("Cybersecurity in cloud computing")	1,725
Wiley	("Economic sustainability of cloud computing") OR ("EDoS attack Prevention") OR ("Cost based attacks in cloud computing") OR ("Cloud Resource exhaustion") OR ("EDoS impact on cloud services") OR ("Smart pricing in cloud computing")	9,860

ii. IC-2: The article was written in the English language.
iii. EC-1: The article was not written in the English language.
iv. EC-2: The article was published before 2015.

4 Correlation between DDoS and EDOS attacks

This section offers an in-depth analysis of various perspectives on the correlation between DDoS and EDoS attacks. The conditions and repercussions of EDoS attacks were also investigated comprehensively. This review then presented an informed perspective on their correlation based on the collected evidence, which is summarised as follows:

"*EDoS attacks constitute a subset of DDoS attacks, specifically aimed at cloud infrastructures. They exploit distinctive features of cloud computing, such as its scalability and pay-per-use payment models. By illicitly consuming cloud resources, EDoS attacks inflict considerable financial damage, as reflected in the increased billing costs incurred by cloud customers.*"

A DDoS attack is a coordinated cyber assault that uses multiple compromised computer systems (botnets or zombies) to target a single network system. This attack aims to inundate the victim with an overwhelming traffic volume while impeding access. On the contrary, EDoS attacks use botnets controlled by attackers against network infrastructures, exhibiting a more subtle and prolonged nature. These attacks inject spurious traffic over prolonged durations to incrementally deplete system resources.

The EDoS attacks increase operational costs for users leveraging the system or cloud services. This process incurs higher charges owing to the auto-scaling functionalities of the cloud, necessitating payment for augmented computational resources to accommodate the increased load. Figure 2 delineates that the operating domain of EDoS attacks is situated beneath that of DDoS attacks, surpassing the scope of regular traffic. This positioning renders EDoS attacks extremely challenging to detect using traditional DDoS defence mechanisms due to their relatively lower attack rates and intensity threshold values [15–18].

Multiple articles have argued that EDoS attacks represent a subtype of DDoS attacks [13, 17, 20]. Another group of articles have articulated that DDoS attacks evolve into EDoS attacks if the process occurs within cloud computing environments. This transformation is attributed to the inherent features of cloud computing, such as auto-scaling and pay-per-use billing models [14, 16, 20–24]. Certain articles have also contended that EDoS attacks are an evolved DDoS attack. Table 4 summarises the most pivotal perspectives on the correlation between EDoS and DDoS attacks, outlining the academic

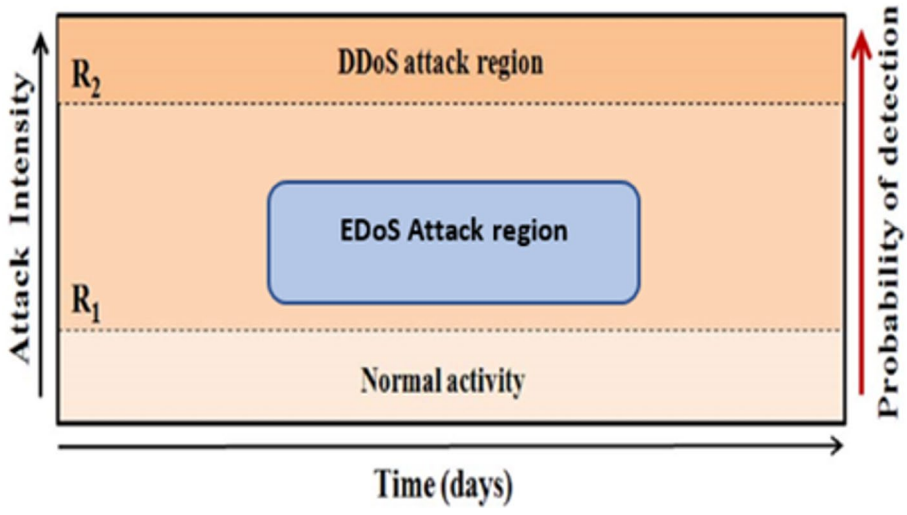

Fig. 2 DDoS and EDoS Attacks Active Regions

Table 4 Summary of the correlation between DDoS and EDoS attacks

Ref.	The correlation between DDoS and EDoS
[13, 17, 20]	EDoS as a type of DDoS
[14, 16, 20–23]	DDoS attacks are transformed into EDoS attacks in cloud environments
[24–28]	EDoS attacks are a new version of DDoS attacks
[15]	EDoS attacks are DDoS attacks with different impacts
[29]	EDoS attacks are similar to DDoS attacks
[30]	EDoS attacks are similar to low-rate DDoS attacks
[31]	EDoS attacks are a new version of general DoS attacks

discussion on their relationship. Furthermore differences and similarities between EDoS and DDoS attacks were summarized as in Table 5 below.

5 Existing solution to address EDoS attacks

Formulating robust defence strategies is essential to preserve the integrity and long-term viability of cloud computing environments against EDoS attacks. Despite extensive study in the larger field, insufficient investigations dedicated to countering EDoS threats have been observed. Hence, this section emphasises the most significant articles within this specialised field.

The scholarly discussion on EDoS attacks has been relatively limited, with only a few articles addressing this challenge. These articles have employed diverse techniques to alleviate the detrimental impact of the adverse effects of EDoS on cloud platforms. Thus, this review detailed these articles, focusing on their methodological frameworks, evaluation criteria, and inherent constraints.

The existing solutions to combat EDoS attacks were categorised into four main classifications: (i) Detection based strategies (ii) Prevention based strategies (iii) Mitigation based strategies (iv) Hybrid strategies.

5.1 Detection based strategies

Detection-based strategies focus on identifying the presence of EDoS attacks through anomaly detection, machine learning, deep learning, time-series analysis, or attribution

Table 5 Presents the comparison of EDoS and DDoS in terms of range of aspects

Aspect	DDoS	EDoS
Primary Objective	Disrupt service availability by overwhelming resources	Drain financial and computational resources over time
Attack Mechanism	Flood target with high-volume traffic from multiple sources	Trigger legitimate-looking requests to exploit pay-per-use and auto-scaling
Visibility	Highly visible due to sudden traffic spikes and service disruption	Often stealthy, mimics normal usage patterns, harder to detect
Attack Duration	Short to medium-term (minutes to hours)	Long-term (hours to days or more)
Impact on Victim	Service outage, reputation damage, potential SLA violations	Increased operational costs, potential financial exhaustion
Typical Target	Web servers, APIs, and network services	Cloud-hosted applications with auto-scaling features
Use of Botnets	Commonly used	Can be used, but may also involve legitimate-looking clients
Resource Consumption Pattern	Abrupt and massive	Gradual and sustained
Cloud Relevance	General threat to all online services	Cloud-specific threat due to billing and elasticity features
Detection Difficulty	Easier due to abnormal traffic volume	Harder due to normal-appearing requests
Mitigation Focus	Rate-limiting, blacklisting, traffic scrubbing	Cost-aware scaling, anomaly detection, economic modeling
Primary Concern	Availability	Sustainability

techniques. These approaches rely on recognizing deviations from typical usage patterns, detecting suspicious behaviors, or profiling attackers using statistical models.

Subsections:

5.1.1 Graphic turing tests (GTT) and crypto-puzzles

The Graphical Turing Test is a variation of the original Turing Test, designed to evaluate a computer system's ability to generate or interpret images in a way indistinguishable from a human. In this test, a human evaluator interacts with both a machine and a human through visual content (e.g., drawings, animations, or rendered scenes). The goal is to determine whether the machine's graphical responses can convincingly mimic human creativity or perception. It is often used in areas like computer graphics, game design, and AI-generated art.This section examines GTT and crypto-puzzle models for improving user defence in cloud settings against EDoS attacks. The objective of these models is to distinguish benign users from attackers. This section also systematically and extensively details several articles to address each individually (see Table 6).

5.1.1.1 EDoS-shield Sqalli et al. identified the EDoS-Shield as a model that prevented cloud environments from EDoS attacks by distinguishing real users from botnet users [1]. This model employed two crucial components: (i) virtual firewall (VF) and (ii) verifier nodes (VNs). A VF maintained whitelists and blacklists, while VNs executed the GTT to distinguish human users from botnets. These whitelists and blacklists of the firewall were then further modified based on the users' responses to the GTT [1]. Figure 3 portrays the working process of this method. The GTT functionality of this model enabled the straightforward identification of legitimate users while obscuring their attackers. This model exhibited a concentrated power of the cloud resources, increasing the security level of the available cloud.

Table 6 Summary of GTT and crypto-puzzle-based models for EDoS attacks

Ref.	Attack type	Approach	Description	Limitations
[32]	sPoW	Crypto-puzzle	The fundamental concept for distinguishing between a legitimate user and an attacker involved employing a crypto-puzzle with a *k*-bit difficulty level. This model could be adjusted based on user responses. Subsequently, the server established a secure channel with an encryption key for data exchange with the user.	• The model was not evaluated using any metrics. • Asymmetric power consumption resulted from the server generating puzzles and consumers solving them. • Attackers could easily overcome the puzzle test through a puzzle accumulation attack. • The model was susceptible to a high false positive rate due to the difficulty or complexity of puzzles, mainly affecting older and disabled users. • No evaluation metrics were performed.
[16]	EDoS-Shield	GTT	User classification involved two primary components: (i) VF and (ii) VNs. The VF possessed whitelists and blacklists, while VNs differentiated between illegitimate users and attackers based on GTT. Subsequently, the whitelists were updated accordingly based on user responses.	• The IP spoofing was not considered. • The GTT was challenging for certain legitimate users. • False positives could result from NAT-IP blocking numerous valid IP addresses. This outcome was attributed to a single-attack IP belonging to the same NAT. • False negatives caused by whitelist IPs could alter their behaviour to compromise the system.
[17]	Enhanced EDoS-Shield	GTT and TTL	The X was enhanced by incorporating the TTL value alongside the IP addresses in the whitelists and blacklists to mitigate the issue of spoofed IP affecting the EDoS-Shield. A counter of unmatched TTL value was also added.	• Given that attackers could utilise several tools to change the TTL value, it was not reliable. • This model demonstrated increased overhead on the VF due to the TTL value recording process. • Similar limitations as EDoS-Shield (except for spoofing IP addresses)
[18]	In-Cloud Scrubber	Crypto-puzzle	The In-Cloud Scrubber was a separate service responsible for generating and verifying a crypto-puzzle. This model presented two working modes: (i) normal and (ii) suspected. The modes were determined based on a predefined bandwidth and resource threshold as follows: • Normal: Both server resources and bandwidth were below the thresholds. • Suspected: Both server resources and bandwidth were above the thresholds (complex puzzle). Resources were excessive, while bandwidth was insufficient (moderate puzzle).	• This model demonstrated end-to-end latency due to the crypto-puzzle. • No evaluation was conducted on response time, CPU use, or false rate. This observation indicated that the reliability and effectiveness of the model was still questionable.

Table 6 (continued)

Ref.	Attack type	Approach	Description	Limitations
[19]	In-Cloud Service	Crypto-puzzle	The article elucidated through experimentation how a DDoS attack could be transformed into an EDoS attack in a cloud environment. Subsequently, a two-component model was proposed to detect and mitigate EDoS attacks: (i) firewall with whitelists and blacklists, and (ii) puzzle server.	• The IP spoofing was not considered. • Higher false rate due to the difficulty of the puzzle server, even for legitimate users. • Attackers could still overcome the puzzle by using puzzle accumulation tools.
[33]	Enhanced DDoS-MS	GTT and crypto-puzzle	The proposed model examined two packets of the source request using two test types. The GTT evaluated the first packet. In contrast, the second randomly selected packet was assessed by crypto-puzzle. This model also included a firewall with two main lists (white and black). Each main list possessed two sub-lists (temporary and permanent).	• This model was time-consuming and required more resources due to GTT and crypto-puzzle. • Even though the model was proposed for detecting DDoS and EDoS attacks, the detection mechanism for EDoS attacks was not explicitly articulated.
[34]	eDDoS-MS	GTT and TTL	This model involved the traditional GTT and TTL methods to control the user access and indefinite traversal of packets, respectively.	• This model demonstrated end-to-end latency due to GTT. • This model demonstrated false rates due to GTT and TTL. • The TTL value could not be trusted.
[26]	EDoS-7	GTT	This behaviour analysis model demonstrates two main components as follows: • SED: The edge device functioned as a firewall, forwarding the incoming request based on the flow table. • GSC: A virtual machine functioning as a VN, which was responsible for classifying the incoming requests into legitimate and attacker by generating and verifying GTT. The SED device was then updated.	• The model employed an identical methodology as EDoS-Shield reported by Sqalli et al., with distinctions solely in the nomenclature of the components utilised [16]. Compared to EDoS-Shield, this model was designed to operate within an SDN-based cloud environment. • This model demonstrated higher response time due to GTT.

Notes: sPoW = Self-Proof of Work; TTL = Time-to-live; VF = Virtual firewall; VNs = Verifier nodes

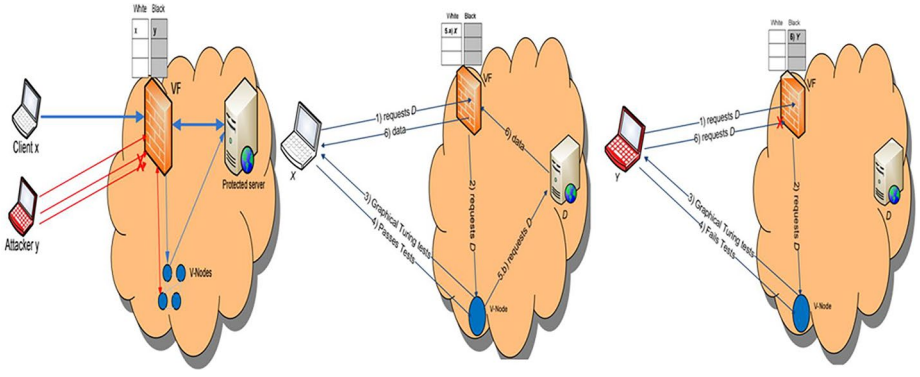

Fig. 3 EDoS sheild working process.source:[1]

Another advantage of this model was that user blocking and unblocking with a VF were conducted in real-time. This procedure ensured the cloud environment remained intact and genuine users were granted prompt access. Conversely, detected attackers were expelled. The structured organisation of this model (VF and VNs) could also accommodate the constrained space requirements of cloud computing, rendering it beneficial for low and high-volume applications. Nonetheless, this step could impede legitimate users' access to cloud services due to the necessity of extra layers, adversely affecting user experience.

Higher processing power was observed when VF and VNs were integrated while serving the GTT processing-based computing demands. This process could negatively interfere with the effectiveness of the cloud ecosystem. Nevertheless, a more significant threat existed from advanced attackers who acquired GTT evasion techniques. This threat required improving the test sophistication and periodic upgrades to maintain relevance. The volatile usage of the whitelists and blacklists through the VF also necessitated revisions to the documentation to represent the user and adversary behaviour accurately. This procedure could jeopardise the reliance on obsolete or incorrect listings. Additionally, legitimate users could be partly or wholly discouraged from employing the cloud service if a certain minimal period was established for using the system after passing the legal GTT. Each time these users were required to undergo GTT, the step could negatively impact the user experience.

5.1.2 sPoW

Khor and Nakao reported the Self-Proof of Work (sPoW) model as a model to safeguard cloud systems against the increasing threat of EDoS attacks [32]. Considering these attacks were frequently executed using botnets, the differences between legitimate user requests and those from botnet attackers could be differentiated through this model. The fundamental concept of this model involved utilising a crypto-puzzle with a variable *k*-bit difficulty level based on the user's response. This process established a secure channel with an encryption key, distinguishing between a legitimate user and an attacker [33]. Therefore, the defence of cloud infrastructures from EDoS attacks was improved using an efficient and highly versatile sPoW model. This model could distinguish between genuine and fraudulent traffic, provide an adjustable difficulty level, and establish a secure channel, rendering it highly effective in reducing the effect of these attacks.

The model revealed several limitations affecting the effectiveness of this solution. Initially, no metrics were developed to evaluate the performance of the model. Secondly, the techniques for generating and resolving puzzles could induce a vertical load imbalance. This review also acknowledged the potential for attackers to overcome the puzzle test using a puzzle accumulation assault. Lastly, high false positive rates were observed in this model due to certain legitimate users' inability to solve straightforward puzzles. Despite this model representing essential progress for cloud security, self-evaluation should still be considered when addressing shortcomings.

5.1.3 Time-To-Live (TTL) and header inspection

In this section two studies are investigated.

5.1.3.1 Enhanced EDoS-shield Al-Haidari et al. proposed the enhanced EDoS-Shield as an improvement model over the conventional EDoS-Shield system. This model addressed the challenges associated with the threats posed by spoof IP that compromised the effectiveness of the previous design [17]. The improved version incorporated time-to-live (TTL) value in whitelists or blacklists while incorporating additional criteria for distinguishing between legitimate users and potential threats using the IP address. Notably, the TTL value facilitated traffic filtration and security against EDoS attacks [18].

Another structural modification involved installing a counter to monitor deviations in TTL values, which could assist in mitigating system abuse (spoofing) [18]. Nonetheless, numerous issues were observed with this model. The recognition and application of this unique aspect (TTL values) constituted an additional issue to the system regarding the duration required to process and authenticate each request. This constraint could increase access latencies to the cloud services for legitimate users due to prolonged average wait periods, aggravating user experience. A more complicated management and system maintenance was also exhibited when the system complexity increased. This outcome indicated that exploring possibilities from the first question was a formidable challenge and could strain resources.

The TTL-based approach necessitated the unrealistic assumption of certain scenarios. One scenario example involved an attacker generating packets with fabricated TTL fields containing values known to the attacker and anticipated during validation [35]. Conversely, this assumption failed against more skilled and sophisticated adversaries. Even though the model aimed to outperform the conventional EDoS-Shield, improvement measures could still be conducted to improve the usability of the solution. Specifically, the whitelist and blacklist verification should not hinder legitimate users by being overly complex or necessitating constant oversight and modification to remain relevant.

5.1.3.2 EDoS source inspection Zekri et al. verified the legitimacy of the requests by introducing an EDoS attack countermeasure based on source inspection, counting, and Turing tests [51]. This model was aided with a test node and VF for request classification and filtration. The article also assured the performance quality (power consumption and response time metrics), ensuring that legitimate requests remained safe from attack traffic. Moreover, verification tests and a high-end firewall were employed to filter out malicious requests, preserving the integrity and performance of the cloud service (without substantial impacts on computing resources) [35–37]. Nonetheless, this service quality-based model required frequent updates in its threshold and list-based approaches to recognise and block new and evolving attack vectors.

5.1.4 Machine learning and deep learning models

This section reviews 11 articles containing models for defending cloud environments from EDoS attacks using machine learning.

Table 7 tabulates the information obtained from these articles for securing cloud services.

5.1.4.1 EDoS detection using Support Vector Machine (SVM) and Neural Network (NN) Abbasi et al. highlighted support vector machine (SVM) and neural network (NN) in a detection model for EDoS threats [20]. The proposed model analysed VM behaviour

Table 7 Summary of machine learning/deep learning-based models for EDoS attacks

Author	Approach	Description	Limitation
[20]	Machine learning	A model for detecting various EDoS attacks using SVM and NN. This model only allowed normal-situation VMs to be scaled up.	• This model required an adequate number of packets for learning. • The detection accuracy decreased when HTTP and DB attacks occurred concurrently. • This model ignored critical evaluation metrics (cost and complexity).
[15]	Machine learning	This model employed ANN and GA algorithms to identify the affected EDoS server by detecting the affected path and determining the affected node (server).	• The article neglected to clarify their proposed strategy effectively.
[35]	Deep learning	The model involved dynamic EDoS detection in SDN and multivariate time series anomaly identification. This model could predict the user resource usage (CPU load and memory use) value based on an unsupervised deep learning algorithm (LSTM).	• This model possessed a false rate. • A higher processing duration WAS needed to detect attacks due to the long sequence length (250). • This model required more resource allocation to the defence system due to the long sequence length. • The performance evaluation revealed low accuracy.
[36]	Deep learning	The MAD-GAN model was used as a multivariate time series anomaly detector. This model could predict the future value of the resource usage by learning from previous and live data based on GAN and deep learning algorithms (LSTM and RNN).	• This model needed more processing time. • This model necessitated more resources for the defence system. • The false alarm rate was relatively high. • This model demonstrated a low detection rate.
[37]	Deep learning	The R-EDoS model consisted of four primary components: (i) GRU (a variant of RNN), (ii) VAE algorithm, (iii) linear Gaussian state-space (connection technique), and (iv) planar-flow (normalising flow technique).	• This model revealed slight difficulties in detecting Yo-Yo and Slowloris attacks.
[38]	Deep learning and DWT	The P-estimation model initially evaluated the attack intensity and selected an appropriate LSTM model based on that evaluation. A suitable (LSTM) model was then selected based on that estimate. The DWT was also applied as a filter to effectively preprocess the data before feeding it to the classification model.	• This model required higher resources due to the long sequence length of the classification algorithm.
[39]	Artificial intelligence (Multihead attention and Machine learning)	This MAN-EDoS model was a multihead attention model containing a multivariate time series to detect EDoS attacks in the network traffic environment. The multihead attention score matrices employed with NN were trained using EDoS data by computing the query, key, and value matrices.	
[27]	Deep learning	Given that LSTM models were ineffective due to the long sequence length, this model only utilised (5) as a sequence length. This model was activated after the ANN model identified the precise moment of an assault, which LSTM was not utilised across all flow periods.	• Model complexity • The period detector (ANN) possessed limited accuracy. • The accuracy could be improved using more advanced algorithms.
[40]	Machine learning	A rapid auto-scale-based detection model, EDoS-BARRICADE was developed using linear SVM and non-linear Kernel-based SVM as classifiers. This model could separate the attacked VM from others.	

Table 7 (continued)

Author	Approach	Description	Limitation
[41]	Flow-based attention and machine learning	A flow-based model was used to create an EDoS detection and prevention system. Furthermore, X by using the attention technique to compute the focused scores in the flow features in the SDN environment. This model was based on a NN.	
[42]	Machine learning	This model detected four EDoS attack types based on DBN and SVM: (i) HTTP flooding, (ii) DB, (iii) TCP SYN, and (iv) UDP flooding. The model consisted of three stages: (i) gathering data packets, (ii) extracting attack features, and (iii) establishing correlations between them. This self-adaptive model could adjusted for better feature extraction and improved performance.	• The three stages involved significant manual work, causing more errors and needing more expertise.

Notes: DWT = Discrete wavelet transformation; NN = Neural network; ANN = Artificial neural network; GA = Genetic algorithm; MAN = Multihead attention network detection; GAN = Generative Adversarial Networks; R-EDoS = Robust EDoS; GRU = Gated recurrent unit; RNN = Recurrent Neural Network; VAE = Variational autoencoder; SVM = Support vector machine; DBN = Deep belief network; TCP SYN = Transmission Control Protocol Synchronize; UDP = User Datagram Protocol

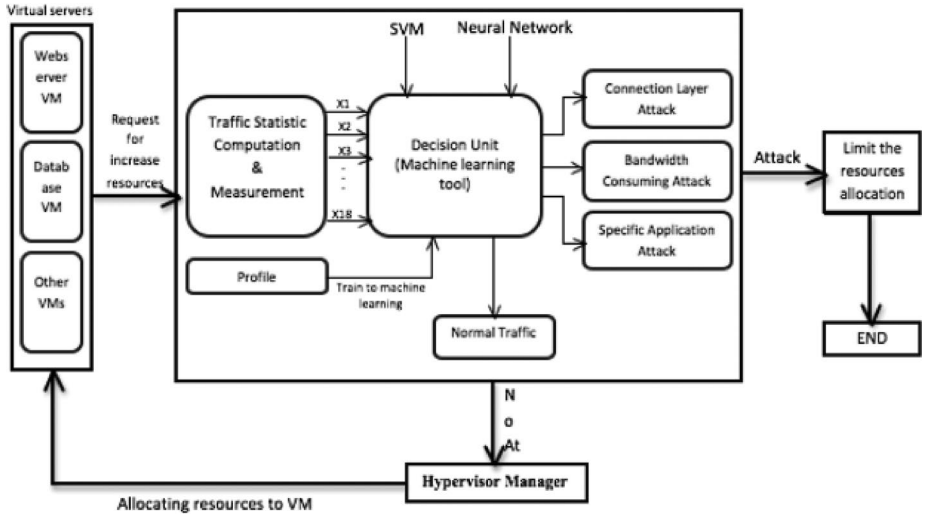

Fig. 4 The architecture of the proposed model by Abbasi et. al. Source: [21]

based on traffic patterns and resource usage to identify HTTP requests, DB, and TCP SYN flood attacks. The proposed model categorises incoming requests by comparing them to the.

behaviour profile of a VM. This process permitted only those from VMs in normal mode to be processed [20]. The model also proficiently utilised machine learning techniques (SVM and NN) for EDoS threat detection across multiple attack types. Even though this process offered a nuanced approach to VM behaviour analysis, lower accuracy was observed under concurrent HTTP and DB attacks. These outcomes highlighted a limitation in addressing complex, multi-vector threats. Figure 4 below, shows the architecture of the SVM-NN proposed mode.

5.1.4.2 EDoS detection using Artificial Neural Network (ANN) and Genetic Algorithm (GA) Nautiyal and Wadhwa computed an artificial neural network (ANN)-based reactive model for EDoS attack detection in cloud environments [15]. The ANN initially analysed network pathways for indicators of economic exploitation. Subsequently, a genetic algorithm (GA) was applied to reduce false positives. Another ANN was then used to

identify a compromised server precisely along the detected attack vector to achieve enhanced detection accuracy [15]. Consequently, this solution demonstrated high-precision detection and mitigation based on advanced deep learning (ANN) and optimisation approaches (GA), emphasising economic implications and minimising false alarms. Nevertheless, the sequential execution required for ANN and GA processing could render this model a latency-prone approach, affecting real-time detection [47]. A significant correlation was also denoted between the effectiveness of the model and network route complexity or attack patterns.

5.1.4.3 Dynamic EDoS detection for SDN-based cloud Dinh and Park addressed multivariate time series anomaly detection for EDoS attack mitigation through a dynamic error thresholding model instead of static thresholding [40]. The proposed model employed an LSTM for unsupervised deep learning to predict user resource consumption (CPU load, memory usage, and TCP connections) based on historical data and live monitoring (SDN controller). This model also presented resource consumption forecasting using anomaly detection with dynamic threshold setting and deep learning [48]. Consequently, the model offered a customised strategy informed by previous data and realtime input. Substantial computational resources and processing time were also necessary in this model, such as an LSTM with an extremely long sequence of 250 packets. Conversely, this model yielded weak results during performance evaluation, raising doubt about its reliability.

5.1.4.4 Robust EDoS (R-EDoS) Dinh and Park proposed a multivariate time series anomaly detection model for the SDN-based cloud networks using RNN called robust EDoS (R-EDoS) [40]. This model was combined with variational autoencoder (VAE), linear Gaussian state-space models, and planar-flow techniques. The model detected anomalies by learning the standard data patterns. Simultaneously, input data was reconstructed while the self-adjusted threshold further reduced the error. This model also solved the vanishing gradient problem, facilitated the identification of intricate patterns of temporal dependence through a gated recurrent unit (GRU), possessed an extensive model structure for detailed anomaly detection, enabled error rate reduction, and maintained interpretability through reconstruction probabilities. Nonetheless, the model could not accurately detect certain attack types (Yo-Yo and Slowloris), indicating a need for enhancement to accommodate a broader range of attack vectors [49].

5.1.4.5 P-Estimation Agarwal et al. pioneered the P-estimation detection model that integrated deep learning with discrete wavelet transformation (DWT) to recognise fraudulent resource consumption (FRC) attacks on cloud services [42]. The model began by assessing the attack intensity to select a suitable LSTM model, employing DWT for data preprocessing and further classification based on web server logs. This model could also be continuously retrained to update the popularity of the web pages. Consequently, the combined novelty of DWT preprocessing and LSTM modelling enabled this strategy to assess attack strength subtly and adaptively respond to the fluctuating characteristics of online traffic, improving detection accuracy. Nonetheless, this model was complex due to several LSTMs for different attack intensities. This feature suggested longer sequence lengths, resulting in slower responses.

5.1.4.6 Multihead attention network detection (MAN)-EDoS Ta and Park enhanced flow-based EDoS attack detection in SDNs utilising the flow-attention model [29]. A multihead attention network detection (MAN) model and multivariate time series analysis were observed in this model to facilitate the processing of flow features from the SDN controller to identify EDoS early. This model also collected the query, key, and value matrices to calculate the attention score for facilitating the identification of EDoS attacks by an NN trained on EDoS-specific data. Meanwhile, attention mechanisms in flow-based contexts transcended the limitations of RNNs, providing runtime adaptability to network flow fluctuations and enhancing detection accuracy. This model also incorporated computational demands to construct and utilise the query, key, and value matrix for calculating the attention score, affecting the efficiency and scalability of the model.

5.1.4.7 Two-phase DL-based EDoS detection Nhu and Park utilised ANN and LSTM as a two-phase deep learning detector for EDoS attacks [27]. This model overcame the disadvantages of traditional LSTM anomaly detection algorithms containing a long sequence and high computational costs. Initially, an ANN was used to identify the attack periods. After this identification, the LSTM could concentrate exclusively on these periods to drastically reduce the sequence length to five packets. Considering that the process narrowed efficiently to specific periods from ANN, computational resources and detection time could be reduced. This outcome rendered the model more efficient than the typical LSTM model. Conversely, the low accuracy of the initial detection phase period of ANN significantly impacted the model. This process undermined the overall effectiveness of the detection strategy [50]. Significantly enhanced procedures could also yield superior performance.

5.1.4.8 EDoS-bARRICADE Jones and Kumar extensively employed auto-scaling techniques with SVM classifiers [45]. The proposed EDoS detection and mitigation approaches emphasised distinguishing between attacked and normal VMs by tracking VM counts over time. An alert was then generated if VM numbers consistently rose over three consecutive periods, prompting the application of the BARRICADE algorithm for flow segregation and subsequent SVM classification into normal or attacked categories. Therefore, the auto-scaling-based detection approach was combined with an additional classification mechanism and SVM in a step-wise manner. This process aided in classifying the EDoS attacks and enhanced cloud protection [51]. Nonetheless, the proposed model could result in abrupt increases in reaction time due to its dependence on the primary trigger associated with the escalation of consecutive VM counts.

5.1.4.9 Flow-based EDoS detection Ta and Park critiqued various current EDoS detection methods [45]. Previous articles demonstrated limited attack traffic features owing to the removal of network infrastructure effects. Thus, the article created an SDN and cloud-optimised EDoS detection model based on attention techniques [52]. This model computed the flow network attention scores from the SDN controller, improving the overall adaptability and accuracy in identifying EDoS flows. The model also integrated the attention mechanism with SDN to monitor alterations in flow characteristics, facilitating real-world application dynamically. Consequently, the proposed model was more accu-

rate and presented higher response times. Nevertheless, the time processing of attention scores necessitated increased computational complexity and demands.

5.1.4.10 Deep belief network (DBN)-SVM for EDoS and DDoS detection Dennis and Priya applied the most optimal deep learning algorithms in a security system to detect EDoS and DDoS threats in cloud computing [42]. The proposed model contained four primary components: (i) feature selection, (ii) deep belief networks (DBNs) for pattern recognition, (iii) SVMs to differentiate between normal and attack traffic, and (iv) hyperparameter optimisation for higher DBN-SVM model performance. Consequently, the article demonstrated high efficiency with a true positive rate (TPR) and true negative rate (TNR) of 99.8% and 99.9%, respectively. These exceptional TPR and TNR rates indicated the successful integration of advanced techniques to produce highly accurate detection of threats. Nonetheless, challenges could occur due to the complexity of implementing and maintaining the sophisticated model [52–54].

5.1.4.11 WEB-TRAP Wang et al. resolved EDoS attacks targeting web systems based on a dynamic defence model (WEB-TRAP) [11]. This model consisted of two major strategies: (i) changing the online resource address for moving target defence and (ii) real-time trap injection for intruder detection. Legitimate clients and costs were also protected by using an online controlling system management. Consequently, this model was an effective cost-reduction strategy for defenders in various attack scenarios by providing dynamically changing resource addresses while injecting traps. Nevertheless, this model could not distinguish between legitimate and malicious clients, affecting genuine users.

5.1.4.12 Data science techniques for EDoS detection Courtney et al. applied various data science techniques (statistical analysis, time series, ANN, and *k*-NN) as a model to detect FRC threats in the cloud [55]. This model acknowledged the constraints of each method when utilised in isolation and proposed an ensemble strategy containing Zipf's Law, Spearman, and overlap for better accuracy. NASA dataset under various attack scenarios was also employed to test the mode for lowering the false positives while enhancing low-intensity attack detection. Consequently, the model offered a more robust solution with lower false positives due to the improved FRC attack detection capabilities. Nonetheless, current cloud traffic was not accurately represented owing to the reliance on an outdated NASA dataset and limited data attributes, impacting the effectiveness of the model.

5.1.5 Attribution techniques

This section reviews two articles involving attribution-based models for protecting cloud environments. Table 8 presents the findings and summaries of these articles.

5.1.5.1 Attribution of FRC Idziorek et al. identified malicious web activity using an attribution methodology [43]. The model analysed weblogs for deviations in client behaviour, such as request volume, session metrics, and chi-square statistics. The client behaviour was also classified using a comparative analysis between the overall attribution and threshold scores [58]. This approach also detected anomalies by systematically examining the main factors of web browsing behaviour, offering a structured method for distinguish-

Table 8 Summary of attribution-based models for EDoS attacks

Author	Approach	Description	Limitation
[43]	Attribution methodology	Various information from weblogs could be extracted from the attribution methodology (statistical method), including request volume, web session, session length, and average session length. The fundamental concept was that the attacker could not predict regular client behaviour on a specific website. Thus, any divergence or deviation was classified as hostile traffic.	• Given that the attacker could mimic the legitimate user behaviour (request rate per session and session length, low accuracy was observed.
[44]	Attribution of EDoS (Markov and Semi-Markov models)	The proposed model was an anomaly-based detector containing two Markov models in parallel to detect EDoS attacks: (i) the Markov Chain Model and (ii) the Hidden Semi-Markov Model. This model relied on the resource usage footprint of users and the user profile patterns derived from weblogs.	• This model lacked evaluation metrics. • A high false rate was denoted.

ing between benign and malicious clients. Nonetheless, the efficiency of the model could be undermined if attackers replicated typical request patterns, potentially evading detection.

5.1.5.2 Attribution of EDoS attacks Karami et al. profiled authorised user resource consumption and detected FRC based on a model using two Markov-based models [44]. This technique modelled user behaviour and costs over time by segmenting resources by size and user request data, identifying malicious activity when the total request values surpassed a predefined threshold [59]. The model also effectively captured resource requests and consumption patterns, providing a sophisticated approach to distinguishing between legitimate and fraudulent activity based on the financial implications of requests. Conversely, the dependence on sustaining minimal frequencies of fraudulent requests to simulate authentic user behaviour could fail to identify sophisticated attackers skilfully integrated with normal activities.

5.2 Prevention-based strategies

These strategies are designed to proactively prevent EDoS attacks by limiting the system's exposure to suspicious traffic or by controlling resource access. The common techniques include static thresholds, access control policies, and resource filtering based on heuristics.

5.2.1 Static threshold models

This section critically reviews eight articles containing models for defending cloud environments from EDoS attacks [69]. These articles demonstrated static threshold-based models in which legitimate users could access the system while attackers and suspicious were blocked. Table 9 tabulates the information obtained from these articles.

5.2.2 Time Spent Profile (TSP)

Koduru et al. established that the time spent profile (TSP) model for detecting EDoS attacks relied on a user's duration on a web page [38]. This model leveraged the difference in the TSP behaviour for sophisticated bots and normal traffic, rendering the behaviour mimicking the process of normal users challenging for an attacker. The mean absolute deviation (MAD) was also employed to categorise normal and malicious traffic

Table 9 Summary of static threshold-based models for EDoS attacks

Author	Concept	Approach	Description	Limitation
[31]	Control virtual resource access to the cloud	Static thresholds for CPU usage, CRPS, UTF, and GTT	The architecture of this model contained VF, LB, DB, and VMInvestigator. Both VF and a blacklist table was used to filter the incoming requests. The user requested that the blacklist be submitted directly to the VMInvestigator for further processing. Meanwhile, the LB managed the equitable distribution of requests to designated virtual machines while monitoring and automatically scaling cloud resources. The VMInvestigator was also responsible for generating GTTs and checking user replies. Furthermore, this X included a UTF that evaluated the user's trustworthiness based on the correct answers to the GTTs within a specified period.	• This model revealed end-to-end latency (more response time). • The entire NAT-IP was blocked due to a single attacker IP. • False error rate owing to static thresholds.
[45]	CloudWatch	Static thresholds	The CloudWatch model monitored the AWS resource and the corresponding real-time applications. Each AWS consumer could establish limits for any service they rent from Amazon based on various factors (price, size, service type. This monitoring model [CloudWatch (alarms)] could send notifications or automatically modify the client's monitored resources based on the predefined rules.	• The cloud elasticity feature was limited. • Clients must recognise their requirements and services to establish limits and rules.
[22]	APART	Pattern recognition and static frequency threshold	The APART model applied pattern recognition through an anomaly-based technique. Moreover, the packet deliveries from various nodes from 400 Hz to 800 Hz presented the attack properties. This model detected severe attacks in the specified range.	• The complexity increased when more components (VF, VMInvestigator, VMscheduler) were added, • The authors failed to describe and explain their model clearly.
[46]	HRF	Static HTTP request threshold	This queuing-based model contained WAF and three lists for classifying users (white, block, and unknown). The model was based on a three-phase mechanism (RF, NA, and RA), in which the fundamental concept was a firewall (WAF), a validating node (S3 bucket) to assess whitelist and blacklist, and Lamada to check thresholds with period while updating the WAF lists. This model also utilised CloudWatch to check all activities and send notifications.	• A specified threshold presented difficulties in identifying the optimal value. • The model was presumed to function correctly if the traffic adhered to a Poisson distribution.
[47]	(EDoS-ADS)	Static thresholds for CPU use and duration timers	The EDoS-ADS model contained three main components based on CPU utilisation and duration: (i) LB, (ii) DB, and (iii) DS. The article proposed four threshold values and two duration timers: scale-up upper and lower values, scale-down upper and lower values, 80%, 75%, 35% and 30%, respectively. This scale up and down were 5 min and 1 min, respectively. Moreover, the model possessed four operation modes (Normal, Attack, Suspicion and Overcrowd) while using URL redirection and GTT.	• This model revealed higher overhead due to the blacklist table. • This model revealed a false positive rate owing to the static threshold. • False negative when an attacker could predict the system scalability limits.

Table 9 (continued)

Author	Concept	Approach	Description	Limitation
[23]	EDoS-ADS	Static thresholds for CPU use, duration timers, CRPS, VM number, and GTT for suspicious users.	A Yo-Yo attack demonstrated a limitation in the EDoS-ADS model proposed by Shawahna et al., wherein the attacker could mislead the EDoS-ADS mechanism without triggering attack mode [38]. The enhanced EDoS-ADS model was then introduced to overcome this constraint. Initially, a challenging process for an attacker to anticipate when to scale up or down was created by producing at least two scaling policies. The process could then be classified as suspicious, prompting the issuance of a GTT to the user if the user behaviour produced periodic high CRPS and periodic zeros. Finally, restricting the number of virtual machines accessible for scaling up was limited to mitigate financial losses.	• This model could only function against Yo-Yo attacks (EDoS variant). • Despite the suggested mechanism that could lower expenses, this model could eliminate the cloud scalability feature while affecting the cloud availability feature.
[24]	EDoS-IDM	Static thresholds for packet size and static time threshold	The ICMP detection and mitigation model was proposed in this statistical method to mitigate the impact of volumetric and typical behavioural ICMP traffic attacks on cloud environments utilising SDN. This EDOS-IDM model employed an *n*-time method, whereas the detected ICMP traffic was only allowed for *n* times if the ICMP packet size was less than or equal to 64 bytes. A cloud gateway for *d*-time also stopped the ICMP traffic after *n*-time.	• Like other statistical solutions, this model necessitated predefining several hard thresholds (packet size and duration). Thus, setting up the threshold value was challenging. • This model revealed a false rate due to hard thresholds. • The model could only detect and mitigate one specific EDoS attack (ICMP attack), neglecting other EDoS attack types.
[48]		TSP as a criterion for EDoS detection	Even though the bot machines were well-developed, the created TSPs differed from the average TSPs observed for real traffic. Therefore, attackers could not predict the average TSP of a page on the target website and devise a strategy accordingly. The TSP of the attacker's queries also differed dramatically from the interpretation of regular requests. Consequently, the MAD of the TSP served as a critical criterion in discriminating between legitimate and illegal traffic. This model also generated TSP and MAD charts, which the administrator could monitor.	• A significant restriction involving human interaction for monitoring and analysing plots was required, which was unfeasible. • The model was more pertinent to e-commerce than EDoS detection.

Notes: APART = Adaptive pattern attack recognition technique; HRF = HTTP request filtering; CRPS = Concurrent request per second; UTF = User trust factor: TSP = Time spent profile; AWS = Amazon web services; WAF = Web application firewall; RF = Request filter; NA = Network analyser; RA = Request analyser. DS = Defence shell; MAD = Mean absolute deviation

based on the extent of TSP deviation from the standard conditions. This model possessed significant promise, given that TSP innovatively provided user interaction patterns. Several pattern examples included time spent on a page for EDoS attack detection and unique model creation that an attacker might find challenging to anticipate and counteract.

The model demonstrated several limitations, and its effectiveness could be diminished by legitimate user behaviour deviations in producing false positives. Other vital attack vectors also frequently get obscured due to the emphasis on time-based metrics. Nonetheless, this model necessitated manual intervention for checking and analysing TSP and MAD charts, making it impractical for real-world applications.

5.3 Migration- based strategies

Mitigation techniques aim to minimize the impact of EDoS attacks once they are detected. These include using firewalls, blacklists, puzzle servers, and scrubbing services to block or deflect attack traffic, ensuring continuity of service.

5.3.1 In-cloud scrubbers and puzzle servers

The most interested studies which utilized In-Cloud Scrubbers and Puzzle Servers are investigated in this section.

5.3.1.1 In-cloud scrubber A model as an additional service called In-Cloud Scrubber was proposed by Kumar et al. [18]. The model prevented EDoS attacks by creating and validating cryptographic puzzles to verify users' legitimacy. This process was performed in two modes (normal and suspected) based on the bandwidth levels and resources. The normal mode involved server resources and bandwidth within the established thresholds. On the contrary, these thresholds were exceeded in the suspected mode. The suspected mode increased or decreased the difficulty of the puzzles with high or low resources, bandwidth, and usage, respectively [19].

The model possessed a notable feature. This model supported normal and suspected modes, determining the current bandwidth and server resources. Thus, this flexibility enabled the maintenance of a normal operating mode, wherein users were presented with low-difficulty puzzles. Higher puzzle difficulty was also displayed when a threat to the system resource occurred or the bandwidth exceeded a specific limit. This escalation functioned as an effective deterrence against attacks by complicating access to the service. Additional resources were also allocated to the service, discouraging prospective assaults. Moreover, different puzzle characteristics (influenced by appeal level to the user and resource with bandwidth availability) enhanced the capacity of the model to address threats while safeguarding the cloud resources.

The model presented several constraints. Lower legitimate user experience was observed when gradually introducing and varying cryptographic puzzles. One notable example was the model erroneously assumed that users seeking access to cloud computing services had malicious intent during high-demand periods. This procedure could lead the users into the suspecting mode, serving real cloud consumers with unnecessary weak-end challenges. Consequently, cloud services became unappealing due to time constraints.

The effectiveness of this model was contingent upon the accurately established normal and perceived thresholds. Given that these thresholds were influenced by the changing level of valid service consumption, establishing excessively high or low thresholds could result in dissatisfied users who have previously resisted superficial scrutiny. Otherwise, the model could neglect actual EDoS attacks. Puzzle solutions also did not preclude the scenario in which complicated forensic teams establish regulatory mechanisms

to automate undistorted puzzle comprehension (nullifying the benefit). Likewise, the model necessitated unanticipated expenditures due to the requirement for additional treasuries and more complex cloud infrastructure.

5.3.1.2 In-cloud service VivinSandar and Shenai created a two-pronged model to detect and mitigate EDoS attacks by introducing a combination of a firewall containing whitelists with blacklists and a puzzle server [19]. The puzzle server was employed to replace the VNs. Overall, this process represented an optimal implementation in which the system was most effective against EDoS attacks. The two-pronged model was constructed based on two stages. Initially, a firewall containing whitelists and blacklists filtered the incoming traffic, preventing the registration of inactive users or facilitating this process through early filtering. This additional security measure further enhanced the effectiveness of the puzzle server. Considering that this stage managed unverified traffic by overwhelming it with diverse computational obstacles, an active defence against possible intruders was presented.

This set of measures enhanced the security framework while adjusting puzzle complexity based on the assessed threat level, providing a robust solution against various intensities of DDoS attacks [20]. Nonetheless, the proposed model was still not optimal. An additional puzzle server created a potential vulnerability in the system, which attackers could penetrate or flood with complex queries. This puzzle server was also dependent on the puzzles. If the puzzle difficulty was not correctly calibrated, the puzzle server failed to function for the defined period. Thus, a sufficiently simplistic possibility could occur, allowing hackers to breach them and obstruct traffic, hindering legitimate users' access.

An updated whitelist and blacklist were challenging, necessitating an accurate depiction of threats and actual traffic patterns. This precise depiction could stabilise the implementation of administrative measures and advance highly sophisticated monitoring systems. Consequently, this two-pronged model required additional computational resources and capital, potentially leading to increased costs for maintaining cloud services. Despite this two-pronged model offering a promising strategy for mitigating EDoS attacks, careful implementation and ongoing management were still needed to balance security needs against usability and cost factors.

5.3.1.3 DDoS-MS The DDoS-MS model prevents DDoS and EDoS attacks through a defence system by inspecting the incoming packets in a two-phase process. A GTT initially assesses the first packet, after which a random packet is selected to obtain a crypto-puzzle for the subsequent evaluation. Firewall partitioning has also recently been implemented with temporary and permanent whitelists and blacklists. These firewalls generally filter and block unauthorised traffic [34]. Additionally, the primary function of the model lies in its multi-layered structure and specific selections during packet analysis. This analysis involves deconstructing a user's first packet using GTT. Concurrently, another packet is randomly chosen for evaluation using the crypto-puzzle. These observations suggest a lower user's capacity for anonymity, facilitating the comparison of genuine users and the passive attack potentials based on GTT and crypto-puzzle [60].

A firewall containing whitelists with blacklists and their corresponding categorisation (temporary and permanent classifications) can facilitate more advanced and dynamic

access control. This comprehensive model enables the system to adapt to various attack methods and intensities, ensuring robust protection against DDoS and EDoS attacks by minimising false positives and negatives through user validation. Conversely, several limitations are observed within the model. The benefits of authenticity derived from this procedure are enduring and more expansive than the gateways traversed by the user. Even though the model is secure, extra latency can be introduced into the system due to the complexity of the two-packet assessment process. Therefore, user experience can weaken when the model filters out innocent users.

The difficulty of GTT and crypto-puzzle must be balanced to avoid any impact on the influx of authentic traffic, which can be challenging to achieve precisely. Meanwhile, the blacklist and whitelist management (temporary and permanent subdivisions) should be constantly updated and maintained to reflect the current threat landscape and user behaviours appropriately. This duty can prove challenging for IT administrators, who must possess the appropriate tools for accurate network tracking. A comprehensive protection model can then necessitate significant resource allocation and infrastructure deployment, affecting the overall cost-effectiveness of the solution.

5.3.1.4 Enhanced DDoS-MS Alosaimi et al. produced the enhanced DDoS-MS model containing several components, including a firewall, VN, puzzle server, intrusion prevention system (IPS), and reverse proxy (RP) [34]. A firewall was utilised to oversee the protection system while the VN performed the GTT and updated the firewall. The IPS monitored packets for malicious activity, masked the victim cloud server using RP, controlled the load balancer (LB), and monitored the incoming traffic volume. Likewise, the client puzzle server limited the activities of suspicious users identified by RP [35]. Consequently, this model improved the previous DDoS-MS by including additional security levels (RP) while factoring the TTL value to check user authenticity.

A principal advantage of this model was its multilevel defence strategy, enhancing the overall robustness of the protective system. The firewall acted as the first layer of defence in executing access control and traffic filtering. Subsequently, the VN further helped discriminate between legitimate users and possible attackers during GTT by updating the rules of the firewall regarding its assessments. The IPS component finally checked packets against established signature patterns, denoting malicious activity as an extra security layer [61]. Overall, the primary conditions for utilising an RP include concealing the target cloud server from the outset of a direct attack while facilitating load balancing and traffic volume monitoring to enhance efficiency and security. The client puzzle server then throttled RP by restricting suspected users, effectively curtailing the opportunities available to the attacker (reducing the impact of user activity).

The model demonstrated several limitations. Numerous processes and components could be challenging in this method, such as system configuration, management, maintenance, firewall integration, VN, puzzle server, IPS, and RP. Hence, each component needed to be laboriously calibrated and synchronised with other elements to attain optimal performance and security, which was resource-intensive. Meanwhile, GTT and client-puzzle were theoretically sound. Nevertheless, both techniques could adversely affect clients in practice due to potential false positives or excessive challenges imposed on valid service users. Another issue was the effectiveness of the IPS for finding

malicious activities. This component was contingent upon the strategies employed in an attack, frequently requiring constant updating and tuning to adapt to evolving threats.

Even though RP was useful for abstracting the cloud server and managing traffic, latency and a bottleneck during high traffic conditions could be denoted. The necessity for all components of this strategy to function cohesively continuously also demanded significant administrative effort and was likely to incur substantial operational costs.

5.3.1.5 EDoS-7 Rao and Nene generated the EDoS-7 model containing a behaviour analysis framework based on two main building blocks [26]. Firstly, the SED functioned as a firewall and filtered the incoming requests through flow tables. Secondly, the GSC served as a VN in a virtual machine to determine legitimate or malicious requests by initiating and verifying GTT. Upon verification, the SED updated the legitimate requests accordingly [26]. A primary advantage of this model was the integration of flow-based request management mechanisms with behavioural analysis to enhance cloud security. The first line of defence to traffic involved the SED component regulating the flow of incoming requests using flow tables. This process effectively managed valid user requests and filtered potential attack routes.

The next stage included the GSC component, in which GTT conducted additional discrimination between authentic and malicious requests. This procedure guaranteed that the SED operation was influenced solely by verified requests, significantly diminishing the likelihood of a successful EDoS attack. The reliance on GTT also introduced another measurement layer (additional metric for security), as human user interactions facilitated the identification of malicious behaviours within cloud resources.

Numerous limitations still exist for the model. Latency within the system was exhibited when over-relying on GTT, impacting the user experience for legitimate users. This process could cause delays or additional issues for users using cloud services. The effectiveness and correctness of SED and GSC were also significantly correlated with the flow table configuration and parameter setup of GTT. Therefore, careful calibration and continuous maintenance were necessary due to the evolving attack patterns and valid user behaviour.

Another issue involved the scaling challenge, which presented an increasing incoming request volume. This process could render the flow-based management (SED) and the verification process (GSC) increasingly challenging and resource-intensive to maintain efficiency and accuracy. Likewise, the virtual machine component in this model could increase overhead within the cloud infrastructure. The scaling factor was also a potential issue, and the overall efficacy of the model hinged on its ability to consistently update and enhance its behavioural analysis to combat the intricate and continuously evolving EDoS attack strategies. Consequently, enormous investments in monitoring, analysis, and response capabilities were necessary.

5.3.1.6 Control virtual access to the cloud Baig and Binbeshr proposed an EDoS model containing four components (VF, LB, DB, and VMInvestigator) [31]. The VF filtered the incoming requests, rejecting those originating blacklisted entities. The VMInvestigator inspected these entities. The LB also managed the request of requests of the virtual machines, whereas the VMInvestigator created GTTs to examine user trustworthiness concerning the accuracy and timeliness of the users' responses. This approach integrated

quasi-static thresholds for CPU utilisation and request rates with user authentication verification through a user trust factor (UTF) metric. Potential threats included EDoS attacks [31].

The suggested multi-layered cloud EDoS defence model possessed numerous strengths, with architecture taking precedence. This technique was robust and observed in high and low-level architectures comprising VF, LB, DB, and VMInvestigator. Consequently, a stratified defence strategy was facilitated, improving the security posture of the cloud environment [62]. The VF also operated using a pre-established table of blacklisted entities, enabling fast filtering of identified threats and reducing the likelihood of malicious traffic affecting cloud resources.

The LB component monitored and dynamically readjusted the utilised cloud resources to prevent the overloading of virtual machines. This process was accomplished by distributing requests evenly for all VMs, ensuring efficient use of resources. Alternatively, the generated GTTs (VMInvestigator) and UTF evaluation introduced a new mechanism for verifying user legitimacy. This procedure strengthened the security mechanism by distinguishing genuine users from prospective attackers through system interaction.

Several constraints were observed with this model. The VF relied on a static blacklist, which could inadequately identify novel attack methodologies (reactive in nature). Advanced attackers could also decrease the effectiveness of the established GTT (VMInvestigator) or the reliability of UTF. Additionally, false positives influencing legitimate users could occur due to the nature of CPU use with request rate thresholds (static) and cloud (dynamic). The complexity of this multi-component model could also pose operational challenges, demanding substantial resources and expertise for balanced security and performance.

5.3.1.7 Adaptive Pattern Attack Recognition Technique (APART) Amazon introduced the adaptive pattern attack recognition technique (APART) as an anomaly detection model based on traffic pattern analysis to identify traffic attacks [36]. This model performed network traffic flow monitoring and static request rate threshold analysis to identify potential threats. One notable threat example involved high-frequency request attacks from multiple sources from 400 Hz to 800 Hz [23]. Nevertheless, this model entailed a significant dependence on static thresholds for evaluating request rates, resulting in numerous false positives that undermined the effectiveness of the model. Most of the algorithms and operational details were also not specified, indicating that further clarification was required.

5.3.1.8 HTTP Request Filtering (HRF) Rao et al. described an HTTP request filtering (HRF) model used in classifying users based on a web application firewall (WAF) containing three different lists (white, black, and unknown categories) [36]. The decision process of HRF was accomplished through three steps: (i) request filter (RF), (ii) network analyser (NA), and request analyser (RA). A verifying node for list management and verification (S3 bucket and Lambda functions) was employed, while activity monitoring and notifications are based on CloudWatch.

The proposed model presented various advantages. One benefit was its organised framework, which enhanced security and connected with Amazon web services (AWS) for real-time monitoring and response. This model effectively prevented EDoS attacks

using the HRF mechanism by considering latency, resource usage, and cost performance metrics. Conversely, the limitation of this model involved the reliance on static lists and thresholds, resulting in its inability to adapt to new attack vectors. The complexity and maintainability of the model also introduced operational challenges and higher costs. Moreover, this evaluation could overlook other critical considerations (user privacy or accuracy in the user classification).

5.3.1.9 EDoS-ADS Shawahna et al. introduced the EDoS-ADS model based on a defensive shell designed to reduce EDoS attacks using static thresholds of CPU usage and scaling periods [38]. This process enabled scaling CPU resources within a predetermined limit and duration, triggering EDoS-shell when thresholds were reached. Various performance and effectiveness metrics of this model included response time, throughput, and cost performance [37]. The model exhibited advantages in its straightforward scalability of resources. Consequently, the defence mechanism was straightforward and cost-effective in managing EDoS threats. The evaluation criteria also addressed vital performance aspects of a system, ensuring operational efficiency.

One limitation observed in the model was the static thresholds, which increased the likelihood of false positives due to rigid threshold parameters. This constraint was also a foundation for attackers regarding the predictability of the scaling limitations of the system, increasing false negative risks and compromising the effectiveness of the model security.

5.3.1.10 Enhanced EDoS-ADS Ko et al. [23]. criticised the EDoS-ADS model proposed by Shawahna et al. [38]., which was vulnerable to Yo-Yo attacks. These attacks were an EDoS attack type that could circumvent this system undetected by not engaging it directly. Thus, the enhanced EDoS-ADS was developed by implementing a more robust defence mechanism. This enhanced system contained three logical phases: (i) dual scaling policies masking the scaling triggers from the attackers, (ii) continuous monitoring for user behaviour to detect anomalies (high request rates or unusual activity patterns, and (iii) virtual machine limits for cost control [24]. The enhanced EDoS-ADS model leveraged a multi-layered strategy, making it difficult for an attacker to anticipate system reactions by strengthening its defence against EDoS attacks [63].

Analysing user behaviour and establishing scaling restrictions could diminish false positives and the financial impacts of allocating unnecessary resources. Nonetheless, this process created complexity and increased operating needs involving monitoring user behaviour and scaling policies. A strict limit on scaling resources could also hinder the management of legitimate traffic at peak periods, leading to possible shortages in service availability and a lower user experience quality.

5.3.1.11 EDoS-IDM The work criticises the Defence System EDoS-ADS of [38], showing its vulnerability to Yo-Yo attacks, a variety of EDoS attacks that could avoid this system without being detected because of not attacking it directly. To solve this research, once more an enhanced version of EDoS-ADS was developed, this time implementing a stronger defence mechanism. This enhanced system is based on three logical phases: it employs dual scaling policies, which would mask the scaling triggers from the attackers; continuous monitoring for user behavior to notice the anomalies in terms of high request

rates or unusual patterns of activities; and capping virtual machines to be scaled for cost control [64]. The enhanced EDoS-ADS derives its power from its multi-layered approach, wherein it is complex for an attack to predict system responses, hence its defense position in an EDoS attack. User behavior analysis and setting scaling limits would clearly reduce false positives and the financial impacts of allocating unnecessary resources.

But this in return will introduce complexity and perhaps higher demands from an operational perspective. This would involve continuously monitoring the behavior of users, several policies for scaling, in which hard scale points may require administrative effort and resources to set. Moreover, setting a hard cap on scaling resources may obstruct the handling of legitimate traffic during peak demands, leading to possible shortages in service availability and a corresponding decrease in the quality of user experience.

5.4 Hybrid/integrated strategies

Hybrid models combine elements of detection, prevention, and mitigation into unified frameworks. These may incorporate entropy-based detection, machine learning, behavioral analysis, honeypots, and game theory for robust defenses.

This section reviews 18 articles on defence mechanisms against EDoS attacks in cloud environments, employing multiple models within each solution. Table 10 summarises these articles.

5.4.1 Zipf's law-entropy

Idziorek and Tannian described a dual-method model for detecting FRC attacks [49]. The strategy combined Zipf's law and entropy to analyse consumption patterns and determine unusual behaviour. Likewise, the anomalies were created by the deviations from the expected traffic patterns or session time. Considering that the consumption patterns and session behaviours were assessed for potential FRC attacks, this model was deemed a holistic approach. Zipf's law was based on quantity indicator, while the entropy-based technique relied on session time for detection accuracy. Nevertheless, high false positive levels were denoted within these methods.

5.4.2 EDoS and DDoS shield

Mary et al. mitigated EDoS attacks using a model combining cloud traceback (CTB), cloud protector (CP), and EDoS-Shield [21]. The EDoS-Shield checked the legitimacy of users. On the contrary, the CTB identified request sources using the DPM algorithm. The CP also used backpropagation and NNs to detect and lower DDoS-EDoS traffic. Thus, this multi-component model was robust as it involved three primary features: (i) legitimacy checking of users, (ii) attack origin identification, and (iii) malicious traffic filtration. Although the model employed by CTB effectively traced the origin of requests, the active detection of malicious traffic was neglected (undermining the overall defence strategy).

5.4.3 ARMOR

Masood et al. employed congestion and admission control to produce the EDoS Armor as a two-tier e-commerce security model [25]. Several phases occurred in this model, including a challenge mechanism for user authentication and categorising legitimate and

Table 10 Summary of hybrid-based models for EDoS attacks

Author	Approach	Description	Limitation
[49]	Zipf's law and entropy	Two methodologies are employed to detect FRC as follows: • Zipf's law to analyse the user consumption pattern. • The entropy-based method to detect anomalous behaviour.	• The first methodology assumed that the normal traffic pattern followed Zipf's law. Any deviation from this law was considered anomalous. • False rates were observed due to Zipf's Law only focusing on quantity metrics (number of requests per session and login). • The entropy-based method emphasised session length, which was inadequate for precise identification. This process produced significant false positive rates.
[21]	GTT and machine learning algorithms (BP-NN)	Several primary components are observed in the model as follows: • Combined EDoS shield, CTB, and CP • The EDoS-Shield was responsible for deciding the user legitimacy. • The CTB was based on the DPM algorithm to detect the source of the request origin. • The CP utilised BP-NN machine learning algorithms.	• Considering that this model employed a similar approach to EDoS-Shield, it possessed identical limitations and failed to enhance EDoS detection. • The CTB and CP were EDoS-Shield extensions that did not contribute to the detection process. These extensions only identified the source of the attack, adding to the complexity of the methodology. • This model presented increased response time.
[25]	Crypto-puzzle, user number limit, and user priorities	This model used a two-tiered approach for e-commerce websites: (i) admission and (ii) congestion controls. The number of concurrent clients using cloud services was limited, and admission control was used to restrict them. In contrast, the congestion control prioritised permitted customers based on a browsing behaviour learning mechanism. Furthermore, the learning method evaluated clients as favourable or unfavourable based on their system activity. A challenge server was also included to decide whether the user was a bot or a human before delivering the request to the control.	• This model presented limited cloud elasticity and availability due to user capacity constraints. • This model presented increased response time for genuine users owing to the learning system and priority upgrading. • The IP spoofing was not considered. • New users could become disinterested due to the difficulty of the login process. • This model appeared to be for optimising the e-commerce page usage rather than detecting or protecting a system against EDoS attacks.
[30]	Flow-based monitoring statistical anomaly detection	This statistical anomaly-detection model consisted of three modules: (i) data preparation, (ii) detection, and (iii) mitigation. The first phase involved flow-based monitoring and collection using a sFlow agent, which was forwarded to the second phase. The second phase collected and examined anomaly detection, extracting the necessary information. Finally, the alarm was generated in the third, blocking the attacking IP.	• The spoofed IP was not addressed.

Table 10 (continued)

Author	Approach	Description	Limitation
[13]	Static game theoretical and honeypots	This model was a game-theoretical approach to EDoS mitigation in a static game scenario functioning as an analytical model to detect the appropriate threshold value based on Nash equilibrium. The attacker and defender (network administrator) were observed as a two-player model. Thus, the honeypots reduced the number of false positives. Meanwhile, the overarching design of the EDoS-Eye project indicated that the edge router served as the entry point for malicious and benign inbound traffic flows. Although the model processed the traffic, it could not distinguish between attack and genuine traffic. The aggregate flow was then equally distributed using an LB.	• The technique for solving games with mixed strategies was particularly complex when dealing with a large payment matrix. • The attack traffic was assumed to follow Poisson distribution. Nevertheless, the system stability limit of the Poisson distribution value was below 1. • Attackers could detect the static honeypots and avoid them.
[50]	Entropy and adaptive thresholds	This model effectively detected unexpected behaviours concerning erroneous signals for resource allocation based on entropy and setting adaptive thresholds. The three main stages of the EDoS detection approach were monitoring with aggregation, detection, and decision-making.	
[11]	Dynamic address changing of resources and real-time injection traps	A WEBTRAP security model could stop EDoS attacks on web-based systems. This model contained two essential components: (i) dynamic modification of online resource addresses for a moving target defence and (ii) real-time trap insertion to detect intruders. An online control-based system governed the trap injection to limit harm to legitimate consumers while cutting costs.	• The security model could not recognise legitimate users and attackers.
[51]	GTT and predefined Thresholds	The model comprised two primary components: (i) VNs and (ii) firewall. The VN and FW utilised GTT and predefined thresholds, respectively. These components collaboratively obstructed unauthorised IPs while allowing the authenticated IPs. The model also measured the time response and computing power consumption to evaluate strategy performance.	• The metrics for the solution performance were not sufficient.
[52]	MLAR protection against EDoS	The suggested model protected the cloud against EDoS assaults by combining periodic authentication, pattern analysis, and data flow management mechanisms. This model could mitigate financial losses caused by EDoS attacks.	• This model required more computing complexity. • This model needed more response time. • This model was not appropriately evaluated, raising concerns regarding its efficiency in a real-time environment.
[53]	Machine learning and fuzzy entropy	This model contained three phases based on machine learning algorithms and fuzzy entropy to mitigate EDoS attacks. The first phase extracted the log file from the data and built the DB for multiple features. The second phase used fuzzy entropy to extract the most proper features. The last phase employed an LNL algorithm to classify the traffic as attacker and authenticated.	• High error rates were observed due to predefined rules. • The model was complex in producing the result.

Table 10 (continued)

Author	Approach	Description	Limitation
[54]	Deep learning, static threshold for requests per hour, and DWT	The scenario of the model was initiated by dividing the webpage into several quantiles based on the popularity index. Subsequently, the number of requests per hour was computed. The DWT was applied as a filtration mechanism before inputting the data into the ANN model for classification.	• The model could not function at an extremely low traffic rate (FRC) below 5%. • Even though a static ANN model (trained once without a feedback loop) was used to detect static FRC efficiently, it was not sufficient for detecting dynamic FRC. • Only a one-hour interval as a unit for traffic analysis was insufficient to derive adequate historical information for detecting FRC attacks.
[14]	Entropy and static thresholds	The model combined various statistical techniques, including Hellinger distance, entropy, and OpenFlow technology. This model contained three modules (data preparation, detection, and mitigation). Network traffic samples were gathered using sFlow agents, whereas network statistics were produced by analysis at a sFlow collector during the data preparation step. The network statistics were also compared to the predetermined criteria during the detection phase. Moreover, the entropy analysis was employed to identify the suspected behaviour. Switching rules were updated in the last step in the mitigation process, and network traffic from suspicious source IP addresses was blocked using OpenFlow controllers.	• Higher error rates were observed due to static thresholds.
[23]	Static thresholds for CPU use, duration timers, CRPS, and VM number alongside GTT for suspicious users	The limitation in EDoS-ADS proposed by Shawahna et al. was proven using a Yo-Yo EDoS-based attack, in which the attacker could confuse the EDoS-ADS mechanism without provoking the attack mode. The article then proposed a model to enhance EDoS-ADS to address this issue. Initially, at least two scaling policies were produced to render it challenging for an attacker to anticipate when to scale up or down. The user behaviour was then examined for periodic high CRPS and zeros to tag it as suspicious, which was delivered to GTT. Finally, the number of VMs available for scaling up was limited to avoid more financial losses.	• Only Yo-Yo EDoS-based attacks were effectively mitigated for this model. • Despite that this model could reduce expenses by limiting the number of VMs, the cloud scalability and availability features could be impacted.
[55]	Statistical, time series, and machine learning	A model was established to detect FRC attacks in a cloud environment containing three components: (i) statistical, (ii) time series, and (iii) automated learning (ANN and *k*-NN). Each approach exhibited differing efficiency levels and ineffectiveness regarding detection in certain instances.	• The article indicated that the model was insufficient for FRC detection. Even though combining all these techniques could be enough, the process was more costly than an FRC attack. • The NASA dataset was outdated, and its accuracy in representing actual cloud transportation patterns was questionable. • The NASA dataset possessed a limited number of attributes that could be derived. • The article was documented in 2021, while the underlying data was from 1995.

Table 10 (continued)

Author	Approach	Description	Limitation
[56]	Obfuscation approach, machine learning, and GTT	This model contained three stages: (i) registration, (ii) login, and (iii) training and testing. A CI-RDA LB obfuscation approach was used for IP spoofing, while an RCDH-ENN classifier distinguished between normal and attack traffic. The GTT was also used in the registration phase of this framework.	• The extra levels for user authentication and GTT necessitated more processing time and could eventually frustrate the users.
[57]	Dynamic game theoretical	This model employed game-theoretical signal modelling, in which a dynamic game scenario was observed between two players: (i) EDoS (attacker) and (ii) system (defender). The model also employed virtual honeypots to explore the suspicious traffic before its termination further.	• A game theoretical model involving mixed strategies produced a complicated model, • Issues occurred when large payment matrices were denoted.
[58]	Semi-Markov, static thresholds (CRPS and CPU usage), TF, GTT, URL redirection, and two timers	The Semi-Markov model contained an edge router, DS, LB, DB, and virtual machine. The DB stored various information, such as the user's behaviour table, TF, and the number of MRPS. Meanwhile, the DS employed URL redirection if the user's CRPS was below the MRPS. Otherwise, GTT was sent to the user. The model also applied two CPU use criteria (80% for upper and 30% for lower) and two durations (5 min for scaling up and 1 min for scaling down).	• The proposed model failed to identify the optimal defence strategy for the system against EDoS attacks. • The chance of executing an EDoS assault with restricted resources under one's control was not considered.
[59]	Binomial probability, TTL, and multi-SYN	This model presented a TCP SYN mitigation approach for cloud environments. The solution leveraged SDN to reduce the impact of system and spoofing-based TCP SYN flooding attacks. Furthermore, the EDOS-Trust Security model employed a binomial probability, TTL, and Multi-SYN to detect TCP SYN and spoofing attacks.	• Several thresholds (packet size and duration) must be defined like other statistical solutions. This process was challenging. • This model demonstrated a false rate due to the thresholds. • Only one specific EDoS attack could be detected (TCP-SYC attacks), neglecting other EDoS attack types.

Notes: EMM = Enhanced Mitigation Mechanism; FLNL = Fuzzy Entropy and Lion Neural Learner; MLAR = Multi-Layered Attack Recognition: TF = Trust Factor; CTB = Cloud Trace Back; CP = Cloud Protector; DPM = Deterministic Packet Marking; LNL = Levenberg Neural Network; RCDH-ENN = Regression Coefficients Deer Hunting-Deep Elman Neural Network; MRPS = Main Requests Per Second

malicious clients through various assessments (image-based challenges or crypto-puzzle within the admission phase. This model also controlled the number of valid connections through a multi-layered protection approach to efficiently improve security against overload from malicious requests. Nonetheless, several limitations could occur from this process. The cloud elasticity with availability could be restricted while the response time for legitimate users could increase due to the learning system. New users could also be deterred owing to the sign-in challenges. Primarily, this model optimised the page usage rather than mitigating EDoS attacks.

5.4.4 EDoS-Enhanced mitigation mechanism (EMM)

Bawa et al. reported an advanced EDoS mitigation model containing a three-module architecture (data preparation, detection, and mitigation [30]. Initially, flow-based monitoring was conducted by the sFlow agent. Data analysis collected critical information, blocking the assaulting IP address upon alert generation. This model also increased sensitivity by incorporating Hellinger distance and entropy into the output. Consequently, this model gathered, analysed, and responded to threats using sophisticated anomaly

detection techniques, rendering it a systematic approach to EDoS mitigation. Conversely, this model was only effective based on the anomaly detection precision. The potential for false positives or negatives in threat identification could also occur.

5.4.5 EDoS-eye

Chowdhury et al. addressed EDoS attacks using a game theory-based mitigation model [13]. This model was based on a two-player, non-cooperative zero-sum game to simulate various attacker or defender scenarios. A Nash equilibrium was also utilised to establish a threshold that diminished attackers' incentives. Concurrently, honeypots were integrated to lower the false positives and obscure the cloud server. Traffic management was also controlled using an edge router and an LB. Despite server protection and lower detection errors observed with honeypots within the game theory-based model, various limitations were presented [64]– [65]. The mechanism was complex while managing large payment matrices was challenging. Attackers could also identify and bypass static honeypots.

5.4.6 Entropy-based EDoS detection

Monge et al. avoided the false resource scaling caused by unexpected activities through an entropy-based EDoS detection model [50]. This model contained three stages: (i) monitoring or aggregation used for collecting and examining the request data, (ii) detection or decision-making (ARIMA prediction and adaptive threshold setting), and (iii) decisions based on identification analysis of potential threats. The dynamic thresholds were also effectively established using entropy and predictive modelling. Therefore, this model was highly fine-grained for EDoS threat detection and preventing unjustified resource scaling. Conversely, the ARIMA rendered the method complex, while accurate adaptive thresholds were challenging. These limitations could reduce the efficiency of the model and its capability to identify EDoS attack activities promptly.

5.4.7 Game theory and honeypots

In this section the most relevant studies that utilized game theory and honeypots are explored.

5.4.7.1 Dynamic game theoretical model Lalropuia and Khaitan explored EDoS attack dynamics between defenders and attackers using a model based on game-theoretical signal modelling and virtual honeypots [57]. This model employed Bayesian Nash equilibrium (BNE) in three strategies (mixed, pooling, separating) to adapt to attacker behaviours, with each strategy differing in the extent of information disclosed regarding the attacker's type to the defender. Consequently, a nuanced defence against EDoS attacks based on game theory and honeypots was observed, potentially enhancing system security. Nevertheless, model management was challenging owing to the increased complexity of implementing mixed strategies in game theory for large payment matrices.

5.4.7.2 Semi-markov model Lalrupuia addressed availability and reliability issues in cloud computing using an analytical model based on the semi-Markov process (SMP) [57]. Request handling, user behaviour assessment, and resource adjustment were managed using an integrated LB, DS, and DB model. The DS assessed user requests through

GTT and redirected them based on MRPS and the user's trust factor (TF), facilitating optimal resource scaling. Consequently, organised request management and systematic resource scaling could improve the reliability and availability of cloud service. Nonetheless, the flexibility and adaptability in dynamic cloud environments could be compromised due to the model complexity, reliance on certain thresholds, and scaling durations.

5.4.7.3 EDoS-TSM Ali Shah et al. proposed a quick statistical anomaly detection model using SDN to mitigate the TCP SYN flooding attacks (EDoS-TSM) [59]. This model combined binomial probability, TTL, and Multi-SYN techniques to identify attacks. The model then eliminated TCP SYN requests using payloads and used an SDN controller to obstruct recognised attack traffic. This process concentrated exclusively on TCP-SYN attack variants [67]. Although this model could quickly detect and mitigate TCP SYN flooding attacks, appropriate threshold values were challenging to determine. The singular emphasis of the model on TCP-SYN attacks also restricted its applicability to other EDoS attack types.

5.4.7.4 Multi-Layered Attack Recognition (MLAR) Arora safeguarded cloud networks from EDoS attacks using a multi-layered attack recognition (MLAR) model [52]. This model was based on periodic authentication, pattern analysis, and data flow control. The model also applied an integrated approach for evaluating the performance of network nodes, assessing data patterns, and authenticating nodes. These processes were critical for effective classification and avoiding financial loss or service level agreement (SLA) violation. Additionally, a multi-layer approach was observed in this model to enhance cloud security by meticulously scrutinising and validating data flow, focusing on detecting and mitigating potential EDoS threats. This systematic approach could then safeguard resources while upholding service quality. Nonetheless, high computational resources and increased response time were required in this complex model, leading to lower system efficiency.

5.4.7.5 Fuzzy entropy and Lion Neural Learner (FLNL) Bhingarkar and Shah classified traffic as legitimate or malicious using the fuzzy entropy and lion neural learner (FLNL) model for feature selection and classification [54]. The model employed fuzzy entropy to extract features from log files and the Levenberg neural network (LNL) for classification. This model also consisted of three stages: (i) relevant feature extraction, (ii) significant feature selection, and (iii) traffic classification. A fuzzy entropy in FLNL was utilised in feature extraction and selection to ensure that only the pertinent data were considered [66]. This process increased the subsequent classification precision within the LNL algorithm, providing an optimised methodological shortcut to identify malicious activities. Even though FLNL effectively classified documents into specific categories, high error rates were observed. This outcome was attributed to the intrinsic complexity and pre-existing constraints in the model structure, reducing its effectiveness.

5.4.7.6 Web traffic analysis Rustogi et al. combined DWT and ANN as a model to detect FRC attacks in cloud services [54]. The website data was segmented based on the popularity index in quantiles. Subsequently, the requests per hour were analysed. The time series data was then refined using DWT by removing high-frequency noise, and an ANN model

was trained with the preprocessed data for classification. This model also denoised data and classified it to effectively detect FRC attacks (below 5% attack traffic) using DWT and ANN, respectively. Given that the model was narrowed down to definite characteristics and data patterns, the efficiency of cloud services in detecting FRC attacks was improved [68]. Nevertheless, one major flaw with this model was the one-hour window for traffic analysis. This process could not yield a sufficient historical record necessary for detecting FRC attacks, impairing the capability of the model.

5.4.7.7 EDoS-EMM Singh et al. employed OpenFlow and statistical analyses (Hellinger distance and entropy) to combat EDoS attacks as an EDoS-enhanced mitigation model (EDoS-EMM) model [14]. This model contained three interconnected modules: (i) data preparation involving collecting and examining network traffic, (ii) detection by matching statistics against thresholds for identifying suspicious activities through entropy analysis, and (iii) mitigation by modifying switching rules of the OpenFlow controllers to block the suspected traffic sources. The model also demonstrated high effectiveness in mitigating HTTP and UDP attacks [69].

A dynamic and efficient approach to defending against EDoS attacks was denoted when combining OpenFlow and statistical methods in EDoS-EMM. This model reduced the financial burdens on consumers while enhancing the security profiles of the cloud services to promote customer confidence. Conversely, the practical application used predetermined threshold values and intricate coordination among these three modules. This process could raise issues regarding the prospects of false positives. Continued adjustments in the detection and mitigation parameters could also pose concerns in adapting to ever-evolving attack patterns.

5.4.7.8 EDoS-DOME Ribin and Kumar created a model by merging regression coefficients of the deer hunting-deep Elman neural network (RCDH-ENN) and.

obfuscated IP spoofing prevention for user classification into white or black lists [56]. This model contained a three-phase detection framework (registration, login, training-testing) while employing CI-.

RDA (load balancing), RCDH-ENN (traffic differentiation), and GTT (enhancing registration security). The model also effectively distinguished between legitimate and malicious traffic, improving security measures. Conversely, increased processing times due to the added authentication layers could cause inconvenience to users.

A review of the existing solutions reveals that most research focuses on mitigation, with limited emphasis on early detection and proactive prevention. Detection strategies have advanced with the use of deep learning and anomaly detection but are often hampered by high false positives. Prevention strategies suffer from rigid threshold dependencies and limited adaptability. Mitigation solutions provide reactive defense but do not address financial or operational sustainability. Hybrid models are emerging as promising solutions, although they tend to be complex and resource-intensive.

Figure 5 illustrates the taxonomy of approaches used to counter EDoS attacks, highlighting the strategic diversity and the need for further exploration in detection and prevention research.

Table 11 presents the quantitative benchmarking of literature reviewed in this research.

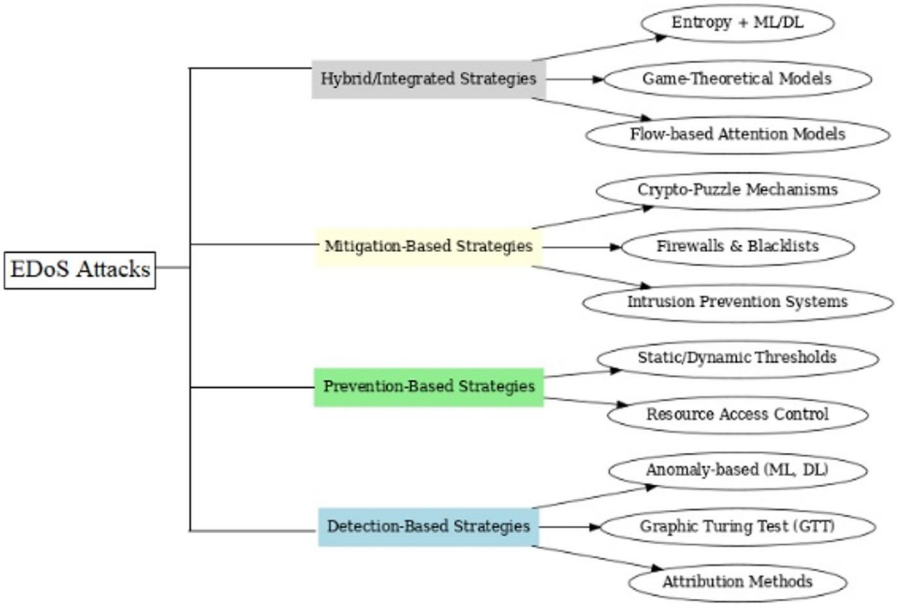

Fig. 5 Taxonomy of approaches used to counter EDoS attacks

Table 11 A snapshot of quantitative benchmarking of reviewed literature

Ref.	Accuracy	F1-Score	Latency	Scalability	Dataset used
[16]	99.46	99.41	Not Specified	High	UNSW-NB15
[21]	94.33	95.51	Low	High	SMD
[29]	97.01	97.05	Low	Not Specified	Real World Business Environment
[33]	95	Not Specified	Not Specified	High	CIC-DDoS2019, UNSW-NB15
[42]	97.3	Not Specified	18ms	High	SMD
[49]	99.0	98.5	Not Specified	High	CAShift

5.5 Analytical synthesis and critical prioritization of EDoS defense methods

Despite the wide range of detection, prevention, mitigation, and hybrid strategies proposed to counter EDoS attacks, their applicability and efficacy vary significantly depending on the underlying cloud service model Infrastructure as a Service (IaaS) or Software as a Service (SaaS).

5.5.1 Suitability for large-scale LaaS environments

IaaS environments are typically more vulnerable to resource exhaustion due to auto-scaling mechanisms and expose a broader surface for low-rate, long-duration EDoS

attacks. Hence, detection accuracy, real-time decision-making, and infrastructure-level adaptability are critical.

5.5.1.1 Recommended strategies Machine/Deep Learning Models: These excel in IaaS due to their ability to analyze large-scale, multivariate data. Models like R-EDoS, MAN-EDoS, and P-Estimation offer time-series-based anomaly detection, essential for identifying sophisticated patterns of resource abuse.

Adaptive Thresholding & Entropy-Based Detection: Suitable for dynamically scaling environments (e.g., EDoS-Enhanced Mitigation Mechanism, Monge et al.), these allow real-time resource allocation decisions that mitigate financial impacts.

GTT with Puzzle Servers (e.g., Enhanced DDoS-MS): Although computationally intensive, they are more applicable in IaaS where the user base is well-controlled and high resource overheads can be tolerated.

5.5.1.2 Limitations High false positive rates in ML models when training data is insufficient.

Entropy-based models can be sensitive to normal traffic bursts, triggering false alarms.

TTL and header inspection techniques are less reliable due to their assumptive nature and can increase latency.

5.5.2 Suitability for SaaS environments

SaaS platforms prioritize user experience and availability, making strategies that introduce latency, complexity, or false positives detrimental. Therefore, lightweight and client-transparent methods are preferable.

5.5.2.1 Recommended strategies Static/Dynamic Threshold Models (e.g., CloudWatch, EDoS-ADS): These are suitable for predictable service usage patterns and offer low overhead, making them viable in SaaS where real-time user service is critical.

WEB-TRAP and TSP-based Models: Useful for web-centric SaaS platforms due to their session analysis and moving target defense, particularly where login-based traps are acceptable.

Hybrid Approaches (e.g., FLNL, MLAR): Limited deployment in SaaS is feasible when combined with lightweight authentication or anomaly scoring for selected user sessions.

5.5.2.2 Limitations Static thresholds cannot adapt well to traffic spikes caused by legitimate marketing events or seasonality.

Game-theory-based models (e.g., EDoS-Eye) are theoretically robust but too complex and computationally heavy for dynamic SaaS applications.

5.5.2.3 Cross-model recommendations Hybrid Strategies (e.g., combining ML, puzzle servers, and adaptive thresholds) are promising for both IaaS and SaaS when tailored to operational constraints.

Attribution Techniques can augment all models by profiling long-term behavioral data, useful in both service models to improve whitelisting/blacklisting decisions. Table 12 presents the comprehensive summary of final prioritization of strategies.

Table 12 Final prioritization summary

Strategy type	Best for	Key methods	Notable limitations
Deep Learning & ML	IaaS	R-EDoS, MAD-GAN, MAN-EDoS	High resource use, false positives
Entropy-Based/Adaptive	IaaS	EDoS-EMM, Zipf's Law + Entropy	Parameter tuning complexity, low traffic misfires
Static Thresholds	SaaS	EDoS-ADS, CloudWatch	Poor adaptability to traffic dynamics
GTT/Crypto-Puzzles	IaaS (some SaaS)	EDoS-Shield, DDoS-MS	Latency, accessibility issues for users
Hybrid (ML + Rules)	Both	FLNL, MLAR, Enhanced EDoS-ADS	High system complexity and integration cost
Game Theory + Honeypots	IaaS (Limited)	EDoS-Eye, Dynamic Game Model	Computational overhead, complexity, detectability

Table 13 Layers, targets and the corresponding explotation mechanisms

Layer	Target	Exploitable mechanism
Application	Serverless functions, APIs	Auto-scaling triggers (e.g., HTTP floods)
Platform	Kubernetes pods, DB queries	Pay-per-use pricing (e.g., DynamoDB RCUs)
Infrastructure	VM instances, load balancers	Scaling policies (e.g., AWS ASG)
Economic	Billing alerts, reserved instances	Silent cost accumulation over time

6 Findings and future research directions on EDOS in cloud environments

This review revealed critical gaps in the detection, prevention, and mitigation of EDoS attacks within cloud environments. Addressing these gaps through targeted research can enhance the sustainability, resilience, and security of cloud services against EDoS threats. Deeper insight into these challenges can lead to more effective strategies for understanding and mitigating EDoS attacks.

6.1 Outline discription of EDoS attacks and attackers

In this section attacker objectives, targeted layers and the EDoS attacks tree are illustrated.

6.1.1 Attacker objectives

- Primary: Force victim's cloud costs to exceed budget ("Sustainability Denial").
- Secondary: Degrade performance via resource starvation (CPU, memory, API quotas).

6.1.2 Attack surface

The targeted layers and the corresponding exploitation mechanism shown in Table 13.

6.1.3 Attack tree (Root → leaf Nodes)

Figure 6 The Root-leaf nodes for EDoS attacks.

6.2 State transition diagram for EDoS attacks

Figure 7 depicts the transtion diagram of EDoS attacks.

Key Transitions:

- T1: Requests exceed scaling threshold (but stay below DDoS radar).
- T2: Billing system accrues costs without real demand.
- T3: Victim's auto-scaling reacts too late (cost already incurred).

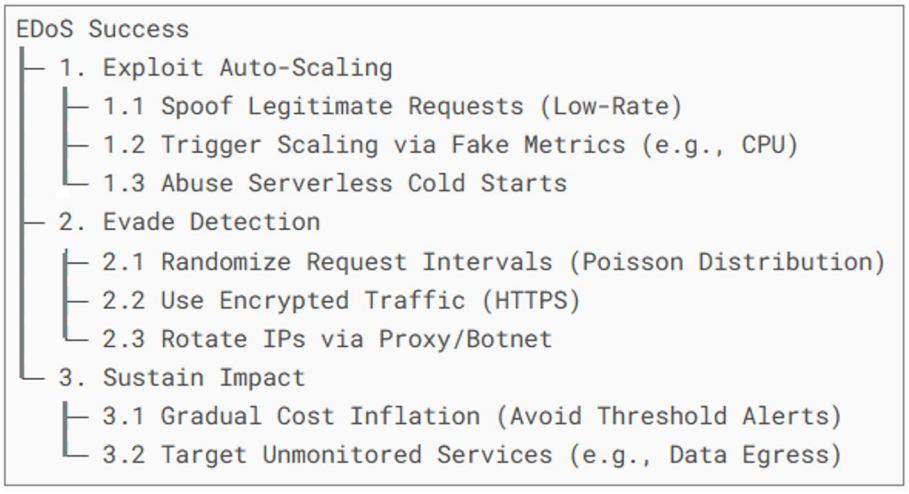

Fig. 6 The root-leaf nodes for EDoS attacks

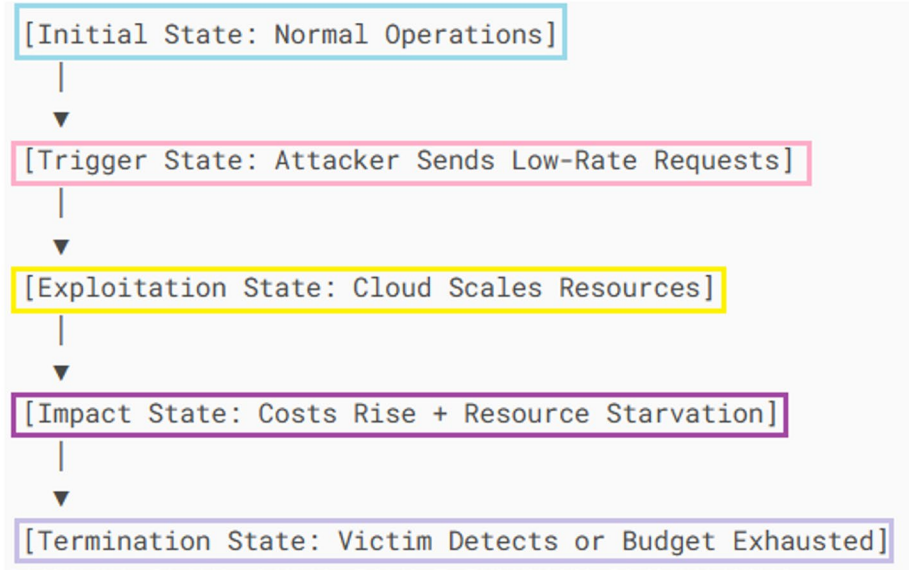

Fig. 7 Transition diagram for EDoS attacks

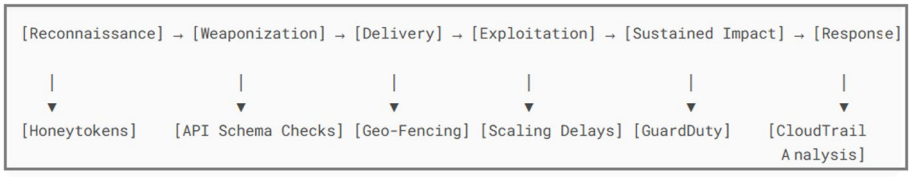

Fig. 8 Mapping of Attack lifecycle and Defense strategies

6.3 Mapping of attack lifecycle and defense strategies

This section presents a comprehensive overview of the EDoS attack lifecycle and highlights defense strategies at each stage. The mapping is encapsulated in Fig. 8, which illustrates how various attack phases correspond to specific countermeasures, offering a

clear visualization of where and how defense mechanisms can be implemented throughout the attack lifecycle.

6.4 Research future directions

Future research directions are highlighted in this section by categorizing the research directions into: Prevention, Detection, Mitigation, Hybrid Strategies and Other Aspects.

6.4.1 Prevention

Existing prevention-based strategies primarily rely on static threshold models to control access and resource utilisation, aiming to block suspicious traffic before it reaches cloud services. However, these approaches face significant limitations. Most studies reviewed focus on rigid thresholds (e.g. CPU usage, HTTP request rates, or packet sizes) which often lead to high false positive rates, blocking legitimate users alongside attackers. Furthermore, static thresholds struggle with dynamic attack behaviours, such as attackers who adapt their traffic to remain below detection limits. Implementation challenges include configuration complexity, the requirement for precise prior knowledge of normal usage patterns, and degradation of cloud elasticity due to hard-coded limits. Additionally, many models are attack-specific (e.g. only for ICMP or TCP SYN floods), limiting generalisability across EDoS variants.

To address these limitations, future prevention strategies should focus on adaptive and intelligent threshold mechanisms, integrating machine learning or statistical learning models that continuously adjust based on evolving traffic patterns to reduce false positives and negatives. Research should explore service-model-aware prevention frameworks tailored to IaaS, PaaS, and SaaS, enabling targeted and efficient defence without compromising usability. Furthermore, studies should investigate hybrid prevention approaches combining static thresholds with behavioural and reputation-based filtering for layered security. Developing cost-aware prevention models that factor in billing structures and economic impacts will help maintain cloud sustainability under attack. Finally, integrating prevention mechanisms seamlessly with detection and mitigation systems in a unified, automated defence architecture can ensure proactive, scalable, and practical protection against diverse and evolving EDoS threats.

6.4.2 Detection

Detection-based strategies aim to identify the presence of EDoS attacks by recognising anomalies in usage patterns, suspicious behaviours, or attacker profiles using statistical analysis, machine learning, and attribution-based techniques. GTT and crypto-puzzle-based solutions, such as EDoS-Shield, sPoW, and In-Cloud Scrubber, have been employed to differentiate legitimate users from attackers. However, these methods often suffer from usability limitations, including high false positive rates that disproportionately affect elderly and disabled users, vulnerability to puzzle accumulation attacks, and latency overheads.

Time-to-live (TTL) and header inspection-based models mitigate IP spoofing by validating TTL values within packet headers. While effective against basic spoofing attempts, these approaches introduce management complexity and remain vulnerable to sophisticated attackers capable of forging or mimicking TTL parameters accurately.

Machine learning and deep learning approaches, including SVMs, ANNs, LSTMs, GANs, and attention-based models, offer adaptive detection capabilities by analysing multivariate time series data, resource usage metrics, and traffic flow features. Despite their potential, such models often face significant challenges, including high computational requirements, impractically long sequence lengths for real-time deployment, limited accuracy against multi-vector or low-rate attacks, and elevated false alarm rates that reduce operational trustworthiness.

Future research of detection techniques may focus on integrating lightweight behavioural analytics with user-aware verification schemes to minimise false positives without imposing additional burdens on legitimate users. For GTT and crypto-puzzle models, adaptive complexity scaling based on real-time user capability profiling can improve inclusivity and usability during attack scenarios. TTL and header inspection-based models can be improved by incorporating multi-layer route validation and cryptographic authentication to counter sophisticated spoofing techniques effectively.

In the context of machine learning and deep learning models, combining lightweight feature extraction with attention-based architectures, transfer learning, and continual model updates could significantly improve detection accuracy while reducing computational overhead. Moreover, integrating these models with real-time orchestration frameworks and continuous feedback mechanisms would enhance their adaptability and responsiveness to evolving EDoS attack patterns to ensure the practical deployment in dynamic cloud environments.

6.4.3 Mitigation

The reviewed mitigation-based strategies, including In-Cloud Scrubbers, puzzle servers, multi-layered models (e.g., Enhanced DDoS-MS), and adaptive techniques (e.g., APART, EDoS-ADS enhancements), show considerable promise but exhibit significant limitations. Common issues include reliance on static thresholds, resulting in false positives or negatives, excessive latency impacting user experience, complex configuration and integration requirements across multiple components, and limited scalability under high demand conditions. Puzzle-based models often introduce user friction, while blacklist/whitelist management remains operationally challenging. Behavioural analysis models, although innovative, require careful calibration and continuous adaptation to avoid undermining legitimate user access and to prevent cost inefficiencies in cloud resource usage.

Future research should focus on developing adaptive and dynamic thresholding mechanisms, integrating AI/ML-based behavioural analytics to replace static configurations while minimising false positives. Additionally, lightweight user verification methods should be designed to preserve user experience without sacrificing security efficacy. Exploring collaborative defence architectures across cloud providers, combined with predictive scaling policies, could also address Yo-Yo and slow-drip EDoS attack variants. Finally, comprehensive cost-benefit analysis models need to be integrated into mitigation frameworks to ensure resource efficiency and operational feasibility in real-world deployments.

6.4.4 Hybrid strategies

Hybrid strategies combine multiple defense techniques to harness their complementary strengths, offering a more effective and resilient approach to mitigating EDoS attacks. Integrating detection, prevention, and mitigation mechanisms build layered defenses that adapt to evolving threats more robustly than individual methods alone.

Many hybrid frameworks dynamically adjust their components in response to observed traffic patterns and attack behaviours. They incorporate adaptive thresholding, traffic profiling, and real-time response capabilities to optimize defense performance. Machine learning often underpins traffic classification, while game theory models the strategic interactions between attackers and defenders. Honeypots are deployed to attract, isolate, and analyze malicious traffic without disrupting legitimate users, enriching threat intelligence.

The literature reflects a clear trend towards hybrid solutions due to their enhanced detection accuracy, reduced false positives, and improved robustness against diverse and sophisticated EDoS attack vectors. However, challenges remain, particularly in balancing computational overhead, minimizing response latency, and ensuring scalability within complex cloud environments.

Future research directions in hybrid strategies include the integration of deep learning models with real-time adaptive control systems, deployment of intelligent and self-evolving honeypots, and leveraging blockchain technology for decentralized verification and mitigation. These innovations have the potential to significantly boost the effectiveness, efficiency, and trustworthiness of hybrid defense frameworks against EDoS attacks in cloud computing.

6.4.5 Other aspects

i. Comprehensive attack coverage: Previous articles predominantly examined specific EDoS attack types (such as TCP SYN flooding [59]– [60]), neglecting various attack vectors. Hence, future studies should provide holistic defence mechanisms by investigating a more comprehensive range of EDoS threats.
ii. Integration of methodologies: Although previous articles displayed diverse approaches (game-theoretical models to anomaly detection), insufficient cohesive models or comparative analyses were observed. This process could lead to fragmented insights. Therefore, future studies should yield more robust solutions by implementing a comprehensive research initiative that compares and synthesises different methodologies.
iii. Practical implementation challenges: Theoretical models usually offer innovative solutions. In contrast, previous articles demonstrated significant challenges (configuration complexity and user inconvenience) in their practical applications. Hence, theoretical models should be effectively examined to ensure their transition into user-friendly, scalable solutions.
iv. Evaluation metrics: Previous articles indicated an inconsistency in the application and comparison of evaluation metrics. Therefore, future studies should assess the effectiveness, efficiency, and impact of proposed EDoS mitigation strategies by adopting a standardised set of comprehensive metrics.
v. Research scope broadening: This review identified several underexplored areas, including the economic impacts of EDoS attacks, legal considerations, and emerging

technologies (AI and blockchain) in combating or facilitating EDoS threats. Hence, future studies should develop more effective countermeasures by expanding research to these areas for valuable insights.

Overall, resolving the identified gaps through future research could substantially improve the security of cloud infrastructures against these threats. Concurrently, previous articles could provide proper tactics for reducing EDoS assaults.

7 Conclusion

This review indicated the significant threat of EDoS attacks within the cybersecurity landscape. The attacks could disrupt critical infrastructure, government services, and national security. These attacks could also disrupt essential services (healthcare, finance, and emergency response), leading to societal and economic disruptions. Thus, EDoS attacks must be addressed to protect individual organisations and safeguard national security and societal well-being. This review also demonstrated inadequate articles concerning EDoS attacks in cloud environments employing diverse mitigation strategies. Meanwhile, several vital articles were presented, outlining various methodologies, evaluation criteria, and the limitations of current solutions. Consequently, the data provided in this review could be a foundational resource for understanding the current state and identifying avenues for future research. This review also enhanced the literature by thoroughly examining knowledge on EDoS assaults, emphasising the necessity for continued research and innovation.

Author contributions

Zubaidi Maytham Sahar Saeed: Wrote the main manuscript.Fuad A. Ghaleb: Reviewed the work and revised it critically for important intellectual content.Anazida Zainal: Reviewed the work and revised it critically and polished the final version. Bander Ali Saleh Al-rimy: Reviewed the work and did the necessary proofreading.

Funding

No funding was received for conducting this study.

Data availability

No datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate

All participants were informed about the purpose of the study and provided informed consent to participate. In addition, participants gave their consent for publication of the findings arising from their data.

Competing interests

The authors declare no competing interests.

Received: 27 February 2025 / Accepted: 8 July 2025

Published online: 22 July 2025

References

1. Qazi ADDINENREFLIST, Kwak F, Khan D, Ali FG, Khan F. Service level agreement in cloud computing: taxonomy, prospects, and challenges. Internet Things. 2024;25:101126.
2. Soni D, Kumar N. Machine learning techniques in emerging cloud computing integrated paradigms: A survey and taxonomy. J Netw Comput Appl. 2022;205:103419.
3. Narayan D. Platform capitalism and cloud infrastructure: theorizing a hyper-scalable computing regime. Environ Plann A: Econ Space. 2022;54(5):911–29.
4. Kshetri N. Privacy and security issues in cloud computing: the role of institutions and institutional evolution. Telecomm Policy. 2013;37(4):372–86.
5. Vinoth S, Vemula HL, Haralayya B, Mamgain P, Hasan MF, Naved M. Application of cloud computing in banking and e-commerce and related security threats. Materials Today: Proceedings. 2022;51:2172–5.
6. Dr. MRC, Sheel Ghule MKP. Cloud computing in banking services. Int J Sci Res Publications. 2014;4(6), ISSN 2250-3153.

7. Alqahtani KS, Albalawi AM, Frikha M. REVIEWING OF CYBERSECURITY THREATS, ATTACKS, AND MITIGATION TECHNIQUES IN CLOUD COMPUTING ENVIRONMENT. J Theoretical Appl. Inform. Technol. 2023;101(6):2058–2066.
8. Al-Haidari F, Sqalli M, Salah K. Evaluation of the impact of EDoS attacks against cloud computing services. Arab J Sci Eng. 2015;40(3):773–85.
9. Ficco M, Palmieri F. Introducing fraudulent energy consumption in cloud infrastructures: A new generation of Denial-of-Service attacks. IEEE Syst J. 2017;11(2):460–70.
10. Charlie. We are under attack: Greatfire.Org; 2015 [Available from: https://en.greatfire.org/blog/2015/mar/we-are-under-attack
11. Wang H, Xi Z, Li F, Chen S, editors. WebTrap: A dynamic defense scheme against economic denial of sustainability attacks. 2017 IEEE Conference on Communications and Security N, Singh (CNS), PMS.2017: Publisher: IEEE.
12. Singh PM, S.; Rehman, S.U. A survey of mitigation techniques against Economic Denial of Sustainability (EDoS) attack on cloud computing achitecture. In Proceedings of the IEEE 3rd International Conference on Reliability, Infocom Technologies and Optimization (ICRITO),; 8-10 October;Noida, India2014. pp. 1–4.
13. Chowdhury FZ, Kiah LBM, Ahsan MAM, Idris MYIB, editors. Economic denial of sustainability (EDoS) mitigation approaches in cloud: Analysis and open challenges. 2017 International Conference on Electrical Engineering and Computer Science (ICECOS); 2017 22-23 Aug. 2017.
14. Singh P, Rehman SU, Manickam S. Comparative analysis of state-of-the-art EDoS mitigation techniques in cloud computing environments. ArXiv Preprint arXiv:190513447. 2019. (Volume 1)
15. Nautiyal S, Wadhwa S, editors. A Comparative Approach to Mitigate Economic Denial of Sustainability (EDoS) in a Cloud Environment. 2019 4th International Conference on Information Systems and Computer Networks (ISCON); 2019 21-22 Nov. 2019.
16. Sqalli MH, Al-Haidari F, Salah K, editors. EDoS-Shield - A Two-Steps Mitigation Technique against EDoS Attacks in Cloud Computing. 2011 Fourth IEEE International Conference on Utility and Cloud Computing; 2011 5-8 Dec. 2011.
17. Al-Haidari F, Sqalli MH, Salah K, editors. Enhanced EDoS-Shield for Mitigating EDoS Attacks Originating from Spoofed IP Addresses. 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications; 2012 25-27 June 2012.
18. Kumar MN, Sujatha P, Kalva V, Nagori R, Katukojwala AK, Kumar M, editors. Mitigating Economic Denial of Sustainability (EDoS) in Cloud Computing Using In-cloud Scrubber Service. 2012 Fourth International Conference on Computational Intelligence and Communication Networks; 2012 3-5 Nov. 2012.
19. VivinSandar S, Shenai S. Economic denial of sustainability (EDOS) in cloud services using HTTP- and XML-based DDoS attacks. For example, Int J Comput Appl. 2012;41(20):11–16.
20. Abbasi H, Ezzati-Jivan N, Bellaiche M, Talhi C, Dagenais MR. Machine learning-based EDoS attack detection technique using execution trace analysis. J Hardw Syst Secur. 2019;3(2):164–76.
21. Mary IM, Kavitha P, Priyadharshini M, Ramana VS. Secure cloud computing environment against ddos and edos attacks. 2014.
22. Thaper R, Verma A, editors. Adaptive Pattern Attack Recognition technique (APART) against EDoS attacks in Cloud Computing. 2015 Third International Conference on Image Information Processing (ICIIP); 2015 21-24 Dec. 2015.
23. Ko HJ, Wang GY, Horng G, -J S W, editors. A Chaotic Attack Offering with Improving Mechanism in Economic Denial of Sustainability. 2020 International Conference on Pervasive Artificial Intelligence (ICPAI); 2020 3-5 Dec. 2020.
24. Ali Shah SQ, Zeeshan Khan F, Ahmad M. The impact and mitigation of ICMP based economic denial of sustainability attack in cloud computing environment using software defined network. Comput Netw. 2021;187:107825.
25. Masood M, Anwar Z, Raza SA, Hur MA, editors. EDoS Armor: A cost effective economic denial of sustainability attack mitigation framework for e-commerce applications in cloud environments. INMIC; 2013 19-20 Dec. 2013.
26. Rao RG, Nene MJ, editors. SEDoS-7: A proactive mitigation approach against EDoS attacks in cloud computing. 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET); 2017 22-24 March 2017.
27. Nhu C-N, Park M. Two-Phase deep Learning-Based EDoS detection system. Appl Sci. 2021;11(21):10249.
28. Somani G, Gaur MS, Sanghi D, editors. DDoS/EDoS attack in cloud: affecting everyone out there! Proceedings of the 8th International Conference on Security of Information and Networks; 2015.
29. Ta V, Park M, MAN-EDoS:. A multihead attention network for the detection of economic denial of sustainability attacks. Electronics. 2021;10(20):2500.
30. Bawa PS, Rehman SU, Manickam S. Enhanced mechanism to detect and mitigate economic denial of sustainability (EDoS) attack in cloud computing environments. Int J Adv Comput Sci Appl. 2017;8(9):51–8.
31. Baig ZA, Binbeshr F, editors. Controlled Virtual Resource Access to Mitigate Economic Denial of Sustainability (EDoS) Attacks against Cloud Infrastructures. 2013 International Conference on Cloud Computing and Big Data; 2013 16-19 Dec. 2013.
32. Khor SH, Nakao A. On-demand cloud_based eddos mitigation mechanism. HotDep (Fifth Workshop on Hot Topics in system dependability 2009.
33. Alosaimi W, Al-Begain K, editors. An Enhanced Economical Denial of Sustainability Mitigation System for the Cloud. 2013 Seventh International Conference on Next Generation Mobile Apps, Services and Technologies; 2013 25-27 Sept. 2013.
34. Alosaimi W, Zak M, Al-Begain K, Alroobaea R, Masud M. Economic denial of sustainability attacks mitigation in the cloud. Int J Communication Networks Inform Secur. 2017;9(3):420–31.
35. Amazon. CloudWatch Developer Guide 2015 [Available from: https://s3.cn-north-1.amazonaws.com.cn/aws-dam-prod/china/pdf/acw-dg.pdf
36. Rao BB, Bulla S, Rao KG, Chandan K, editors. HRF (HTTP request filtering): a new detection mechanism of EDOS attack on cloud. 2019 International Carnahan Conference on Security Technology (ICCST); 2019: IEEE.
37. Shawahna A, Abu-Amara M, Mahmoud ASH, Osais Y. EDoS-ADS: an enhanced mitigation technique against economic denial of sustainability (EDoS) attacks. IEEE Trans Cloud Comput. 2020;8(3):790–804.
38. Koduru A, Neelakantam T, B MS S, editors. Detection of Economic Denial of Sustainability Using Time Spent on a Web Page in Cloud. 2013 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM); 2013 16-18 Oct. 2013.
39. Dinh PT, Park M, editors. Dynamic Economic-Denial-of-Sustainability (EDoS) Detection in SDN-based Cloud. 2020 Fifth International Conference on Fog and Mobile Edge Computing (FMEC); 2020 20-23 April 2020.

40. Dinh PT, Park M, editors. Economic Denial of Sustainability (EDoS) Detection using GANs in SDN-based Cloud. 2020 IEEE Eighth International Conference on Communications and Electronics (ICCE); 2021 13-15 Jan. 2021.
41. Dinh PT, Park MR-EDS. Robust economic denial of sustainability detection in an SDN-Based cloud through stochastic recurrent neural network. IEEE Access. 2021;9:35057–74.
42. Agarwal A, Prasad A, Rustogi R, Mishra S. Detection and mitigation of fraudulent resource consumption attacks in cloud using deep learning approach. J Inform Secur Appl. 2021;56:102672.
43. Kanimozhi V, Jacob TP. Artificial intelligence outflanks all other machine learning classifiers in network intrusion detection system on the realistic cyber dataset CSE-CIC-IDS2018 using cloud computing. ICT Express. 2021;7(3):366–70.
44. Ribin Jones S, Kumar N, editors. EDoS-BARRICADE: A Cloud-Centric Approach to Detect, Segregate and Mitigate EDoS Attacks. International Conference on Communication, Computing and Electronics Systems; 2021: Springer.
45. Ta QV, Park M, editors. Economic Denial of Sustainability (EDoS) attack detection by attention on flow-based in Software Defined Network (SDN). 2022 International Conference on Information Networking (ICOIN); 2022 12-15 Jan. 2022.
46. Britto Dennis J, Shanmuga Priya M. Deep belief network and support vector machine fusion for distributed denial of service and economical denial of service attack detection in cloud. Concurrency Computation: Pract Experience. 2022;34(1):e6543.
47. Idziorek J, Tannian M, Jacobson D, editors. Attribution of fraudulent resource consumption in the cloud. 2012 IEEE fifth international conference on cloud computing; 2012: IEEE.
48. Karami M, Chen S, editors. Attribution of economic denial of sustainability attacks in public clouds. International Conference on Security and Privacy in Communication Systems; 2016: Springer.
49. Idziorek J, Tannian M, editors. Exploiting Cloud Utility Models for Profit and Ruin. 2011 IEEE 4th International Conference on Cloud Computing; 2011 4-9 July 2011.
50. Sotelo Monge MA, Maestre Vidal J, García Villalba LJ. Entropy-based economic denial of sustainability detection. Entropy. 2017;19(12):649.
51. Zekri M, El Kafhali S, Hanini M, Aboutabit N. Mitigating economic denial of sustainability attacks to secure cloud computing environments. Trans Mach Learn Artif Intell. 2017;5(4):473-481.
52. Arora Y, editor. Multi-Layered Attack Recognition (MLAR) Model to Protect Cloud From EDOS Attacks. 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC); 2018 15-17 Dec. 2018.
53. Bhingarkar S, Shah D. FLNL: fuzzy entropy and Lion neural learner for EDoS attack mitigation in cloud computing. Int J Model Simul Sci Comput. 2018;9(06):1850049.
54. Rustogi R, Agarwal A, Prasad A, Saurabh S, editors. Machine Learning Based Web-Traffic Analysis for Detection of Fraudulent Resource Consumption Attack in Cloud. 2019 IEEE/WIC/ACM international conference on web intelligence (WI); 2019: IEEE.
55. Courtney L, Li X, Xu R, Coffman J, editors. Data Science Techniques to Detect Fraudulent Resource Consumption in the Cloud. 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC). 2021: Publisher: IEEE.
56. Ribin Jones S, Kumar N. An efficient EDoS-DOME system in cloud computing using obfuscated IP spoofing techniques and RCDH-ENN detection techniques. Appl. Nanosci. 2021;13:1703–1
57. Lalropuia K, Khaitan V. Game theoretic modeling of economic denial of sustainability (EDoS) attack in cloud computing. Probab Eng Inf Sci. 2021;36:1–25.
58. Lalropuia K. Availability and reliability analysis of cloud computing under economic denial of sustainability (EDoS) attack: a semi-Markov approach. Cluster Comput. 2021;24(3):2177–91.
59. Shah SQA, Khan FZ, Ahmad M. Mitigating TCP SYN flooding based EDOS attack in cloud computing environment using binomial distribution in SDN. Comput Commun. 2022;182:198–211.
60. Samriya JK, Kumar S, Kumar M, Xu M, Wu H, Gill SS. Blockchain and reinforcement neural network for trusted cloud-enabled IoT network. IEEE Trans Consum Electron. 2023;70(1):2311–22.
61. Gill, S. S., Golec, M., Hu, J., Xu, M., Du, J., Wu, H.,… Uhlig, S. Edge AI:A taxonomy, systematic review and future directions. Cluster Computing, 2025;28(1):1–53.
62. Samriya JK, Kumar S, Kumar M, Wu H, Gill SS. (2024). Machine learning-based network intrusion detection optimization for cloud computing environments. IEEE Trans Consum Electron. 70(4)
63. Yadav, A. S., Kumar, S., Karetla, G. R., Cotrina-Aliaga, J. C., Arias-Gonzáles, J.L., Kumar, V.,… Tatkar, N. S. A feature extraction using probabilistic neural network and BTFSC-net model with deep learning for brain tumor classification. Journal of Imaging, 2022;9(1):10.
64. Kumar, S., Kumar, S., Ranjan, N., Tiwari, S., Kumar, T. R., Goyal, D.,… Rafsanjani,M. K. Digital watermarking-based cryptosystem for cloud resource provisioning. International Journal of Cloud Applications and Computing (IJCAC), 2022;12(1):1–20.
65. Kumar S, Samriya JK, Yadav AS, Kumar M. To improve scalability with boolean matrix using efficient gossip failure detection and consensus algorithm for PeerSim simulator in IoT environment. Int J Inform Technol. 2022;14(5):2297–307.
66. Kumar S, Srivastava S, Kumar S, Saini AK, Verma N, Kapila D. Filtering big data with an optimized hybrid algorithm for IoT-based data selection. J Intell Syst Internet Things, 2024;12(2):150–162.
67. Kumar N, Kumar S. A salp swarm optimization for dynamic resource management to improve quality of service in cloud computing and IoT environment. Int J Sens Wirel Commun Control. 2022;12(1):88–94.
68. Kumar N, Kumar S. Virtual machine placement using statistical mechanism in cloud computing environment. Int J Appl Evolutionary Comput (IJAEC). 2018;9(3):23–31.
69. Nadu T, Pradesh A. A holistic mathematical cyber security model in fog resource management in computing environments.

Publisher's note