Modeling and Detection of Flooding-Based Denial-of-Service Attack in Wireless *Ad Hoc* Network Using Bayesian Inference

N. Nishanth and A. Mujeeb

***Abstract*—Wireless *ad hoc* networks are widely useful in locations where the existing infrastructure is difficult to use, especially during the situations like flood, earthquakes, and other natural or man-made calamities. Lack of centralized management and absence of secure boundaries make these networks vulnerable to various types of attacks. Moreover, the mobile nodes used in these networks have limited computational capability, memory, and battery backup. Flooding-based denial-of-service (DoS) attack, which results in denial of sleep attack, targets the mobile node's constrained resources which results in excess consumption of battery backup. In SYN flooding-based DoS attack, the attacker sends a large number of spoofed SYN packets which not only overflow the target buffer but also creates network congestion. The present article is divided into three parts: 1) mathematical modeling for SYN traffic in the network using Bayesian inference; 2) proving the equivalence of Bayesian inference with exponential weighted moving average; and 3) developing an efficient algorithm for the detection of SYN flooding attack using Bayesian inference. Based on the comprehensive evaluation using mathematical modeling and simulation, the proposed method can successfully defend any type of flooding-based DoS attack in wireless *ad hoc* network with higher detection accuracy and extremely lower false detection rate.**

***Index Terms*—Bayesian inference, denial-of-service (DoS) attack, denial of sleep attack, SYN flooding attack, wireless *ad hoc* network.**

I. Introduction

WIRELESS *ad hoc* networks are self-organizing wireless networks formed by mobile node without using any fixed infrastructure. These networks have lower deployment and less operational expenses which enables its widespread use in everyday life. Ease of application, flexibility, and scalability enable the use of these networks in a wide variety of applications like disaster management, military battlefield, and Internet of Things (IoT) based networks [1]. Lack of centralized management and absence of secure boundaries make these networks vulnerable to various types of attacks. Also, the mobile nodes used in these networks have limited computational capability, memory, and battery backup. As a result, the defense mechanism designed for traditional network may be infeasible for these networks [2]. The ultimate goal of the security solutions for wireless *ad hoc* networks is to provide security services, such as authentication, confidentiality, integrity, availability, and nonrepudiation to mobile users [3]. Out of these security services, availability refers to the survivability of the network in the presence of attacks. Threat to availability is termed as denial-of-service (DoS) attack. The main outcome of DoS attacks is either to completely exhaust certain resources like memory and battery backup or to bring down an entire network so that the legitimate users are not able to get access to services offered by the network. As a result, the assurance of survivability of the network is compromised. Packet flooding is one of the resource consuming DoS attacks and is spreading faster, causing more damages [4]. Packet flooding also results in denial of sleep attack by targeting the mobile node's constrained resources which results in excessive consumption of battery backup. A typical wireless sensor node (Tmote Sky) consumes 64.68 mW in receive mode and 0.114 mW in sleep mode [5]. Thus, by using two standard 3000-mAh AA batteries, the battery backup of the sensor node will last 3300 d in sleep mode, but only 5.8 d in receive mode. DoS attack prevents the victim node from entering to sleep mode and, as a result, the attack can degrade the lifetime of the node in an exponential manner. As these attacks can be conducted anywhere and at anytime with varying intensity, particularly in the wireless *ad hoc* networking environment, the detection and prevention of such attacks are of prime importance. By measuring the traffic intensity or volume of traffic coming through a particular link, DoS attacker can be easily identified. Flooding attack can be launched from different layers of TCP/IP protocol stack. A popular and clever DoS attack launched from transport layer is SYN flooding attack [6]. Route Request (RREQ) flooding and Hello flooding are flooding attacks launched from network layer. The present work discusses SYN flooding-based DoS attack and its detection algorithm which is tested using a standard dataset. The detection algorithm can be extended for the detection of other types of flooding attacks like RREQ flooding.

A. SYN Flooding-Based DoS Attack

SYN flooding attack exploits the vulnerability in TCP specifications [7]. In order to establish a connection with the server node, the source node sends a SYN packet to the server node and the connection is considered to be in half-open state. The server

Manuscript received August 26, 2019; revised December 11, 2019, March 2, 2020, and March 27, 2020; accepted March 28, 2020. *(Corresponding author: N. Nishanth.)*

N. Nishanth is with the Department of Electronics and Communication Engineering, TKM College of Engineering, Kollam, Kerala, India (e-mail: nishtkm@gmail.com).

A. Mujeeb is with the International School of Photonics, Cochin University of Science and Technology, Kochi, India (e-mail: mujeebpoovar@gmail.com).

Digital Object Identifier 10.1109/JSYST.2020.2984797

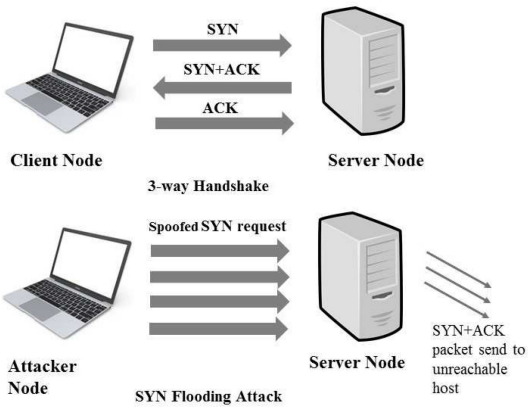

Fig. 1. Illustration of three-way handshake and SYN flooding attack.

has a backlog queue to maintain all these half-open connections. If the server node is ready to give the connection, it responds with a SYN-ACK packet. On receiving the SYN-ACK packet, the source node sends the final ACK packet to the server node and hence completes the three-way handshake process. In SYN flooding-based DoS attack, an attacker may send a large number of spoofed SYN packets (fake packets) to the victim server. Since the SYN request is spoofed, the victim server never receives the final ACK packet from the client to complete the three-way handshake which is shown in Fig. 1. Since the backlog queue of the victim server is of finite size, flooding of spoofed SYN requests can easily exhaust the victim server's backlog queue, causing all new incoming legitimate SYN request to be dropped. Moreover, thousands of these flooding packets cause congestion in the network. In a wireless *ad hoc* network, the attacker sends a large number of SYN packets to a remote node which excessively consume battery backup and computational resources available at the victim node as well as relay nodes which results in complete exhaustion of the battery resources of these nodes. As a result, these nodes are removed from the network, thereby affecting the normal operation of the network. The attacker deployed in the present network model is assumed to have unlimited resources, especially unlimited battery backup and computational resources compared to other nodes in the network.

B. Proposed Idea

In this article, a novel method is proposed for modeling the SYN traffic in the network using Bayesian inference, and the mean of Bayesian inference is used as a metric for SYN arrival rate (SAR) in the incoming traffic which is discussed in Section III. Since exponential weighted moving average (EWMA) is a widely used method for the detection of flooding attack, the proposed article proved the equivalence of mean of Bayesian inference with EWMA using three lemmas which is illustrated in Section IV. Finally, an efficient algorithm for detecting SYN flooding-based DoS attack in wireless *ad hoc* network based on proposed mathematical model is illustrated in Section V. A comparative study of the proposed algorithm with state-of-the-art techniques like Fixed Threshold Algorithm (FTA) and Adaptive Threshold Algorithm (ATA) is also carried out in Section VI.

II. Related Works on Attack Detection

DoS attack detection mechanism suggested for the Internet can be used for attack detection in wireless *ad hoc* network, but attacker traceback mechanism suggested for Internet is not effective for these networks due to limited resources and computational ability of nodes [8]. Siris and Papagalou [9] suggested an ATA and cumulative sum (CUSUM) algorithm for the detection of SYN flooding-based DoS attack. Since the threshold is set adaptively based on mean SAR, the ATA misinterprets attack traffic as normal traffic after some samples of attack traffic due to persistent attack which further results in increased false alarm rate and lower accuracy. Geetha and Sreenath [10] proposed a mechanism for early detection of SYN flooding attack in mobile *ad hoc* network (MANET) by monitoring the number of SYN packets, SYN-ACK packets, and final ACK packets exchanged between a node and a multimedia server. The number of half-open connections was calculated by the multimedia server and a decision is made as malicious node if the number of half-open connection is greater than a threshold. Mohammadi and Ghaffari [11] proposed a method for defending flooding-based DoS attack in MANET by including a misbehavior detection system and a flooding detection system in the network. But these methods needed a dedicated server node for monitoring the traffic. Wang *et al.* [12] proposed a traffic prediction model to predict SYN traffic with the past traffic and then CUSUM algorithm is used for the detection of SYN flooding attack. Bhalodiya and Vaghela [13] suggested a threshold-based detection scheme for detecting RREQ flooding attack in MANET. In this scheme, if the rate of RREQ from a sender node is lower than a threshold, then the node is considered as normal, otherwise the sender node is identified as malicious. But these methods are suffering from higher false positives and misdetection due to seasonal variations in the network traffic. Conti *et al.* [14] proposed a framework for detection of distributed DDoS attacks in software-defined network using CUSUM and ATA. The performance evaluation of this method is done using Defense Advanced Research Projects Agency (DARPA) dataset. Zargar *et al.* [15] conducted a survey of destination-based detection and traceback schemes for DDoS attack. Many researchers used entropy as an effective metric for detecting anomaly in the network traffic [16]. Application of chaos theory and Lyapunov exponent is also effective in detecting flooding-based DDoS attack [17],[18]. Kim *et al.* [19] studied the effect of data flooding attack in wireless *ad hoc* network and suggested a period-based defense mechanism which enhances the throughput of the burst traffic. Lyamin *et al.* [20] suggested a real-time detection of jamming-based DoS attack in vehicular *ad hoc* network (VANET) by proposing a hybrid detector. It aimed at detecting the loss of beacon messages in multiple collision times in a detection period of length T compared to the model-based detector [21] which can only detect loss of exactly one beacon message. Wei *et al.* [22] suggested to incorporate queue management algorithm with the detection schemes used for defending DoS attack to improve network's defensive capabilities. Pham *et al.* [23] proposed a flooding detection based on encounter record method for differentiating flooding attack and legitimate burst transmission in delay tolerant network. However, this method is suffering

from increased network overhead and power consumption due to the exchange of encounter record between the nodes and usage of digital signatures. Faghihniya *et al.* [24] proposed balanced ad hoc on demand distance vector (AODV) (B-AODV) method in VANET for the detection of RREQ flooding attack without using additional control packets for detecting malicious nodes. In this method, an adaptive threshold is computed based on network conditions and node behavior which is further used for the acceptance or rejection of RREQ packet. But this method is also suffering from higher false alarm rate since legitimate nodes are misdetected as an attacker if the RREQ rate is greater than the computed threshold.

III. Modeling of SYN Traffic Using Beta Distribution

In the present article, a novel method for modeling of SYN traffic is done using Beta distribution by sampling the network traffic at regular intervals [25], [26]. Let p be the probability of getting a SYN packet in a given sample. The exact value of p for normal traffic is generally not known initially. So we assume that p is uniformly distributed in the interval (0,1). After getting each sample, p is updated through Bayesian inference. Let A be the event of getting k SYN packets in a sample of n packets. Each sample consisting of n packets can be considered as a sequence of n Bernoulli trial. Here, k is the number of SYN packets obtained (success) out of the n packets captured. In the present modeling of TCP traffic, the SYN packets are considered as successful and other packets as failures. Let $f(p)$ be the initial probability density function (pdf) of p. After observing each sample, the posterior distribution of p is to be determined

$$f(p|A) = \frac{P(A|p)f(p)}{\int_0^1 P(A|p)f(p)dp} \tag{1}$$

where

$$P(A|p) = p^k(1-p)^{n-k}. \tag{2}$$

Evaluating (1) gives

$$f(p|A) = \frac{(n+1)!p^k(1-p)^{n-k}}{(n-k)!k!}. \tag{3}$$

But the general expression for Beta distribution [25] having parameters α and $\boldsymbol{\beta}$ is

$$\text{Beta}(\alpha, \boldsymbol{\beta}) = \frac{p^{\alpha-1}(1-p)^{\boldsymbol{\beta}-1}}{B(\alpha, \boldsymbol{\beta})} \tag{4}$$

where $B(\alpha, \boldsymbol{\beta})$ plays a role of normalizing constant

$$B(\alpha, \boldsymbol{\beta}) = \frac{\Gamma(\alpha)\Gamma(\boldsymbol{\beta})}{\Gamma(\alpha+\boldsymbol{\beta})} \tag{5}$$

where α can be considered as one more than the number of success and β can be considered as one more than the number of failures. As a result, $\alpha = k+1$, $\boldsymbol{\beta} = n-k+1$. Thus, (4) and (5) give

$$\text{Beta}(\alpha, \boldsymbol{\beta}) = \frac{p^k(1-p)^{n-k}}{B(\alpha, \boldsymbol{\beta})} \tag{6}$$

and

$$B(\alpha, \boldsymbol{\beta}) = \frac{(n-k)!k!}{(n+1)!} \tag{7}$$

since $\Gamma(\alpha) = (\alpha-1)!$. Thus, (3) is equivalent to (4). In the subsequent samples, the posterior pdf evaluated will become the prior pdf for evaluating the next posterior pdf. Now, the mean value μ of the Beta distribution is as follows:

$$\mu = \frac{\alpha}{\alpha+\boldsymbol{\beta}}. \tag{8}$$

In the proposed traffic modeling, μ can be evaluated as follows:

$$\mu = \frac{k+1}{n+2}. \tag{9}$$

Before starting the experiment, we have $k = n = 0$. Thus

$$\mu = \frac{1}{2} \tag{10}$$

which justifies the assumption of prior distribution is uniform in the interval (0,1)

$$U(0,1) \equiv B(1,1). \tag{11}$$

When Beta distributions are used as prior distribution, the posterior distribution is also Beta distributed since it is self-conjugate. In this way, p for the normal traffic is updated through Bayesian learning. In the proposed solution, μ can be considered as a metric for representing SAR which is defined as the ratio of SYN packets to the total number of captured packet in a sample. SAR can be further used as a metric for detecting SYN flooding attack.

IV. Equivalence of Mean of Beta Distribution and EWMA

In EWMA, let ω_N be the SAR at Nth sampling instant and $\bar{\omega}_{N-1}$ be the average SAR of the traffic till $(N-1)$th samples. Then, average SAR for N samples is given by [9]

$$\bar{\omega}_N = \beta\omega_N + (1-\beta)\bar{\omega}_{N-1} \tag{12}$$

where β is the weightage given to SAR of the current sample (EWMA factor) and $(1-\beta)$ is the weightage given to average SAR till $(N-1)$th sample. Value of β is carefully selected from the interval (0,1) based on the traffic condition.

Lemma 4.1: For large number of samples, EWMA of SAR ($\bar{\omega}_N$) for normal traffic approaches to the sum of arithmetic average of instantaneous SAR and arithmetic average of its variation.

Proof: Let $\omega_1, \omega_2, \ldots, \omega_N$ be the instantaneous SAR for the sample1, sample2, $\ldots$, sample$N$, respectively

$$\bar{\omega}_1 = \beta\omega_1 + (1-\beta)\bar{\omega}_0 \tag{13}$$

where $\bar{\omega}_0$ is the initial average SAR before starting the experiment. We should initialize $\bar{\omega}_0$ to an arbitrarily small value instead of 0 in order to reduce false alarm rate during the initial phase of experiment. Thus, let $\bar{\omega}_0 = \gamma$

$$\begin{aligned}\bar{\omega}_2 &= \beta\omega_2 + (1-\beta)\bar{\omega}_1 \\ &= \beta\omega_2 + (1-\beta)[\beta\omega_1 + (1-\beta)\gamma] \\ &\vdots \\ \bar{\omega}_N &= \beta\omega_N + (1-\beta)\bar{\omega}_{N-1}\end{aligned}$$

$$= \beta\omega_N + \beta(1-\beta)\omega_{N-1} + \beta(1-\beta)^2\omega_{N-2} + \beta(1-\beta)^3\omega_{N-3} + \cdots + (1-\beta)^N\gamma.$$

Let ω be the simple average SAR for normal traffic. Hence, the instantaneous SAR for each sample varies about ω. Let the variation of SAR about ω be $\Delta_1, \Delta_2, \ldots, \Delta_N$, respectively, for the sample 1, sample 2, . . . ,sample N. As a result

$$\begin{aligned}\bar{\omega}_N &= \beta(\omega+\Delta_N) + \beta(1-\beta)(\omega+\Delta_{N-1}) + \beta(1-\beta)^2(\omega+\Delta_{N-2}) + \cdots + (1-\beta)^N\gamma \\ &= \beta\omega[1+(1-\beta)+(1-\beta)^2+\cdots+(1-\beta)^N] + \beta[\Delta_N + (1-\beta)\Delta_{N-1} + \cdots + (1-\beta)^N\gamma.\end{aligned}$$

For a large number of samples, $(1-\beta)^N\gamma \approx 0$ and is neglected

$$\bar{\omega}_N = \beta\omega\left[\frac{1-(1-\beta)^{N-1}}{1-(1-\beta)}\right] + \beta[\Delta_N + (1-\beta)\Delta_{N-1} + \cdots + (1-\beta)^{N-1}\Delta_1].$$

Subsequently, $(1-\beta)^{N-1} \approx 0$ and hence neglected

$$\bar{\omega}_N = \beta\omega\left[\frac{1}{\beta}\right] + \beta[\Delta_N + (1-\beta)\Delta_{N-1} + \cdots + (1-\beta)^N\Delta_1].$$

Let Δ be the simple average of variation of SAR about ω. Since Δ is very small compared to ω for normal traffic, $\Delta_1 = \Delta_2 = \cdots = \Delta_{N-1} \approx \Delta$. Then

$$\begin{aligned}\bar{\omega}_N &= \omega + \beta\Delta[1+(1-\beta)+(1-\beta)^2+\cdots+(1-\beta)^N] \\ &= \omega + \beta\Delta\left[\frac{1}{1-(1-\beta)}\right] \\ \bar{\omega}_N &= \omega + \Delta. \qquad (14)\end{aligned}$$

Thus, from (14), it is clear that for a large number of samples, EWMA boils down to the sum of arithmetic average of instantaneous SAR and arithmetic average of its variation. ■

Lemma 4.2: For a large number of samples taken from normal traffic, the mean of Beta distribution is equivalent to EWMA.

Proof: Let $k_1, k_2, \ldots, k_N$ be the number of SYN packets (success) received during each samples of fixed size, say *n*. The mean of Beta distribution is modeled as a metric for SAR. The mean of Beta distribution after getting first sample is

$$\mu_1 = \frac{k_1+1}{n+2}$$

$$\mu_2 = \frac{k_1+k_2+1}{2n+2}$$

$$\vdots$$

$$\mu_N = \frac{k_1+k_2+\cdots+k_N+1}{Nn+2}.$$

For a large value of *N*, mean of Beta distribution after *N* samples

$$\mu_N \approx \frac{k_1+k_2+\cdots+k_N}{Nn}. \qquad (15)$$

Let k be the arithmetic average of SYN packets (success) for *N* samples. Then, $k_1 = k+\delta_1, k_2 = k+\delta_2, \ldots, k_N = k+\delta_N$, where $\delta_1, \delta_2, \ldots, \delta_N$ are variation of SYN packets from mean value *k*

$$\begin{aligned}\mu_N &= \frac{k+\delta_1+k+\delta_2+\cdots+k+\delta_N}{Nn} \\ &= \frac{Nk}{Nn} + \frac{\delta_1+\delta_2+\cdots+\delta_N}{Nn}.\end{aligned}$$

Let δ be the simple average of variation of SYN packets about k. Since δ is very small compared to k for normal traffic, $\delta_1 = \delta_2 = \cdots = \delta_{N-1} \approx \delta$. Then

$$\begin{aligned}&= \frac{k}{n} + \frac{N\delta}{Nn} \\ \mu_N &= \frac{k}{n} + \frac{\delta}{n} \qquad (16)\end{aligned}$$

where $\frac{k}{n} = \omega$ and $\frac{\delta}{n} = \Delta$. Thus, the mean of Beta distribution after *N* samples is given by

$$\mu_N = \omega + \Delta \qquad (17)$$

where ω is the arithmetic average of instantaneous SAR and Δ is the arithmetic average of variation in instantaneous SAR. Hence, from (14) and (17), mean of Beta distribution is equivalent to EWMA for normal traffics. ■

Lemma 4.3: For persistent attack, mean of Beta distribution is equivalent to EWMA.

Proof: Considering the case of attack traffic merged with normal traffic and assuming that the attack persists for d number of samples out of N samples

$$\begin{aligned}\bar{\omega}_N &= \beta(\omega+\phi_d) + \beta(1-\beta)(\omega+\phi_{d-1}) + \cdots \\ &\quad + \beta(1-\beta)^{N-d-1}(\omega+\phi_1) \\ &\quad + \beta(1-\beta)^{N-d}(\omega+\Delta_{N-d}) + \cdots \\ &\quad + \beta(1-\beta)^{N-1}(\omega+\Delta_1) + (1-\beta)^N\gamma\end{aligned}$$

where ω is the mean SAR for normal traffic, ϕ_d is the variation of SAR about mean value at Nth sampling instant during attack, and Δ_{N-d} is the variation of SAR about mean value at $(N-d)$th sampling instant during normal traffic with $\Delta \ll \phi$.

Assume that attack started after t samples and persisted for d samples such that $t+d = N$. Now

$$\begin{aligned}\bar{\omega}_N &= \beta(\omega+\phi_d) + \beta(1-\beta)(\omega+\phi_{d-1}) + \cdots \\ &\quad + \beta(1-\beta)^d(\omega+\phi_1) + \beta(1-\beta)^{d+1}(\omega+\Delta_t) \\ &\quad + \beta(1-\beta)^{d+2}(\omega+\Delta_{t-1}) + \cdots \\ &\quad + \beta(1-\beta)^{N-1}(\omega+\Delta_1) + (1-\beta)^N\gamma\end{aligned}$$

$$\begin{aligned}\bar{\omega}_N &= \beta\omega\left[1+(1-\beta)+(1-\beta)^2+\cdots+(1-\beta)^{d-1}\right]\\ &\quad+\beta\left[\phi_1+(1-\beta)\phi_2+(1-\beta)^2\phi_3+\cdots+(1-\beta)^{d-1}\phi_d\right]\\ &\quad+\beta(1-\beta)^d\omega\left[1+(1-\beta)+(1-\beta)^2+\cdots+(1-\beta)^t\right]\\ &\quad+\beta(1-\beta)^d\left[\Delta_1+(1-\beta)\Delta_2+\cdots+(1-\beta)^{t-1}\Delta_t\right]\\ &\quad+(1-\beta)^N\gamma\\ \bar{\omega}_N &= \beta\omega\left[\frac{1-(1-\beta)^{d-1}}{1-(1-\beta)}\right]+\beta\phi\left[\frac{1-(1-\beta)^{d-1}}{1-(1-\beta)}\right]\\ &\quad+\beta(1-\beta)^d\omega\left[\frac{1-(1-\beta)^{t-1}}{1-(1-\beta)}\right]\\ &\quad+\beta(1-\beta)^d\Delta\left[\frac{1-(1-\beta)^{d-1}}{1-(1-\beta)}\right]\\ &\quad+(1-\beta)^N\gamma\\ \bar{\omega}_N &\approx \omega+\phi+\beta(1-\beta)^d\omega+\beta(1-\beta)^d\Delta \qquad (18)\end{aligned}$$

since, for a large value of d, $\beta(1-\beta)^d \approx 0$

$$\bar{\omega}_N \approx \omega+\phi$$

since SAR during attack is very much high compared to SAR of normal traffic. To check the equivalence of EWMA with mean of Bayesian inference in the presence of attack traffic

$$\begin{aligned}\mu_N &= \frac{k_1+k_2+\cdots+k_N+1}{Nn+2}\\ \mu_N &= \frac{k_1+k_2+\cdots+k_N}{Nn}\\ &= \frac{(k+\delta_1)+\cdots+(k+\delta_t)+(k+\rho_1)+\cdots+(k+\rho_d)}{Nn}\\ &= \frac{k}{n}+\frac{\delta_1+\delta_2+\cdots+\delta_t+\rho_1+\rho_2+\cdots+\rho_d}{Nn}\\ &= \frac{k}{n}+\frac{\delta_1+\delta_2+\cdots+\delta_t}{Nn}+\frac{\rho_1+\rho_2+\cdots+\rho_d}{Nn} \qquad (19)\end{aligned}$$

where $\delta_1, \delta_2, \ldots, \delta_N$ are the variation of SYN packets during normal traffic and assuming $\delta_1, \delta_2, \ldots, \delta_t \approx \delta$. Also, $\rho_1, \rho_2, \ldots, \rho_d$ are the variation of SYN packets during attack and assuming $\rho_1, \rho_2, \ldots, \rho_d \approx \rho$. Thus, $\delta \ll \rho$

$$\mu_N = \omega+\frac{t\delta}{Nn}+\frac{d\rho}{Nn} \qquad (20)$$

where $\omega = \frac{k}{n}$ and for $d \approx N$ and $t \approx 0$

$$\mu_N \approx \omega+\phi$$

where $\frac{\rho}{n} = \phi$. Thus, mean of Bayesian inference is equivalent to EWMA for attack traffic also. That is, the equivalence of EWMA and mean of Beta distribution for attack traffic is true only when d is approximately equal to N (persistent attack). ■

V. Proposed Method for Attack Detection

It is proved that EWMA is equivalent to the mean of Bayesian inference for normal traffic as well as for persistent attack traffic. A method to detect the presence of attack in the incoming traffic is addressed in the present article. For normal traffic, mean of Beta distribution is computed for each sample to estimate the normal statistics of mean SAR. The problem with mean of Beta distribution is that the denominator term increases rapidly compared to the numerator term. During an attack, the increase in numerator term is made insignificant by the increase in denominator term and thus failed to detect the change in traffic. So for attack detection, mean of Beta distribution is slightly modified to detect the changes in the incoming traffic. As the number of SYN packets is more than the normal statistics for "*k*" consecutive samples (indication of anomalous traffic), the computed mean till the last sample of normal traffic is stored. Subsequently, the mean for anomalous traffic was recomputed by introducing a $\boldsymbol{\gamma}$ factor in (8) to obtain a modified formula as in the following equation:

$$\mu = \frac{\alpha}{\alpha+\boldsymbol{\gamma\beta}} \qquad (21)$$

where $\boldsymbol{\gamma} \ll 1$ and, thus, $\mu \approx 1$ (during attack). After attack, the mean has been computed using (8) with the stored value as the previous value for computation. Since mean of Beta distribution is not updated during the attack traffic, the normal statistics of the traffic is unaffected. The algorithm for the detection of flooding attack is given in Algorithm 1. Initially, 40–50 samples of size n is captured from the network traffic with probability $p = 1$ in order to estimate the normal statistics of the given traffic. Once the normal statistics of the traffic is obtained, capturing of samples of size n is done with probability $p < 1$ in order to ensure sufficient sleeping periods for mobile nodes. It is preferred to have the probability value $0.5 < p < 1$ in order to ensure sufficient sleeping time without affecting the accuracy of attack detection. In order to fix the value of p, each node uses a random number generator to generate a random number x which is uniformly distributed in the interval (0,1) with a benchmark of 0.5 for x. If the generated number x is less than 0.5, then $p = 0$ and the node remains in the sleep mode. If the generated number x is greater than 0.5, then $p = x$ and hence obtains the probability value in the range of $0.5 < p < 1$. Capturing of samples with probability $p < 1$ is continued till a violation of normal statistics is observed. Once a violation is detected, capturing of samples are done with probability $p = 1$ to avoid any misdetection of attack samples.

The proposed algorithm given in Algorithm 1 overcomes the limitations of ATA with higher detection rate and accuracy. Moreover, the battery life of mobile nodes is increased due to the presence of sufficient sleep times for the mobile nodes.

VI. Simulation Results

The proposed algorithm for detecting SYN flooding-based DoS attack has been tested using the 1999 DARPA dataset which is one of the standard datasets available for SYN flooding attack [27]–[29]. The DARPA dataset was developed by the

Algorithm 1: Algorithm for Attack Detection by Modifying the Mean of Beta Distribution.

1: Begin
2: Initialize the probability of capturing a sample of the traffic as $p < 1$, number of consecutive violations of normal statistics as $k = 0$
3: Samples of size n is captured with probability p during each unit of time
4: Compute the number of SYN packets for the captured sample
5: Compute the mean of Beta distribution, $\mu = \frac{\alpha}{\alpha+\boldsymbol{\beta}}$ for the captured sample
6: **if** the computed mean value is greater than the normal statistics **then**
7: Set $p = 1$, $k = k + 1$ and Go to Step No.12
8: **end if**
9: **if** mean value is less than the normal statistics **then**
10: Go to Step No.2
11: **end if**
12: **if** the computed mean value is greater than the normal statistics for $k \geq 5$ number of consecutive samples (anomaly) **then**
13: Save the computed mean before anomaly and then recompute the mean during anomaly using modified formula, $\mu = \frac{\alpha}{\alpha+\boldsymbol{\gamma\beta}}$ where $\boldsymbol{\gamma} \ll 1$ and thus, $\mu \approx 1$ during attack.
14: After attack, the saved mean is restored and updated using Step No.5
15: **end if**
16: **if** $k < 5$ **then**
17: Go to Step No.3
18: **end if**
19: End

Lincoln Laboratory at MIT and it consists of three weeks of training data in which the first and third week contains no attack, while the second week contains different types of labeled attacks. It is available in the form of "tcpdump" file. C program is written to estimate the statistics of SYN, FIN, and RST packets by capturing the samples consisting of 250 packets from normal and attack traffics, respectively. The algorithm for collecting the statistics of these packets is presented in Fig. 2.

Based on the protocol field of IP header, the incoming packets were classified as TCP, UDP, and ICMP. The protocol field of IP header contains the values 06, 11, and 01 for TCP, UDP, and ICMP packets, respectively. The incoming TCP packets were classified as SYN packets, FIN packets, RST packets, etc., based on the flag bits present in the TCP header. SAR is defined as the ratio of the number of SYN packets to the total number of captured packet (here, 250 packets). Similarly, FIN arrival rate is defined as the number of FIN packets to the total number of captured packet. In this way, we can define reset arrival rate. The statistics of SYN, FIN, and RST packets for Friday of week one normal traffic is shown in Fig. 3. For normal traffic, it is observed that the number of SYN packets (connection starting packets)

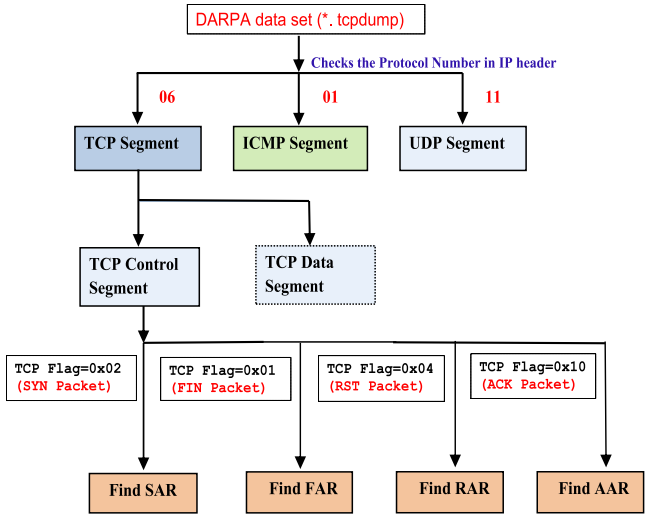

Fig. 2. Steps involved in 1999 DARPA dataset evaluation.

is almost the same as the number of FIN packets (connection closing packets), and the number of RST packets is negligibly small.

The statistics of SYN, FIN, and RST packets for Friday of week two attack traffic is as shown in Fig. 4 where it is observed that SAR increases abnormally during SYN flooding-based DoS attack.

A. Performance Parameters

The parameters used for the evaluation of performance for different algorithms is defined as follows [30].

1) *True Positive (TP)*: Actual positive (attack) predicted as positive (attack).
2) *True Negative (TN)*: Actual negative (normal) predicted as negative (normal).
3) *False Positive (FP)*: Actual negative (normal) predicted as positive (attack).
4) *False Negative (FN)*: Actual positive (attack) predicted as negative (normal).
5) *Accuracy (A)*: The ratio of true values to total observations, calculated as follows:

$$A = \frac{\text{TP} + \text{TN}}{N} \tag{22}$$

where *N* is the number of samples.

6) *True Positive Rate*: The ratio of true positives to the number of observations predicted as positive, termed as sensitivity, recall, or hit rate

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{23}$$

7) *False Positive Rate*: The ratio of false positives to the sum of false positives and true negatives, known as fall-out

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}. \tag{24}$$

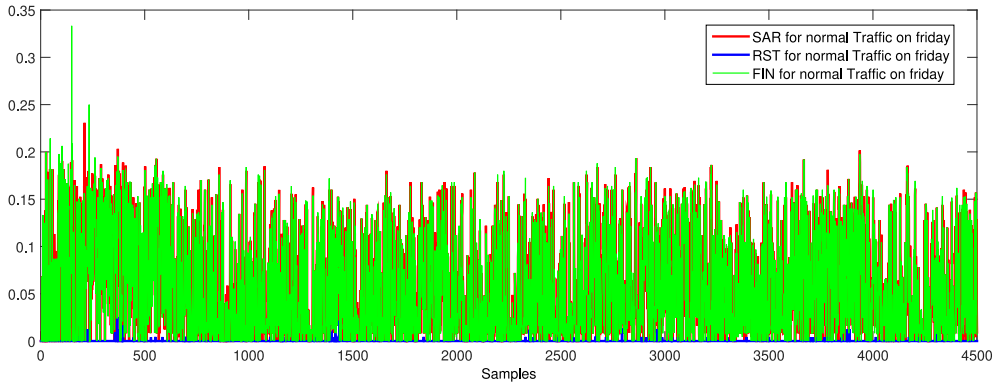

Fig. 3. Statistics of SYN, FIN, and RST packets for Friday of week one normal traffic.

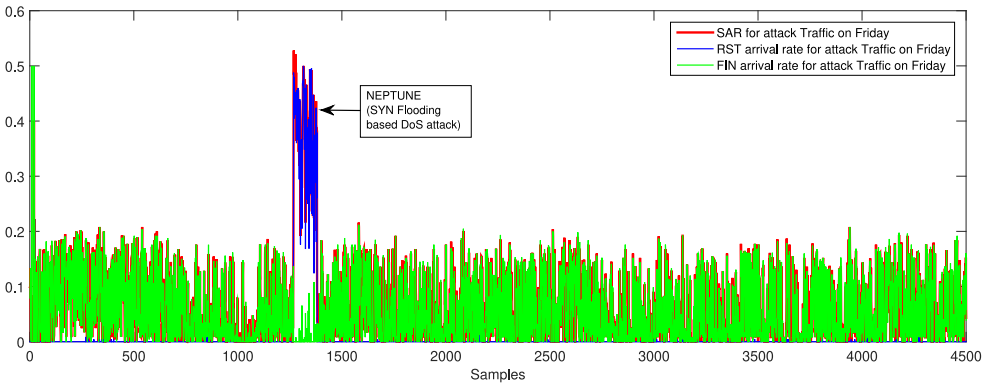

Fig. 4. Statistics of SYN, FIN, and RST packets for Friday of week two attack traffic.

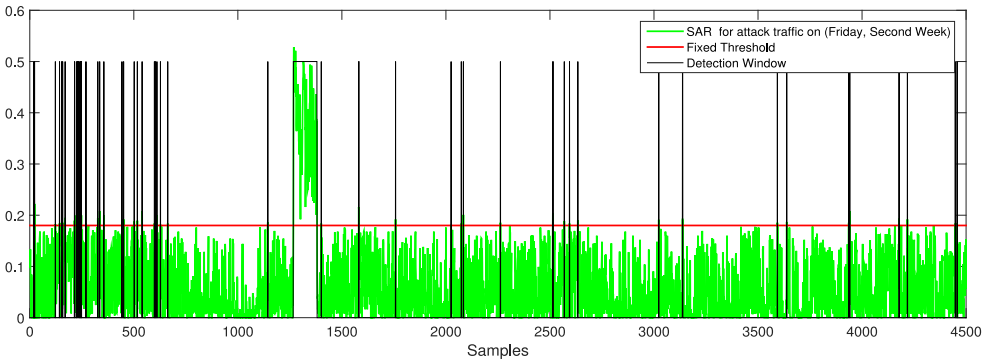

Fig. 5. Detection window using FTA applied to Friday of week two attack traffic.

8) *False Detection Rate*: (1-accuracy) is termed as FDR and is calculated as follows:

$$\text{FDR} = \frac{\text{FP} + \text{FN}}{N}. \tag{25}$$

9) *Precision (P)*: It is the ratio of correctly predicted positive observations to the total predicted positive observations

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{26}$$

The confusion matrix C is used to characterize the performance of the detection algorithm and is given by

$$C = \begin{bmatrix} \text{TP} & \text{FN} \\ \text{FP} & \text{TN} \end{bmatrix}.$$

A comparison of the proposed method with the state-of-the-art techniques like FTA and ATA for attack detection is carried out. The detection window for FTA is plotted in Fig. 5 for the Friday of week two attack traffic for a fixed threshold of 0.19.

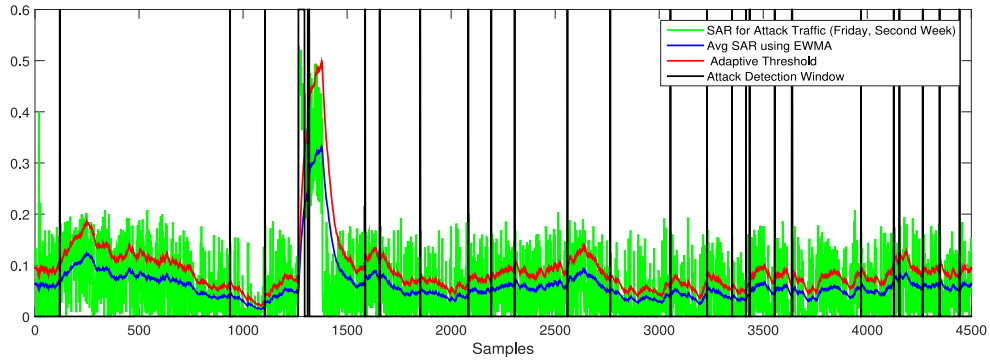

Fig. 6. Detection window using ATA applied to Friday of week two attack traffic.

The confusion matrix evaluated for FTA is given by

$$C = \begin{bmatrix} 102 & 11 \\ 55 & 4332 \end{bmatrix}.$$

Fig. 5 reveals that FTA has higher false positive rate and higher detection rate. The limitation of FTA is the determination of an optimum threshold for attack detection. Finding the threshold for any network needs a continuous monitoring of the network for a long time. Selecting a suboptimal value of threshold would result in higher misdetection or false alarm rate. Here, the fixed threshold for FTA is identified as 0.19 and is valid only for the given dataset. When the dataset changes, a new threshold is to be determined. The detection scheme based on ATA is plotted in Fig. 6 for the Friday of week two attack traffic for $\beta = 0.02$, $\alpha = 0.5$, and $k = 5$.

The confusion matrix evaluated for ATA is given by

$$C = \begin{bmatrix} 36 & 77 \\ 38 & 4349 \end{bmatrix}.$$

Fig. 6 reveals that misdetection rate is more for ATA since statistics of SAR are computed for normal as well as attack traffic. The limitation of ATA is the determination of an optimum value for α, β, and k for attack detection.

B. Testing of the Proposed Lemma

EWMA and mean of Beta distribution is calculated for Friday of week one normal traffic and week two attack traffic of the 1999 DARPA dataset. A typical plot for EWMA and mean of Beta distribution of the Friday traffic of week one is shown in Fig. 7.

The proposed lemma has been tested for both week one (normal) and week two (attack) traffic. The consistency of the proposed lemma has also been verified with data of two normal days of week one. The results are shown in Table I. The simulation results are consistent with the proposed lemmas. In Table I, for testing the lemma for Friday of week two attack traffic, we have considered only the SYN flood attack portion in the total traffic. Thus, the mean of Beta distribution is equivalent to EWMA for a large number of attack samples.

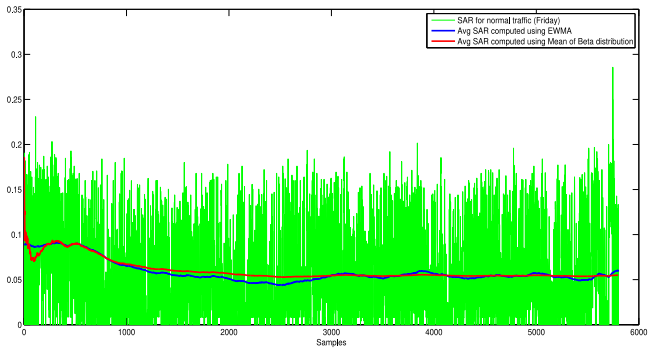

Fig. 7. Comparison of EWMA and mean of Beta distribution for Friday of week one normal traffic.

TABLE I
PROPOSED LEMMAS VERIFIED WITH DARPA DATASET

Dataset Used	ω (EWMA)	Δ (EWMA)	ω_n (EWMA)	ω (Beta)	Δ (Beta)	ω_n (Beta)
1999 DARPA dataset, Week 1, Normal (Thursday)	0.065078	0.048907	0.11399	0.064901	0.49006	0.11391
1999 DARPA dataset, Week 1, Normal (Friday)	0.055205	0.045105	0.10031	0.054951	0.045123	0.10007
1999 DARPA dataset, Week 2, attack (Friday)	0.3747	0.0657	0.4404	0.3746	0.0657	0.4403

C. Detection of SYN Flooding Attack Using the Proposed Algorithm

For detecting the anomaly involved in network traffic, mean of Beta distribution has been modified as per Algorithm 1. Plot for SAR, mean of Beta distribution, and the mean modified for attack detection for Friday of week two attack traffic are shown in Fig. 8 for $k = 5$ and $\boldsymbol{\gamma} = 0.01$. As the mean of Beta distribution is insensitive to variations in network traffic, it is evident that the modified mean successfully detects the anomaly involved in network traffic with higher accuracy. The confusion matrix evaluated for the proposed scheme based on the modified mean of Beta distribution is given by

$$C = \begin{bmatrix} 113 & 0 \\ 0 & 4387 \end{bmatrix}.$$

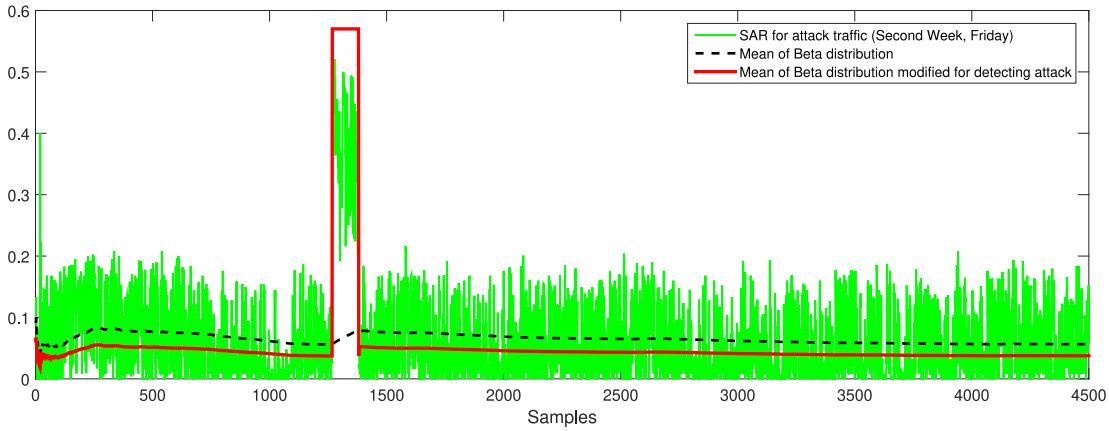

Fig. 8. Proposed method applied to Friday of week two attack traffic for attack detection.

TABLE II
COMPARISON OF PROPOSED METHOD WITH EXISTING METHODS

Detection Scheme	TP	TN	FP	FN	A(%)	TPR(%)	Precision(%)	FPR(%)	FDR(%)
Fixed Threshold Algorithm	102	4332	55	11	98.53	90.26	64.97	1.25	1.47
Adaptive Threshold Algorithm	36	4349	38	77	97.44	31.85	48.65	0.86	2.56
Modified Mean of Beta Distribution	113	4387	0	0	100	100	100	0	0

Thus, the proposed algorithm has successfully detected the anomaly involved in network traffic with higher TPR and precision. A comparison of various parameters of the proposed algorithm with the existing methods is shown in Table II.

VII. CONCLUSION AND FUTURE WORK

Flooding-based DoS attack is a major problem in all variants of wireless *ad hoc* network namely wireless sensor networks, MANET, VANET, and flying area networks. In the present work, the SYN traffic is modeled using Bayesian inference and an efficient algorithm is developed for detecting persistent flooding-based DoS attack applicable for different types of wireless *ad hoc* networks by providing necessary modification in the mean of Beta distribution. The proposed algorithm ensures sufficient sleeping time for the mobile nodes which further ensures the efficient utilization of battery resources. It can also detect Hello flooding attack, RREQ flooding attack, SYN flooding attack, data flooding attack, and UDP flooding attack without a dedicated server node but with higher TPR and precision. The proposed algorithm is useful for offline evaluation of any dataset or traffic extracted from any network by fixing optimum values for various parameters of the algorithm. With necessary changes in the proposed algorithm, it is recommended for the detection of burst attack traffic and gradual increase of attack traffic. Here, additional source of evidences like packet drop statistics, statistics of battery consumption, collision in MAC layer, etc., are to be considered in addition to the number of consecutive violations of normal statistics of SAR. The future work is intended based on additional source of evidence combined with data fusion techniques like Dempster–Shafer evidence theory.

REFERENCES

[1] S. U. Jan, S. Ahmed, V. Shakhov, and I. Koo, "Toward a lightweight intrusion detection system for the internet of things," *IEEE Access*, vol. 7, pp. 42450–42471, Mar. 2019, doi: 10.1109/ACCESS.2019.2907965.
[2] R. J. Cai, X. J. Li, and P. H. J. Chong, "An evolutionary self-cooperative trust scheme against routing disruptions in MANETs," *IEEE Trans. Mobile Comput.*, vol. 18, no. 1, pp. 42–55, Jan. 2019.
[3] M. L. Rajaram, E. Kougianos, S. P. Mohanty, and U. Choppali, "Wireless sensor network simulation frameworks: A tutorial review," *IEEE Consumer Electron. Mag.*, vol. 6, no. 2, pp. 63–69, Apr. 2016.
[4] W. Khalid, N. Ahmed, M. Khalid, A. U. Din, A. Khan, and M. Arshad, "FRID: Flood attack mitigation using resources efficient intrusion detection techniques in delay tolerant networks," *IEEE Access*, vol. 7, pp. 83740–83760, Jun. 2019, doi: 10.1109/ACCESS.2019.2924587.
[5] Mica2 Datasheet, CrossBow Corporation, San Jose, CA, USA. Accessed May 2006. [Online]. Available: http://www.xbow.com/
[6] N. Nishanth and P. Venkataraman, "Mobile agent based TCP attacker identification in MANET using the Traffic History (MAITH)," in *Proc. 13th IEEE Int. Conf. Commun. Technol.*, 2011, pp. 1130–1134.
[7] R. Mohammadi, R. Javidan, and M. Conti, "SLICOTS: An SDN-based lightweight countermeasure for TCP SYN flooding attacks," *IEEE Trans. Netw. Service Manage.*, vol. 14, no. 2, pp. 487–497, Jun. 2017.
[8] W. Wei, H. Song, H. Wang, and X. Fan, "Research and simulation of queue management algorithms in ad hoc networks under DDoS attack," *IEEE Access*, vol. 5, pp. 27810–27817, 2017, doi:10.1109/ACCESS.2017.2681684.
[9] V. A. Siris and F. Papagalou, "Application of anomaly detection algorithms for detecting SYN flooding attacks," *Comput. Commun.*, vol. 29, no. 9, pp. 1433–1442, 2006.
[10] K. Geetha and N. Sreenath, "Detection of SYN flooding attack in mobile ad hoc networks with AODV protocol," *Arabian J. Sci. Eng.*, vol. 41, no. 3, pp. 1161–1172, 2016.
[11] P. Mohammadi and A. Ghaffari, "Defending against flooding attacks in mobile ad hoc networks based on statistical analysis," *Wireless Personnel Commun.*, vol. 106, pp. 365–376, May 2019.
[12] S. Wang, Q. Sun, H. Zou, and F. Yang, "Detecting SYN flooding attacks based on traffic prediction," *Secur. Commun. Netw.*, pp. 1131–1140, 2012.
[13] S. Bhalodiya and K. Vaghela, "Enhanced detection and recovery from flooding attack in MANET using AODV routing protocol," *Int. J. Comput. Appl.*, vol. 125, no. 4, pp. 10–15, 2015.

[14] M. Conti, A. Gangwal, and M. S. Gaur, "A comprehensive and effective mechanism for DDoS detection in SDN," in *Proc. IEEE 13th Int. Conf. Wireless Mobile Comput., Netw. Commun.*, 2017, pp. 1–8.
[15] S. T. Zargar, J. Joshi, and D. Tipper, "A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks," *IEEE Commun. Surveys Tut.*, vol. 15, no. 4, pp. 2046–2069, Oct.–Dec. 2013.
[16] H. Rahmani, N. Sahli, and F. Kamoun, "DDoS flooding attack detection scheme based on F-divergence," *Comput. Commun.*, vol. 35, no. 11, pp. 1380–1391, 2012.
[17] Y. Chen, X. Ma, and X. Wu, "DDoS detection algorithm based on pre-processing network traffic predicted method and chaos theory," *IEEE Commun. Lett.*, vol. 17, no. 5, pp. 1052–1054, May 2013.
[18] X. Ma and Y. Chen, "DDoS detection method based on chaos analysis of network traffic entropy," *IEEE Commun. Lett.*, vol. 18, no. 1, pp. 114–117, Jan. 2014.
[19] H. Kim, R. Chitti, and J. Song, "Novel defense mechanism against data flooding attacks in wireless ad hoc networks," *IEEE Trans. Consum. Electron.*, vol. 56, no. 2, pp. 579–582, May 2010.
[20] N. Lyamin, D. Kleyko, Q. Delooz, and A. Vinel, "Real-time jamming DoS detection in safety-critical v2v c-its using data mining," *IEEE Commun. Lett.*, vol. 23, no. 3, pp. 442–445, Mar. 2019.
[21] N. Lyamin, A. Vinel, M. Jonsson, and J. Loo, "Real-time detection of denial-of-service attacks in IEEE 802.11p vehicular networks," *IEEE Commun. Lett.*, vol. 18, no. 1, pp. 110–113, Jan. 2014.
[22] W. Wei, H. Song, H. Wang, and X. Fan, "Research and simulation of queue management algorithms in ad hoc networks under DDoS attack," *IEEE Access*, vol. 5, pp. 27810–27817, 2017.
[23] T. N. D. Pham, C. K. Yeo, N. Yanai, and T. Fujiwara, "Detecting flooding attack and accommodating burst traffic in delay-tolerant networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 1, pp. 795–808, Jan. 2018.
[24] M. J. Faghihniya, S. M. Hosseini, and M. Tahmasebi, "Security upgrade against RREQ flooding attack by using balance index on vehicular ad hoc network," *Wireless Netw.*, vol. 23, pp. 1863–1874, Aug. 2016.
[25] A. Jøsang and R. Ismail, "The Beta reputation system," in *Proc. Bled Electron. Commerce Conf.*, Jun. 17–19, 2002, pp. 1–14.
[26] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. New York: McGraw-Hill, 1991.
[27] MIT Lincoln Laboratory, "Intrusion detection attacks database." Available: https://archive.ll.mit.edu/ideval/docs/attackDB.html
[28] Y. Gu, K. Li, Z. Guo, and Y. Wang, "Semi-supervised K-means DDoS detection method using hybrid feature selection algorithm," *IEEE Access*, vol. 7, pp. 64351–64365, May 2019, doi: 10.1109/ACCESS.2019.2917532.
[29] J. Cheng, M. Li, X. Tang, V. S. Sheng, Y. Liu, and W. Guo, "Flow correlation degree optimization driven random forest for detecting DDoS attacks in cloud computing," *Secur. Commun. Netw.*, vol. 2018, Nov. 2018, Art. no. 6459326.
[30] M. Sokolova, N. Japkowicz, and S. Szpakowicz, "Beyond accuracy F-score and ROC: A family of discriminant measures for performance evaluation," in *Proc. Australian Conf. Artif. Intell.*, 2006, vol. 4304, pp. 1015–1021.

N. Nishanth is working toward the Ph.D. degree with International School of Photonics, Cochin University of Science and Technology, Cochin, Kerala, India, under the guidance of Prof. A Mujeeb.

He is currently working as an Associate Professor in the Department of Electronics and Communication Engineering, TKM College of Engineering, Kollam, Kerala, India. His area of interest is wireless *ad hoc* networks and its security.

A. Mujeeb received the M.Sc. degree in physics with applied electronics specialization, M.Phil. degree in applied sciences, and Ph.D degree in optoelectronics.

He is a former Director and is currently working as a Professor with the International School of Photonics, Cochin University of Science and Technology, Cochin, Kerala, India. He has served as Member Syndicate and Senate of different State Universities. He has also served as the Director with LBS Centre for Science and Technology, under the Government of Kerala. His areas of specialization are wireless communication, optoelectronics, and speckle metrology.