**IAES International Journal of Artificial Intelligence (IJ-AI)**
Vol. 15, No. 4, August 2026, pp. 3154~3163
ISSN: 2252-8938, DOI: 10.11591/ijai.v15.i4.pp3154-3163 

# Utilizing Poisson modeling and advanced machine learning techniques for enhanced detection of slowloris attacks

**Prathima Mabel John[1], Mamatha Ram[2], Vani Kurugod Ashwathanarayana Reddy[3], Santhosh Babu[4], Rama Mohan Babu Kasturi Nagappasetty[5]**
[1]Department of Information Science and Engineering, Dayananda Sagar College of Engineering, Visvesvaraya Technological University, Bengaluru, India
[2]Department of Computer Science and Engineering, BMS College of Engineering, Visvesvaraya Technological University, Bengaluru, India
[3]Department of Aritficial Intelligence and Machine Learning, BNM Institute of Technology, Visvesvaraya Technological University, Bengaluru, India
[4]Department of Electronics and Telecommunication Engineering, Dayananda Sagar College of Engineering, Visvesvaraya Technological University, Bengaluru, India
[5]Department of Computer Science and Engineering (Data Science), Dayananda Sagar College of Engineering, Visvesvaraya Technological University, Bengaluru, India

**Article Info**

***Article history:***

Received Jun 13, 2025
Revised May 26, 2026
Accepted Jun 19, 2026

***Keywords:***

Machine learning
Monte Carlo simulation
Poisson arrival process
Slowloris
Software defined network

**ABSTRACT**

This paper presents a secure framework for detecting and mitigating slowloris attacks in software defined networks (SDN) using Poisson arrival modeling to generate attack traffic. Slowloris attacks imitate legitimate traffic patterns, making them difficult to identify. The framework includes three modules: a data generation module, a feature selection module based on the blended statistical and information gain (BSIG) approach, and a traffic classification module using machine learning and deep learning algorithms. Poisson arrival modeling captures the stochastic nature of attack traffic by introducing variability in packet inter-arrival times, thereby simulating realistic network conditions. Monte Carlo simulations establish anomaly thresholds, while the BSIG method identifies key features, such as inter-arrival variability and CPU utilization. Support vector machine (SVM), k-nearest neighbors (KNN), and random forest (RF) classifiers achieve high precision in distinguishing legitimate traffic from attack traffic. The integration of Poisson modeling with advanced analytical methods improves detection capability while maintaining scalability for similar stealth attacks. The proposed methodology provides high detection accuracy and adapts to dynamic network environments, enhancing web server defenses against slowloris attacks in real-world scenarios. The framework also reduces false positives, supports efficient resource utilization, and strengthens proactive security management in modern SDN infrastructures under varying conditions.



***Corresponding Author:***

Prathima Mabel John
Department of Information Science and Engineering, Dayananda Sagar College of Engineering
Visvesvaraya Technological University
Kumarswamy Layout, Bengaluru 560078, India
Email: prathimamabel-ise@dayanandasagar.edu

## 1. INTRODUCTION

The coordinated proliferation of web applications has led, particularly in industries placing high levels of dependence on zones such as e-commerce, health care, finance, and social interaction, to turning

web servers into prime targets for cyberassaults. Among these, such attacks as slowloris, which are distributed denial of service (DDoS) attacks, are rather clever in that they unauthorizedly address seldom reports whereas only use very little bandwidth. It was so stealthy that it could hardly be detected, thus due to which slowloris was able to become, for instance, very effective.

The complexity involved in the recognition of slowloris attacks is based on their capability of generating traffic that, statistically, resembles normal web traffic. Using the Poisson arrival process, the slowloris attacker will treat traffic generation with a teenage level of care, introducing a high degree of randomness and unpredictability in how fast he delivers packets. The attack traffic then fits in with legitimate traffic, hence making common detection processes impractical. Without their aid, there is a need for more complex frameworks that will embrace statistical modeling, feature selection, and machine learning for effective detection.

This study brings to the fore the importance of the Poisson arrival process in modeling slowloris attack traffic. The combination of stochastic models, feature selection algorithms, and machine learning has been found to be effective for enhancing the detection and classification of slowloris attacks. Researchers studied an application of the machine learning methods to detect the slowloris attacks [1]–[8]. Their work pointed out the disadvantages of the static threshold-based approach and suggested a dynamic traffic modelling approach as an effective means to distinguish between legitimate and attack traffic. A Poisson based modelling was suggested to be able to better capture the stochastic nature of the attack traffic and thus improve the accuracy of the detection system.

The authors [9]–[12]7/28/2026 3:03:00 PM introduces a deep learning framework for the real-time detection and mitigation of DDoS attacks in an software defined network (SDN)-enabled environment. With advanced traffic analyses feature employed, this framework achieves high accuracy with low latency, coupled with automatic mitigation strategies. The authors [13]–[15] focus on slow HTTP/2 denial of service (DoS) attacks, as the authors of the study highlight their potential for causing disruption given their features within the protocol itself. The authors put forward real-time detection techniques through the analysis of an event sequence to successfully mitigate these threats. The need for adaptive security in evolving attack vectors is underlined in the study. The findings call for further investigation into a robust defense mechanism to address such vulnerabilities.

The intelligent system for slow DoS detection using machine learning (ISSDM), is an intelligent system that supports the diagnosis and mitigation of slow DoS attacks, in particular, the slowloris type within SDNs. It acquires features such as response times, connection counts, timeouts, and pattern match through the generation of traffic data of real-world configurations built upon the setup of Apache2 and Nginx servers, Zodiac FX OpenFlow switch, and the Ryu controller. The dynamic threshold value is estimated for classifying attacks with Monte Carlo simulation, distinguishing legitimate from attack traffic via header inspection using regular expressions. To enhance detection accuracy, the focus selection integrates blended statistical and information gain (BSIG), which chooses the most relevant interviewed features for classification. When put through several machine and deep learning models, these features lead to better detection performance in identifying slow DoS attacks as opposed to more traditional feature selection methods like information gain, T-test, and Chi-square. This survey provides strong justification for the fact that the modeling and detecting of low-rate DDoS attacks; for instance, slowloris is distinguished by a wide use of Poisson arrival processes in [16]–[24]. Being statistical traffic models, along with machine learning and advanced simulation techniques, the latter works contribute to the construction of very robust systems that could spot stealthy attack patterns in dynamic networking environments.

The Poisson distribution, extensively utilized in the analysis of network traffic, is mathematically grounded in coming up with the random but controlled arrival patterns of HTTP requests generated during slowloris attacks. While legitimate traffic might just maintain a consistent pattern between requests, with inter-arrival times not having high variability, the Poisson process introduces variances that imitate real-world randomness. Thus, inserting this approach into the detection framework creates realistic traffic modeling and later accurate traffic analysis.

The proposed framework comprises of three major modules. Through the data generation module, certain key server parameters, including CPU usage, memory utilization, and network throughput alongside packet inter-arrival times, are monitored. These parameters are continuously being analyzed and anomalies are detected through some thresholds using Monte Carlo simulations in order to identify deviations from normal server behavior. The feature selection module, with the help of the BSIG technique, extracts the most relevant features for classification such as inter-arrival time variability, packet arrival rates, and CPU utilization that are directly influenced by the Poisson arrival process. The traffic classification module makes use of several machine learning and deep learning algorithms to classify traffic as legitimate or attack in nature. For this purpose, support vector machine (SVM), decision tree (DT), k-nearest neighbors (KNN), random forest (RF), naïve Bayes (NB), and multilayer perceptron (MLP) algorithms are employed.

By integrating the advanced analytical techniques within a Poisson arrival model, this framework acts directly toward elevating detection capabilities of slowloris kinds of attacks and creates a scalable and systematic methodology to tackle similar types of stealth attacks. The envisaged approach promises high detection accuracies coupled with dynamic network adaptabilities. The paper explains the design, implementation, and evaluation of the envisaged approach, providing a functioning possibility of the SDN for web server protection against real-world application-layer DDoS attacks.

## 2. PROPOSED METHODOLOGY

The method proposed in this paper is a modified approach of work presented in [25] called "An intelligent system for slow DoS detection using machine learning - Poisson arrival rate-based (ISSDM-PA)". This system is designed to detect slow DoS attacks generated by a Poisson arrival process. The system consists of three main modules called a data generation module; feature selection module; and traffic classification module.

Key differences in the modified approach as follows:

i) Poisson arrival process integration: the Poisson process is explicitly used to model attack traffic, generating randomized inter-arrival times for HTTP GET requests, which mimics the stealthy behavior of slow DoS attacks.
ii) Monte Carlo simulation for thresholds: the threshold values for server parameter monitoring are derived using Monte Carlo simulation to predict the normal behavior of the server under the influence of Poisson-generated attack traffic.
iii) Poisson-related features: the inter-arrival times and rate of packet arrivals—which are core characteristics of a Poisson process—are considered key features in detecting slow DoS attacks, and are selected via the BSIG method.
iv) Poisson-aware classification: the machine learning models are trained to identify attack traffic that exhibits Poisson-like characteristics, ensuring that the system can effectively detect attacks generated with a Poisson arrival rate.

### 2.1. Poisson arrival process

The Poisson arrival process is a mathematical model used to describe independent random events happening at a constant average rate. The process is formally defined by the Poisson distribution for the number of events during a fixed time interval and the exponential distribution for the time between events. Its properties include independent increments, stationary increments, and a memoryless inter-arrival time distribution. The Poisson process is widely used in fields like queueing theory, telecommunications, traffic engineering, and reliability analysis.

A Poisson process is a stochastic process that models events happening randomly over time or space under certain conditions. The key assumptions of a Poisson process are follows:

i) Events are independent: the occurrence of one event does not affect the probability of the occurrence of another event.
ii) Stationarity: the rate of occurrence of events is constant over time (i.e., the arrival rate does not change).
iii) No simultaneous events: the process assumes that two events cannot occur at the exact same instant (although in real life, extremely close events can happen).
iv) Memoryless property: the time until the next event occurs is independent of the time since the last event.

The Poisson process is characterized by a rate parameter (denoted by λ) which defines the expected number of events per unit time. This rate is often referred to as the Poisson arrival rate.

### 2.2. Poisson arrival rate: mathematical formulation

Let N(t) denote the number of events occurring up to time t in a Poisson process with rate λ. The distribution of N(t) follows a Poisson distribution and is given by (1).

$$P(N(t) = k) = \frac{(\lambda t)^k e^{-\lambda t}}{k!} \quad (1)$$

Where λ is the arrival rate, representing the average number of events per unit time; $t$ is the time period considered; $k$ is the number of events that occurred during the time period $t$; and e is Euler's number (approximately 2.718). The mean (expected value) number of events occurring in a time interval t is as in (2).

$$E[N(t)] = \lambda t \quad (2)$$

The variance of N(t) is also as in (3).

$$Var\big(N(t)\big) = \lambda t \quad (3)$$

### 2.3. Key properties of the Poisson process

The key properties of Poisson process are as described as follows.

i) Inter-arrival times: the times between consecutive events in a Poisson process follow an exponential distribution with parameter λ, where the probability that the time between events is greater than some value t is (4).

$$P(T > t) = e^{-\lambda t} \quad (4)$$

The expected time between events (mean inter-arrival time) is 1/λ.

ii) Increment independence: the number of events occurring in disjoint time intervals are independent of each other. For example, the number of events that occur from time 0 to 10 seconds is independent of the number of events that occur from time 10 to 20 seconds.

iii) Poisson distribution of events: the number of events occurring in a fixed time interval t follows a Poisson distribution, which means that 'the probability of observing exactly k events in time t is given by (1)'.

### 2.4. Poisson arrival process in networks

The Poisson arrival process in networking is a stochastic (random) model where packets or data units arrive at a network node (e.g., a router, switch, or server) at a rate that is constant on average. The arrival times of packets are independent of each other, and the number of packets arriving within any fixed time period follows a Poisson distribution.

i) Arrival rate (λ): the Poisson arrival rate λ represents the average number of packets arriving per unit time (e.g., packets per second).

ii) Inter-arrival time: the time between consecutive packet arrivals follows an exponential distribution, meaning that the time between any two consecutive arrivals is memoryless.

In mathematical terms, if the number of packets arriving in time t is denoted as N(t), then the number of packets arriving follows a Poisson distribution is given by (1). The exponential inter-arrival time between packets is given by (4).

#### 2.4.1. Slowloris attack and Poisson arrival rate

The slowloris attack is a form of DoS attack designed to target web servers, specifically exploiting how HTTP connections are handled by the server. This attack works by keeping connections to a server open for as long as possible and sending data extremely slowly, preventing the server from closing those connections, eventually exhausting its resources and causing it to become unresponsive. The Poisson arrival rate comes into play in modeling how the attacker can generate requests in a seemingly "random" but controlled manner, mimicking the behavior of legitimate traffic but at an intentionally slow pace, making detection difficult.

#### 2.4.2. How Poisson arrival rate fits with slowloris

The Poisson arrival rate can be used to model the rate at which requests are sent during the slowloris attack. A Poisson process is a random process where events (in this case, HTTP requests or chunks of an HTTP request) occur independently over time, at a constant average rate λ. In the context of the slowloris attack as follows:

i) The attacker can generate requests or data chunks with a Poisson arrival rate. This means the attacker is sending data (e.g., HTTP request headers or partial data) at a rate determined by a Poisson distribution, where the average number of packets sent per unit of time is λ.

ii) The exponential inter-arrival times between requests are a characteristic of the Poisson process. This imposes an attacker model which sends slow, periodic, and unpredictable requests, making the attack less likely to be detected in comparison to a constant-rate attack.

### 2.5. Poisson process and inter-arrival times

In a Poisson process, the inter-arrival time generally follows an exponential distribution and describes the time for sending packet requests one after another. This is important when modeling the "random" yet controlled nature of slowloris. If the Poisson arrival rate is λ (packets per second), the time between sending two requests is exponentially distributed with the probability density function:

$$P(T > t) = e^{-\lambda t}$$

Where $T$ is the inter-arrival time (time between two packets), $t$ is time we're interested in (e.g., time until the next chunk is sent), and λ is average rate at which the attacker sends packets (chunks of an HTTP request).

For example, if the attacker sets a Poisson arrival rate of 0.1 packets per second (i.e., one packet every 10 seconds on average), the inter-arrival time between packets will follow an exponential distribution with λ=0.1. This means that some requests may be sent quickly (with a very short inter-arrival time), while others will be sent with a longer gap between them. The overall behavior, however, will still maintain an average arrival rate of λ.

### 2.6. Slowloris with Poisson arrival rate: attack model

Consider an attacker targeting a web server. The attacker generates N simultaneous connections to the server and uses the Poisson arrival rate to control how frequently the attacker sends small chunks of data on each connection. Let's model the attacker's behavior as follows:

i) Connection setup: the attacker opens N connections to the web server, one for each attack thread. The number of connections N depends on the server's maximum allowed connections.
ii) Data arrival rate: the attacker sends data (chunks of the HTTP request) at a Poisson arrival rate λ. This means that the time between sending each new chunk follows an exponential distribution. The attacker doesn't flood the server with a constant stream of data; instead, the chunks arrive at a variable rate, which on average is λ.
iii) Request partialization: instead of sending the full HTTP request at once, the attacker sends the headers of the HTTP request in very small chunks, spaced out with an exponential inter-arrival time based on the Poisson rate.
iv) Keep alive connections: each incomplete request keeps the connection alive by periodically sending small chunks. Since the chunks arrive randomly but at a steady average rate, the server cannot easily distinguish this attack from legitimate traffic.
v) Server exhaustion: over time, the server's connection pool will fill up. If the server has a limit of 100 concurrent connections, and the attacker uses 100 connections with a Poisson rate of λ=0.1 requests per second, it will take an average of 10 seconds to send a single chunk to each of the 100 connections. However, as long as the attacker continues sending data at this slow rate, the server is unable to close those connections, leaving legitimate users with no available resources.

#### 2.6.1. Why Poisson arrival helps

The motivation behind choosing Poisson arrival process is its randomness, unpredictability, controlled rate and its slow and gradual nature. Its significance is described as follows:

i) Randomness and unpredictability: Poisson arrival times introduce randomness in the attack, making it harder to detect since the request rate isn't constant or predictable. It's not a simple flood of packets, but rather packets arriving at random intervals, keeping attack less noticeable to intrusion detection systems.
ii) Controlled rate: the rate of attack can easily be controlled. The Poisson arrival rate λ can be adjusted by the attacker to keep the number of requests on the verge of being undetectable, while keeping connections to the server open for as long as possible.
iii) Slow and gradual: the packet inter-arrival times are so exponential that the attack is a very slow one. The attacker does not flood the server all at once; rather, he spreads it out in time to keep its connection pool filled over a long period of time.

Slowloris-style attacks, in which packets are sent deliberately slow and in an unpredictable manner. This method looks at incoming traffic characteristics and tries to detect anomalies with such characteristic's indicative of an attack in progress. Timings and frequencies of requests can allow for the discerning of normal user behavior from abusers trying to exhaust server resources. This permits proactive detection, thus increasing the resistivity of a server against advanced attack vectors used to disrupt the availability of service.

#### 2.6.2. Data generation module

The entire process starts with the Ryu controller's data generation module. In this module, the CPU and memory utilization of the server, along with network throughput, packet arrival, packet skewness, and connection count, among other factors, are monitored and collected continuously through the tools iostat and netstat. These parameters have a great deal of importance since they are utilized in the identification of traffic anomalies and attack patterns. The server is monitored over a period of time, where CPU utilization and packet inter-arrival time threshold values are set through Monte Carlo simulation. The simulation predicts the statistical behavior of the server while specifying thresholds exactly commensurate with normal server activity. The Poisson arrival rate is incorporated into the data generation process to model how the attacker

generates slow traffic using the Poisson distribution. The arrival rate λ of attack traffic (such as HTTP GET requests or chunks of requests) is controlled by the attacker to follow a Poisson process. This results in traffic arriving at a random, but controlled, rate that mimics legitimate traffic, making it more challenging to detect. Figure 1 shows the data generation process.

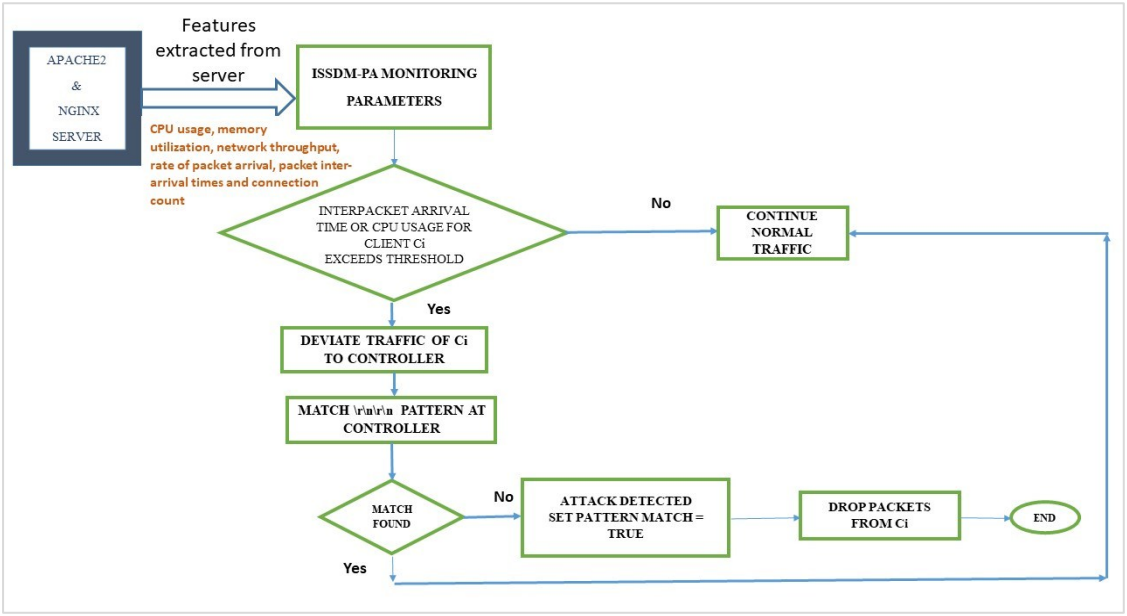


Figure 1. Flowchart of data generation process of ISSDM-PA

As packets arrive, they require processing by the server, increasing CPU utilization. In case of legitimate traffic, under normal conditions, a server processes incoming packets at a regular rate. For example, the average CPU utilization might be around 40%, with short bursts occurring when packets arrive in quick succession. In a slow DoS attack using a Poisson arrival process, the packets arrive at a very low rate (e.g., λ=0.1 packets per second) but in a way that can still overwhelm the server. The CPU utilization could rise to 80% or more as the server struggles to process an increasing number of half-open connections or long-lasting sessions, even though the attack is subtle and uses minimal bandwidth. Monte Carlo simulation set a threshold of 80% for a sample size of 400.

On the other hand, legitimate traffic might have varying inter-arrival times, but typically within a range (e.g., 1-5 seconds between packets). During slow DoS attack (Poisson modeled traffic with λ=0.1) the inter-arrival time will be longer, with packets arriving approximately every 10 seconds, but still maintaining a random, irregular pattern. This can be hard to distinguish from normal traffic by traditional detection systems. Monte Carlo simulation set a threshold of 10s for a sample size of 400. If the collected parameters exceed these predefined threshold values, the (ISSDM-PA) initiates inspection of header of the incoming HTTP GET requests. The HTTP headers are inspected to check if they match known slow DoS attack patterns, with the help of regular expressions [25]. If a match is identified, the traffic flow is classified as attack traffic, and a corresponding entry is generated with the relevant features.

### 2.6.3. Feature selection module

After generating the dataset, the feature selection module utilizes the technique called BSIG [25] to choose the most relevant features for classification of traffic. The Poisson arrival process introduces variability in the packet arrival times (inter-arrival times), which are incorporated as statistical features in the dataset. For example, fluctuating random bursts of traffic that can be denoted by Poisson-distributed inter-arrival times would help in differentiating between benign and malicious traffic. The BSIG method then selects the following features for classification: the rate of packet arrivals, inter-arrival time variability, and CPU utilization of the server.

### 2.6.4. Traffic classification using machine learning and deep learning

A range of machine learning and deep learning models are used by the traffic classification module, to classify traffic into legitimate and attack categories once the conclusive features are identified as described

by the BSIG method. In classifying the traffic, the following machine learning algorithms are used: KNN, SVM, DT, RF, NB, and MLP. These algorithms are trained on the feature set generated by the Poisson arrival-based attack traffic and legitimate traffic data. The classification process is aimed at distinguishing between attack traffic, which follows a Poisson arrival pattern with irregular inter-arrival times, and legitimate traffic, which typically has more consistent packet arrival patterns.

## 3. EXPERIMENT AND RESULTS

Figure 2 depicts the experimental setup's structure. Two Apache2 servers and one Nginx server are connected through two Zodiac FX switches within the data plane, and there are twenty clients total—ten of which are attackers and ten of which are legal clients. The ISSDM module is operated by a Ryu controller in the control location. In this work, the initial step in detecting an assault is data production. As previously mentioned, information about CPU usage, RAM utilization, network speed, packet arrival rate, packet inter-arrival periods, and connection count is gathered from both valid and malicious traffic. To determine thresholds for CPU usage and packet inter-arrival periods, a Monte Carlo simulation is performed. For a data set size of 700, BSIG identifies and selects the features utilized for training detection models like KNN, SVM, RF, DT, MLP, and NB. Table 1 displays the attack detection findings and accuracy, while Figure 3 displays the graph.

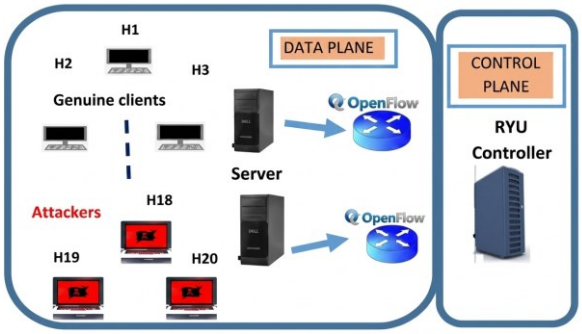


Figure 2. Topology used for ISSDM – PA [25]

Table 1. Accuracy and F1-score achieved by models trained using the BSIG feature set

| Algorithm | Accuracy | F1-score |
|---|---|---|
| DT | 0.85 | 0.90 |
| SVM | 1 | 1 |
| KNN | 0.89 | 0.93 |
| RF | 1 | 1 |
| NB | 0.85 | 0.92 |
| MLP | 0.91 | 0.91 |

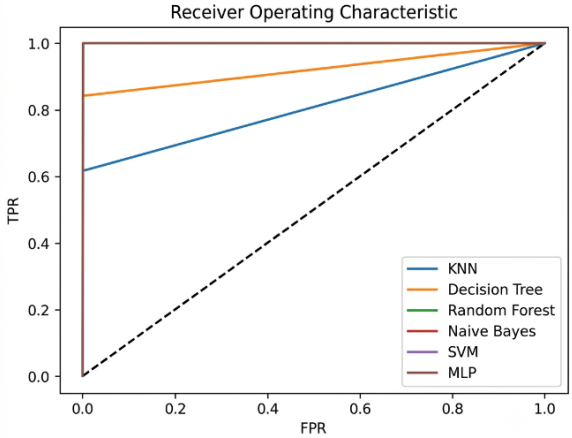


Figure 3. ROC curves of models trained using the features selected by BSIG

## 4. CONCLUSION

The proposed methodology ISSDM-PA uses a Poisson arrival process for effective simulation and detection of stealthy slow DoS attacks. Attack traffic is modeled using Poisson distributions in order to mimic legitimate traffic patterns such as varied inter-arrival times and randomized bursts of packets, thus allowing detection systems of conventional design to elaborate the differences between malicious and normal traffic. The methodology utilizes Monte Carlo simulations to define dynamic thresholds per server parameter that considers variations based on the tracked Poisson-modeled traffic. These thresholds are further used to identify the operational irregularities with respect to resource usage, connection counts, and packet arrival patterns, all of which describe related slow DoS attacks. The procedure utilizes the BSIG feature-selection process in order to recognize salient features, such as the inter-arrival time, variability, and resource usage, which are central to differentiate between normal and Poisson-based malicious attack traffic. Those features are passed into classifiers of machine learning like NB, SVM, RF, and MLP, where they guarantee accurate detections. This variability introduced by the Poisson arrival process exhibits stochastic characteristics that offer its reflection on real-world customer behaviors, improving the robustness against evolving strategies of attack. It not only brings in its vision of improving detecting slow DoS attacks with higher accuracies but also reduces false positives via ad hoc practice adjusting to network conditions dynamically. With a balanced combination of realistic attack simulation, advanced feature selection, and machine learning, ISSDM-PA proposes a scalable and efficient solution to modern-day network security dilemmas. This would be a big step further in the detection of low-rate stealthy attacks which evade traditional security measures.

## FUNDING INFORMATION

Authors state no funding involved.

## AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

| Name of Author | C | M | So | Va | Fo | I | R | D | O | E | Vi | Su | P | Fu |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Prathima Mabel John | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ | ✓ | ✓ | | | ✓ | |
| Mamatha Ram | | ✓ | | | | ✓ | | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| Vani Kurugod | ✓ | | ✓ | ✓ | | | ✓ | | | ✓ | ✓ | | ✓ | ✓ |
| Ashwathanarayana Reddy | | | | | | | | | | | | | | |
| Santhosh Babu | ✓ | | ✓ | ✓ | | | ✓ | | | ✓ | ✓ | | ✓ | ✓ |
| Rama Mohan Babu | ✓ | | ✓ | ✓ | | | ✓ | | | ✓ | ✓ | | ✓ | ✓ |
| Kasturi Nagappasetty | | | | | | | | | | | | | | |

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

## CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

## DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, [PMJ], upon reasonable request.

## REFERENCES

[1] X. Wang, T. Dan, Y. Feng, Z. Qin, B. Xiong, and Y. Liu, "An LDoS attack detection method based on FSWT time–frequency distribution," *Expert Systems with Applications*, vol. 256, Dec. 2024, doi: 10.1016/j.eswa.2024.125006.

[2] A. Reed, L. Dooley, and S. K. Mostefaoui, "SA-IDS: a single attribute intrusion detection system for Slow DoS attacks in IoT networks." *Internet of Things*, vol. 30, Mar. 2025, doi: 10.1016/j.iot.2025.101512.

[3] N. M. Y.-Naula, C. V.-Rosales, and J. A. P.-Díaz, "SDN/NFV-based framework for autonomous defense against slow-rate DDoS attacks by using reinforcement learning," *Future Generation Computer Systems*, vol. 149, pp. 637–649, Dec. 2023, doi: 10.1016/j.future.2023.08.007.
[4] F. S. de L. Filho, F. A. F. Silveira, A. De M. B. Junior, G. V.-Solar, and L. F. Silveira, "Smart detection: an online approach for DoS/DDoS attack detection using machine learning," *Security and Communication Networks*, vol. 2019, pp. 1–15, Oct. 2019, doi: 10.1155/2019/1574749.
[5] N. Muraleedharan and B. Janet, "A deep learning based HTTP slow DoS classification approach using flow data," *ICT Express*, vol. 7, no. 2, pp. 210–214, Jun. 2021, doi: 10.1016/j.icte.2020.08.005.
[6] C. L. Calvert and T. M. Khoshgoftaar, "Impact of class distribution on the detection of slow HTTP DoS attacks using big data," *Journal of Big Data*, vol. 6, no. 1, Dec. 2019, doi: 10.1186/s40537-019-0230-3.
[7] R. F. Kapourchali, R. Mohammadi, and M. Nassiri, "P4httpGuard: detection and prevention of slow-rate DDoS attacks using machine learning techniques in P4 switch," *Cluster Computing*, vol. 27, no. 6, pp. 8047–8064, 2024, doi: 10.1007/s10586-024-04407-5.
[8] M. Wang, Y. Lu, and J. Qin, "A dynamic MLP-based DDoS attack detection method using feature selection and feedback," *Computers & Security*, vol. 88, Jan. 2020, doi: 10.1016/j.cose.2019.101645.
[9] V. Hnamte, A. A. Najar, H. N.-Nguyen, J. Hussain, and M. N. Sugali, "DDoS attack detection and mitigation using deep neural network in SDN environment," *Computers and Security*, vol. 138, Mar. 2024, doi: 10.1016/j.cose.2023.103661.
[10] N. M. Y.-Naula, C. V.-Rosales, J. A. P.-Díaz, and D. F. Carrera, "A flexible SDN-based framework for slow-rate DDoS attack mitigation by using deep reinforcement learning," *Journal of Network and Computer Applications*, vol. 205, 2022, doi: 10.1016/j.jnca.2022.103444.
[11] D. Y. Zebua, C. S. C. Waruwu, K. Zebua, M. Zai, A. Lase, and O. L. Ofel, "Simulasi serangan DOS menggunakan SLOWHTTPTEST," *Jurnal Komputer Teknologi Informasi Sistem Informasi*, vol. 4, no. 2, pp. 409–419, 2025, doi: 10.62712/juktisi.v4i2.402.
[12] V. Rios, P. Inacio, D. Magoni, and M. Freire, "Detection of slowloris attacks using machine learning algorithms," in *Proceedings of the ACM Symposium on Applied Computing*, Apr. 2024, pp. 1321–1330, doi: 10.1145/3605098.3635919.
[13] N. Tripathi, "Delays have dangerous ends: slow HTTP/2 DoS attacks into the wild and their real-time detection using event sequence analysis," *IEEE Transactions on Dependable and Secure Computing*, vol. 21, no. 3, pp. 1244–1256, 2024, doi: 10.1109/TDSC.2023.3276062.
[14] R. K. Batchu and H. Seetha, "A generalized machine learning model for DDoS attacks detection using hybrid feature selection and hyperparameter tuning," *Computer Networks*, vol. 200, 2021, doi: 10.1016/j.comnet.2021.108498.
[15] M. Yeasir, M. Morshed, and M. Fakrul, "A practical approach and mitigation techniques on application layer DDoS attack in Web server," *International Journal of Computer Applications*, vol. 131, no. 1, pp. 13–20, 2015, doi: 10.5120/ijca2015907209.
[16] V. N. Okorogu and Okafur, "Improving traffic management in a data switched network using an adaptive discrete time Markov modulated Poisson process," *European Journal of Engineering Research and Science*, vol. 1, no. 2, pp. 24–31, 2021.
[17] A. F. G.-Fernandez and J. Xiao, "Trajectory Poisson multi-Bernoulli mixture filter for traffic monitoring using a drone," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 1, pp. 402–413, 2024, doi: 10.1109/TVT.2023.3310742.
[18] B. Y. Lichtzinder, A. Y. Privalov, and V. I. Moiseev, "Batch Poissonian arrival models of multiservice network traffic," *Problems of Information Transmission*, vol. 59, no. 1, pp. 63–70, 2023, doi: 10.1134/S0032946023010064.
[19] D. Salimzyanova and E. Lisovskaya, "Identification of network traffic using neural networks," in *Communications in Computer and Information Science*, 2024, pp. 76–90, doi: 10.1007/978-3-031-65385-8_6.
[20] Q. Yang and X. Fu, "An extended queueing model for vehicles at signalized intersections considering the platoon correlated arrivals," *Physica A: Statistical Mechanics and its Applications*, vol. 635, 2024, doi: 10.1016/j.physa.2023.129483.
[21] R. Ramanathan and V. Varadharajan, "Performance approximations of Markov modulated Poisson process arrival queue with Markovian service using matrix geometric approach," *Yugoslav Journal of Operations Research*, vol. 34, no. 2, pp. 245–256, 2024, doi: 10.2298/YJOR230715027R.
[22] A. A. A.-Shareha, M. M. Abualhaj, A. Alshahrani, and B. Al-Kasasbeh, "A four-state Markov model for modelling bursty traffic and benchmarking of random early detection," *International Journal of Data and Network Science*, vol. 8, no. 2, pp. 1151–1160, 2024, doi: 10.5267/j.ijdns.2023.11.019.
[23] A. I. Alattar, A. N. Al-Shamani, and M. S. Mohmood, "Issues in IoT traffic modelling and analysis," in *2nd International Conference on Engineering and Science to Achieve the Sustainable Development Goals*, 2024, doi: 10.1063/5.0199702.
[24] F. Zhang, J. Lu, X. Hu, and Q. Meng, "A stochastic dynamic network loading model for mixed traffic with autonomous and human-driven vehicles," *Transportation Research Part B: Methodological*, vol. 178, 2023, doi: 10.1016/j.trb.2023.102850.
[25] P. M. John and R. M. B. K. Nagappasetty, "An intelligent system to detect slow denial of service attacks in software-defined networks," *International Journal of Electrical and Computer Engineering*, vol. 13, no. 3, pp. 3099–3110, 2023, doi: 10.11591/ijece.v13i3.pp3099-3110.

## BIOGRAPHIES OF AUTHORS

**Dr. Prathima Mabel John** is an associate professor at Dayananda Sagar College of Engineering, Visvesvaraya Technological University (VTU), Bengaluru, Karnataka, India. She has received her Ph.D. from VTU, Belagavi, Karnataka, India. She received her Bachelor of Engineering and master of Technology degree in Computer Science and Engineering from VTU, Belagavi, Karnataka, India. She has about 17 years of experience in teaching and industry together. Her areas of interest are computer networks, software defined networks, mobile networks, network security, and machine learning. She can be contacted at email: prathimamabel-ise@dayanandasagar.edu.

**Dr. Mamatha Ram** received Ph.D. and M.Tech. degree in Computer Science and Engineering from Visvesvaraya Technological University, Belgaum, B.E. degree in Computer Science and Engineering from Kuvempu University. She is having 26 years of teaching experience. Presently working as associate professor in Department of Computer Science and Engineering at BMS College of Engineering, Bangalore. Her research interests include information security and cancelable biometrics. She can be contacted at email: krm.ise@bmsce.ac.in.

**Dr. Vani Kurugod Ashwathanarayana Reddy** is associate professor at BNM Institute of Technology, Visvesvaraya Technological University (VTU), Bengaluru, Karnataka, India. She received her Bachelor of Engineering and Master of Technology degree in Computer Science and Engineering from VTU, Belagavi, Karnataka, India. She is currently pursuing Ph.D. from VTU, Belagavi, Karnataka, India. She has about 17 years of experience in teaching. Her areas of interest are computer networks, QoS in software defined networks, mobile networks, serverless computing, and machine learning. She can be contacted at email: vanireddy@bnmit.in.

**Dr. Santhosh Babu** working as an assistant professor in the Department of Electronics and Telecommunication Engineering, Dayananda Sagar College of Engineering has about 18 years of teaching experience. He received his B.E. degree in Medical Electronics from Visvesvaraya Technological University and M.Tech. degree in Visual Information in Embedded Systems from Indian Institute of Technology, Kharagpur, West Bengal. He received doctoral degree in Medical Image Security from Visvesvaraya Technological University. He has published research papers in refereed international journals and research papers in the proceedings of various international conferences. He received funds from various funding agencies in the cryptography and security domain. He is authored various book chapters in the area of cryptography and artificial intelligence. He can be contacted at email: santhoshmehtre@gmail.com.

**Dr. Rama Mohan Babu Kasturi Nagappasetty** is currently working as professor in the Department of Computer Science and Engineering (Data Science) at Dayananda Sagar College of Engineering, Bengaluru, India. He obtained his B.Tech. in Computer Engineering from Mangalore University, India, M.S from BITS-PILANI, India and Ph.D. from Dr. MGR University, India. His areas of interest are computer networks, wireless mobile networks, software defined networks, and network security. He can be contacted at email: ramamohanbabu-csds@dayanandasagar.edu.