Detection and Prevention of DDoS Attacks using ML in Cloud Environment

# Achyut Katiyar

*Computer Science Department*

*SRM University AP*

Guntur, India

achyut [katiyar@srmap.edu.in](mailto:katiyar@srmap.edu.in)

Krishna Mihir Tatavarthi *Computer Science Department SRM University AP*

Guntur, India krishnamihir [t@srmap.edu.in](mailto:t@srmap.edu.in)

Chandra Sai Reddy Donthireddy *Computer Science Department SRM University AP*

Guntur, India

saireddy [donthireddy@srmap.edu.in](mailto:donthireddy@srmap.edu.in)

# Kaushik Aadithya Chiratanagandla

*Computer Science Department SRM University AP*

Guntur, India kaushikaditya [nag](mailto:nagandla@srmap.edu.in)[andla@srmap.edu.in](mailto:andla@srmap.edu.in)

Sameer Thadimarri *Computer Science Department SRM University AP*

Guntur, India

sameer [thadimarri@srmap.edu.in](mailto:thadimarri@srmap.edu.in)

Dr.Sambit Kumar Mishra *Computer Science Department SRM University AP*

Guntur, India [sambitkumar](mailto:sambitkumar.m@srmap.edu.in)[.m@srmap.edu.in](mailto:.m@srmap.edu.in)

***Abstract*—Cloud computing delivers computing services over the internet, including storage, processing power, and application. It allows users to access these resources from anywhere with an internet connection and pay only for what they use, providing flexibility and cost-effectiveness. The aim of this work is to address the security concerns associated with cloud computing. There are many security risks, and it is susceptible to attacks, particularly Distributed Denial of Service (DDoS), which floods the network with attack packets that are difficult to identify. In DDoS attacks, multiple machines attack users by sending packets with large data overhead, making resources unavailable. There are many approaches available, but one of the approaches for preventing DDoS on cloud computing is to use a Captcha- Based Verification System. This system requires users to solve challenges to prove they are human, which blocks automated bots and detects suspicious traffic patterns. But the determined attackers can bypass the captcha, so additional security measures such as Machine Learning algorithms are applied to analyze traffic patterns and detect anomalies that may indicate an attack. Another approach is to use IP Blocking or Rate-Limiting to limit the number of requests from a suspicious IP address. Regular security audits and penetration testing can help identify and address weaknesses in the cloud infrastructure. The simulated results show the detection and prevention of DDoS attacks using machine learning in a cloud environment.**

***Index Terms*—DDoS, Captcha-Based Verification System, IP Blocking, Rate Limiting.**

1. INTRODUCTION

Cloud computing has brought many benefits to businesses, such as cost-effectiveness and scalability. However, it has also introduced new risks and threats to the ecosystem. Here are some of the major threats to cloud networks: One major threat is data breaches. Cloud networks store sensitive and critical data, making them a prime target for hackers. Unauthorized access to cloud networks can lead to data breaches, which can result in identity theft, financial loss, and reputational damage. Malware attacks can infect cloud networks and cause signifi- cant damage to the network and its users. Malware can be used to steal data, spy on users, or launch attacks on other networks.

Account hijacking is another threat to cloud networks. Cloud accounts can be hijacked through phishing attacks or other means, allowing hackers to access sensitive data and resources. Denial of Service (DoS) attacks can disrupt cloud services, rendering them unavailable to users. These attacks can be carried out through a range of methods, including flooding the network with traffic or exploiting vulnerabilities in the network infrastructure.

One of the most significant threats to cloud networks today is Distributed Denial of Service (DDoS) attacks. DDoS attacks involve a large number of devices or computers flooding a network with traffic, overwhelming its capacity and causing it to become unavailable to users. DDoS attacks can be launched from anywhere in the world and are difficult to trace back to their source, making them a popular tool for cybercriminals. They can be used to extort money from businesses by threat- ening to disrupt their services, or to carry out political attacks against government or corporate websites.

Cloud service providers employ various techniques such as rate limiting, traffic filtering, and the use of content delivery networks (CDNs) to mitigate the threat of DDoS attacks. However, as DDoS attacks continue to evolve and become more sophisticated, it remains a major threat to the cloud ecosystem.

1. WHAT IS DDOS?

A Distributed Denial of Service (DDoS) attack is a type of cyberattack where a network or website is overwhelmed with traffic, making it inaccessible to its intended users. In a DDoS attack, the attacker typically uses a network of computers or devices to flood the targeted system with requests or data packets, effectively blocking legitimate traffic and causing the system to slow down or crash. DDoS attacks are often carried out using botnets, which are networks of infected devices that can be controlled remotely to carry out attacks without the knowledge of their owners.

DDoS attacks can have serious consequences for their victims, including financial losses, damage to reputation, and even legal liability. DDoS attacks can be difficult to prevent and mitigate, as they often originate from multiple sources and are designed to evade detection and countermeasures. To defend against DDoS attacks, organizations may use a range of techniques, including firewalls, intrusion detection systems, and content delivery networks. In some cases, organizations may also employ specialized DDoS mitigation services or work with Internet Service Providers (ISPs) to filter out malicious traffic.

The motivations behind DDoS attacks can vary widely. Some attackers carry out DDoS attacks for financial gain, such as by extorting businesses for ransom or using attacks to distract from other cybercriminal activities. Other attackers may carry out DDoS attacks as a form of protest or activism, often targeting websites of governments, corporations, or other organizations they disagree with. DDoS attacks can also be used as a tool for cyberwarfare, with nation-states or other groups using attacks to disrupt critical infrastructure or com- munications systems.

As the internet continues to play an increasingly important role in society and commerce, the threat of DDoS attacks is likely to persist. To protect against these attacks, organizations must remain vigilant and take proactive measures to secure their networks and systems. This may include investing in advanced security technologies, training employees on cyber- security best practices, and working with law enforcement and other organizations to share threat intelligence and coordinate response efforts. By taking these steps, organizations can help reduce their risk of falling victim to a DDoS attack and minimize the impact of any attacks that do occur.

1. DATASET

The data is provided in CSV format, where each line in the file represents a single record of data. The research used datasets containing information on 1955 rows of data. Each dataset includes 19 attributes. They are

1. Source IP: This attribute represents the IP address of the device that originated the network traffic. An IP address is a unique identifier assigned to each device on a network that uses the Internet Protocol for communication.
2. Source PORT: This attribute represents the port number on the source device that was used for the network traffic. A port number is a 16-bit integer that is used to identify a specific process or service running on a device that is communicating over the internet.
3. Destination IP: This attribute represents the IP address of the device that received the network traffic.
4. Destination PORT: This attribute represents the port num- ber on the destination device that was used for the network traffic.
5. TYPE Protokol: This attribute represents the protocol used for the network traffic, such as TCP (Transmission Con- trol Protocol) or UDP (User Datagram Protocol). A protocol

is a set of rules that govern the format and transmission of data over a network.

1. Total Length: This attribute represents the total length of the network traffic packet in bytes. This includes the header information and the payload (data) portion of the packet.
2. Target: This attribute represents the classification target of the network traffic, which could be benign or malicious. In a supervised learning scenario, this attribute is used as the output variable, while the other attributes are used as input variables to training a machine learning model that can predict the target classification of new network traffic.
3. METHODOLOGY

We used some ML models like Logistic Regression, naive Bayes & K Neighbours Classifier.

1. *Logistic Regression*

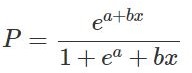


Fig. 1. Equation for Logistic Regression

Logistic Regression is a popular machine-learning algorithm used for classification tasks. Unlike linear regression, which is used for predicting continuous numeric values, logistic regres- sion models the probability of a binary outcome (e.g., yes/no, true/false) based on one or more input variables. In logistic regression, the output of the dependent variable is a binary variable, which is either 0 or 1. The input or independent variables can be continuous, discrete or categorical. Logistic regression works by applying a transformation to the output of a linear model, which maps the output to a probability value between 0 and 1. This transformation is typically done using the sigmoid function, which has an S-shaped curve.

1. *K-Nearest Neighbour*

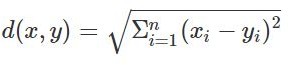


Fig. 2. Equation for KNN

K-Nearest Neighbors (KNN) is a popular algorithm in ma- chine learning used for classification and regression problems. It is a simple, non-parametric algorithm that can be used for both supervised and unsupervised learning. In KNN, the ”k” refers to the number of nearest neighbours used to classify a new data point. The algorithm works by calculating the distance between the new data point and the existing data

points in the dataset. It then selects the ”k” nearest neighbours and assigns the new data point to the class that is most frequent among those neighbours.

1. *Naive Bayes*

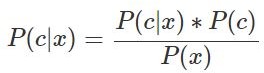


Fig. 3. Equation for Naive Bayes

Naive Bayes is a machine learning algorithm used for classification tasks. It is a variant of the Naive Bayes algorithm that makes the assumption that the features are normally distributed. This assumption simplifies the calculation of prob- abilities by assuming that the distribution of each feature is independent of the others. The algorithm works by calculating the probability of each class given the input features and then selecting the class with the highest probability as the predicted output. The probability is calculated using Bayes’ theorem, which states that the probability of a hypothesis (in this case, a class) given some observed evidence (input features) is proportional to the probability of the evidence given the hypothesis times the prior probability of the hypothesis.

1. RESULTS

The results you provided show the performance of three different machine learning models: Naive Bayes, K-Nearest Neighbors (KNN), and Logistic Regression and gives two correlation matrix. The training accuracy is a measure of how well the model has learned from the training data, while the testing accuracy is a measure of how well the model can generalize to new, unseen data. In the Naive Bayes model, the training accuracy is 0.5591810620601407, which means that the model correctly classified only 55.9% of the training data. This indicates that the model may not have learned the underlying patterns and relationships in the data very well. The testing accuracy of 0.5524296675191815 is even lower, which suggests that the model does not generalize well to new data. In contrast,The KNN model has a very high training accuracy of 0.982085732565579, indicating that it has learned the patterns in the data very well. The testing accuracy of 0.9744245524296675 is also very high, which suggests that the model can generalize well to new data. The KNN algorithm is a type of instance-based learning, where the model classifies new data points based on the closest k points in the training set. This makes KNN a simple but powerful algorithm that can be effective in a wide range of applications. However, the performance of the algorithm can be sensitive to the choice of distance metric and the value of

k. The Logistic Regression model also has a high training

accuracy of 0.982085732565579 and a testing accuracy of 0.9744245524296675, which indicates that it has learned the patterns in the data well and can generalize to new data. Logistic Regression is a type of linear model that models the probability of a binary outcome based on the input features. It is a simple and interpretable algorithm that can be useful in cases where the goal is to understand the relationship between the input features and the outcome variable. However, Logistic Regression assumes a linear relationship between the input features and the log odds of the outcome, which may not always be the case in real-world data. Additionally, Logistic Regression can be sensitive to outliers and may not perform well when there are non-linear relationships between the input features and the outcome variable.

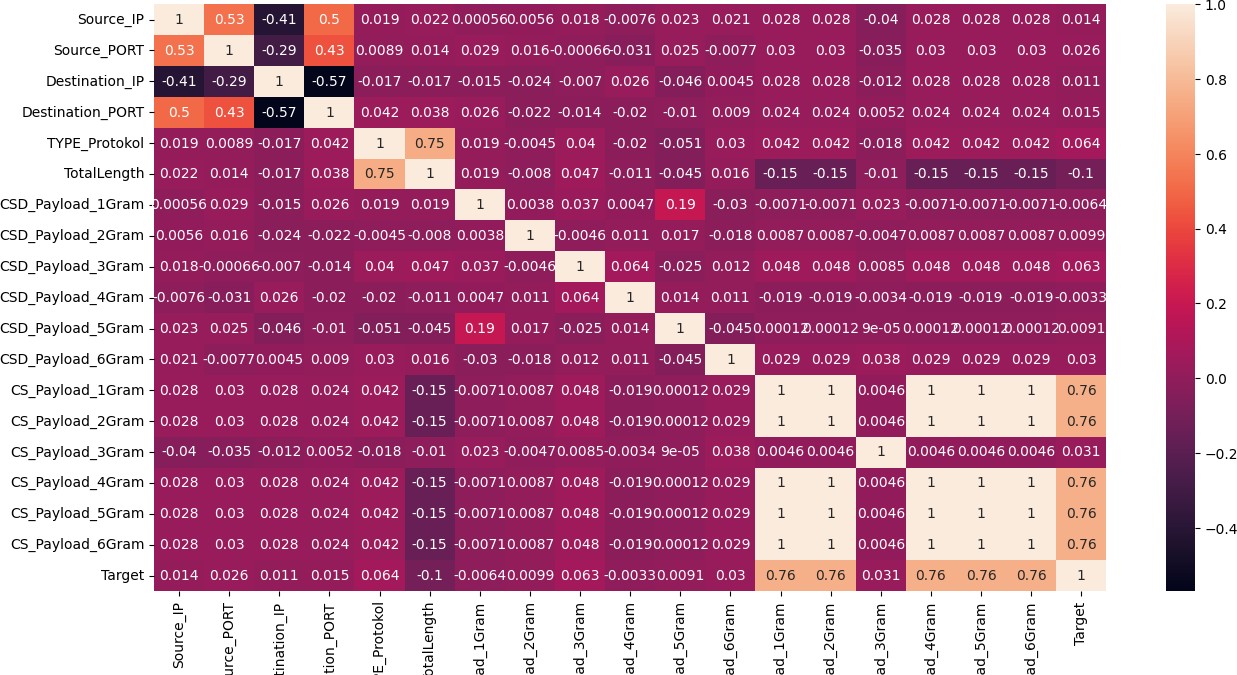


Fig. 4. Correaltion Matrix

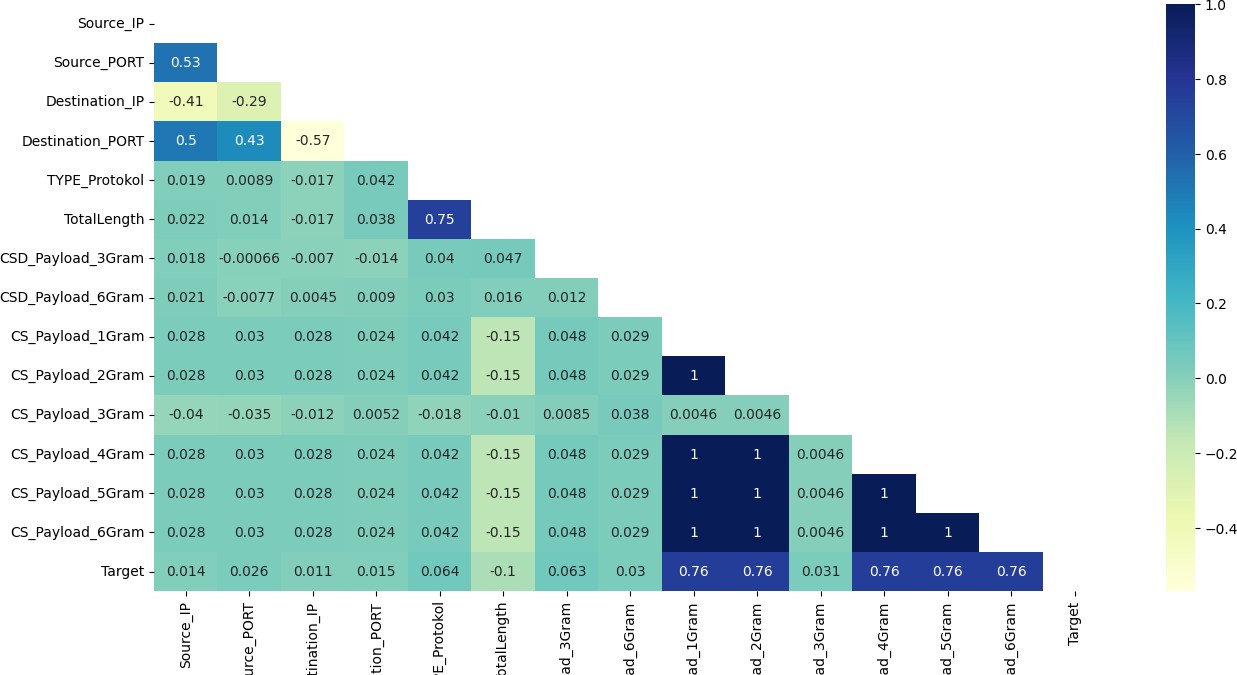


Fig. 5. Improved Correlation Matrix

1. IP BLOCKING

IP blocking is a method of preventing specific IP addresses from accessing a website or network. It can be implemented in a variety of methods, including firewalls, routers, access control lists, and specialised DDoS protection services. The goal is to identify and block the IP addresses that are gen- erating malicious traffic, such as requests that exceed typical levels, requests with specified patterns or payloads, or requests from known botnets or attackers. This can be done manually or automatically, using either established rules or machine learning techniques.

While IP filtering can be effective against certain types of DDoS assaults, such as those relying on a small number of

attacker IP addresses, it has several limitations and potential drawbacks.

While IP blocking can be effective against certain forms of DDoS assaults, such as those involving a small number of attacking IP addresses, it does have some limitations and potential downsides. For example, attackers can utilise IP spoofing, which includes faking the source IP address of attack traffic in order to avoid IP blocking, or botnets, which contain a large number of distributed compromised computers that can create traffic from various IP addresses. Furthermore, IP blocking can cause false positives, in which normal users are prohibited from accessing the target owing to misconfigured or outdated filters, or collateral damage, in which innocent hosts are mistakenly labelled as attackers and blocked.

1. *Relevance of IP-BLOCKING in Cyber Security*

In the vast ocean of Cyber Security, IP blocking acts as a crucial tool by blocking Information transmission from IP addresses associated with unnecessary things, such as spam or cyberattacks, IP blocking can be used as a technique to prevent these kinds of attacks. This can help protect networks and websites against illicit access, data theft, and other security breaches. By making use of IP filtering, access to resources can also be restricted according to location. This can help prevent assaults that originate from specific locations by restricting access to content for that area.

The ability for organizations to keep track of who has access to their networks and resources and to protect against a number of risks makes IP blocking, in general, a key component of cybersecurity.

As a result, IP blocking should be part of a comprehensive defence plan that includes traffic analysis, anomaly detection, load balancing, and cloud-based solutions. Traffic analysis, for example, can assist in identifying and mitigating DDoS as- saults based on their signature or behaviour, whereas anomaly detection might signal unexpected traffic patterns that may suggest an ongoing attack. To avoid overloading a single point of failure, load balancing can distribute traffic across numerous servers or data centres, whilst cloud-based solutions can provide scalable and adaptable resources to absorb and filter out attack traffic. A combination of these steps can help boost the target’s resistance and responsiveness to DDoS attacks while minimising the impact on legitimate users.

To develop an effective DDoS defense strategy, organi- zations should also consider the specific characteristics and risks of their business and sector, and tailor their approach accordingly. For example, financial institutions may face more sophisticated and targeted attacks than online retailers, and may need to invest in more advanced technologies and per- sonnel to prevent and respond to DDoS incidents. Similarly, government agencies may have different regulatory and com- pliance requirements than private companies, and may need to coordinate with law enforcement and other stakeholders to mitigate DDoS threats.

1. *Relevance of IP-blocking to DDOS*
   * Determine the origin of the DDoS attack: Examine net- work traffic with network monitoring tools to determine the IP addresses causing the malicious activity.
   * Configure the firewall: Once the IP addresses responsible for the attack have been identified, configure the firewall to block traffic from these addresses. This can be accom- plished by adding the IP addresses to a blacklist.
2. *Implementation of IP-BLOCKING to prevent DDoS*
   * Determine the origin of the DDoS attack: Examine net- work traffic with network monitoring tools to determine the IP addresses causing the malicious activity.
   * Configure the firewall: Once the IP addresses responsible for the attack have been identified, configure the firewall to block traffic from these addresses. This can be accom- plished by adding the IP addresses to a blacklist. But it is important to keep to note that IP filtering always doesn’t prevent DDoS attack.
   * Enable SYN cookies: To prevent SYN flood attacks, enable SYN cookies on all platforms. SYN cookies are a defence measure that can aid in the prevention of DDoS assaults by restricting the number of connections from a specific IP address.
   * Limit the rate of incoming traffic: Set limits on how much traffic can be transmitted to a specific IP address in a given time frame. By limiting the amount of traffic that may be delivered from a specific IP address, this can assist avoid DDoS attacks.
   * Use load balancers to disperse traffic across numerous servers to avoid a single point of failure. By making it more difficult for attackers to overwhelm a single server, this can help avoid DDoS attacks.
   * Rate limitation should be implemented on your servers to avoid excessive traffic from a single IP address. By limiting the amount of traffic that may be delivered from a specific IP address, this can assist avoid DDoS attacks.
   * Use intrusion detection systems (IPS): Implement intru- sion prevention systems (IPS) to monitor network traffic to detect and block DDoS attacks in real time. IPS can detect and restrict traffic from IP addresses that have been linked to DDoS assaults.
   * Monitor traffic patterns on a regular basis to spot any odd traffic surges. This allows you to detect DDoS at- tacks early and take precautionary measures before major damage is done.
   * Maintain software updates: Keep all software up to date with the most recent patches and security fixes. This can help prevent DDoS assaults by resolving known software vulnerabilities.
3. RATE LIMITING

In order to control the volume of incoming traffic, numerous applications, including cloud-based software, use a mechanism called rate limiting. It serves as a safeguard against overtaxing a system or network by limiting the quantity of requests that

may be handled at one time. Rate limiting’s major goal is to keep a system or network from becoming overloaded with too many requests, thereby ensuring that it is still functional and accessible to authorised users.

In the subject of cybersecurity, rate limitation is essential for stopping various attacks that try to disrupt or jeopardise the security of the system. One such attack is a Distributed Denial of Service (DDoS) attack, in which a malicious actor saturates a target system with a massive amount of traffic in an effort to bring it down or render it unavailable to authorised users. By restricting the amount of requests the system can process in a given period of time and keeping it from becoming overwhelmed, rate limiting can assist lessen the effects of DDoS attacks.

A brute-force attack, in which an attacker repeatedly tries to guess the password to gain unauthorised access to a system or account, is another attack that can be stopped by rate restric- tion. Rate restriction can make it harder for attackers to execute successful brute-force assaults by restricting the number of login attempts that can be made in a given period of time.

Moreover, rate restriction can assist in preventing API abuse, which occurs when a user or bot sends an exces- sive amount of calls to an API quickly. Resource depletion, service interruptions, and potential security flaws can all be caused by API overuse. API providers can safeguard their systems against overload and guarantee that their services are still available to authorised customers by implementing rate restriction, which limits the pace at which requests are made to their API.

In conclusion, rate restriction is an effective method that may be applied to defend against a variety of assaults that seek to disrupt or jeopardise system security. By using it, a system or network could be protected from DDoS attacks, brute-force attacks, API abuse, and other dangerous practises that could jeopardise the availability and integrity of the system or network.

Depending on the particular needs and specifications of the application or system, rate limiting is a flexible and adaptable technique that can be used in a variety of situations. In order to reduce their impact on the system’s performance or security, rate restriction can be used, for example, to limit the amount of requests that a specific IP address or user account may submit. Additionally, depending on the nature of the application and the risks involved, rate restriction can be applied to various forms of traffic, such as HTTP requests, UDP packets, or DNS queries.

Rate limitation can help manage the use and consumption of resources, such as bandwidth, CPU, or memory, in addition to thwarting assaults. Rate limitation can stop resource depletion and make sure the system is reliable and responsive to autho- rised users by controlling the amount of incoming traffic.

The ability to gain useful insights about the usage patterns and behaviour of the application’s users is another advan- tage of rate limiting. Application owners can see potential anomalies, such as spikes or dips in traffic, that may signify malicious behaviour or system problems by tracking and

analysing the rate of requests over time. The rate limitation rules can be adjusted using this information to account for shifting circumstances and keep the system secure.

Rate restriction should be utilised along with other security measures like firewalls, intrusion detection systems, and en- cryption but should not be considered a cure-all for all security and performance problems. To avoid unwanted consequences like false positives, slowness, or denial of service to legitimate customers, rate limiting should also be carefully developed and thoroughly tested before being used.

Overall, rate limiting is a crucial part of contemporary security and performance strategies, offering a powerful and adaptable way to defend systems and applications against a variety of risks and difficulties. Organisations may make sure that their systems are resilient, dependable, and secure by integrating rate limitation with other protective strategies and industry best practices.

*A. Implementation of Rate Limiting to prevent DDoS*

* Identifying the important essential assets and resources that must be secured from a DDoS attack.

Ex: Web servers, Databases or any other critical services of cloud infrastructure.

* Monitor and analyze traffic patterns for any unusual activities that may include DDoS attack or any type of other attacks. To monitor and analyze the traffic patterns we need to use Tools such as, Wireshark etc.
* Set up rate limiting rules to limit the amount of traffic can be sent from specified IP addresses or network ranges. This can be accomplished with the use of firewalls, load balancers, or other network devices that allow rate limitation.
* Limit the rate of incoming traffic: Set limits on how much traffic can be transmitted to a specific IP address in a given time frame. By limiting the amount of traffic that may be delivered from a specific IP address, this can assist avoid DDoS attacks.
* Verify the rate restriction rules to confirm they are ef- fective at mitigating DDoS attacks while not interfering with genuine traffic. This can be achieved by simulating a DDoS assault with application such as Apache jmeter or the OWASP Zed assult Proxy.
* Rate limitation should be implemented on your servers to avoid excessive traffic from a single IP address. By limiting the amount of traffic that may be delivered from a specific IP address, this can assist avoid DDoS attacks.
* Use intrusion detection systems (IPS): Implement intru- sion prevention systems (IPS) to monitor network traffic to detect and block DDoS attacks in real time. IPS can detect and restrict traffic from IP addresses that have been linked to DDoS assaults.
* Over time, Monitor and change the rate limiting rules to ensure that they continue to be effective in mitigating DDoS attacks. This could include updating the threshold for different categories of traffic or the rate limiting rules based on the most recent security concerns.
* Consider using other security measures, such as Captcha- Based Verification systems or Machine Learning algo- rithms, to supplement rate limiting and provide additional layers of protection against DDoS attacks.
* Overall, the key to properly implementing rate limiting is finding an appropriate balance between preventing malicious traffic and enabling legal data to transit via the cloud infrastructure. Cloud providers may help ensure that their stay available and safe in the safe in the face of DDoS attacks by closely monitoring traffic patterns and modifying rate limiting policies as needed.

CONCLUSION

Due to the increasing frequency and sophistication of cyber threats, it is now more crucial than ever to detect and block DDoS attacks using machine learning (ML) in a cloud context. Security teams can swiftly detect and neutralise DDoS assaults before they do substantial harm because to the ability of ML algorithms to analyse vast volumes of data in real-time and discover aberrant patterns and behaviours.

Due to the scalability and flexibility of cloud services, ML- based DDoS detection and prevention can be very useful in a cloud setting. Cloud-based ML solutions can swiftly react to changing attack patterns and reduce the effects of DDoS attacks thanks to their capacity to instantly spin up extra resources and intelligently disperse traffic over many servers. However, the quality of the data and the precision of the algorithms are what determine how well ML-based DDoS detection and prevention works. To ensure that ML models are optimised for the unique features of the cloud environment, it

is crucial to regularly monitor and improve them.

Overall, employing ML for DDoS attack detection and mitigation in a cloud setting is a viable strategy for bolstering cybersecurity defences. The importance of ML-based solutions in defending infrastructure and cloud-based assets from DDoS attacks is projected to increase as the threat landscape changes.

1. REFERENCES
2. Detecting Distributed Denial of Service (DDoS) Attacks using Machine Learning Techniques” by Mohammed Alreshoodi and Ahmed Al-Eroud
3. An efficient approach for detecting DDoS attacks using machine learning techniques” by Amira E. Abdelhameed and Hanan S. Kamal
4. DDoS attack detection using machine learning tech- niques: A review” by Mehtab Singh and Ajay Kumar
5. DDoS attack detection using machine learning algo- rithms: A comparative study” by Priti Verma and Ram Gopal Raj.
6. A machine learning-based approach to detecting DDoS attacks in cloud computing” by Shaofeng Duan, Junhua Ding, and Yulong Zhang
7. A Novel Approach to DDoS Attack Detection Using Machine Learning Techniques” by S. Swetha and S. Sreeja
8. Efficient DDoS Attack Detection Using Machine Learn- ing Techniques in Software-Defined Networks” by S. L. Pandian and K. K. Thirunavukkarasu
9. Detecting DDoS Attacks in Software-Defined Networks Using Machine Learning Techniques” by B. O. Oladejo and O. A. Oyebade
10. A Comparative Study of Machine Learning Algorithms for DDoS Attack Detection” by A. Varghese, S. S. Kumar, and S. S. Soman
11. A survey of machine learning techniques for DDoS attack detection and mitigation” by J. Sharma, V. Tyagi, and M. Sharma