Detecting LDoS Attacks based on Abnormal Network Traffic

Kai Chen, HuiYu Liu and XiaoSu Chen

School of Computer Science & Technology, Huazhong University of Science and Technology

Wuhan, Hubei 430074, P.R. China

[e-mail: kchen@hust.edu.cn] [e-mail: liuhuiyu@hust.edu.cn] [e-mail: x_s_chen@hust.edu.cn]

*Corresponding author: Huiyu Liu

Received January 12, 2012; revised February 21, 2012; revised May 11, 2012; revised June 11, 2012; accepted July 17, 2012; published July 25, 2012

Abstract

By sending periodically short bursts of traffic to reduce legit transmission control protocol (TCP) traffic, the low-rate denial of service (LDoS) attacks are hard to be detected and may endanger covertly a network for a long period. Traditionally, LDoS detecting methods mainly concentrate on the attack stream with feature matching, and only a limited number of attack patterns can be detected off-line with high cost. Recent researches divert focus from the attack stream to the traffic anomalies induced by LDoS attacks, which can detect more kinds of attacks with higher efficiency. However, the limited number of abnormal characteristics and the inadequacy of judgment rules may cause wrong decision in some particular situations. In this paper, we address the problem of detecting LDoS attacks and present a scheme based on the fluctuant features of legit TCP and acknowledgment (ACK) traffic. In the scheme, we define judgment criteria which used to identify LDoS attacks in real time at an optimal detection cost. We evaluate the performance of our strategy in real-world network topologies. Simulations results clearly demonstrate the superiority of the method proposed in detecting LDoS attacks.

Keywords: low-rate, denial of service, network security, abnormal traffic

http://dx.doi.org/10.3837/tiis.2012.07.007

1. Introduction

Denial-of-service (DoS) attacks present a serious threat to today's Internet services [1]. Usually, DoS attacks are executed against a distributed model, called distributed denial of service (DDoS), to achieve their purposes. Traditional DDoS attacks, such as FDoS (flood denial of service) or FDDoS (distributed FDoS), typically send a huge amount of traffic to a victim to overwhelm its resources. They disrupt the availability of a system or network by pressure of huge traffic. The challenge in detecting them is the need to deal with the huge traffic in networks [2], which makes the process of detection and prevention very difficult. The harm caused by a denial of service attack, both technologically and economically, cannot be eliminated for a long period after the attacks have ended. A successful DoS attack causes, not only pure economic loss, but also long-term damage to businesses, including customers loss, lower reliability of businesses services, and loss of reputation to the businesses.

Moreover, a new kind of DoS attack was proposed in 2003 by Kuzmanovic and Knightly [3]. According to them, just by sending a well-timed pulse periodically may cause transmission control protocol (TCP) traffic to decline seriously. Since 2004, more instances of this kind of attack have been identified, the most well-known of which are the RoQ [4] and the PDoS attacks [5]. They impact the target TCP data traffic using *defects* of the congestion control mechanism in computer network, such as timeout retransmission mechanism, queue management mechanism, and congestion window control mechanism. Inasmuch as this kind of attack is able to provide attacks with a similar power as an FDDoS attack but with a much lower average attack traffic, i.e., lower rate, it is hence named *low-rate denial of service* (LDoS).

An LDoS attack can exist in various patterns, for example, waging with a fixed or alterable attack cycle directed at different congestion strategies, such as *Additive Increase Multiplicative Decrease* (AIMD)[3][5], *Fast Retransmit* (FR)[4], or *Active Queue Management* (AQM)[6][7], and using multiform attack data types (UDP, ICMP, TCP, DNS, and SYN[8]). Recently, more kinds of LDoS attack, such as LoRDAS attack [9][10][11][12][13] which is launched against application servers and is able to reduce their availability, and another kind of LDoS attack against network firewalls by triggering the last-matching rules of the security policy of a firewall which require the most processing time by the firewall [14]. LDoS attack could also be launched in Ad-hoc network [15], and as a completely distinct MAC layer protocol is adopted in this environment, the form and effect of the attack could be different. Hence, the LDoS attack is more difficult to detect than an FDDoS attack.

To detect the LDoS attack, a series of detecting methods have been developed, which can be classified into the following three categories [16]: frequency-domain based methods, time-domain based methods, and traffic character-based methods.

Frequency-domain based detection methods extract periodic features from attack traffic by spectral analysis, such as CDF [17], Wavelet Analysis [5][18], and UDP frequency domain-based detection method [19]. The CDF, as a distributed collaborative filtering detection method based on power spectral density, has a higher detection rate, but may spend a large number of storage resources for using three main tables to record incoming packets. Wavelet Analysis aims principally at an AIMD-targeted attack, but is ineffective against a non-AIMD-targeted LDoS attack. UDP frequency domain-based detection methods need *Time/Frequency* transforms, whose reaction is less efficient.

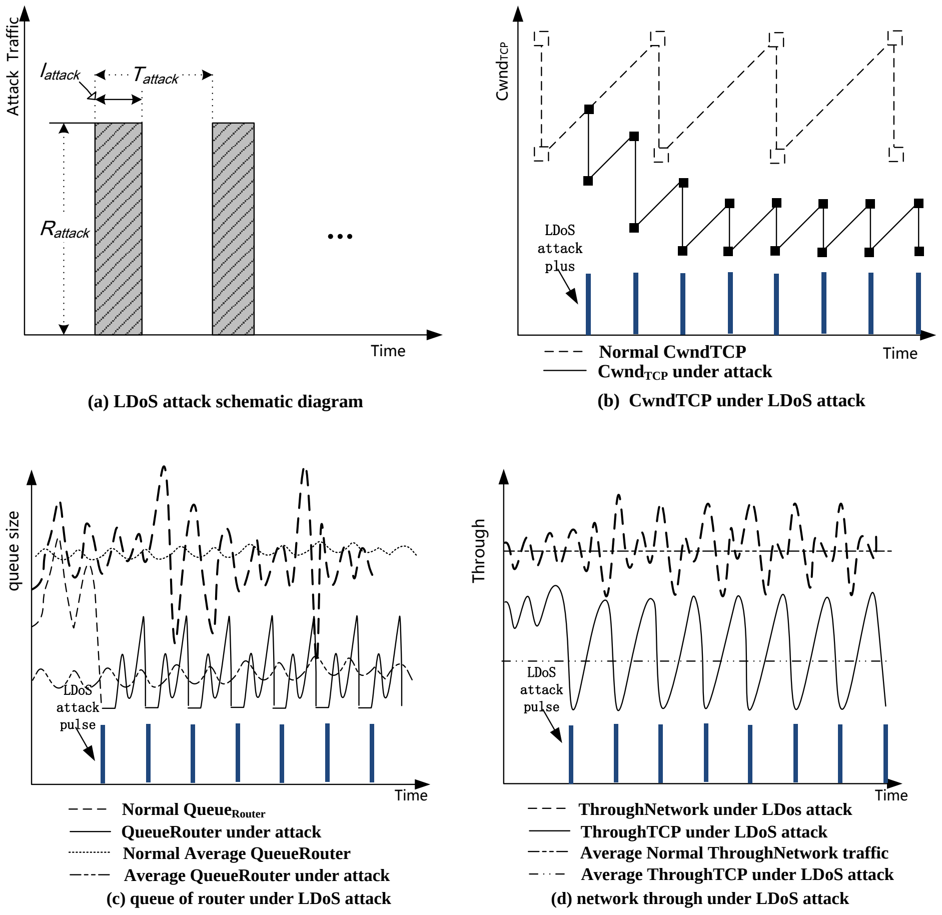
A time domain-based detection method [20][21][22] focuses on the period of attack to recognize the abnormal characteristics of a flow on a given time domain, and then identifies them if they exist. DTW [20] and HAWK [21] are typical methods, in which the following deficiencies exist: (1) limitations in an LDoS attack mode, (2) low detection efficiency and more resource intensive, and (3) the weak real-time.

Traffic character-based detection methods [23][24][25][26][27][28][29][30]detect LDoS attacks by searching and identifying the characters in a network traffic with LDoS attacks. The method based information metrics [30], one of them, not only can detect LDoS attacks, but also can trace back attackers. However, the limitation on particular kinds of LDoS attacks makes them less effective than other methods, although they are low-consumption methods. The Vanguard, another one of them, can obtain preferable detection results on most kinds of LDoS attacks, but may misjudge when network traffic busts or becomes silent. Also, it cannot differentiate between a special LDoS attack and a general DoS attack.

This paper proposes a detecting strategy that recognizes various LDoS attacks with higher efficiency and lower detection cost. We introduce some judgment criteria which are designed to distinguish LDoS attacks from the normal traffic and from other attacks. The rest of the paper is organized as following: In Section 2 we describe the model of LDoS attacks. In Section 3 we propose a strategy that can detect LDoS attacks with high efficiency and low detection cost. We verify the performance of the proposed strategy by comparing the results obtained from performance evaluation in Section 4 and conclude the paper in Section 5.

2. Model and Definitions

In this section, we present a unified model to describe LDoS attacks. By sending plenty of attack data periodically, LDoS attacks congest the target repeatedly, exploiting the scaling behavior of the legit TCP senders' congestion control window, which causes TCP traffic going into a process of burst decline and climbing back frequently. Afterwards, the target network or host *denies servicing* because the average legit traffic decline significantly. When LDoS attacks continue, three arguments,T_{attack},l_{attack}, and R_{attack}, are preconfigured, where T_{attack} refers to the periods of attack, l_{attack} is the duration of the attack, and R_{attack} is the intensity of attack pulse (the average velocity of the attack data). The three arguments relate to the result of an LDoS attack. Let $Cwnd_{TCP}$ denote the size of congestion control window of legit TCP senders, $Query_{Router}$ and $\overline{Query_{Router}}$ express the queue length and average queue length of the key router in the victim network, respectively. Use $Through_{TCP}$ and $\overline{Through_{TCP}}$ as the TCP traffic and average TCP traffic of the victim, respectively. Let $Through_{Network}$ be the network traffic. An LDoS attack sketch is shown in **Fig. 1 (a)**. The influence of LDoS attacks on the TCP congestion control window and the average queue length of the key router are shown in **Fig. 1 (b)** and **Fig. 1 (c)** respectively. **Fig. 1 (d)** shows LDoS attack effect on TCP traffic, average TCP traffic and network traffic.



(a) LDoS attack schematic diagram

(b) CwndTCP under LDoS attack

(c) queue of router under LDoS attack

(d) network through under LDoS attack

Fig. 1. Schematic diagram of LDoS attack principle and results

Fig. 1 shows that LDoS attacks reduce the average TCP traffic substantially by influencing the size of the congestion control window of legit TCP senders and the length of the router queue. Let $L(T_{attack}, l_{attack}, R_{attack})$ be the LDoS attack function, then the results of LDoS attacks are shown as Eq. (1) and Eq. (2), where ↓ expresses decline and ↓↑ expresses undulation.

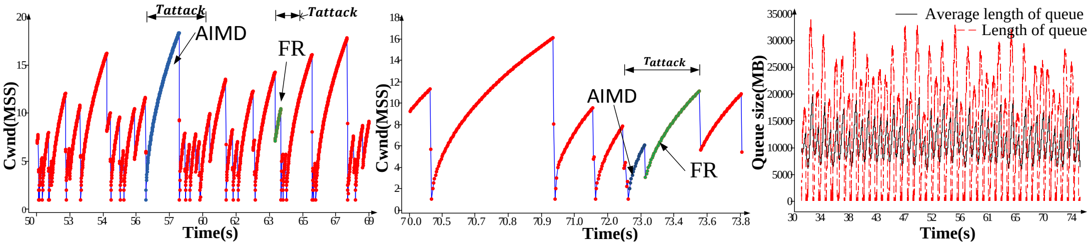
$$L(T_{attack}, l_{attack}, R_{attack}) \rightarrow Cwnd_{TCP} \downarrow\uparrow \rightarrow Through_{TCP} \downarrow\uparrow \rightarrow \overline{Through_{TCP}} \downarrow \quad (1)$$

$$L(T_{attack}, l_{attack}, R_{attack}) \rightarrow Query_{Router} \uparrow\downarrow \rightarrow Through_{Network} \downarrow\uparrow \rightarrow Through_{TCP} \downarrow\uparrow \rightarrow \overline{Through_{TCP}} \downarrow \quad (2)$$

The parameters of function $L(T_{attack}, l_{attack}, R_{attack})$ are analyzed as fellows.

- **Values for $\boldsymbol{T_{attack}}$:** Ideally, the value of a T_{attack} should be the *Retransmission Timeout (RTO)* of TCP flows accurately and set dynamically. So the congestion control window and the throughput may always keep the minimum values (size of the congestion control window is *1* and throughput is *0*), which means the best attacking result, namely, synchronization attack. However, the real-time value of *RTO* is difficult to obtain, so LDoS attackers usually set $T_{attack} = RTO_{min} + 2{\sim}3RTT_{avg}$ to get a relatively good result, where RTO_{min} is the minimum *RTO*, whose general value is *1s*. RTT_{avg} is the average value of *Round-Trip Time (RTT)* of all TCP connections.
- **Values for $\boldsymbol{l_{attack}}$:** Considering the concealment, LDoS attack pulse should be a short burst of traffic. However, for a favorable effect, it generally persists long enough to cause router queue overload to abandon legit TCP packets. An attacker always sets $l_{attack} \in \{RTT_1, RTT_2, \ldots RTT_k\}$, where *K* is the number of TCP flows in a network. In addition, $min\{RTT_i\} \leq l_{attack} \leq \zeta T_{attack}$ *i=0,…k* , ,$\zeta \leq 0.5$.
- **Values for $\boldsymbol{R_{attack}}$:** While an LDoS attack occurs, the higher of R_{attack} makes the more bandwidth loss, but the less concealment of attack simultaneously. Especially if $\overline{R_{attack}} \geq C$, the LDoS attacks are approximate the FDDoS, where $\overline{R_{attack}}$ is the average of R_{attack} and *C* is the bottleneck bandwidth. Typically, $\overline{R_{attack}}$ is related to the LDoS attack pattern and the background flow and, usually, $R_{attack} \ll \frac{C \times T_{attack}}{l_{attack}}$.

Generally, LDoS attacks are classified by the congestion scheme they target. However, to distinguish the scheme which an LDoS attack aims at is hard, because that several congestion schemes may work concurrently when an LDoS attack occurs. **Fig. 2** shows that the AIMD and the FR are both induced during an attack (**Fig. 2 (a)**), possibly in turn in an attack periods (**Fig. 2 (b)**), as well as the length of the router queue flaps violently (**Fig. 2 (c)**). However, whatever the congestion scheme initiates during LDoS attacks, the traffic of a victim becomes abnormal.



(a) two-congestion scheme (b) two-congestion scheme induced in turn (c) router queue

Fig. 2. Concurrence of congestion scheme during an LDoS attack

As described previously, LDoS attacks may lead to anomalies on a victim's traffic. Furthermore, if the victim is a host, the traffic of the network on which the host is located becomes abnormal at the same time. Therefore, this study focuses on the traffic on the key router of a network.

3. Proposed Detection Strategy

This section describes several abnormity of network traffic, and then introduces a new LDoS detection scheme based on them.

First, we propose the following three hypotheses:

H$_1$: No attacks exist in the network

H$_2$: Non-LDoS attacks, which are attacks but LDoS ones, exist in the network

H$_3$: Only LDoS attacks exist in the network

The ACK traffic distribution patterns, the volatility of the ACK traffic and *Attack Effectiveness* are similar when H_1=*true* and H_2=*true*, but significant different when H_3 = *True*.

Known by the principle of LDoS attacks, LDoS attacks cause the targets denial of service by affecting legitimate TCP traffic, which reflecting in the variations of ACK traffic. For increasing detection efficiency, we detect anomalous TCP traffic by observing the variations in ACK traffic instead. ACK traffic is given in $\{x_k, k=1,2...\}$, which is *independent identically distributed* (*i.i.d.*), and follows a certain probability distribution, where x_k is the statistical value of ACK traffic with serial number *k*.

3.1. Abnormity of ACK traffic distribution

Let $P(x|H_i)$ represent the probability distribution of $\{x_k\}$ at hypothesis H_i. Some studies have shown that network traffic obeys self-similar distribution in a large time scale [31][32][33], and can be simulated by a Poisson Model in sub-second [34]. In the meantime, many experiments show that ACK traffic is x^2 distributed asymptotically when the network is free flowing and that it is approximate to exponential distribution when it is busy. According to the *Central Limit Theorem*, on a certain condition, the distribution of the sum of huge random variables, which are statistically independent, tends to *Normal Distribution*. Based on the conclusion, other distributions can be approximated by *Normal Distribution*. Therefore, we get the following expressions:

$$H_1 = True \rightarrow P(x|H_1) \sim N(\mu, \sigma^2) \tag{3}$$

$$H_2 = True \rightarrow P(x|H_2) \sim N(\mu', \sigma'^2) \tag{4}$$

$$H_3 = True \rightarrow P(x|H_3) \sim N(\hat{\mu}, \hat{\sigma}^2) \tag{5}$$

where *N( )* denotes the *Normal Distribution* which $\{x_k\}$ obey; μ, μ' and $\hat{\mu}$ denote the average of $\{x_k\}$; and σ^2, σ' and $\hat{\sigma}^2$ express the variance of $\{x_k\}$.

Different pattern of ACK traffic distribution correspond different states of network. ACK traffic distribution patterns can be used to distinguish in which state the network is located. With a given ACK traffic mean, we descript ACK traffic distribution patterns under three hypotheses holding respectively in **Fig. 3**.

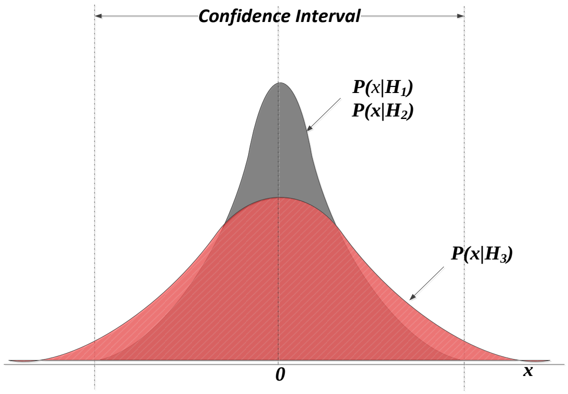


Fig. 3 Schematic diagram of $P(x|H_1)$、$P(x|H_2)$ and $P(x|H_3)$

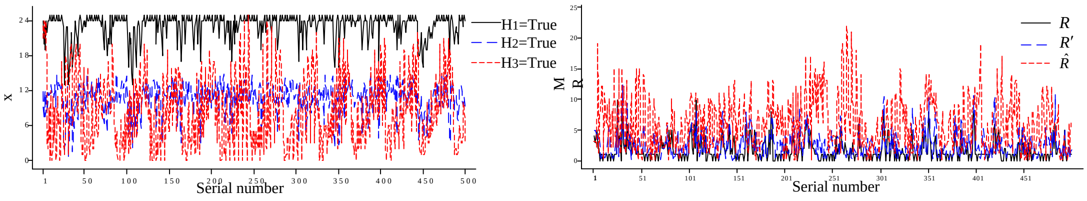
In **Fig. 3**, $P(x|H_1)$ and $P(x|H_2)$ are approximately the same, because that when H_2=*True* and H_1 = *True*, ACK traffic changes weakly and the degree of the changes are similar, that is, $\sigma^{'} \approx \sigma$. While H_3 = *True*, ACK traffic changes dramatically and the degree of changes are obvious differences from that when H_1 = *True*, namely, $\hat{\sigma} \gg \sigma$, that is the reason of deviation between $P(x|H_3)$ and $P(x|H_1)$. Based on the deviation, we can get such an interval *CI*, outside which $P(x|H_1)$ and $P(x|H_2)$ falls barely but $P(x|H_3)$ falls much more. The interval *CI* is called the right confidence interval with the significance that if the x crossing the interval meets certain conditions, then H_3 = *True*.

3.2. Abnormity Of Ack Traffic Fluctuation Range

In addition to the variation of *Probability Distribution*, the degree of discrete of ACK traffic shows quite a difference between those with and without LDoS attacks. Set *mr* as the *moving range* of ACK traffic between times t_i and t_j, where t_j=t_i+Δt, Δt is a small positive real number, and *i* and *j* are any integer which meets *j*>*i*. Then, *mr* is calculated based on the following condition:

$$mr = |x_j - x_i| \tag{6}$$

Fig. 4 shows the ACK traffic's *fluctuations* diagram and *Moving Range* schematic under the three hypotheses holding respectively in the same network background where R, $R^{'}$ and $\hat{R}$ denote the *mr* when H_1=*True*, H_2=*True*, and H_3=*True*, respectively.



(a) Statistical value of ACK traffic

(b) *mr* of ACK traffic

Fig. 4. Statistical values of ACK traffic and *mr* of ACK traffic

Based on **Fig. 4**, we can recognize that, generally, $R^{'}$ is similar to R and both of them are smaller, but $\hat{R}$ is much larger than R.

3.3. Difference of Attack Effectiveness

While H_1=*True*, the variance of legitimate TCP traffic changes slightly in a certain term of time scale. While H_2=*True*, most other attacks usually *press* legitimate TCP traffic to decline, hence, the attack flows usually have the feature of a *large traffic*. While H_3=*True*, legitimate TCP traffic changes greatly, but average traffic of attack data is relatively small. In view of the difference between legitimate TCP traffic and attack traffic on the three hypotheses, this paper proposes the *Attack Effectiveness* based on *entropy difference*.

Definition of Attack Effect (*Se*): The entropy value of the loss of legit TCP traffic is defined as *Attack Effect*.

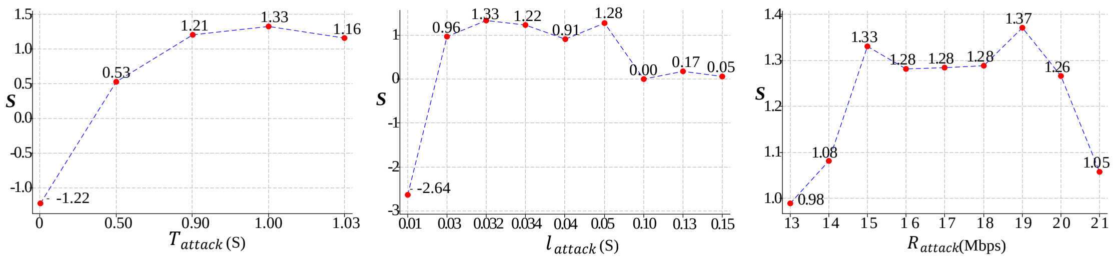
Definition of Attack Cost (*Sc*): The entropy value of the average attack traffic is defined as *Attack Cost*.

Definition of Attack Effectiveness (*S*): The difference between attack effect and attack cost is defined as *Attack Effectiveness*.

Respectively, we denote the average of normal and abnormal legit TCP traffic as V_1 and V_2. Let ***Z*** be the type of potential LDoS attack data. The average of ***Z*** traffic is denoted as U_1 and U_2 when H_3=*False* or *True*, respectively, and set ***Se*** = $\log_2(V_1\text{-}V_2)$ and ***Sc*** = $\log_2(U_2\text{-}U_1)$. Then, ***S*** is calculated based on the following condition:

$$\boldsymbol{S} = \boldsymbol{Se} - \boldsymbol{Sc} = log_2(\frac{V_1 - V_2}{U_2 - U_1}) \tag{7}$$

From Eq. (7), it can be seen that, at the same *Attack Effect*, the higher the *Attack Effectiveness* is, means the smaller the *Attack Cost*, and more concealment of attack. When H_1 = *True*, the legitimate TCP traffic and ***Z***-flow change slightly, as well as, ***Se*** and ***Sc*** are very low and the difference between them is small which causes the *Attack Effectiveness* to be low. When H_2 = *True*, ***Z*** -flow spikes and legitimate TCP traffic dump, although ***Se*** and ***Sc*** are high but the difference between them is also small, which cause the *Attack Effectiveness* to be low equally. When H_3 = *True*, the legitimate TCP traffic reduces highly, but ***Z*** -flow increases less, the difference between ***Se*** and ***Sc*** is much large. So the *Attack Effectiveness* is high. **Fig. 5** shows the relationship between the *Attack Effectiveness* and LDoS attack parameters under the same network environment. In the current network environment, RTO_{min}=*1*s, RTT_{avg}=*0.03*s, the link bottleneck bandwidth is *C=10*Mb. Because that different mode of LDoS attack can be considered as the performance of different combinations of parameters practically, to simplify the analysis, we do not consider the LDoS attack mode temporarily, and only examine the different values of the attack parameters shown in **Fig. 5**.



(a) The relationship between T_{attack} and ***S*** with l_{attack}=0.03s, R_{attack}=20Mbps

(b) The relationship between l_{attack} and ***S*** with T_{attack}=1s, R_{attack}=20Mbps

(c) The relationship between R_{attack} and ***S*** with T_{attack}=1s, l_{attack}=0.03s

Fig. 5. Diagrams of the relations of attack parameters and attack effectiveness

As can be seen from **Fig. 5**, when $T_{attack}=0$s, $l_{attack}=0.032$s and $R_{attack}=20$Mbps, it gets ***S*** = -1.22, and when $T_{attack}=1$s, $l_{attack}=0.01$s and $R_{attack}=20$Mbps, it gets ***S*** = -2.64. These two cases, can be considered as attack free and only background traffic in the network, that is, $H_1=True$. When $T_{attack}=l_{attack}=1$s, $R=20$Mbps, i.e., the FDoS attack occurring, that is, $H_2=True$, it gets ***S*** = -0.07. And for others parameters' values which match the requirements of LDoS attack, the corresponding ***S*** are mostly relatively high and all of them are greater than *0*. Especially when $T_{attack}=1$s, $l_{attack}=0.032$s and $R_{attack}=19$Mbps, which are the optimal values of LDoS attack parameters of this network as defined in section 2, it gets ***S***=1.37, which is the maximum in **Fig. 5**. So when $H_3=True$, the *Attack Effectiveness* is relatively high and is quite different from those when $H_1=True$ and $H_2=True$. Although in different network environments, the attack effectiveness may be different from the values in **Fig. 5** slightly, but they still obeys to the relationship generally. We denote ***S*** as , $\mathcal{S}_{H_1}$, $\mathcal{S}_{H_2}$ and $\mathcal{S}_{H_3}$ while each of the three hypothesis holds, then $\left(\mathcal{S}_{H_1} \approx \mathcal{S}_{H_2}\right) < \mathcal{S}_{H_3}$.

3.4. Method Of Detection

Based on the explanation above, this paper proposes a new LDoS attack detection method and it involves the following steps.

- **Sampling of traffic:** ACK traffic, TCP traffic, and other traffics on the key router are sampled using preset sampling parameters.
- **Processing samples and measurements:** Perform detection algorithms to process and to measure the samples obtained in the previous step.
- **Attack judging:** Based on the judgment criterion, this paper analyzes the measurement results to determine an LDoS attack.

The detection method is described hereunder in detail.

Ⅰ **Sampling traffic**

By considering the detection efficiency, we regard the number of incoming ACK packets on the key router per unit time as ACK traffic. In an *observation time window* (*WS)* with length T_s, we record the number of ACK bags, the traffic of legit TCP with payload, and the traffic of non-TCP as the observed values at an interval Δt on the key router, denoted as I_{ACK}, I_{TCP}, and I_{other}, respectively. Thus, we get three time series of observation, O_{ACK}, O_{TCP}, and O_{other}, which are given in Eq. (8), Eq. (9), and Eq. (10), respectively.

$$O_{ACK} = \{I_{ACK_i} | i = 1,2 \dots, \left\lceil \frac{T_S}{\Delta t} \right\rceil\} \tag{8}$$

$$O_{TCP} = \{I_{TCP_i} | i = 1,2 \dots, \left\lceil \frac{T_S}{\Delta t} \right\rceil\} \tag{9}$$

$$O_{Other} = \{I_{Other_i} | i = 1,2 \dots, \left\lceil \frac{T_S}{\Delta t} \right\rceil\} \tag{10}$$

Taking account of detection accuracy and real-time capability, T_s should be an appropriate value to gain enough observed values as well as small detection delay.

Ⅱ **Processing samples and measurement**

Based on the preceding analysis in part B, we construct three indicators for detecting LDoS, (1) the fluctuation degree of ACK traffic, (2) the departure degree of ACK traffic distribution, and (3) attack effectiveness. Below, we describe sampling processing and measurement based on these three indicators.

(1) Measure the fluctuation of ACK traffic

The fluctuation degree of ACK traffic can be evaluated by the degree of dispersion of continuous samples. We introduce an algorithm, referred to as *Moving Range*, to calculate the fluctuation degree. *Moving Range* algorithm is in the following:

① Establish *sliding window* (*SW)*, which is composed of samples in ACK traffic sample sequence. The size of *SW*, i.e., L_{SW}, should be configured with $L_{SW} \geq 2$.

② Obtain the difference between maximum and minimum values of samples' means in *SW*, and record the difference value as mr_l, where, $1 \leq l \leq M$ and *M* is the number of samples.

③ Add a sample in *SW* whenever it is obtained, and remove the *oldest* one at the same time;

④ Repeat step ② until the last sample and, finally, the *difference sequence set* (*MR*) of $\{mr_l\}$ is formed.

Let *n* be the sample size and *n* is a natural number, based on the O_{ACK} to generate ACK traffic samples st_j using *simple time series techniques*, where $j \in \{1,2,\dots,\lceil T_S/\Delta t \rceil\}$, and using the *arithmetic average* to calculate the sample mean of each sample. Following the steps above, we construct ACK traffic samples sequence ST_{ACK}, as shown in Eq. (11).

$$ST_{ACK} = \left\{ st_j | st_j = \{I_{ACK_i}, I_{ACK_{i+1}} \dots I_{ACK_{i+n-1}}\}, i = (j-1)*n+1, j \in \{1,2,\dots,\left\lceil \frac{T_S}{n\Delta t} \right\rceil\} \right\} \tag{11}$$

ACK traffic sample means sequence $\overline{ST_{ACK}}$ is given as in Eq. (12).

$$\overline{ST_{ACK}} = \left\{ \overline{st_j} \mid \overline{st_j} = \frac{1}{n} \sum_{i=k}^{k+n-1} I_{ACK_i}, k = (j-1)*n+1, j \in \{1,2,\dots,\left\lceil \frac{T_S}{n\Delta t} \right\rceil\} \right\} \tag{12}$$

We construct the *MR* as shown in Eq. (13).

$$MR = \{mr_l | mr_l = max(\overline{st_j}, \overline{st_{j+1}}, \dots \overline{st_{j+L_{DW}-1}}) - min(\overline{st_j}, \overline{st_{j+1}}, \dots \overline{st_{j+L_{DW}-1}}), j \in \{1,2,\dots,\left\lceil \frac{T_S}{n\Delta t} \right\rceil\} \tag{13}$$

In Eq. (13), mr_l expresses the variation degree (difference degree) of ACK traffic between consecutive time points. The lager *mr*, means that there are more varieties and more influence on ACK traffic.

(2) Measure the departure degree of ACK traffic distribution

We employ the *exponentially weighted moving average (EWMA) algorithm* to establish the statistics of distribution of ACK traffic. The main idea of *EWMA* is that the latest observations may carry the most information. Therefore, newer statistical variable has more weight. The statistic of *EWMA* is a weighted linear combination of samples in such a way that:

$$EWMA = \sum_{i=0}^{t} W_i x_i^{ACK} \tag{14}$$

where w_i is the *i*-th sample weight, x_i^{ACK} is the *EWMA* statistic of the *i*-th sample, *t* is the total number of samples, and $\sum_i^t W_i = 1$. Eq. (14) can be transformed as shown:

$$x_{i+1}^{ACK} = x_i^{ACK} + \lambda\varepsilon_i = \lambda\acute{x}_i^{ACK} + (1-\lambda)x_i^{ACK} \tag{15}$$

where ε_i is the *Error* between the actual value and the estimated value of the *i*-th sample, $\acute{x}_i^{ACK}$ is the actual value of the *i*-th sample, λ is the *weight operator*, which has a valid range of $0<\lambda\leq 1$, whereas the larger λ represents the more influence by the *recent* sample.

EWMA has two typical characteristics, namely, the time domain expressed in Eq. (14) and the smoothing effect shown in Eq. (15). By adjusting weight operator λ, the influence of historical data and the smoothing function of slight fluctuations (noise) can be controlled. To balance smoothness and sensitivity of the newest data, we usually set $\lambda\in[0.8,1]$.

Based on the discussion in Part A of this section, a *confidence interval* (*CI*) has to be set to measure the degree of deviation of ACK traffic distribution. *CI* is calculated as following:

$$CI = [\mu - k\sigma_T\sqrt{\lambda/(2-\lambda)}, \mu + k\sigma_T\sqrt{\lambda/(2-\lambda)}] \tag{16}$$

where μ is the sample mean of training data, and σ_T is the deviation of training data standard deviation. *k* is a constant called *confidence operator* which has a direct bearing on the size of *CI*. Considering the accuracy of detection, we set $k=3$. When $H_1=True$ or $H_2=True$, ACK traffic samples are out of the *CI* with small probability (According to the calculation of *significance level*, the theoretical value is 0.3% when $k=3$). If the probability of ACK traffic samples outside the *CI* is more than a threshold, the distribution of ACK traffic offset. The more samples there are outside, the larger deviation is.

(3) Measure the attack effectiveness

We select the data at an appropriate time (less link-state changed at the time) when $H_1=True$ as training data. From the training data, we obtain V_1 and U_1 in accordance with Eq. (7). Inasmuch as LDoS attacks can use many types of data, we regard any data as attack data except the legit TCP with payload. Also, we obtain V_2 and U_2 from the test data.

A large number of experiments based on different network environment and different type of LDoS attacks show that when $H_1=True$, the value of $\boldsymbol{S}$ is incomputable ($V_2\geq V_1$ or $U_1\leq U_2$) or smaller than the value as $H_3=True$. When $H_2=True$, a large amount of attack data is injected into the network, so attack traffic increases, whereas legit TCP traffic reduces. To achieve a certain attack effect, increases in attack traffic are much larger than reductions in legit TCP traffic (such as an attack by seizing bandwidth and queue of routers). So the value of $\boldsymbol{S}$ is also smaller than the value as H_3=*True*.

Ⅲ Judge LDoS Attacks

In the above two steps, we measured the three anomaly features on data traffic led by an LDoS attack. However, they may bring misjudgments based on the above-mentioned abnormal characteristics only. Some of the scenarios are shown below.

(1) x^{ACK} is still outside *CI* at a certain probability under a normal situation;

(2) Normal data flow may lead to network congestion;

(3) Changes in network structure or huge normal traffic may cause upheaval in TCP traffic;

(4) Other attacks cause drastic changes in TCP traffic.

To reduce misjudgment, based on the results of processing and measurement of samples, we give some judgment criteria in the following:

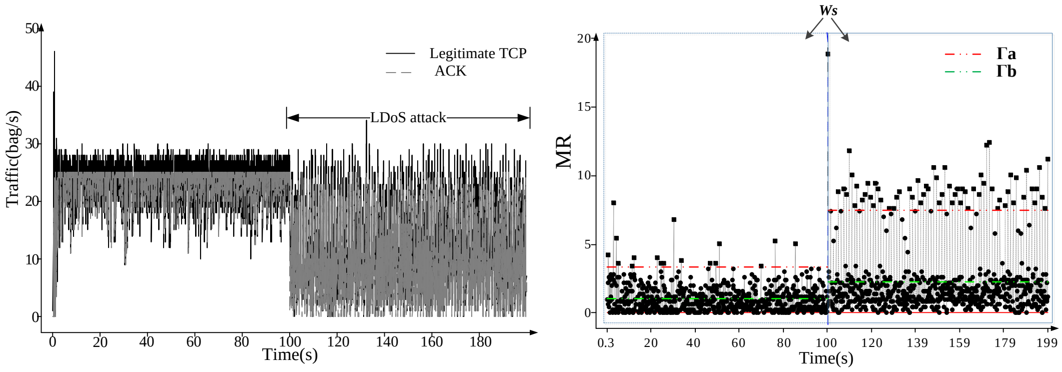
(1) Judgment criteria based on the fluctuation degree of ACK traffic

There are two typical characteristics of ACK flow fluctuation caused by an LDoS attack, namely, ① ACK flow rises slowly and descends immediately in every cycle, and ②frequency and amplitude are relatively high. Further, when an LDoS attack occurs, the *MR* reflects two corresponding typical characteristics: ① the proportion of **Þ** consecutive *mr*, which is less than $\mathit{\Gamma}_a$ in *MR*, is relatively low, and ② the proportion of *mr*, which is greater than $\mathit{\Gamma}_b$, is relatively high. $\mathit{\Gamma}_a$, $\mathit{\Gamma}_b$, and **Þ**, which are given advance, are called *control factors* of *MR*. After repeat trials, we found that **Þ** = 9 is an appropriate value. $\mathit{\Gamma}_a$ and $\mathit{\Gamma}_b$ are calculated as shown in Eq.17 and Eq.18, where σ_{MR} is the standard deviation of *MR*.

$$\mathit{\Gamma}_a = \frac{1}{\left\lceil \frac{T_s}{n\Delta t} \right\rceil} \sum_{l=1}^{\left\lceil \frac{T_s}{n\Delta t} \right\rceil} mr_l \tag{17}$$

$$\mathit{\Gamma}_b = \mathit{\Gamma}_a \ + 3\sigma_{MR} \tag{18}$$

ACK traffic and its *MR* are shown in **Fig. 6**, in which attack free at *WS* of 0s ~ 100s and an LDoS attack at *WS* of 100s~200s.



(a) TCP and ACK traffic

(b) *MR* of ACK traffic

Fig. 6. Diagram of TCP and ACK traffic and the *MR* of ACK traffic with an LDoS attack at 100s~200s

In a *MR*, each consecutive **Þ** *mr* less than $\mathit{\Gamma}_a$, form a chain. And every *mr* in the chains is denoted as $\overleftarrow{\boldsymbol{mr}}$. If an *mr* is greater than $\mathit{\Gamma}_b$, it is denoted as $\overrightarrow{\boldsymbol{mr}}$. Based on the two characteristics of *MR* sequence and certain *control factors*, we define judgment criterion 1.

Judgment Criterion 1 (JC1): Check the following conditions: ① the proportion of $\overleftarrow{mr}$ in *MR* is less than Λ_a and ② the proportion of $\overrightarrow{mr}$ in *MR* is greater than Λ_b. If both conditions are true, an LDoS attack is very likely to exist in a *WS*.

We call Λ_a and Λ_b in *Judgment Criterion 1* as *MR detection operators*, which are predetermined constants accessed from the training data.

(2) Judgment criteria about the departure degree of ACK traffic distribution

The two typical characteristics of ACK traffic fluctuation are also reflected on the probability distribution. It mainly manifests as *EWMA* statistic values of ACK traffic *cross* a confidence interval frequently. The *crossed* value is defined as the *Abnormal Point* (AP), and the others called *Normal Point* (NP). They are defined as follows:

Definition: If a $x^{ACK} \notin CI$, the x^{ACK} called AP, otherwise, called NP. A set of consecutive APs constitute a G_{AP}.

EWMA sequence diagram is shown in **Fig. 7**, in which attack free in *WS* of 0s~100s and an LDoS attack in *WS* of 100s~200s.

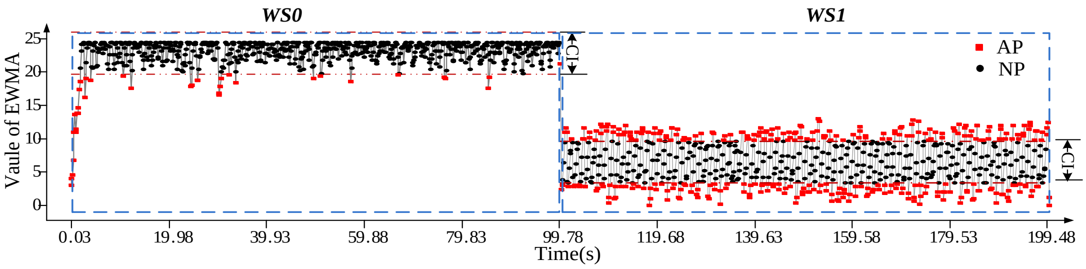


Fig. 7. *EWMA* sequence diagram of ACK traffic with LDoS attack at 100s~200s

Based on the characteristics of distribution of ACK traffic EWMA statistics on *CI*, we define judgment criterion 2.

Judgment Criterion 2 (JC2): If the proportion of G_{AP} in the EWMA statistic sequence of *ACK* traffic is greater than Λ_c, and the proportion of NP is less than Λ_d, then a probable LDoS attack exists in the *WS*.

In *Judgment Criterion 2*, Λ_c and Λ_d refers to *EWMA detection operators*, which are predetermined constants.

(3) Judgment criteria on attack effectiveness

Lower attack cost and relatively high attack effect are the differences between an LDoS attack and other attacks, by which we define judgment criterion 3 and 4.

Judgment Criterion 3 (JC3): If $\boldsymbol{S} \geq \Lambda_e$, then a probable LDoS attack exists in the *WS*.

Judgment Criterion 4 (JC4): In a *WS*, if an LDoS attack occurs, then $V_1 \geq V_2$.

Λ_e refers to attack effectiveness detection operators accessed from the training data.

To reduce misjudgments, we stipulate that it must meet the JC4 criterion and at least two of JC1~JC3 criteria in a *WS*, then an LDoS attack can be confirmed in the *WS*.

3.5. Algorithm Complexity

Inasmuch as there is huge traffic in a real network and because there is a necessity to detect an attack quickly, low algorithm complexity is necessary in designing a detection method. The complexity of the algorithm of the method proposed by this paper is analyzed as follows:

(1) Step Ⅰ is only sampling. The number of observed values is fixed under a given size of *WS* and sampling interval. If the observed values number is *N*, then space complexity and time complexity are all *O (N).*

(2) Step Ⅱ mainly obtain the *EWMA* statistic series and *MR* series of ACK traffic. Based on Eq. (11) to Eq. (15), the computation methods are all linear and only samples are stored. The amount of operation and samples depends on *N* in step (1). Therefore, the least space and time complexity are *O (N)*.

(3) Step Ⅲ is the judgment process. Nothing but detection operators and control factors are stored, so the space complexity is *O (1)*. This step only includes comparisons and analyses, which are linear dependents. Time complexity is *O (N)*.

Table 1. Algorithm complexity of different methods of LDoS attacks detection

Detection method	Algorithm complexity
Proposed scheme in this paper	*O (N)*
Vanguard	*O (N)*
WCM	*O (N)*
STM	*O(NlogN)*
DTM	*O(N2)*

Based on the analyses above, space complexity and time complexity of the proposed algorithm in this paper are linear dependents. As such, space complexity and time complexity of the algorithm are *O (N)*. We compare it with other LDoS detection methods [27] in **Table 1**.

4. Experiments and Evaluations

In the view of detection and analysis technology, the proposed scheme in this paper belongs to *anomaly detection* which has a low rate of false negative but a high rate of false-positive compared with *misuse detection*. Therefore, we designed three groups of experiments to verify the scheme. The first set of experiments proves the validity in detecting an LDoS attack. The second and third sets of experiments evaluate the *Error detection rate* of the scheme.

The first set experiments simulate LDoS attacks based on Network Simulator 2 (NS2) [35]. The second set experiments use LBNL [36] and WIDE [37] datasets to evaluate misjudgment when attack free. The third set of experiments evaluates the *misjudgment* under non-LDoS attacks with DARPA datasets.

The first set of experiments

The first set of experiments build the experiment system based on the NS2 simulator platform. The network topology is shown in **Fig. 8**.

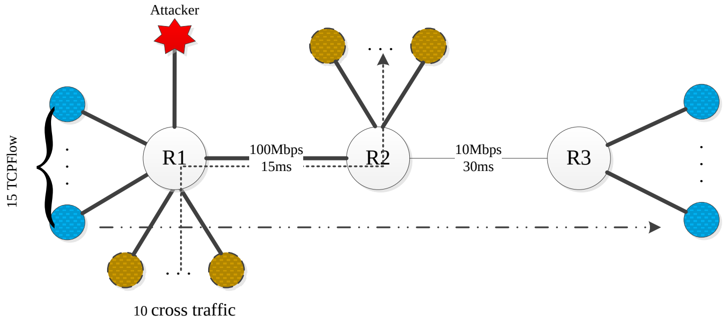


Fig. 8. Network structure diagram of the first set of experiments

R1, R2, and R3 are three routers in **Fig. 8**, where R2 is the *key router*. The link between R2 and R3 is the bottle link, whose bandwidth is 10 Mbps and whose delay time is 30ms. All the other links' bandwidth is 100 Mbps and the delay time is 15ms. The network includes 25 legitimate TCP flows, among which 10 TCP flows act as the *cross traffic*. All TCP flows use *New Reno* as the congestion control algorithm and RTO_{min}=1*s*. The active queue management mechanism is *Random Early Detection (RED)* algorithm. Other arguments of the network use the default values of NS2. Emulation time is from 0 s to 350s, and the TCP flows last from 0 s to 350 s.

The arguments of the LDoS attack are as follows: $T_{attack}=1.0s$, $R_{attack}=\{30,40\}Mbps$ $l_{attack}=\{150,200,250,1000\}ms$. The size of WS, i.e. T_s, is 100 s. The attack data types are *UDP*, *ICMP*, and useless *TCP*, respectively.

This set of experiments includes the experiment on training data (NO.I), the sudden drop of legit TCP flows (NO.II), and LDoS attack (NO.III to VIII). The details of the first set experiments are shown in **Table 2**.

Table 2. The details of the first set experiments

NO.	Legit TCP connection	LDoS attack time	l_{attack}	R_{attack}	Attack data type
I	0~350s 15 TCP connection	none	none	none	none
II	0~200s 15 TCP connection	none	none	none	none
	200~350s 2 TCP connection	none	none	none	none
III	0~350s 15 TCP connection	200~350	150	30	UDP
IV	0~350s 15 TCP connection	200~350	200	30	ICMP
V	0~350s 15 TCP connection	200~350	250	30	UDP
VI	0~350s 15 TCP connection	200~350	150	40	TCP
VII	0~350s 15 TCP connection	200~350	200	40	UDP
VIII	0~350s 15 TCP connection	200~350	250	40	ICMP

Each experiment occupies two *WS*, with WS1=[150 s, 250 s] and WS2=[250 s, 350 s].

The traffic of legit TCP and ACK with LDoS attacks and *WS* is shown in **Fig. 9**.

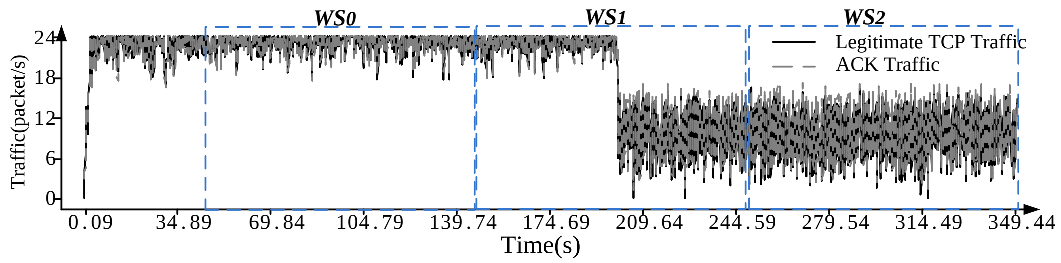
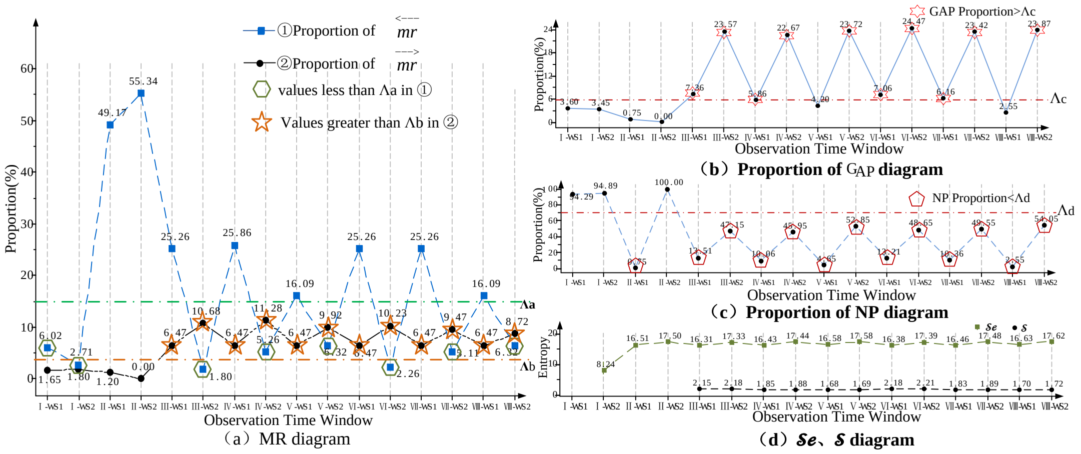


Fig. 9. Legit TCP and ACK traffic with LDoS attacks in the first set of experiments

By exercising the data of the first set of experiments as the training data, we get $\sigma_T = 2.26$ and $\sigma_{MR} = 1$. Other arguments are set as follows: Λ_a=3%, Λ_b=3%, Λ_c=5.71%, Λ_d=70% and Λ_e=0. Considering the detection sensibility, we set the weight coefficient *λ=0.95* in this set of experiments. We build *WS1=[150s,250s]* and *WS2=[250s,350s]* of each experiment, then obtain 16 *WS*s, such as *I- WS1,I- WS2,II- WS1,II- WS2*, and so on. The analysis results of the *MR* of ACK traffic are shown in **Fig. 10(a)**, G_{AP} ratio and *NP* ratio are shown in **Fig. 10(b)** and **Fig. 10 (c)**, and ***S e*** and ***S*** are shown in **Fig. 10(d)**.



（a）MR diagram

（b）Proportion of GAP diagram

（c）Proportion of NP diagram

（d）Se、S diagram

Fig. 10. The detection diagram of the first set of experiments

The detection results are shown in **Table 3**.

Table 3. Detection results of the first set of experiments

WS	I		II		III		IV	
	WS1	WS2	WS1	WS2	WS1	WS2	WS1	WS2
JC1	F	F	F	F	F	T	F	T
JC2	F	F	F	F	T	T	F	T

JC3	-	-	-	-	T	T	T	T
JC4	-	T	T	T	T	T	T	T

WS	**V**		**VI**		**VII**		**VIII**	
	WS1	WS2	WS1	WS2	WS1	WS2	WS1	WS2
JC1	F	T	F	T	F	T	F	T
JC2	F	T	T	T	T	T	T	T
JC3	T	T	T	T	T	T	T	T
JC4	T	T	T	T	T	T	T	T

Based on the regulation, we determine that the experiment III, IV, V, VI, VII, and VIII contain LDoS attacks. The detection result corresponds with actual values.

The second set of experiments

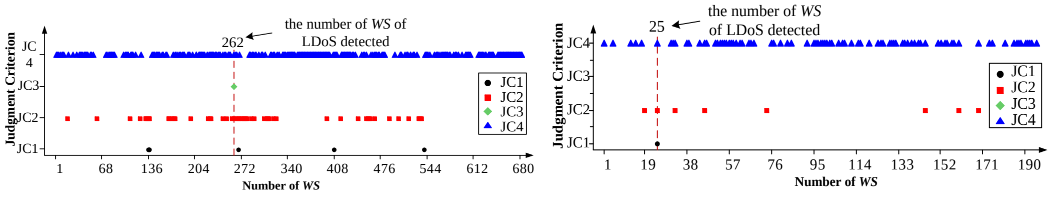
The second set of experiments evaluates miscarriage of justice in a normal network (without any attack). We choose twenty datasets randomly from LBNL and thirteen datasets from WIDE as test datasets.

Set T_s*=100 s*, sampling interval Δt*=0.1 s*, and sample size *n=5*. Considering the detection sensibility, we set λ=0.95. According to the length of a *WS* and the sampling period, we get 684 and 195 *WS*s in LBNL and WIDE datasets, respectively. Also, we employ separately the data at first 600 s and 300 s from LBNL and WIDE datasets, respectively. We employ *trimmed mean* algorithm to obtain σ_T and σ_{MR} of each dataset, and set the detection operators, as shown in **Table 4**.

Table 4. Detection operators

Datasets	Λ_a	Λ_b	Λ_c	Λ_d	Λ_e
LBNL	26.32%	7.89%	10.26%	80.00%	0.00
WIDE	2.54%	1.69%	6.72%	80.00%	0.00

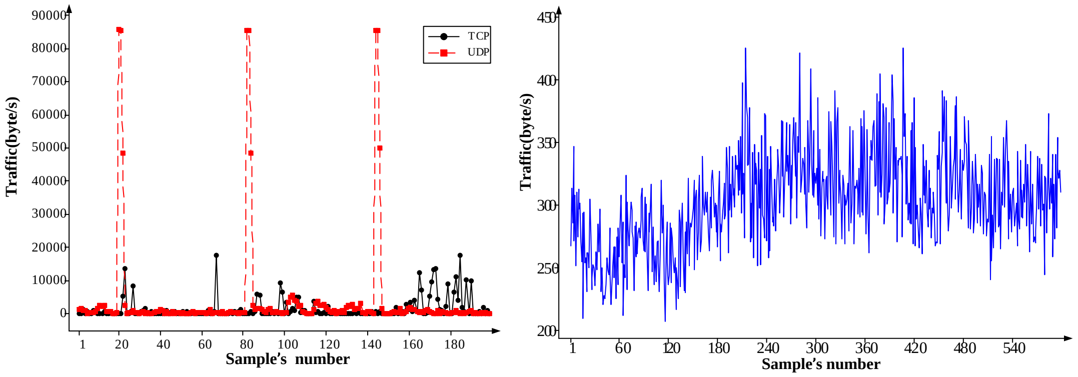
The judgment results are shown in **Fig. 11**.



(a) LBNL dataset judgment results (b) WIDE dataset judgment results

Fig. 11. The detection results of the second set of experiments

The misjudgments of LBNL and WIDE in this set of experiments are 0.14% and 0.51% separately. The reason for the misjudgment of LBNL datasets is because UDP traffic is much like the LDoS attack traffic in *WS*262, which is shown in **Fig. 12 (a)**. The misjudgment of WIDE is because that the fluctuation of ACK traffic in *WS* 25, is very similar as causing by LDoS attacks, as shown in **Fig. 12(b)**.



(a) TCP and UDP traffic in *WS262* of LBNL (b) ACK traffic in *WS25* of WIDE

Fig. 12. Diagram of misjudgment reason of the second set of experiments

The misjudgments of Vanguard are 1.62% and 2.04% [27] of LBNL and WIDE datasets, respectively. The main reason for the misjudgments is the sudden drop in legit TCP traffic, causing its distribution and the ACK traffic to change. Then, it causes $r_{d/a} \downarrow$ ^$\delta_f \uparrow$, which is the judgment condition of LDoS attack of Vanguard. However, a sudden drop in TCP traffic always exists in a normal network, thereby producing a certain rate of false positives.

The third set of experiments

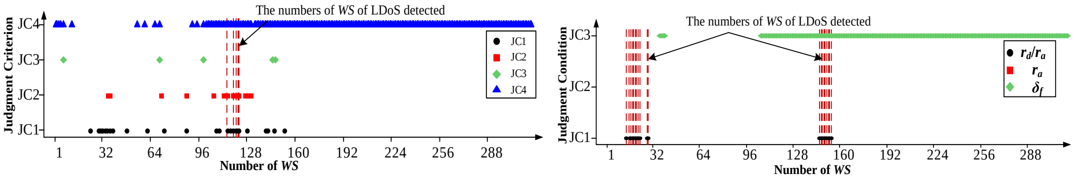
The third set of experiments evaluates miscarriage of justice during other attacks. This set of experiments adopts MIT Lincoln Laboratory's DARPA99 datasets [38] as test dataset. The first and the third weeks of the training data do not contain any attacks. The fourth and fifth weeks of data are the "test data". We choose the 0s~7200s in the inside tcpdump data of Tuesday of the first week as training data, and choose the inside tcpdump data of Monday of the fifth week as test data. There are 84 instances of 16 types of attacks distributed throughout the test data.

Set the length of detection window $T_s=250\ s$, the sampling interval $\Delta t=0.5\ s$, the sample size n=5. We get $\sigma_T = 5.32$, $\sigma_{MR} = 2.09$, and all detection operators from the training data. The parameters and the detection results are shown in **Table 5** and in **Fig. 13**, respectively. For comparison, we also show the parameters and the detection results of the Vanguard.

Table 5. Parameters and detection results of the two schemes

Detection method	WS number	Parameters	LDoS attack times	False positive rates
Our proposed scheme	317	σ_T =5.32, σ_{MR} =2.09, Λ_a =38.90%, Λ_b = 10.04%,Λ_c=3.27%,Λ_d=80.00%, Λ_e=0	5	1.57%

Vanguard	317	α_a=0,$\alpha_{d/a}$=4.98,α_δ=0.167,β=1.6,B=25. r_a=600,$r_{d/a}$=2.506,δ_f=0.5	14	4.42%



(a) Detection results of scheme this paper proposed

(b) Detection results of Vanguard with JC1∨{JC2∧JC3}where JC1=r_d/r_a↑ and JC2=r_a↓ and JC3=δ_f↑

Fig. 13. Detection results of our proposed scheme and Vanguard. In fig (b), r_d for the TCP data rate in bps, r_a for the TCP ACK rate in bps, and δ_f for the absolute change in the TCP data-rate distribution

There are misjudgments in our proposed scheme. This is because of the dictionary attacks at *WS*115, *WS*119, *WS*121, *WS*122, and *WS*123. The dictionary attack is a remote-to-local user attack, in which an attacker tries to gain access to some machines by making repeated guesses at possible usernames and passwords. In Drapa99 datasets, the sample dictionary attack consists of 40 login attempts, with a 4-second delay between each attempt. Therefore, the action is the same as that of an LDoS attack. Vanguard detects 20 times attacks, including 6 times DDoS attacks. Inasmuch as Vanguard can also detect DDoS attacks, it misjudged 14 times. Moreover, all misjudgments are caused by a sudden jump in legit traffic.

5. Conclusions and Future Work

In this paper, we addressed an important issue of detecting LDoS attacks. Based on the abnormal features of network traffic caused by LDoS attacks, three detection indexes were built as well as three judgment criteria were given. Our scheme adopt the Moving Range algorithm to compute the degree of ACK traffic influence, the Exponentially Weighted Moving-Average algorithm to compute the departure degree of ACK traffic distribution, and the entropy difference to describe attack effectiveness. Extensive comparative studies have been conducted to compare the detection rate and the false positive rate of our scheme, Vanguard and other detection methods for various real-world network topologies. Results show that that our method is very efficient to detect various types of LDoS attacks. Moreover, the proposed method has linear complexity, which makes its real-time detection practical, with an approximate time-length of WS delay.

Some of the parameters in this paper, such as confidence intervals and the relevant parameters in the detection criteria mainly obtain values from the training data. But this non-self-adaptive method may cause a certain error in LDoS attack detection. In the next work, from the views of reducing the detection error and enhancing the detection performance, we will focus on the adaptive mechanism of parameters. As the data sets used in this paper not including LDoS attacks, the experiments are not so sufficient. So we will incorporate appropriate data sets in the next research.

While we addressed issues and presented strategy related to LDoS attack detection, what needs to be done in the immediate future is to address the problem collectively with judgment criterion to increase detection precision. This is especially crucial to consider LDoS attack detection on different levels (e.g., application layer) to establish universality and tridimensional (Multi-layer applicable) LDoS attack detection scheme. This study is currently underway.

References

[1] V. D. Gligor, “A note on denial-of-service in operating systems,” *Software Engineering, IEEE Transactions on*, vol.10, no.3, pp.320-324, May.1984.Article (CrossRef Link).

[2] Ruoyu Yan, Qinghua Zheng and Haifei Li, “Combining Adaptive Filtering and IF Flows to Detect DDoS Attacks within a Router,” *KSII Transactions on Internet and Information Systems*, vol.4, no.3, pp.428-451, Jun.2010. Article (CrossRef Link).

[3] A. Kuzmanovic and E.W. Knightly, "Low-rate TCP-targeted denial of service attacks: the shrew vs. the mice and elephants," in *Proc. of SIGCOMM*, pp.75-86, 2003. Article (CrossRef Link).

[4] M. Guirguis, A. Bestavros, and I. Matta, “Exploiting the tran-sients of adaptation for RoQ attacks on internet resources,” in *Proc. of the 12th IEEE International Conference on Network Protocols*, pp.184–195, Oct.2004. Article (CrossRef Link).

[5] Xiaopu Luo and R. K. C. Chang, “On a new class of pulsing denial-of-service attacks and the defense,” in *Proc. of the Network and Distributed System Security Symposium*, pp.1-19, Feb.2005. Article (CrossRef Link).

[6] Zhang Jing, Hu Huaping and Liu Bo, “Robustness of RED in Mitigating LDoS Attack,” *KSII Transactions on Internet and Information Systems*, Vol.5, no.5, May.2011. Article (CrossRef Link).

[7] Changwang Zhang, Jianping Yin, Zhiping Cai and Weifeng Chen, “RRED: Robust RED Algorithm to Counter Low - rate Denial -of -Service Attacks,” *IEEE Communication Letter*, vol.14, no.5, pp.489-491, May.2010. Article (CrossRef Link).

[8] Jon Postel. RFC 793: Transmission Control Protocol, September 1981. Available from ftp://ftp.rfc-editor.org/in-notes/rfc793.txt as of Aug.2003. Article (CrossRef Link).

[9] G. Maciá-Fernández, J. E. Díaz-Verdejo, and P. Garcia-Teodoro, “Evaluation of a low-rate DoS attack against iterative servers,” *Computer Networks*, pp. 1013-1030, vol.51, no.4, 2007. Article (CrossRef Link).

[10] G. Maciá-Fernández, J. E. Díaz-Verdejo, and P. Garcia-Teodoro, “LoRDAS: A low-rate DoS attack against application servers,” in *Proc. CRITIS’07*, vol.5141, pp.197–209, 2008. Article (CrossRef Link).

[11] G. Maciá-Fernández, J. E. Díaz-Verdejo, and P. Garcia-Teodoro, “Evaluation of a low-rate DoS attack against application servers,” *Computer Security*, vol.27, pp.335–354, 2008. Article (CrossRef Link).

[12] G. Maciá-Fernández, Rafael A, Rodriguez-Gomez and Jesus E. Diaz-Verdejo, "Defense techniques for low-rate DoS attacks against application servers, " *Computer Networks*, vol.54, no.15, pp.2711–2727, Oct. 2010. Article (CrossRef Link).

[13] Maciá-Fernández, G., J.E. Díaz-Verdejo and P. García-Teodoro, "Mathematical model for low-rate dos attacks against application servers," *Information Forensics and Security*, vol.4, no.3, pp.519 – 529, Sep.2009. Article (CrossRef Link).

[14] Salah K, Sattar K, Sqalli M, et al, "A potential low-rate DoS attack against network firewalls," *Security and Communication Networks*, vol.4, no.2, pp.136–146, Feb.2011. Article (CrossRef Link).

[15] He Yanxiang, "LDoS attack in ad-hoc network," in *Proc of 6th International Conference on Wireless On-Demand Network Systems and Services*, pp.251-257, Feb.2009. Article (CrossRef Link).

[16] Guirguis Mina, Bestavros Azer and Matta Ibrahim, "On the impact of low-rate attacks," in *Proc. Communications*, pp.2316-2321, Jun.2006. Article (CrossRef Link).

[17] Chen Y and Hwang K, "Collaborative detection and filtering of shrew DDoS attacks using spectral analysis," *Journal of Parallel and Distributed Computing*, vol.66, no.9, pp.1137–1151, Sep.2006. Article (CrossRef Link).

[18] He Yanxiang, Cao Qiang, Liu Tao, Han Yi and Xiong Qi, "A low- rate dos detection method based on feature extraction using wavelet transform," *Journal of Software*, vol.20, no.4, pp.930 -941, Apr.2009. Article (CrossRef Link).

[19] S. Sarat and A. Terzis, "On the effect of router buffer sizes on low-rate denial of service attacks," in *Proc. IEEE ICCCN 05*, pp.281–286, 2005.Article (CrossRef Link).

[20] H. Sun, J. C. S. Lu, and D. K. Y. Yau, "Defending against low-rate TCP attacks: dynamic detection and protection," in *Proc. of the 12th IEEE International Conference on Network Protocols*, pp.196-205, Oct.2004. Article (CrossRef Link).

[21] Y. K. Kwok, R. Tripathi, Y. Chen, and K. Hwang, "HAWK: Halting anomalies with weighted choking to rescue well-behaved TCP Sessions from Shrew DDoS attacks," in *Proc. of the 3rd International Conference on Computer Network and Mobile Computing*, pp.423-432, Aug.2005. Article (CrossRef Link).

[22] WU Zhi-jun, ZENG Hua-long, and YUE Meng, "Approach of detecting LDoS attack based on time window statistic," *Journal on Communications*, vol.31, no.12, pp.55-62, Dec.2010. Article (CrossRef Link).

[23] S Athuraliya, V H Li, S H Low, Q Yin. REM, "Active queue management," *IEEE Network*, pp.48-53, vol.15, no.3, 2001. Article (CrossRef Link).

[24] A. Shevtekar, K. Anantharam, and N. Ansari, "Low rate TCP denial-of-service attack detection at edge routers," *IEEE Communications Letters*, vol.9, no.4, pp.363–365, 2005. Article (CrossRef Link).

[25] Y. Xu and R. Guerin, "On the robustness of router-based denial-of-service (DoS) defense systems," *ACM SIGCOMM Computer Communication Review*, vol.35, no.3, pp.47–60, 2005. Article (CrossRef Link).

[26] Xiaopu Luo, Edmond W.W. Chan, Rocky K.C. Chang, "Vanguard: A new detection scheme for

a class of TCP-targeted denial-of-service attacks," in *Proc. of the 10th IEEE/IFIP Network Operations and Management Symposium*, pp.507-518, Apr.2006. Article (CrossRef Link).

[27] Xiapu Luo, Edmond W. W. Chan, and Rocky K.C.Chang, "detecting pulsing denial-of-service attacks with nondeterministic attack intervals," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, Jan.2009. Article (CrossRef Link).

[28] Sean McPherson and Antonio Ortega, "Detecting low-rate periodic events in Internet traffic using renewal theory," in *Proc. of ICASSP'2011*. pp.4336-4339, May.2011. Article (CrossRef Link).

[29] WU Zhijun, and PEI Baosong, "The detection of LDoS attack based on the model of small signal," *ACTA ELECTRONICA SINICA*, vol.39, no.6, Jun.2011. Article (CrossRef Link).

[30] Yang Xiang, Ke Li, and Wanlei Zhou, "Low-Rate DDoS attacks detection and traceback by using new information metrics," *IEEE Transactions on Information Forensics and Security*, vol.6, no.2, pp.426-437, Jun.2011. Article (CrossRef Link).

[31] W.E.Leland,M. S. Taqqu, W. Willinger, and D.V Wilson, "On the self-similar nature of Ethernet traffic," in *Proc.of ACM Sigcomm'93*, pp.183-193, Oct.1993. Article (CrossRef Link).

[32] K. Park and W. Willinger, "Self-similar network traffic and performance evaluation," *John Wiley & Sons*, JAN. 2002. Article (CrossRef Link).

[33] Thomas K, Mart M, Michalis F, et al, "Long-range dependence-ten years of Internet traffic modeling," *IEEE Internet Computer*, vol.8, no.5, pp.57-64, Sept-Oct, 2004. Article (CrossRef Link).

[34] T.Karagiannis, M.Molle, M.Faloutsos, and A. Broido, "A nonstationary poisson view of Internet traffic," in *Proc.of INFOCOM 2004*, pp.1558-1569, Mar.2004. Article (CrossRef Link).

[35] K. Fall, K. Varadhan, "The NS manual," http://www.isi.edu/nsnam/ns/, 2009. Article (CrossRef Link).

[36] Lawrence Berkeley National Laboratory (LBNL) and ICSI, "LBNL's internal enterprise traffic," http://www.icir.org/enterprise-tracing, 2005. Article (CrossRef Link).

[37] MAWI Working Group, "Packet traces from WIDE backbone," http://tracer.csl.sony.co.jp/mawi, 2006. Article (CrossRef Link).

[38] Cyber Systems and Technology Group, "1999 DARPA Intrusion Detection Evaluation Data Sets,"http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/data/1999data.html,1999. Article (CrossRef Link).

Kai Chen received his M.S. degree in Huazhong University of Science and Technology, in 2008. Currently he is pursuing the PhD degree in the School of Computer Science & Technology at Huazhong University of Science and Technology. His current research interests include computer network application, computer network security and computer network protocol analysis.

Huiyu Liu is a lecturer of School of Computer Science & Technology at Huazhong University of Science and Technology. In 2011, He received the Ph.D. degree in Systems Architecture from Huazhong University of Science and Technology. His research interests include network security, cloud computing, web service and semantic network.

Xiaosu Chen is a professor of Huazhong University of Science and Technology. His research interests include computer network application, computer network security, computer network protocol analysis, and image recognition.