Article

Not peer-reviewed version

Reinforcement Learning and Anomaly Detection as a Defense for Edge-Based DDoS Attacks

Tan De Xuan , Low Kang Sheng , Chua Hee Yuan , Leong Jin Hao , Lee Sze Min , Oscar Teh Yong Xin , Zhang Shao Jie , Siva Raja A/L Sindiramutty *

Posted Date: 3 September 2025

doi: 10.20944/preprints202509.0269.v1

Keywords: edge computing; DDoS mitigation; reinforcement learning; anomaly detection; cybersecurity architecture

Article

Reinforcement Learning and Anomaly Detection as a Defense for Edge-Based DDoS Attacks

Tan De Xuan, Low Kang Sheng, Chua Hee Yuan, Leong Jin Hao, Lee Sze Min, Oscar Teh Yong Xin, Zhang Shao Jie and Siva Raja A/L Sindiramutty *

Taylor's University

* Correspondence: magan.shiva91@gmail.com

Abstract

Edge computing has emerged as a solution to minimize cloud computing's latency by placing servers physically closer to the users. However, unlike centralized servers utilized in cloud computing, edge servers are limited by resources and as a decentralized network, is vulnerable to cyberattacks such as Distributed Denial of Services (DDoS) attack. This paper proposes a model based on AWS's architecture with additional components. The added components are reinforcement learning (RL) based resource management for optimizing VM instances and granting defence systems with more resource, the implementation of Akamai Prolexic alongside with an AI-driven threat detection and response system to mitigation malicious traffic before it reaches internal system. The proposed model is finically costly and difficult to implement for small-medium enterprises however, in large enterprises like AWS and Google Cloud, the performance gains several advantages such as adaptability, scalability, a proactive defence system, etc.

Keywords: edge computing; DDoS mitigation; reinforcement learning; anomaly detection; cybersecurity architecture

1. Background/Introduction and Problem

1.1. Background

In recent years, edge computing has emerged as a transformative approach to data processing, particularly in applications involving the Internet of Things (IoT), autonomous vehicles, and smart cities. Unlike traditional cloud computing, which centralises data processing in remote data centres, edge computing brings computation and storage closer to the data source—at the "edge" of the network. This proximity allows for lower latency, faster response times, reduced bandwidth usage, and improved reliability in real-time systems (Bigelow, 2021; Hussain et al., 2024).

With the rise of IoT devices, the amount of data generated at the edge is growing exponentially. Processing data locally at the edge allows organisations to respond to events in milliseconds, critical for time-sensitive operations such as emergency response systems, industrial automation, and self-driving vehicles (Powell & Smalley, 2024; Jun et al., 2024). As a result, edge computing is becoming increasingly integrated into critical infrastructure and business operations.

1.2. Problem Statement

The distributed and decentralised nature of edge computing introduces a broad range of security challenges. Unlike cloud environments, where security policies and monitoring tools are typically centralised and well-established, edge environments often consist of numerous devices deployed in uncontrolled or semi-trusted locations. These devices may have limited processing power, making it difficult to implement traditional security mechanisms such as strong encryption, intrusion detection systems, or advanced authentication protocols (Kelly, Pitropakis, Mylonas, McKeown, & Buchanan, 2021; Kiyani et al., 2024).

Among these threats, Distributed Denial of Service (DDoS) attacks have emerged as one of the most common, disruptive and technically challenging, especially when targeting protocol vulnerabilities and resource exhaustion in edge systems (Sheikh, et al., 2025; Krishnan et al., 2021). The distributed nature of edge computing exposes the system to several DDoS-related complications:

- Edge devices are highly vulnerable to protocol-level DDoS attacks, such as the HTTP/2 Rapid Reset attack, due to limited processing capacity and lack of built-in protections (Pardue & Desgats, 2023; Linqiang et al., 2024).
- Decentralized architecture lacks unified DDoS detection and response, making it difficult to coordinate mitigation across multiple edge nodes in real time.
- Attackers can exploit unprotected edge points to create entry vectors, amplifying traffic towards central cloud services and causing systemic disruption (Thomas, 2025; Manchuri et al., 2024).
- Edge environments often lack adaptive traffic filtering, allowing stealthy or low-rate DDoS attacks to bypass traditional threshold-based defences.
- Logistical constraints in updating or patching edge nodes result in prolonged exposure to known vulnerabilities exploited in modern DDoS strategies.

This report focuses on a key security issue in edge computing: the vulnerability of distributed edge systems to DDoS attacks, especially protocol-level threats like the HTTP/2 Rapid Reset.

1.3. Why Existing Solutions Fall Short

Conventional DDoS defenses—built for centralized cloud environments—struggle in edge scenarios due to limited device resources, lack of unified monitoring, and rigid rule-based filtering. These solutions often fail to detect or mitigate fast-evolving, low-rate, or distributed attacks targeting the edge (Snellman & Lamartino, 2023; Ravichandran et al., 2024).

To address these gaps, this report explores adaptive approaches such as reinforcement learning-based resource management, AI-powered anomaly detection at the edge (AETDR), and integration with platforms like Akamai Prolexic for scalable protection (Akamai Adds Behavioral DDoS Engine to App & API Protector, 2024; Riza et al., 2025).

2. Case Study Analysis

2.1. HTTP/2 Rapid Reset DDoS Attack

2.1.1. Introduction

In August and September of 2023, Cloudflare experienced a tremendous Distributed Denial of Service (DDoS) attack that exploited an unexplored vulnerability in the HTTP/2 protocol labelled CVE-2023-44487. The DDoS attack exceeded the previous record (as of the time), reaching over 201 million requests per second (RPS) (Todd, 2023). The event was similarly replicated on other well-known edge service providers such as Google Cloud and Amazon Web Services (AWS), interrupting numerous services and revealing the major deficiencies of the common HTTP/2 protocol (Pardue & Desgats, 2023; Seng et al., 2024).

2.1.2. Technical Overview of the Attack

2.1.2.1. HTTP/2 Protocol Features

HTTP/2 ensures multiplexing by offering several parallel streams in one TCP connection. RST_STREAM frame is a notable mechanism as it enables either party to reset particular streams without resetting the entire connection. This makes administration of streams flexible, it also avails the possibility of abuse (Pardue, 2024; Sindiramutty, Jhanjhi, Tan, Lau, et al., 2024).

2.1.2.2. Exploitation Method

The flaw lies in how the attacker could immediately request a new stream after cancelling a stream while not ever having received a response from the server (from the sever side, it is resetting a stream, thus *Rapid reset attack*) (Pardue & Desgats, 2023; Sindiramutty et al., 2024). This approach let the attackers to circumvent standard stream concurrency limits but did so by using disproportionate amounts of server resources, creating a very strong cost asymmetry as shown in Figure 1 Attackers took advantage of this RST_STREAM mechanism in that they:

- Opening a massive number of HTTP/2 streams.
- Cancelling them instantly with RST_STREAM frames.
- Repeating the cycle fast enough to overwhelm target servers.

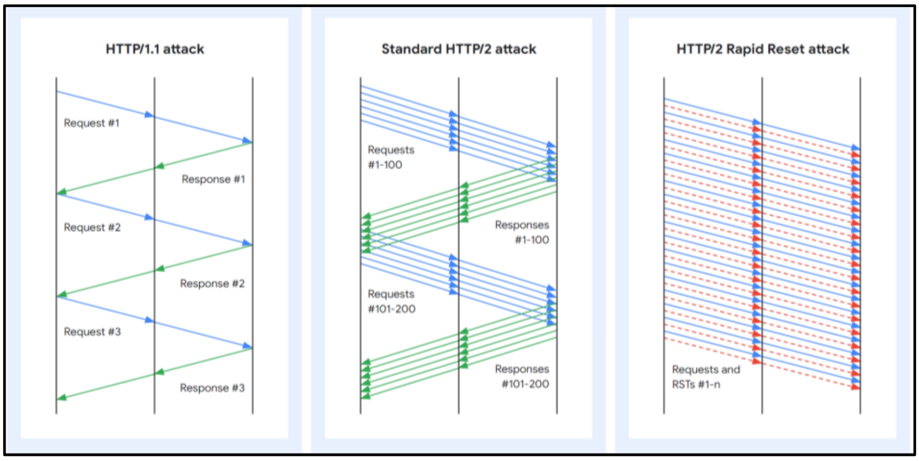

Figure 1. HTTP Protocol Attack Comparison (Losio, 2023).

2.1.2.3. Impact of the Attack

The following table summarizes the impact and details from the Rapid Reset DDoS attack on Cloudflare.

Table 1. Cloudflare Rapid Reset DDoS Attack Impact Summary (Pardue & Desgats, 2023).

Botnet Size	Approximately 20,000 nodes (as evidenced by the per-node traffic patterns in P3)
Peak Traffic	Over 201 million RPS
Affected Services	Google, AWS and Cloudflare experienced disturbances, but mitigation rapidly stabilized the systems.
Scale	The attack routed the traffic from more than a third of the web towards targets for a while.

Figure 2 reflect the violent nature during the starting of the attack and the efficiency of the countermeasures applied by Cloudflare.

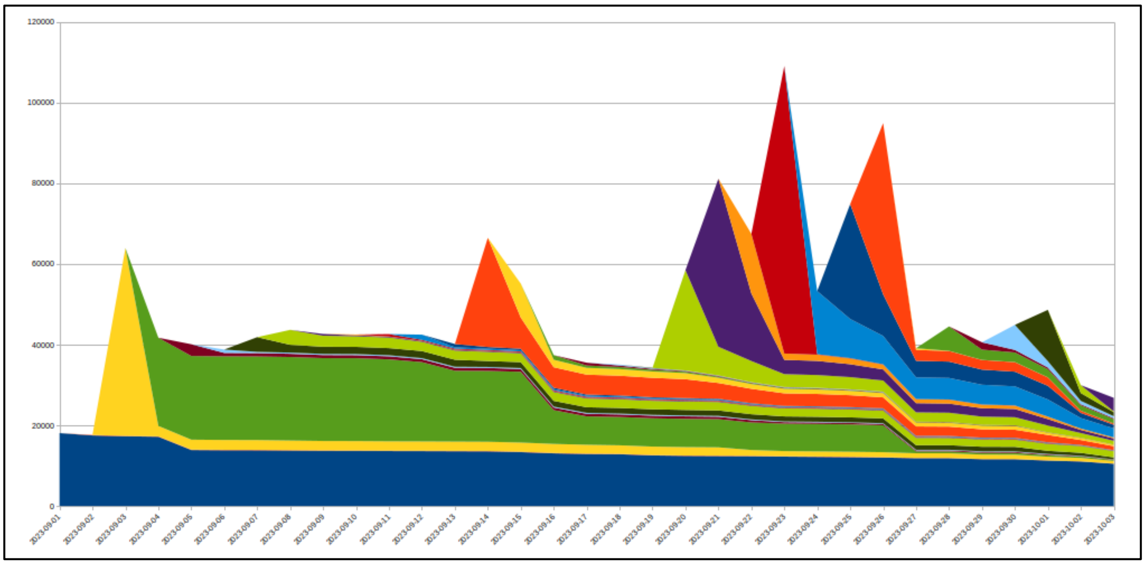

Figure 2. Churn of IPs in the Cloudflare Rapid Reset attacks during September 2023 (Pardue & Desgats, 2023).

2.1.3. Analysis of Failures

2.1.3.1. Protocol-Level Design Issues

Performance was the primary objective of HTTP/2 parallel streams, but it did not have built-in protections against abuse of control frames. The specification had no limit specified for frequency, volume of RST_STREAMs usage (Pardue, 2024).

2.1.3.2. Server Implementation Weaknesses

There were failures in validation or throttling abusive RST_STREAM usage in some server implementations (Pardue & Desgats, 2023; Sindiramutty et al., 2024). This resulted in massive resource utilization involving activities such as connection tracking, memory allocation.

2.1.3.3. Inadequate Defensive Measures

Common DDoS protections concentrate on volume and session limits rather than intelligent application-level patterns such as re-use of rapid stream reset (Pardue & Desgats, 2023).

2.1.4. Prevention and Mitigation Strategies

2.1.4.1. Protocol Enhancements

The HTTP/2 specifications need to be improved to incorporate rate-limiting mechanisms or monitoring mechanisms for the control frames (Birchard & Rath, 2023). These changes would allow early identification of misuse as well as anti-protocol level abuse. Additionally, a revised version of the protocol would enhance its perception of anomalous or malicious behaviour, especially with respect to high frequency reset frames.

2.1.4.2. Server-Side Improvements

HTTP/2 libraries like nghttp2, Apache, Nginx, could be updated to recognise and block abuse-borne patterns (Pardue & Desgats, 2023; Sindiramutty, Prabagaran, Jhanjhi, Ghazanfar, et al., 2024). This involves creating thresholds to monitor abnormal RST_STREAM behaviour thus enabling a server to perform preventive measures before services are interrupted. Such server-level upgrades are critical in enhancing defences against exploitation of protocols.

2.1.4.3. Application and Network-Level Defenses

On the application and network level, it is very necessary to enable advanced analytics that can identify patterns of attacks in real time (Pardue & Desgats, 2023; Sindiramutty et al., 2024). Anomaly-based detection should feed into throttling mechanisms at the connection level to contain the effect of bad traffic. In addition, establishing Layer 7 (application-level) rate control and perpetual traffic monitoring will assist in accomplishing strong and active defensive measures against evolving DDoS attacks.

2.2. Case 2: CLDAP Reflection/Amplification DDoS Attack on AWS

2.2.1. Introduction

In February 2020, AWS experienced a huge DDoS attack with a peak bandwidth of 2.3 Tbps, despite that, AWS managed to keep its services running. The use Connection-less Lightweight Directory Access Protocol (CLDAP) reflection and amplification technique made this an exceptional DDoS attack that garnered public notice (Boyaci, 2024; Sindiramutty, Tan, & Wei, 2024). As CLDAP relies on UDP and responds in large amounts to simple queries, it became exceptionally susceptible to reflection and amplification attacks.

2.2.2. Technical Overview of the Attack

2.2.2.1. CLDAP & Exploit Method

The CLDAP is typically used to access and query Microsoft Active Directory servers. CLDAP exploitation allows the attackers to insert the victim's IP address on the CLDAP requests (spoofing), causing large amounts of unwanted traffic to the victim while overloading the CLDAP destination server.

The attacker sent small queries to publicly accessible CLDAP servers using false information that made it look like the victim was sending them. As a result, the servers replied with bigger packets, amplifying the traffic to the victim. Consequentially, the attacks bandwidth reached a maximum of 2.3 Tbps (Nicholson, 2020; Waheed et al., 2024). Despite how big the traffic was, AWS successfully handled the attack without interrupting customer services (Labs, 2022).

Taken from AWS Shield threat report, Figure 3 illustrate the traffic peaks corresponding to the bandwidth peaks of 2.3 Tbps during the attack. Additionally, this showcases AWS's ability to swiftly respond to cyberattacks.

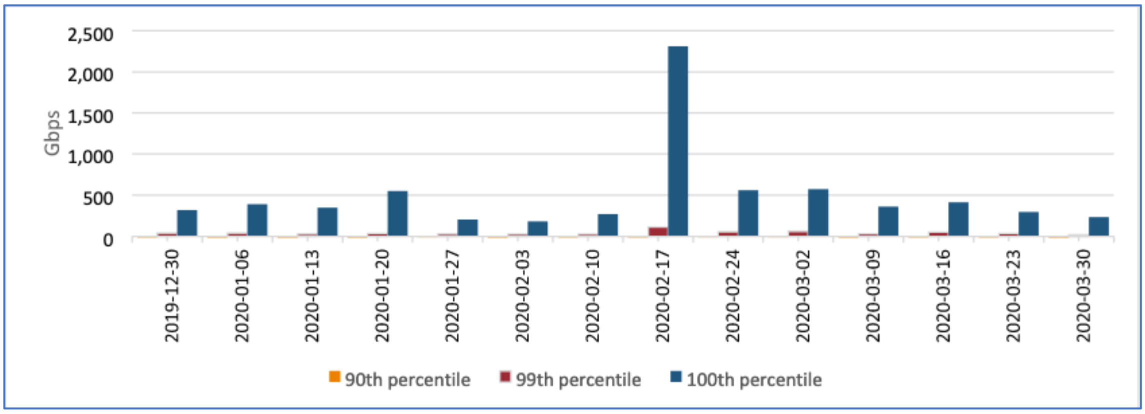

Figure 3. AWS Volumetric events for resources during first quarter of 2020.

2.2.3. Analysis of Failures

Publicly accessible CLDAP servers lacks proper configurations, allowing any individual on the internet to send queries to them (Olzak, 2022; Weiqi et al., 2024). These servers were not set up to authenticate or monitor connections. This is evident in Figure 4 where a staggering 48% of exploited reflectors were under 3 months old, with the most vulnerable group (36%) being servers aged 1 week to 3 months. This high churn rate of short-lived, misconfigured servers gave attackers a constantly renewable pool of amplification tools.

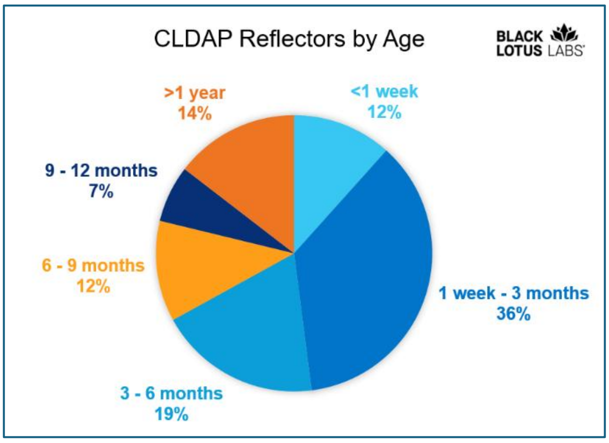

Figure 4. Age Distribution of CLDAP reflectors.

Without the use of IP spoofing protection, the attackers could easily impersonate other online users. The design of the CLDAP protocol does not require verification of sources and it allowed up to 50 times the original query / traffic (Boyaci, 2024), enabling malicious actors to carry out attacks with minimum effort. Many companies had CLDAP traffic available without the proper protection, log-keeping or updates (Labs, 2022; Wen et al., 2023). Due to problems in access management, network design and device setup, the attack was able to cause much more damage.

2.2.4. Prevention and Mitigation Strategies

There are several approaches you can use to defend against CLDAP reflection and amplification attacks.

Table 2. CLDAP preventive measures.

Filtering Packets	Filtering forged IP addresses is possible by configuring tools and settings according to BCP 38
Configurations and updates	Network administrators can restrict or block access to CLDAP services (UDP port 389) (What is a CLDAP reflection DDoS Attack?, n.d.) from the internet, as it is usually only needed inside the local network. Updating systems and monitoring for unregular activity can prevent exploitation from taking place.
External Tools	AWS Shield and Cloudflare can be used to protect against DDoS attacks and ensure your network does not go down due to large-scale traffic surges

doi:10.20944/preprints202509.0269.v1

3. Proposed Secure System

3.1. AWS Architecture enhancement proposal

The following architectural diagram, retrieved from the AWS white paper, illustrates AWS's proposed architectural system for optimal DDoS resilience.

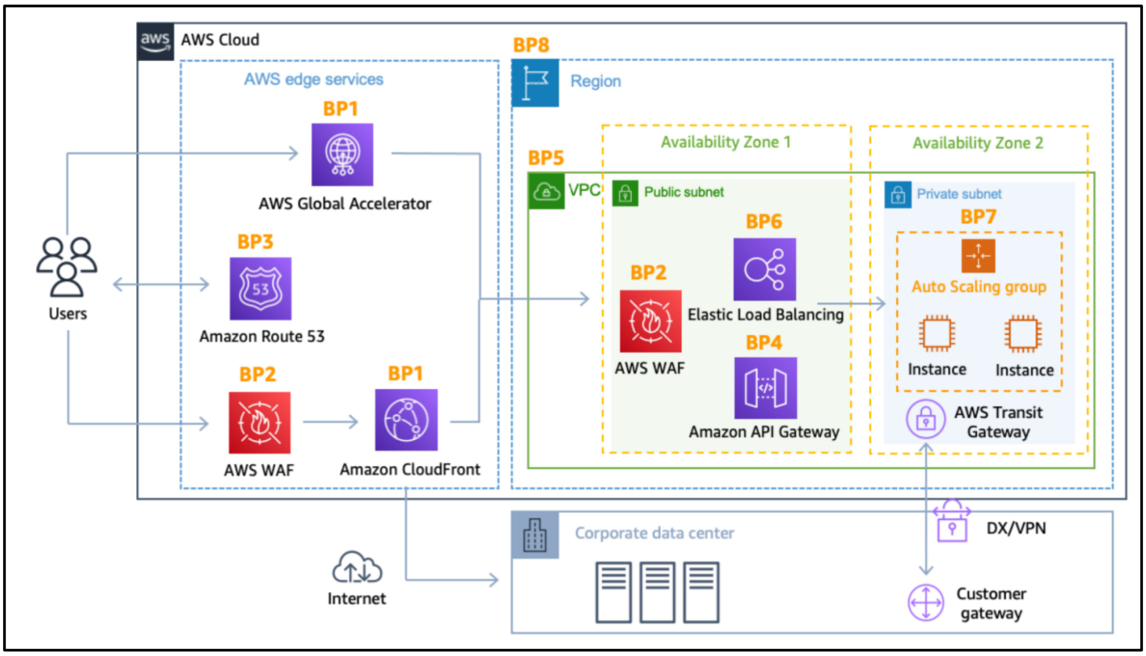

Figure 5. Architectural diagram by AWS.

The proposed system attaches three additional parts to the AWS architecture diagram to improve the system's DDoS resiliency. A *reinforced learning model* has been implemented to optimize edge server resources; *Akamai* and an *AI-Driven Edge Threat Detection and Response System* are implemented for filtering traffic.

3.2. RL-based Resource Management System for DDoS Resiliency

The architectural diagram, shown in Figure 6, included auto-scaling for resource management, which could scale up both horizontally and vertically depending on the users' specifications.

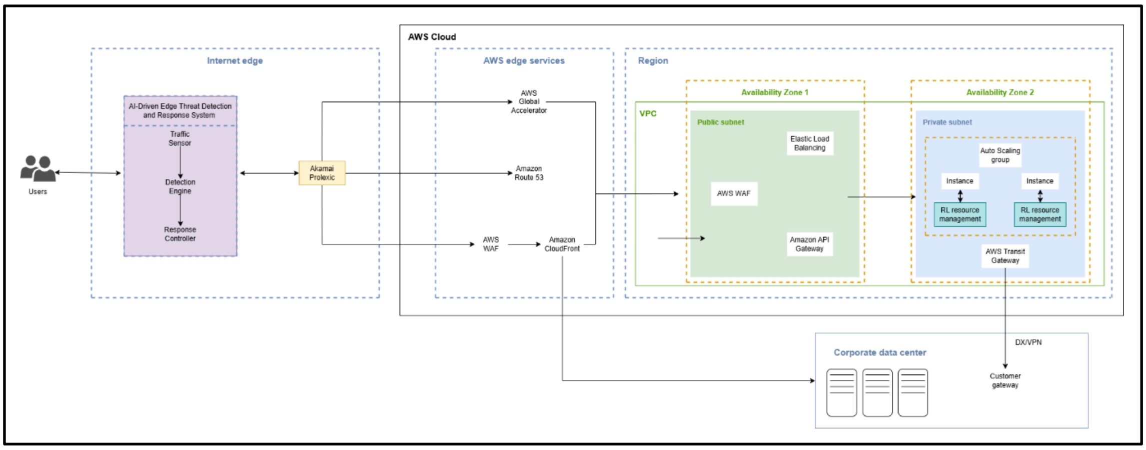

Figure 6. Proposed System Architecture diagram.

The use of machine learning models has been proposed as a solution for resource management. For instance, the work of K.Surya and V. Mary Anita Rajam (2022) attempted to predict resource contention to reduce edge server overloading, thus aiding resource availability. Another proposed solution for resource management is the work of Saifur Rahman and his fellow peers (2024), where an algorithm encourages edge devices to provide computational resources and offload tasks to other mobile edge servers when the current one is overloaded.

Reinforced Learning (RL), a subclass of machine learning, allows learning and optimisation through various interactions and feedback via the environment (Taleb, Benzaïd, Addad, & Samdanis, 2023; Xun et al., 2025). An RL-based resource management system focuses on minimising resource usage, increasing the resources available to tackle spikes in traffic. This provides DDoS mitigation services, such as AWS CloudFront, with additional resources to utilise.

3.2.1. Technical Details

3.2.1.1. Algorithms

The Dyna-Q algorithm would be utilised to optimise resources. AWS offers various services, including Software as a Service and hosting user applications on different operating systems. The main objective is to optimise the ratio of resources used and performance to ensure resources are available to receive the initial stages of a DDoS Attack. Dyna-Q is a model-based RL; thus, a well-designed model with a strong reward function must be provided for an accurate optimisation plan. The following is a sample of a dense reward function (Mukeshreddy, 2024; Ying et al., 2024):

– 0.5 for increased RAM (500 MB)
+ 0.7 for decreased RAM (500 MB)
– 0.6 for increased GPU (1%)
+ 0.8 for decreased GPU (1%)
+ 1 for increased performance (0.1 GHz)
– 1.1 for decreased performance (0.1 GHz)

While a model requires extensive resources for development, it hastens the learning process of the RL model (Rivlin, 2019). Deep Q network is an alternative that could be used in the initial testing phase. It is a model-free RL, thus eliminating the requirement of a model. However, a well-defined reward function would still be crucial to the RL (Amin, 2024).

3.2.1.2. Framework

Each AWS instance specialises in different elements, such as computing (E2C C8g instances) or memory-oriented (E2C R8g instances) instances (Amazon EC2 Instance types, n.d.). The instances host various user applications with distinct resource requirements and factors, including the operating system and the application's features. Thus, the RL must have access to each instance's resources to address their unique condition, preferably directly integrated with the instance part of the AWS framework, as shown in Figure xx. The use of relevant API's including the Amazon EC2 API, to retrieve the instance's information would be a possible alternative to direct integration. Similarly, the most optimal configuration would be set on the instance through APIs.

3.2.1.3. Protocol

AWS Gateway API service allows the creation of APIs, including HTTP(S) REST protocols for communication to services like AWS Lambda, Amazon EC2, and Amazon DynamoDB (What is Amazon API Gateway?, n.d; Ahmed et al., 2022). To factor in simplicity, similar APIs would be utilised by the RL to perform 'actions' to the 'environment' in order to receive 'feedbacks/rewards' where actions = change resource values, environment = the current instance, and feedbacks/rewards = performance changes defined in the reward function. *This only applies if the RL is not directly integrated with the instances to capture data close to the source.*

3.2.2. Upside of RL-based Resource Management System

The benefit provided by the mentioned enchantment is increased resources to be utilised by DDoS mitigation services. Although additional resources could be achieved through additional installations of hardware servers, this software solution would be effective on all current and future servers in service. Thus increasing the cost efficiency and DDoS resiliency of AWS. Other benefits would be efficient use of memory on demanding tasks (Next generation dynamic resource management, 2025; Attaullah et al., 2022) and improved flexibility of resource allocation (Signh, 2024; Azeem et al., 2021).

3.3. Akamai Prolexic

Akamai Prolexic uses a multi-layered algorithmic approach to neutralise threats like CLDAP reflection attacks. Its core engine leverages behavioural anomaly detection, which combines unsupervised machine learning and entropy analysis to detect deviations in UDP traffic (Ahmed & Bella, 2024). For protocol-specific threats such as CLDAP amplification, it monitors packet size asymmetry (>70:1 request/response ratios) and source IP spoofing density (What is a CLDAP reflection DDoS Attack?, n.d.). This enables near real-time signature generation to block attack vectors within seconds which significantly reduces false positives compared to static threshold-based systems (Akamai Adds Behavioral DDoS Engine to App & API Protector, 2024). A supplementary correlation engine may apply graph-based techniques such as Label Propagation to link attack sources and spoofed IPs (Zuckerman, Sakazi, Ozery, & Goren, 2023; Brohi et al., 2020, Aldughayfiq, 2023). Not only that, it also auto-generates ACL rules for mitigation (Prolexic, n.d.).

3.3.1. Technical Implementation

3.3.1.1. Algorithm

In theory, during an attack, Akamai's ML engine flags anomalies. BGP Flowspec will immediately reroute traffic to the nearest scrubbing centre where protocol-specific rules discard malicious payloads. Legitimate traffic is encrypted and delivered to AWS via Direct Connect or VPN. Post incident stage, Akamai's Attack Visualizer can use graph neural networks to trace spoofed IPs to botnets, which provide actionable indicators of compromise for AWS security groups. This integration functions as an externalized filtering later to ensure attack traffic never consumes AWS bandwidth.

3.3.1.2. Frameworks and Protocols

The Cloud Control Fabric (CCF) coordinates mitigation across Akamai's global scrubbing centers. It can integrate with AWS via BGP Flowspec (RFC 8955) to divert attack traffic to Akamai's edge where malicious packets are scrubbed (Loibl, Hares, Raszuk, McPherson, & Bacher, 2020; Hanif et al., 2022). Clean traffic is then routed back to AWS through encrypted GRE/IPsec tunnels (AES-256) terminating at ALB or CloudFront (What is AWS Direct Connect?, n.d.). This process is enforced through a Zero Trust architecture where mutual TLS and cryptographic tokens validate all traffic entering AWS (Rose, Borchert, Mitchell, & Connelly, 2020; Humayun et al., 2022). Only requests from Akamai's Anycast IP ranges are permitted in order to eliminate direct-to-origin attacks.

3.3.2. Why use Akamai?

Akamai leverages its purpose-built scrubbing network to mitigate attacks within 20 Tbps of volumetric capacity as stated in its official reports (Akamai Announces Next Generation DDoS Defense Platform, 2022). By contrast, AWS's largest publicly confirmed mitigated attack reached 2.3 Tbps during the 2018 Github incident (Cimpanu, 2020; Jabeen et al., 2023).

For network layer (L3/L4 threats), Akamai advertises sub-3-second mitigation times owing to edge level traffic diversion while AWS Shield Advanced, which operates on shared cloud infrastructure, targets rapid response but provides no public SLA for specific mitigation timeframes (Best DDoS Protection Services: Top 8 Solutions in 2025, n.d.; Khan et al., 2021).

The mitigation time for AWS may potentially extend to minutes during novel or complex attacks. Critically, Akamai's fixed fee model eliminates cost variability during attacks while AWS imposes bandwidth fees ($0.01 - $0.15/GB) for mitigated traffic alongside scaled resource costs, which creates potential expense volatility during sustained incidents (AWS Shield Pricing, n.d.).

In layman's terms, Akamai is capable of:

- Akamai's global scrubbing infrastructure operates at an internet edge that can absorb attacks before they impact AWS environments (Kaneko, 2023).
- Second, machine learning precisely isolates malicious traffic to avoid AWS's collateral damage during volumetric floods (Ahmed & Bella, 2024).
- Third, Akamai is certified for NIST 800-53 and ISO 27001, which are crucial for finance or healthcare workloads (Information Security Compliance, n.d.).
- Finally, the fixed pricing eliminates variable AWS costs, especially during multi-terabit attacks.

3.4. AI-Driven Edge Threat Detection and Response System (AETDR)

The architecture of AETDR suggests an expert security component in between the user and the point of entry to the cloud. Traditional cloud-based DDoS defence solutions such as AWS Shield and web application firewalls (WAFs) rely on central architectures. Effective in most cases, these central solutions might introduce latency when acting against fast-evolving or evasive attacks (Yaegashi, Hisano, & Nakayama, 2020; Muzafar & Jhanjhi, 2019).

To counter this weakness, we propose an AI-driven, edge-based detection and response system that operates independently of specific cloud vendors. With the placement of detection logic at the edge (such as CDN points of presence or reverse proxies), AETDR can identify and disrupt out-of-the-norm traffic patterns before they reach the cloud infrastructure, naturally decreasing response latency.

3.4.1. Technical Implementation

3.4.1.1. Algorithms

The AETDR system employs anomaly-based detection using unsupervised machine learning algorithms. Instead of relying on signature-based techniques, AETDR learns baseline traffic behaviours and flags deviations that may indicate slow-rate or stealthy DDoS attacks.

Key algorithms include:

- Isolation Forests: Efficient for identifying outliers in high-dimensional traffic features (Ripan, Islam, Alqahtani, & Sarker, 2022).
- Autoencoders: Neural models trained to reconstruct input traffic; anomalies cause higher reconstruction error (Kashyap, 2024; Muzammal et al., 2020).

These models are suitable for edge deployment due to their lightweight design and ability to detect uncommon and complex traffic behaviours / correlations without labelled data (Bergmann & Stryker, 2023). They analyse features such as connection duration, request rates, header entropy, and payload sizes.

Future implementations could incorporate Deep Q-Learning (DQL) to dynamically adapt mitigation actions based on observed traffic environments. However, the current version focuses on robust detection via anomaly modelling.

3.4.1.2. Framework

The AETDR system consists of three containerised components (Uddin, Kumar, & Chamola, 2023):

1. Traffic Sensor: Utilizes tools like Zeek or Suricata to collect network metadata at the edge.

2. Anomaly Detection Engine: Applies trained machine learning models (developed in Scikit-learn or PyTorch) to classify incoming traffic.
3. Response Controller: Integrates with proxies (e.g., NGINX), load balancers, or firewalls to enforce real-time mitigation actions.

Each component is deployable in lightweight virtual machines, CDN edge nodes, or reverse proxies. This modularity and vendor-agnostic design enable proactive filtering before attacks reach cloud resources.

3.4.1.3. Protocol

The system's components communicate via REST APIs or gRPC for low-latency, high-performance interactions (Kamiński, et al., 2022; Saeed et al., 2022). A typical detection flow includes (Liu & Lang, 2019):

- Step 1: Traffic Sensor extracts and transmits features to the Detection Engine.
- Step 2: Detection Engine classifies traffic as benign or suspicious.
- Step 3: Response Controller updates access control lists to block or verify users.

In passive monitoring scenarios (e.g., when inline deployment isn't feasible), mirrored traffic is analysed without impacting production services. An optional Admin API allows for model updates, system logs access, and retraining workflows.

Future implementations could incorporate Deep Q-Learning (DQL) to dynamically adapt mitigation actions based on observed traffic environments. However, the current version focuses on robust detection via anomaly modelling.

3.4.2. Key Strengths of AETDR

The AI-Powered Edge Threat Detection and Response System (AETDR) offers a low-latency, proactive defence with anomaly detection placed at the network edge. It employs unsupervised machine learning techniques like Isolation Forests and Autoencoders to identify new and complex attacks without signatures. With its modular, vendor-independent architecture, it is highly deployable across multiple environments. Overall, AETDR enhances cloud security by blocking threats in early stages and adapting to shifting patterns of attacks.

4. Implementation Challenges & Feasibility

4.1. Key Implementation Challenges

4.1.1. Dependency on Reward Function (RL)

The effectiveness of RL depends heavily on the design of the reward function. An inadequately constructed reward system may hinder the agent's capacity to acquire desired behaviors resulting in less-than-ideal performance.

For instance, if the reward system is not in line with the objectives of the task, the agent might learn to take advantage of weaknesses instead of achieving the desired results, this misalignment is referred to as the 'reward hacking' issue (DigitalDefynd, 2025). This dependence highlights the importance of carefully designing reward mechanisms.

Other than that, ensuring a balanced reward structure that optimizes for both resource efficiency and service availability requires extensive testing which is difficult to simulate accurately in volatile real-world scenarios like DDoS attacks (Yang, Zheng, Li, Tomizuka, & Liu, 2024; Gill et al., 2022).

4.1.2. Data and computational requirements (RL)

For reinforcement learning to be effective, a thorough engagement with the environment is crucial. To improve their strategies, RL algorithms require a large amount of data about the results of actions (Acharya, 2023). However, gathering this data can be expensive in terms of both time and resources. This dependence typically results in increased processing costs and the need for sophisticated simulation settings particularly in complex real-world scenarios.

Furthermore, RL models frequently rely on GPU powered infrastructure which incurs costly hardware and maintenance costs (What are the Reinforcement Learning Advantages and Disadvantages?, 2024).

4.1.3. Data Quality and Feature Extraction (AETDR)

Good network characteristics (e.g, entropy, connection duration) are essential for anomaly detection to work. Improper configuration can lower the detection capabilities of sensors such as Zeek or Suricata (Buckman, 2025), while encryption will introduce an increased height of complication.

This increases the risk of undetected threats, or false alarms that may interfere with legitimate traffic. Maintaining effective feature extraction at the edge, where data may be noisy or limited, requires continual tuning and oversight.

4.1.4. False Positives and Service Denial (AETDR)

Anomaly based models, particularly those deployed at the edge must strike a balance between preventing malicious activity and enabling authentic traffic. Misclassifications can result in denying service for legitimate users, particularly in industries where constant availability is crucial. Continuous tuning and contextual rule validation are required to minimize collateral damage.

4.1.5. Integration Complexity and Traffic Flow Management (Akamai Prolexic)

Deploying Akamai Prolexic requires extensive integration with cloud infrastructure such as AWS. This comprises routing inbound traffic to Akamai's scrubbing centers using BGP Flowspec, decrypting and filtering traffic, and returning clean data via encrypted GRE/IPsec tunnels.

Each component, flowspec, encryption, and tunnel management must be properly set to avoid creating new vulnerabilities or generating availability concerns. Furthermore, coordinating this arrangement across multi clouds or hybrid systems adds complexity (Loibl, Hares, Raszuk, McPherson, & Bacher, 2020).

4.1.6. Signature Accuracy and ML Reliability (Akamai Prolexic)

Akamai's detection system uses machine learning and behaviour-based algorithms to detect anomalies quickly, but these models must be continuously tuned to balance sensitivity and specificity. Too aggressive a threshold may result in false positives, prohibiting legitimate users, while a lenient configuration may allow sophisticated attacks to pass. Furthermore, attackers can test and adapt their behaviour to match legitimate patterns, thereby reducing detection efficacy over time (Zuckerman, Sakazi, Ozery, & Goren, 2023).

4.2. System Limitations

4.2.1. Cost (RL)

RL systems require very high-performance computing, especially when using GPU instances (Sagar, 2021). These costs can be difficult to justify for systems with low frequency of DDoS attacks or small-scale applications. Beyond infrastructure, there are also costs associated with training, model tuning, and long-term maintenance.

4.2.2. Cost (Akamai Prolexic)

Although Akamai uses a fixed-fee pricing model to control cost volatility during attacks, its services may still be prohibitively expensive for smaller organizations. The integration and licensing fees, especially for sustained multi-terabit attack protection, may limit accessibility outside enterprise-scale deployments. For some clients, alternative solutions with flexible pricing may be more appropriate (Ahmed & Bella, 2024).

4.2.3. Scalability (AETDR)

Scaling containerized detection systems across a large network requires orchestration and standardization, which can be very difficult to manage. Any misalignment in these deployments could create blind spots. Furthermore, retraining and updating distributed models to retain consistent detection capacity raises the operational burden.

4.2.4. Vendor Lock-in (Akamai Prolexic)

Though designed to be vendor agnostic, practical use of Akamai Prolexic often leads to tight integration with Akamai infrastructure and operational models. Switching to another provider may need rethinking routing rules, revalidating tunnels, and retraining teams, all of which discourage mobility and could conflict with long-term strategic goals focused on cloud independence.

4.3. Ethical considerations

4.3.1. Accountability (RL)

The autonomy of RL based systems introduces uncertainty regarding responsibility when things go wrong. Tracing accountability becomes difficult when service disruptions, misallocations or data loss occur as a result of poor RL decisions. To facilitate post-incident inquiry, organizations must create monitoring and logging tools that provide transparency into the RL agent’s logic.

4.3.2. Privacy and Data Handling (AETDR)

Privacy issues arise when communication is analyzed at the edge, especially when payload inspection is involved. Compliance with data protection laws, such as the General Data Protection Regulation (GDPR), a set of rules placing limitation on what organizations can do with personal data (Burgess, 2020) as well as the Personal Data Protection Act of 2010, is crucial. To achieve these standards and provide effective threat detection, systems must anonymize or encrypt data.

4.3.3. Transparency and External Routing Trust (Akamai Prolexic)

Routing traffic to third-party scrubbing centers means relinquishing partial control of inbound data paths. Even if encryption is enforced, organizations in regulated industries may still be required to explain why sensitive data is passing through external networks. Compliance with security certifications such as ISO 27001 is helpful but internal audits and controls are still required to retain stakeholder trust.

4.3.4. Certification and Regulatory Alignment (Akamai Prolexic)

Akamai’s conformance with international standards makes it suitable for regulated businesses. Enterprises are still responsible for ensuring that third party services adhere to internal compliance frameworks. This involves integrating audit logs, assuring end to end encryption and implementing fallback policies just in case Akamai services are unavailable or out of policy scope.

5. Evaluation & Discussion

This section will compare the proposed system with two widely used DDoS mitigation systems, the edge-based autonomous defense system used by Cloudflare and the HTTP/2 Rapid Reset attack response system proposed by Google Cloud.

The comparison is approximately how the reinforcement learning (RL)-based resource management, anomaly-based edge threat detection (AETDR) and instance-level resource control using APIs enhance or supplement the solutions that are already deployed.

5.1. *Cloudflare General DDoS Defence Architecture*

In an attempt to limit DDoS attacks Cloudflare maintains a global distributed edge network which uses the dosd daemon and low-level kernel filters (e.g., XDP, eBPF) to detect and prevent attacks at or near the source (Yoachimik, 2021), architecture seen in Figure 7. It is a very fast, scale, and efficient architecture and the attack traffic is absorbed before it reaches the origin.

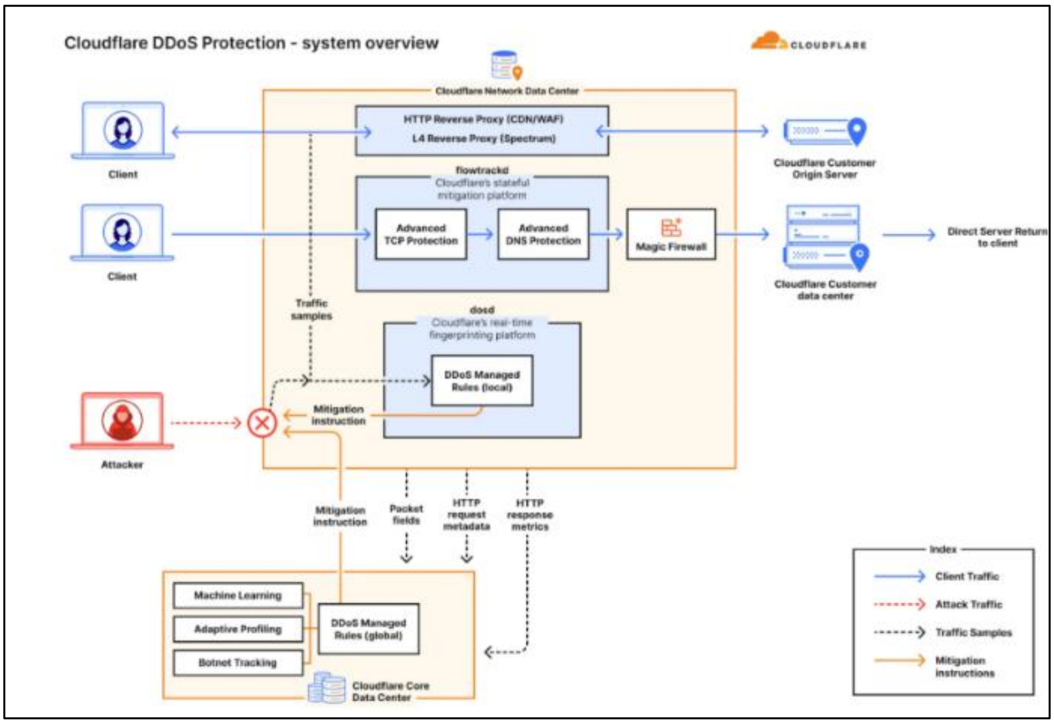
Cloudflare DDoS Protection - system overview
CLOUDFLARE
Cloudflare Network Data Center
HTTP Reverse Proxy (CDN/WAF)
L4 Reverse Proxy (Spectrum)
Client
Client
Cloudflare Customer Origin Server
flowtrackd
Cloudflare's stateful mitigation platform
Advanced TCP Protection
Advanced DNS Protection
Magic Firewall
Cloudflare Customer data center
Direct Server Return to client
Traffic samples
dosd
Cloudflare's real-time fingerprinting platform
DDoS Managed Rules (local)
Mitigation instruction
Attacker
Mitigation instruction
Packet fields
HTTP request metadata
HTTP response metrics
Machine Learning
Adaptive Profiling
Botnet Tracking
DDoS Managed Rules (global)
Cloudflare Core Data Center
Index
Client Traffic
Attack Traffic
Traffic Samples
Mitigation instructions

Figure 7. Cloudflare DDoS Protection Architecture diagram.

5.1.1. RL-based resource management

The scaling would complement the current static provisioning that Cloudflare currently displays, whereby the system can automatically increase the amount of compute or memory resources as the attack pressure increases.

Even though the world capacity of Cloudflare is huge, it cannot always optimize the resource-to-performance trade-offs in real time. The gap can be closed by RL as it learns where and when to add infrastructure though reward feedback. However, in the case with Cloudflare, the benefit is quite small since the traffic is already distributed and absorbed at scale (Soam, 2024).

5.1.2. AETDR (anomaly-based edge detection)

It is an improvement of the signature-based or rate-threshold rules since it can detect hidden or unknown traffic patterns through the application of unsupervised learning functions. This can complement the detection system at Cloudflare in detecting changing or low-and-slow attacks, which can evade fixed heuristics. AETDR comes with a degree of proactive intelligence at the cost of a little bit more computational complexity and is particularly strong against behaviours that have never been seen before.

5.1.3. Akamai Prolexic

The scrubbing provided by Cloudflare is entirely contained within their own network, however in the proposed solution, Akamai Prolexic is an external, high-capacity DDoS scrubbing layer. Prolexic employs worldwide network and BGP Flowspec to redirect malicious traffic to its scrubbing centers, filter it using machine learning, and then returning clean traffic over GRE/IPsec tunnels. It is a design that offloads volumetric threats prior to hitting the cloud layer and offers predictable SLA-backed mitigation speeds and pricing. Prolexic offers more flexibility than in-house scrubbing provided by Cloudflare, as well as compliance certifications and better alignment to cloud-native deployments, such as AWS or hybrid deployments.

5.2. Google Cloud Countermeasure against HHTP/2 Fast Reset Attack

In 2023, Google Cloud stopped an HTTP/2-based distributed denial of service (DDoS) attack record using protocol-level rate limiting, connection behaviours analysis and infrastructure isolation. They employed a system, which relied on per-stream constraints, distributed behavioural filters proxied and load balancers, shown in Figure 8 (Sharma, 2023; Kobialka, 2023).

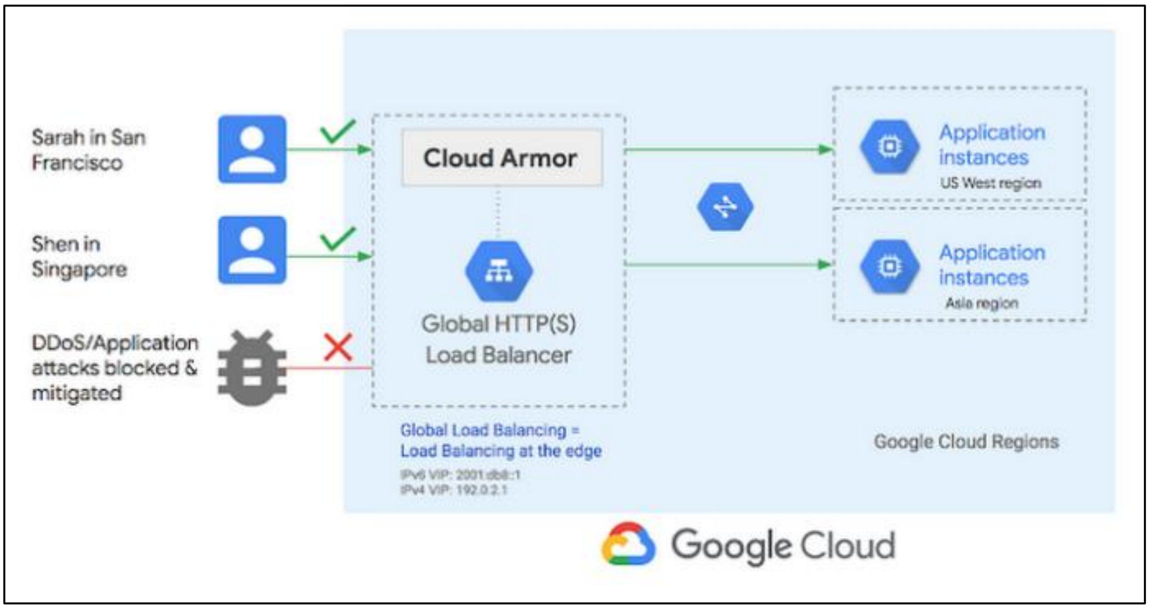

Figure 8. Google Cloud Armor.

5.2.1. RL-based resource management

Adaptive resource allocation in the backend as proposed by RL is the resource scaling, but Google offers a fixed infrastructure that is pre-allocated. Dynamic scaling of instances during the attack peaks can minimize the cost and guarantee the highest resilience of the system. This can be beneficially employed particularly on multi-tenant or resource-constrained environments.

5.2.2. AETDR (anomaly-based edge detection)

AETDR is an improvement as it can detect abnormalities not stipulated in the protocols. As opposed to the traditional stream reset limits, AETDR learns traffic baselines and notifies on deviations per-flow in real time, offering more protection against zero-day or low-and-slow attack vectors. Effective when inserted at CDN or reverse proxy nodes close to the origin.

5.2.3. Akamai Prolexic

Google uses self-edge and self-protocol stack mitigation, in other words, Google uses its own edge severs and protocol stack rather than relying on external tools. Conversely, the proposed system uses Akamai Prolexic 20+ Tbps scrubbing network that soaks up the attack traffic outside the origin infrastructure through BGP routing (Gold, 2022; Gal, 2021).

The scrubbing layer enforces behavioural and entropy-based filters, dynamically creates ACL rules and is securely integrated with AWS via encrypted tunnels. This Offloads mitigation burden, prevents bandwidth usage at application layer, and enhances compliance in sectors which mandate certified filtering (i.e. NIST 800-53, ISO 27001).

In short, Prolexic offers vendor-neutral, modular protection with predictable performance guarantees when compared to the tightly integrated mitigation offered by Google.

5.3. Advantages of the proposed Methodology

The proposed architecture has the following main advantages in comparison with the traditional DDoS protection system:

5.3.1. Adaptability and Elasticity

Although only in theory, the RL-based scaling policy offers the system to scale in a real-time basis depending on the level of attacks by making intelligent trade-offs involving performance and resource availability. This dynamic action as compared to the static provisioning leads to increased resilience and u extent of use of infrastructure resources.

5.3.2. Intelligence and Proactivity

AETDR enables the system to detect not only the signatures of attacks that are already known about but also previously unknown signatures or lower profile patterns via anomaly detection. This changes the defence mode to proactive instead of reactive whereby the threats are identified at an early phase before services degradation can be attained.

5.3.3. Vendor Independence and Modularity

The architecture is platform-agnostic and can be integrated with different cloud environments, hence it is elastic to be implemented on multi-cloud or hybrid architecture.

5.3.4. Scalability to Future Data

The learning-based system scales with data, i.e., its defences become continually stronger as it observes increasingly more attack behaviour whereas the rule-based systems become weaker over time, unless updated.

6. Conclusions

As edge computing continues to gain traction in critical infrastructure and real-time systems, its decentralized architecture introduces new security challenges—particularly its vulnerability to Distributed Denial of Service (DDoS) attacks. This report has demonstrated how traditional cloud-based security solutions are insufficient in protecting edge environments, as exemplified by the HTTP/2 Rapid Reset attack that overwhelmed major cloud providers.

To address these limitations, we proposed a multi-faceted defence strategy tailored for edge computing systems. Reinforcement learning-based resource management provides dynamic and efficient scaling to absorb traffic surges. Akamai Prolexic offers robust, cloud-integrated DDoS scrubbing capabilities with protocol-level threat detection. Meanwhile, the AI-powered Edge Threat Detection and Response (AETDR) system enables real-time, decentralized anomaly detection directly at the edge.

Together, these solutions highlight a shift toward intelligent, distributed, and adaptive security models. By deploying lightweight and proactive mechanisms closer to the attack surface, organisations can significantly enhance the resilience of their edge infrastructures against both current and emerging threats. Future work should continue to refine these approaches, integrating automated response systems and aligning them with industry standards to ensure scalable, vendor-agnostic protection for next-generation edge networks.

References

Acharya, K. (2023, Febraury 25). Challenges in Deep Reinforcement Learning. Retrieved from Medium: https://medium.com/@lotussavy/challenges-in-deep-reinforcement-learning-46ec2eaab3c2

Akamai Adds Behavioral DDoS Engine to App & API Protector. (2024, October 16). Retrieved from Akamai: https://www.akamai.com/newsroom/press-release/akamai-adds-behavioral-ddos-engine-to-app-api-protector

Akamai Announces Next Generation DDoS Defense Platform. (2022, October 25). Retrieved from Akamai: https://www.akamai.com/newsroom/press-release/akamai-announces-next-generation-ddos-defense-platform

Amazon EC2 Instance types. (n.d.). Retrieved from AWS: https://aws.amazon.com/ec2/instance-types/

AWS Shield Pricing. (n.d.). Retrieved from AWS: https://aws.amazon.com/shield/pricing/

Best DDoS Protection Services: Top 8 Solutions in 2025. (n.d.). Retrieved from Radware: https://www.radware.com/cyberpedia/ddospedia/best-ddos-protection-services-top-8-solutions-in-2025/

Boyaci, I. (2024, Feburary 5). Network-Based Cyber Attacks and Best Practices for Mitigation. Retrieved from Medium: https://medium.com/@ibrahimboyaci302/network-based-cyber-attacks-and-best-practices-for-mitigation-6f9b7dea6df1

DigitalDefynd. (2025). 10 Pros and Cons of Reinforcement Learning [2025]. Retrieved from digitaldefynd: https://digitaldefynd.com/IQ/reinforcement-learning-pros-cons/

Information Security Compliance. (n.d.). Retrieved from Akami: https://www.akamai.com/legal/compliance

Kamiński, Ł., Kozłowski, M., Sporysz, D., Wolska, K., Zaniewski, P., & Roszczyk, R. (2022). Comparative review of selected Internet communication protocols. Cornell University.

Kelly, C., Pitropakis, N., Mylonas, A., McKeown, S., & Buchanan, W. J. (2021). AComparative Analysis of Honeypots on Different Cloud Platforms. sensors.

Liu, H., & Lang, B. (2019). Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey. Applied Sciences.

Loibl, C., Hares, S., Raszuk, R., McPherson, D. R., & Bacher, M. (2020, December). RFC 8955. Retrieved from datatracker: https://datatracker.ietf.org/doc/html/rfc8955

Prolexic. (n.d.). Retrieved from Akamai: https://www.akamai.com/products/prolexic-solutions

Rahman, S., Hussain, M., Shah, S. I., Ali, Z., Kahn, M. A., Nowakowski, G., . . . Muhammad, F. (2024). Resource Management Across Edge Server in Mobile Edge Computing. IEEE Access, 181579-181589.

Ripan, R. C., Islam, M. M., Alqahtani, H., & Sarker, I. H. (2022). Effectively predicting cyber-attacks through isolation forest learning-based outlier detection. Security and Privacy Volume 5 Issue 3.

Rose, S., Borchert, O., Mitchell, S., & Connelly, S. (2020). Zero Trust Architecture NIST SP 800-207.

Sheikh, Sheikh, A. M., Islam, M. R., Habaebi, M. H., Zabidi, S. A., Najeeb, A. R., & Kabbani, A. (2025). A Survey on Edge Computing (EC) Security Challenges: Classification, Threats, and Mitigation Strategies. Future internet.

Surya, K., & Rajam, .. M. (2022). Novel Approaches for Resource Management Across Edge Servers . International Journal of Networked and Distribued Computing (2023), 20-30.

Taleb, T., Benzaïd, C., Addad, R. A., & Samdanis, K. (2023). AI/ML for beyond 5G systems: Concepts, technology enablers & solutions. Computer Networks.

Uddin, R., Kumar, S. A., & Chamola, V. (2023). Denial of services attacks in edge computing layers: Taxonomy, vulnerabilities, threats and solutions. Ad Hoc Networks.

What are the Reinforcement Learning Advantages and Disadvantages? (2024, October 18). Retrieved from Birchwood University: https://www.birchwoodu.org/reinforcement-learning-advantages-and-disadvantages/

What is a CLDAP reflection DDoS Attack? (n.d.). Retrieved from Akamai: https://www.akamai.com/glossary/what-is-a-cldap-reflection-ddos-attack

What is Amazon API Gateway? (n.d.). Retrieved from AWS: https://docs.aws.amazon.com/apigateway/latest/developerguide/welcome.html

What is AWS Direct Connect? (n.d.). Retrieved from AWS: https://docs.aws.amazon.com/directconnect/latest/UserGuide/Welcome.html

Yaegashi, R., Hisano, D., & Nakayama, Y. (2020). Light-Weight DDoS Mitigation at Network Edge with Limited Resources. Tokyo: Cornell University.

Yang, Y., Zheng, Z., Li, S. E., Tomizuka, M., & Liu, C. (2024). The Feasibility of Constrained Reinforcement Learning Algorithms: A Tutorial Study⋆. arXiv Cornell University.

Ahmed, Q. W., Garg, S., Rai, A., Ramachandran, M., Jhanjhi, N. Z., Masud, M., & Baz, M. (2022). AI-Based Resource Allocation Techniques in Wireless Sensor Internet of Things Networks in Energy Efficiency with Data Optimization. *Electronics*, *11*(13), 2071. https://doi.org/10.3390/electronics11132071

Attaullah, M., Ali, M., Almufareh, M. F., Ahmad, M., Hussain, L., Jhanjhi, N., & Humayun, M.

(2022). Initial stage COVID-19 detection system based on patients' symptoms and chest X-Ray images. *Applied Artificial Intelligence*, *36*(1). https://doi.org/10.1080/08839514.2022.2055398

Azeem, M., Ullah, A., Ashraf, H., Jhanjhi, N., Humayun, M., Aljahdali, S., & Tabbakh, T. A. (2021). FOG-Oriented secure and lightweight data aggregation in IOMT. *IEEE Access*, *9*, 111072–111082. https://doi.org/10.1109/access.2021.3101668

Aldughayfiq, B., Ashfaq, F., Jhanjhi, N. Z., & Humayun, M. (2023, April). Yolo-based deep learning model for pressure ulcer detection and classification. In *Healthcare* (Vol. 11, No. 9, p. 1222). MDPI.

Hanif, M., Ashraf, H., Jalil, Z., Jhanjhi, N. Z., Humayun, M., Saeed, S., & Almuhaideb, A. M. (2022). AI-Based wormhole attack detection techniques in wireless sensor networks. *Electronics*, *11*(15), 2324. https://doi.org/10.3390/electronics11152324

Humayun, M., Jhanjhi, N. Z., Niazi, M., Amsaad, F., & Masood, I. (2022). Securing Drug Distribution Systems from Tampering Using Blockchain. *Electronics*, *11*(8), 1195. https://doi.org/10.3390/electronics11081195

Jabeen, T., Jabeen, I., Ashraf, H., Jhanjhi, N. Z., Yassine, A., & Hossain, M. S. (2023). An intelligent healthcare system using IoT in wireless sensor network. *Sensors*, *23*(11), 5055. https://doi.org/10.3390/s23115055

Khan, N. A., Jhanjhi, N. Z., Brohi, S. N., Almazroi, A. A., & Almazroi, A. A. (2021). A secure communication protocol for unmanned aerial vehicles. *Computers, Materials & Continua/Computers, Materials & Continua (Print)*, *70*(1), 601–618. https://doi.org/10.32604/cmc.2022.019419

Muzafar, S., & Jhanjhi, N. Z. (2019). Success stories of ICT implementation in Saudi Arabia. In *Advances in electronic government, digital divide, and regional development book series* (pp. 151–163). https://doi.org/10.4018/978-1-7998-1851-9.ch008

Muzammal, S. M., Murugesan, R. K., Jhanjhi, N. Z., & Jung, L. T. (2020). SMTrust: Proposing Trust-Based Secure Routing Protocol for RPL Attacks for IoT Applications. *2020 International Conference on Computational Intelligence (ICCI)*, 305–310. https://doi.org/10.1109/icci51257.2020.9247818

Gill, S. H., Razzaq, M. A., Ahmad, M., Almansour, F. M., Haq, I. U., Jhanjhi, N. Z., ... & Masud, M. (2022). Security and privacy aspects of cloud computing: a smart campus case study. *Intelligent Automation & Soft Computing*, *31*(1), 117-128.

Hussain, K., Rahmatyar, A. R., Riskhan, B., Sheikh, M. a. U., & Sindiramutty, S. R. (2024). Threats and Vulnerabilities of Wireless Networks in the Internet of Things (IoT). *2024 IEEE 1st Karachi Section Humanitarian Technology Conference (KHI-HTC)*, *2*, 1–8. https://doi.org/10.1109/khi-htc60760.2024.10482197

Jhanjhi, N. (2024). Comparative analysis of frequent pattern mining algorithms on healthcare data. In 2024 IEEE 9th International Conference on Engineering Technologies and Applied Sciences (ICETAS) (pp. 1-10). IEEE. https://doi.org/10.1109/ICETAS62372.2024.11119839

Jhanjhi, N. Z. (2025). Investigating the influence of loss functions on the performance and interpretability of machine learning models. In S. Pal & Á. Rocha (Eds.), Proceedings of 4th International Conference on Mathematical Modeling and Computational Science. ICMMCS 2025. Lecture Notes in Networks and Systems, vol 1399 (pp. 100-110). Springer. https://doi.org/10.1007/978-3-031-91005-0_43

Jun, A. Y. M., Jinu, B. A., Seng, L. K., Maharaiq, M. H. F. B. Z., Khongsuwan, W., Junn, B. T. K., Hao, A. a. W., & Sindiramutty, S. R. (2024). Exploring the Impact of Crypto-Ransomware on Critical Industries: Case Studies and Solutions. *Preprint.org*. https://doi.org/10.20944/preprints202409.1325.v1

Kiyani, F. F., Hamid, B., Humayun, M., Sindiramutty, S. R. a. L., & Chowdhury, S. (2024). Discovery of Influential Publications Using Research Article's Usage Context. *2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC)*, 1–7. https://doi.org/10.1109/etncc63262.2024.10767576

Krishnan, S., Thangaveloo, R., Rahman, S. B. A., & Sindiramutty, S. R. (2021). Smart Ambulance Traffic Control system. *Trends in Undergraduate Research*, *4*(1), c28-34. https://doi.org/10.33736/tur.2831.2021

Linqiang, Y., Sindiramutty, S. R. a. L., Ashraf, H., Muzammal, S. M., Balakrishnan, S. a. P., Gupta, S., & Kavita, N. (2024). Intelligent Household Waste Classification System Based on Machine Learning. *2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC)*, 760–768. https://doi.org/10.1109/etncc63262.2024.10767563

Manchuri, A., Kakera, A., Saleh, A., & Raja, S. (2024). pplication of Supervised Machine Learning Models in Biodiesel Production Research - A Short Review. *Borneo Journal of Sciences and Technology*. https://doi.org/10.35370/bjost.2024.6.1-10

Ravichandran, N., Tewaraja, T., Rajasegaran, V., Kumar, S. S., Gunasekar, S. K. L., & Sindiramutty, S. R. (2024). Comprehensive Review Analysis and Countermeasures for Cybersecurity Threats: DDoS, Ransomware, and Trojan Horse Attacks. *preprint.org*. https://doi.org/10.20944/preprints202409.1369.v1

Riza, A. Z. B. M., Jennsen, L., Anggani, P., Rafeen, A. I., Ruth, P. N. J., Sookun, D., Sookun, V., Yusri, N. a. Z. B. M., Sern, L. J., Luximon, L., Omer, M. L., & Sindiramutty, S. R. (2025). Leveraging Machine Learning and AI to Combat Modern Cyber Threats. *Preprints.org*. https://doi.org/10.20944/preprints202501.0360.v1

Seng, Y. J., Cen, T. Y., Raslan, M. a. H. B. M., Subramaniam, M. R., Xin, L. Y., Kin, S. J., Long, M. S., & Sindiramutty, S. R. (2024). In-Depth Analysis and Countermeasures for Ransomware Attacks: Case Studies and Recommendations. *Preprints.org*. https://doi.org/10.20944/preprints202408.2261.v1

Sindiramutty, S. R., Jhanjhi, N., Tan, C. E., Lau, S. P., Muniandy, L., Gharib, A. H., Ashraf, H., & Murugesan, R. K. (2024). Industry 4.0. In *Advances in logistics, operations, and management science book series* (pp. 342–405). https://doi.org/10.4018/979-8-3693-1363-3.ch013

Sindiramutty, S. R., Jhanjhi, N. Z., Tan, C. E., Tee, W. J., Lau, S. P., Jazri, H., Ray, S. K., & Zaheer, M. A. (2024). IoT and AI-Based Smart Solutions for the Agriculture Industry. In *Advances in computational intelligence and robotics book series* (pp. 317–351). https://doi.org/10.4018/978-1-6684-6361-1.ch012

Sindiramutty, S. R., Jhanjhi, N. Z., Tan, C. E., Yun, K. J., Manchuri, A. R., Ashraf, H., Murugesan, R. K., Tee, W. J., & Hussain, M. (2024). Data security and privacy concerns in drone operations. In *Advances in information security, privacy, and ethics book series* (pp. 236–290). https://doi.org/10.4018/979-8-3693-0774-8.ch010

Sindiramutty, S. R., Prabagaran, K. R. V., Jhanjhi, N. Z., Ghazanfar, M. A., Malik, N. A., & Soomro, T. R. (2024). Security Considerations in Generative AI for web Applications. In *Advances in information security, privacy, and ethics book series* (pp. 281–332). https://doi.org/10.4018/979-8-3693-5415-5.ch009

Sindiramutty, S. R., Prabagaran, K. R. V., Jhanjhi, N. Z., Murugesan, R. K., Brohi, S. N., & Masud, M. (2024). Generative AI in network security and intrusion detection. In *Advances in information security, privacy, and ethics book series* (pp. 77–124). https://doi.org/10.4018/979-8-3693-5415-5.ch003

Sindiramutty, S. R., Tan, C. E., & Wei, G. W. (2024). Eyes in the sky. In *Advances in information security, privacy, and ethics book series* (pp. 405–451). https://doi.org/10.4018/979-8-3693-0774-8.ch017

Waheed, A., Seegolam, B., Jowaheer, M. F., Sze, C. L. X., Hua, E. T. F., & Sindiramutty, S. R. (2024). Zero-Day Exploits in Cybersecurity: Case Studies and Countermeasure. *Preprints.org*. https://doi.org/10.20944/preprints202407.2338.v1

Weiqi, X., Hooi, S. T. C., Sindiramutty, S. R. a. L., Asirvatham, D. a. L., Kumar, D., & Verma, S. (2024). Surface Anomaly Detection Using Machine Learning Technique. *2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC)*, 1–7. https://doi.org/10.1109/etncc63262.2024.10767562

Wen, B. O. T., Syahriza, N., Xian, N. C. W., Wei, N. G., Shen, T. Z., Hin, Y. Z., Sindiramutty, S. R., & Nicole, T. Y. F. (2023). Detecting cyber threats with a Graph-Based NIDPS. In *Advances in logistics, operations, and management science book series* (pp. 36–74). https://doi.org/10.4018/978-1-6684-7625-3.ch002

Xun, A. T., En, L. a. Z., Shen, L. T., Xin, A. N., Soon, W. H., Jun, W. Z., Ramachandra, H., Xinghao, G., Khant, N. M., Weitao, F., & Sindiramutty, S. R. (2025). Building Trust in Cloud Computing:Strategies for Resilient Security. *Preprints.org*. https://doi.org/10.20944/preprints202501.0716.v1

Ying, X., Murugesan, R. K., Sindiramutty, S. R., Wei, G. W., Balakrishnan, S., Kumar, D., & Verma, S. (2024). Scene Text Recognition using Deep Learning Techniques. *024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC)*, 1–9. https://doi.org/10.1109/etncc63262.2024.10767484

Rivlin, O. (2019, August 1). Model Based Policy Optimization. Retrieved from Medium: https://medium.com/data-science/model-based-policy-optimization-d7e099c73d8

Burgess, M. (2020, March 24). What is GDPR? The summary guide to GDPR compliance in the UK. Retrieved from Wired: https://www.wired.com/story/what-is-gdpr-uk-eu-legislation-compliance-summary-fines-2018/

Cimpanu, C. (2020, June 17). AWS said it mitigated a 2.3 Tbps DDoS attack, the largest ever. Retrieved from ZDNET: https://www.zdnet.com/article/aws-said-it-mitigated-a-2-3-tbps-ddos-attack-the-largest-ever/

Nicholson, P. (2020, June 24). AWS hit by Largest Reported DDoS Attack of 2.3 Tbps. Retrieved from A10: https://www.a10networks.com/blog/aws-hit-by-largest-reported-ddos-attack-of-2-3-tbps/

Yoachimik, O. (2021, March 18). A deep-dive into Cloudflare's autonomous edge DDoS protection. Retrieved from Cloudflare: https://blog.cloudflare.com/deep-dive-cloudflare-autonomous-edge-ddos-protection/

Sagar, R. (2021, July 19). Is Cost-Effective Deep Reinforcement Learning Possible? Retrieved from Analytics India magazine: https://analyticsindiamag.com/ai-features/cost-effective-deep-reinforcement-learning/

Gal, E. (2021, July 28). Software-Defined Networking Concept Adoption at Akamai. Retrieved from Akamai: https://www.akamai.com/blog/performance/software-defined-networking-concept-adoption-at-akamai

Bigelow, S. J. (2021, December 8). What is edge computing? Everything you need to know. Retrieved from TechTarget: https://www.techtarget.com/searchdatacenter/definition/edge-computing

Labs, B. L. (2022, October 24). CLDAP Reflectors On The Rise Despite Best Practice. Retrieved from LUMEN: https://blog.lumen.com/cldap-reflectors-on-the-rise-despite-best-practice/

Gold, J. (2022, October 25). Akamai to boost network-layer DDoS protection with new scrubbing centers. Retrieved from CSO: https://www.csoonline.com/article/573961/akamai-to-boost-network-layer-ddos-protection-with-new-scrubbing-centers.html

Olzak, T. (2022, November 22). How CLDAP Reflectors Enable DDoS Attacks & Ways to Reduce Your Exposure. Retrieved from spiceworks: https://www.spiceworks.com/it-security/cyber-risk-management/articles/defending-against-cldap-reflection-attacks/

Sharma, D. (2023, June 26). Cloud Armor : Protect your application from DDoS attack. Retrieved from Medium: https://medium.com/google-cloud/cloud-armor-protect-your-application-from-ddos-attack-3feb7c62661e

Kaneko, H. (2023, July 28). The Power of Proximity: Local DDoS Scrubbing Centers Enhance Security. Retrieved from Akamai: https://www.akamai.com/blog/security/the-power-of-proximity-local-ddos-scrubbing-centers-enhance-security

Zuckerman, O., Sakazi, I., Ozery, Y., & Goren, G. (2023, October 20). Detect and Remediate Attacks: Practical Applications for Machine Learning. Retrieved from Akamai: https://www.akamai.com/blog/security/detect-and-remediate-attacks-with-akamai-hunt

Pardue, L., & Desgats, J. (2023, October 10). HTTP/2 Rapid Reset: deconstructing the record-breaking attack. Retrieved from CloudFlare: https://blog.cloudflare.com/technical-breakdown-http2-rapid-reset-ddos-attack/

Snellman, J., & Lamartino, D. (2023, October 11). How it works: The novel HTTP/2 'Rapid Reset' DDoS attack. Retrieved from Google Cloud: https://blog.cloudflare.com/technical-breakdown-http2-rapid-reset-ddos-attack/

Kobialka, D. (2023, October 12). Google, Amazon Face Massive Denial-of-Service Attack. Retrieved from MSSP Alert: https://www.msspalert.com/news/google-amazon-face-massive-denial-of-service-attack

Todd, D. (2023, October 12). HTTP/2 Rapid Reset Zero-Day Largest DDoS Attack in Internet History. Retrieved from SecureWorld: https://www.secureworld.io/industry-news/rapid-reset-record-ddos-attack

Birchard, D., & Rath, S. (2023, October 16). How Akamai Protects Customers from HTTP/2 Rapid Reset DDoS Attacks. Retrieved from Akamai: https://www.akamai.com/blog/security/akamai-protects-customers-http2-rapid-reset-ddos-attacks

Losio, R. (2023, November 5). Cloudflare, Google and AWS Disclose HTTP/2 Zero-Day Vulnerability. Retrieved from InfoQ: https://www.infoq.com/news/2023/11/http2-rapid-reset-vulnerability/

Bergmann, D., & Stryker, C. (2023, November 23). What is an autoencoder? Retrieved from IBM: https://www.ibm.com/think/topics/autoencoder

Pardue, L. (2024, Jan 11). HTTP/2 Rapid Reset: Deconstructing the record-breaking attack. Retrieved from APNIC: https://blog.apnic.net/2024/01/11/http-2-rapid-reset-deconstructing-the-record-breaking-attack/

Powell, P., & Smalley, I. (2024, May 3). Edge computing use cases: Eight ways organizations are leveraging edge computing. Retrieved from IBM: https://www.ibm.com/think/topics/edge-computing-use-cases

Mukeshreddy. (2024, September 7). Understanding Models in Reinforcement Learning: A Dive into Dyna-Q. Retrieved from Medium: https://medium.com/@mukeshreddy662369/understanding-models-in-reinforcement-learning-a-dive-into-dyna-q-c2b58c982b84

Amin, S. (2024, September 14). Deep Q-Learning (DQN). Retrieved from Medium: https://medium.com/@samina.amin/deep-q-learning-dqn-71c109586bae

Soam, A. (2024, November 6). Cloudflare DDoS Protection : How Does it Shield Your Site from Cyber Threats? Retrieved from Kennies: https://kenniesit.com/kb/security-services/cloudflare-ddos-protection-how-does-it-shield-your-site-from-cyber-threats/

Ahmed, A., & Bella, A. (2024, November 7). Akamai's Behavioral DDoS Engine: A Breakthrough in Modern DDoS Mitigation. Retrieved from Akamai: https://www.akamai.com/blog/security/akamais-behavioral-ddos-engine-breakthrough-in-modern-ddos-mitigation

Kashyap, P. (2024, December 5). A Comprehensive Guide to Autoencoders. Retrieved from Medium: https://medium.com/@piyushkashyap045/a-comprehensive-guide-to-autoencoders-8b18b58c2ea6

Signh, S. (2024, December 16). Kubernetes Dynamic Resource Allocation: A Leap in Resource Management. Retrieved from Medium: https://medium.com/@simardeep.oberoi/kubernetes-dynamic-resource-allocation-a-leap-in-resource-management-c39fdca6b99e

Thomas, C. (2025, April 10). Distributed Edge Computing: Unlocking the Power of Decentralized Networks To Drive Innovation. Retrieved from Suse: https://www.suse.com/c/distributed-edge-computing-unlocking-the-power-of-decentralized-networks-to-drive-innovation/

Buckman, B. (2025, May 18). What is Suricata? The Cybersecurity Tool You Need to Know. Retrieved from Huntress: https://www.huntress.com/cybersecurity-education/cybersecurity-101/topic/what-is-suricata

Next generation dynamic resource management. (2025, June 13). Retrieved from Google Cloud: https://cloud.google.com/compute/docs/dynamic-resource-management

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.