Tang *et al. Hum. Cent. Comput. Inf. Sci.* (2020) 10:6
https://doi.org/10.1186/s13673-020-0210-9

Human-centric Computing and Information Sciences

RESEARCH Open Access

Low-rate DoS attack detection based on two-step cluster analysis and UTR analysis

Dan Tang[1], Rui Dai[1*], Liu Tang[1] and Xiong Li[2]

*Correspondence:
dairui@hnu.edu.cn
[1] College of Computer Science and Electronic Engineering, Hunan University, Changsha, China
Full list of author information is available at the end of the article

Abstract

Low-rate denial of service (LDoS) attacks send attacking bursts intermittently to the network which can severely degrade the victim system's Quality of Service (QoS). The low-rate nature of such attacks complicates attack detection. LDoS attacks repeatedly trigger the congestion control mechanism, which can make TCP traffic extremely unstable. This paper investigates the network traffic' characteristics, in which variance and entropy are used to evaluate the TCP traffic's characteristics, and the ratio of UDP traffic to TCP traffic (UTR) is also analyzed. Thus, a detection method combining two-step cluster analysis and UTR analysis is proposed. Through two-step cluster analysis which is one of the machine learning algorithms, network traffic is divided into multiple clusters and then clusters subjected to LDoS attacks are determined using UTR analysis. NS2 simulation platform and test-bed network environment aim to evaluate the detection approach's performance. To better assess the effectiveness of the method, public dataset WIDE is also utilized. Experimental results with a good performance prove that the proposed detection approach can accurately detect LDoS attacks.

Keywords: LDoS attacks, Attack detection, Machine learning, Two-step cluster analysis, UTR analysis

Introduction

Denial of service (DoS) attack [1, 2] is a common attack vector, which generally seeks to exhaust the limited network resources, resulting in the legitimate users' requests not being processed. DoS attacks are becoming more widespread, targeting IoT networks [3, 4], SDN networks [5, 6], cloud computing environments [7, 8] and cyber-physical systems [9]. Aiming to combat DoS attacks, many methods have been proposed, in which a common detection method is based on abnormal statistical characteristics.

Another type of DoS attack is the low-rate denial of service (LDoS) attack [10–12] that is hard to be accurately detected due to its low-rate nature. Many LDoS attacks have emerged, such as Shrew attacks [13], LoRDAS attacks [14], slow DoS attacks(e.g. Slow Next, SlowComm) [15, 16], etc. These attacks have the same characteristics, that is, they do not need to maintain sustained high-speed attack traffic to cause damage. Among all these attacks, TCP-targeted LDoS attacks are one of the most common LDoS attacks. To reduce the TCP's throughput, the attacker sends packet bursts

at intervals, causing the response and adjustment of the congestion control mechanism. Because the adaptive response and adjustment caused by the attack are legal, it increases the difficulty of attack detection. When the network is congested, the data packet will be lost which will cause the network traffic to compete for the limited network resources [17]. Since UDP traffic is not impacted by the congestion control mechanism, it can compete for more network resources than TCP traffic when congestion occurs. Therefore, LDoS attacker typically uses UDP traffic as an attacking burst [18, 19]. Figure 1 shows the model of TCP-targeted LDoS attacks [20], three parameters (T, L, R) are used to describe the attack model where T denotes the period, L represents the burst length, and R is the burst rate. To maximize the attack outcome, the attacker sets the value of T based on the retransmission timeout (RTO). In each attack period T, the attack burst length L is much smaller than T, which means that the average attack rate ($R \times \frac{L}{T}$) of LDoS attacks is relatively small. Therefore, the method based on abnormal statistical characteristics cannot effectively detect this attack.

To detect the TCP-targeted LDoS attack, a method using two-step cluster analysis [21] and UTR analysis is proposed. The cluster analysis is used to divide the network traffic with similar characteristics into the same cluster since the network traffic subjected to LDoS attacks has similar characteristics. Then UTR analysis is used to determine which clusters have suffered LDoS attacks. The major contributions are as follows.

- Network traffic's characteristics under LDoS attacks are analyzed using variance and entropy.
- Network traffic with different characteristics is divided into different clusters through cluster analysis and UTR analysis is then used to identify clusters containing LDoS attacks.
- The detection method has been evaluated in NS2 simulation platform, test-bed network environment and public dataset WIDE [22]. Experimental results prove that the method has a good performance in detecting LDoS attacks.

The paper's organization is as follows. Related researches on LDoS attack detection are presented in "Related work" section. Network traffic's characteristics are investigated in "Analysis of network traffic characteristics" section. The detection approach is described in "Proposed approach" section. The detection method is evaluated

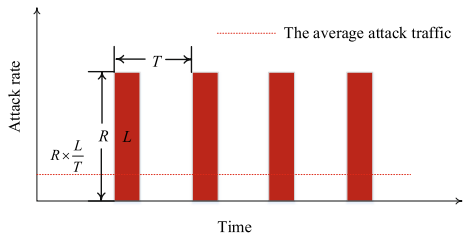


Fig. 1 The model of TCP-targeted LDoS attacks

via experiments in different platform and the experimental results are analyzed in "Experiments and results analysis" section. The whole paper is summarized and the future work is introduced in "Conclusion and future work" section.

Related work

To combat LDoS attacks, researchers have proposed many defense strategies [23–25] including two categories. The first is a feature-based defense strategy, and the second is an anomaly-based defense strategy.

The feature-based defense strategy

Although LDoS attacks have great concealment, their periodicity and impulsivity provide the basis for detection. The feature-based defense strategy implements detection by analyzing the principles and features of LDoS attacks.

Wu et al. [26] proposed a method based on the sequence matching, which uses the Smith–Waterman algorithm and double threshold rules to detect LDoS attacks. Yue et al. [27] proposed a method to detect the LDoS attack traffic using the wavelet energy spectrum and the combined neural network. Sun et al. [28] proposed the DTW detection method using dynamic time-wrapping to perform feature matching on the sampled network traffic, and the DRR algorithm is used to limit the attack flow on the router for the bandwidth allocation and resource protection. For defending against LDoS attacks in wireless sensor networks, Cao et al. [29] developed AccFlow, in which the flow's packets with larger loss rate are more aggressively to be dropped by AccFlow since attacking flows are featured with high loss rates.

The anomaly-based defense strategy

When LDoS attacks occur, the features of the network will be changed. The anomaly-based defense strategy implements the LDoS attack detection according to the anomaly of the network features.

Zhang et al. [30] proposed using wavelet multi-scale analysis and adaptive KPCA to detect LDoS attacks, in which the KPCA method is used as an anomaly detection model. Wu et al. [19] proposed a method to detect LDoS attack flows according to the network multifractal, in which LDoS attacks are confirmed according to the D-value. The anomaly of network multifractal means the occurrence of LDoS attacks since the attack flow will change this characteristic. Periodic LDoS attack flows and normal network flows have different frequency domain features. Many detection methods based on frequency domain feature anomalies have been proposed [31–33]. For example, Chen et al. [18] proposed to combine power spectrum analysis and information entropy to detect and mitigate LDoS attacks.

To accurately detect LDoS attacks, new countermeasures still need to be designed. Machine learning is recently applied to LDoS attack detection [34–36]. This paper proposed an LDoS detection method using machine learning, in which two-step cluster analysis is used to evaluate TCP's characteristics and TCP traffic with similar characteristics is divided into the same cluster. Then clusters suffered LDoS attacks are determined through UTR analysis.

Analysis of network traffic characteristics

From the perspective of network architecture, network behavior can be researched through network traffic, because network behavior can often be reflected in the characteristics of the network traffic [37]. Network traffic can be defined as a stochastic process $\{X(t), t = n\triangle t, n \in z^+\}$, where Δt is the sampling time interval and $X(t)$ represents the number of data packets that reached the detection point in the period $(t - \Delta t, t]$. Define the data packet sequence within a time length $Time_{DU}$ as the detection unit, that is, the detection unit is $\{X(i+1), \ldots, X(i+N)\}$, where $N = \frac{Time_{DU}}{\Delta t}$. The proposed detection approach starts with the packet process of network traffic to realize the detection of LDoS attacks.

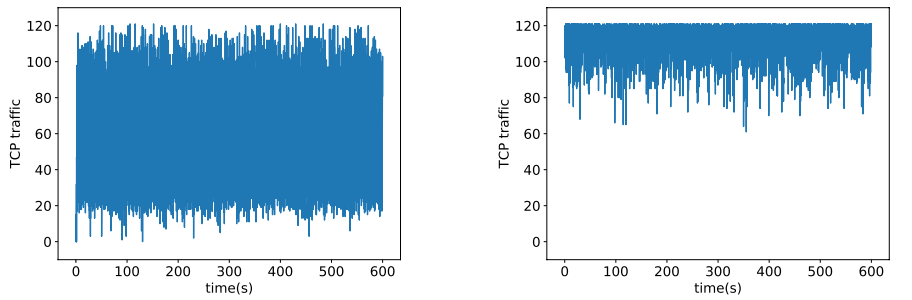
LDoS attacks exploit the congestion control mechanism's defects to limit TCP traffic on the network. When the congestion control mechanism is triggered, the congestion window will be reduced to one package and wait for an RTO time before attempting to retransmit. If the attacker can acquire the exact value of RTO in each round, then (s)he can send the attack packages to the victim system when TCP source-end retransmits TCP packages. Consequently, this leads to failure for each retransmission and the capacity of the congestion window will always be one and the throughput of TCP traffic will be zero. However, the exact value of RTO in each round is hard to acquire. Generally, an LDoS attack will repeatedly trigger the congestion control mechanism for achieving the attack effect of reducing TCP throughput. Therefore, TCP traffic during the LDoS attack will fall into a vicious cycle of drop-recovery-drop. Compared with TCP traffic in normal network scenarios, TCP traffic in LDoS attack scenarios becomes more discrete and unstable.

Variance is used to measure the TCP's discrete characteristics. The formula of variance is Eq. 1, where m is the mean value of the number of data packets in a detection unit and N is the length of the detection unit. Shannon entropy is used to evaluate the TCP's randomness and uncertainty. The formula of the Shannon entropy is Eq. 2, where p_i is the probability of the value of the number of the data packet in a detection unit and N is the length of the value sequence. If the detection unit $DU = \{15, 10, 12, 15, 12, 15, 11, 10, 12, 12\}$, then the value sequence is $value = \{15, 10, 12, 11\}$, where 0.3 represents the probability of 15 appearing in the detection unit, and other values are similar.

$$V = \frac{1}{N}\sum_{i=1}^{N}(X_i - m)^2 \tag{1}$$

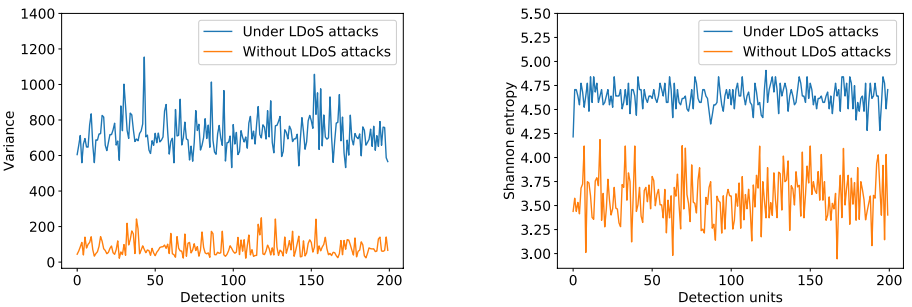
$$H = -\sum_{i=1}^{N} p_i \log_2 p_i \tag{2}$$

Figure 2 shows TCP traffic in different network scenarios, in which (a) is the scenario that LDoS attacks occur and (b) is the scenario that no LDoS attacks occur. As shown in the figure, TCP traffic's characteristics in these two scenarios are different. TCP traffic subjected to LDoS attacks is unstable and in a cycle of drop-recovery-drop because of repeated triggers of the congestion control mechanism. TCP traffic without LDoS attacks is stable and concentrates on a high level. Figure 3 shows variance and Shannon



a TCP traffic under LDoS attacks. **b** TCP traffic without LDoS attacks.

Fig. 2 TCP traffic in different network scenarios



a Variance of TCP traffic. **b** Shannon entropy of TCP traffic.

Fig. 3 Variance and Shannon entropy of TCP traffic in different network scenarios

entropy of TCP traffic in different network scenarios, in which (a) shows the variance of TCP traffic and (b) shows the Shannon entropy of TCP traffic. Compared to TCP traffic without LDoS attacks, variance and Shannon entropy of TCP traffic under LDoS attacks are significantly larger.

Proposed approach

According to the analysis in the preceding section, the detection method using two-step cluster analysis and UTR analysis is proposed. Figure 4 shows the detection process. First, network traffic is collected for LDoS attack detection and the packet sequence is divided into the detection units. Then variance and Shannon entropy of each detection unit are calculated for evaluating the TCP's characteristics. And two-step cluster analysis is used to divide detection units into multiple clusters. TCP traffic with the same characteristics is gathered into the same cluster, that is, TCP traffic that may have suffered LDoS attacks will be divided into the same cluster. Finally, UTR analysis is used to determine which clusters have suffered an LDoS attack.

Two-step cluster analysis

Two-step cluster analysis [21] includes balanced iterative reducing and clustering using hierarchies (BIRCH) algorithm and agglomerative hierarchical clustering algorithm. The

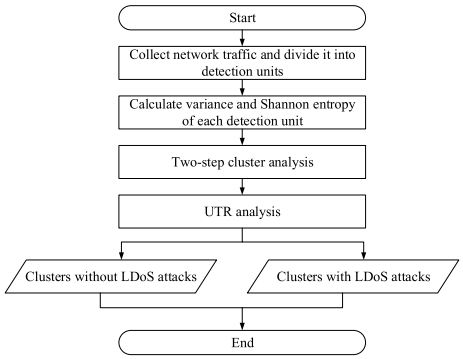


Fig. 4 Process diagram of the proposed detection method

first step is to investigate the distance of all records to build a cluster feature (CF) tree and records with high similarity will be divided into the same node. The second step is to further merge the subclusters until the optimal number M of clusters using the agglomeration hierarchical clustering algorithm, in which subclusters are obtained through CF tree's leaf nodes. In the proposed method, the optimal number M of clusters is determined based on the within-group mean square error (WMSE).

BIRCH algorithm

BIRCH algorithm uses cluster features that can effectively compress the data size to describe the information of each cluster. The CF of a cluster containing N D-dimensional data points $\{\vec{x}_n, n = 1, 2, \ldots, N\}$ is a triple $\vec{CF} =< N, \vec{\Sigma}_l, \Sigma_s >$, where N is the number of data points, $\vec{\Sigma}_l = \sum_{i=1}^{N} \vec{x}_n$ is the linear sum of the data points, and $\Sigma_s = \sum_{i=1}^{N} \vec{x}_n^2$ is the sum of the square of each data point.

BIRCH algorithm builds a CF tree based on the distance of all records and records with closer distance will be divided into the same node. The closer the records are, the higher their similarity. The centroid of a cluster that contains N data points is Eq. 3. In this paper, the proposed method uses centroid Euclidian distance to measure the distance between clusters and the formula is Eq. 4. Algorithm 1 shows the BIRCH algorithm and CF tree structural details can refer to [38]. The input of the algorithm is M 2-dimensional data points which are the variance and Shannon entropy of detection units. The output of the algorithm is N subclusters, which are obtained based on the leaf nodes of the CF tree.

$$c = \frac{1}{N} \sum_{i=1}^{N} \vec{x}_n \tag{3}$$

$$distance(c_1, c_2) = \sqrt{(c_1 - c_2)^2} \tag{4}$$

Algorithm 1 BIRCH algorithm

Input: M data points
Output: N subclusters
  repeat
    Calculate the cluster feature $\vec{CF}$ of data points
    Insert record to construct CF tree
  until Insert all records into the CF tree

Agglomerative hierarchical clustering algorithm

Subclusters obtained from the CF tree's leaf nodes will be merged one by one through the agglomerative hierarchical clustering algorithm until the optimal number M of clusters. WMSE is used to determine the value of M and the formula of WMSE is Eq. 5. As the number of data points in the cluster decreases with the number of clusters increases, the data points become more concentrated and the WMSE value also decreases. The optimal number M is determined based on the "elbow" of WMSE's change with the number of clusters because when WMSE slowly decreases, further increasing the number of clusters cannot enhance the effect. Algorithm 2 shows the aggregation hierarchical clustering algorithm. Firstly the centroid of each subcluster and the proximity matrix are calculated. When the two closest subclusters are merged, the centroid of the new cluster and the proximity matrix are updated. Subclusters are merged until there are only M clusters.

$$WMSE = \frac{1}{N}\sum_{i=1}^{N}(x_i - c)^2 \tag{5}$$

Algorithm 2 Agglomerative hierarchical clustering algorithm

Input: N subclusters
Output: M clusters
  Calculate the centroid of each subcluster
  Calculate the proximity matrix
  repeat
    Merge the two closest subclusters
    Update the centroid of the new cluster
    Update the proximity matrix
  until M clusters

UTR analysis

UTR analysis is performed on the M clusters obtained from the two-step cluster analysis to determine which clusters suffered LDoS attacks. UTR_{DU} is the ratio of UDP traffic to TCP traffic in a detection unit and the formula is Eq. 6, which is the number of all UDP traffic packets divided by the number of all TCP traffic packets in the detection unit. UDP_{DU} is the number of UDP traffic packets and TCP_{DU} is the number of TCP traffic packets in the detection unit. $UTR_{cluster}$ is the ratio of UDP traffic to TCP traffic in a cluster and the formula is Eq. 7, which is the UTR_{DU}'s mean value in the cluster.

$$UTR_{DU} = \frac{SUM(UDP_{DU})}{SUM(TCP_{DU})} \tag{6}$$

$$UTR_{cluster} = \frac{1}{N}\sum_{i=1}^{N} UTR_{DUi} \tag{7}$$

Since UDP traffic is not impacted by the congestion control mechanism, it can compete for more network resources than TCP traffic when congestion occurs. Therefore, an LDoS attacker typically uses UDP traffic as an attacking burst [18]. Figure 20 shows the UTR frequency distribution in the WIDE dataset [22], in which no LDoS attacks occur. In normal scenarios, UDP traffic and TCP traffic will remain at a roughly constant ratio. When LDoS attacks occur, the throughput of TCP traffic will be severely reduced. Although the average rate of LDoS attack flows will not change significantly, UTR will change significantly. Algorithm 3 shows UTR analysis, where Ω is the threshold that is obtained by training data without LDoS attacks. The threshold Ω is determined based on the mean value of UTR and the standard deviation value of UTR and the formula is Eq. 8. The value of z will be discussed in "Results analysis" section. If the UTR of a cluster is larger than the threshold Ω, it is determined that the cluster has suffered LDoS attacks.

$$\Omega = Mean(UTR) + z \times Std(UTR) \tag{8}$$

Algorithm 3 UTR analysis

Input: M clusters, threshold Ω
Output: Determine which clusters suffered LDoS attacks
repeat
  Calculate UTR_i for each cluster
  if $UTR_i > \Omega$ **then**
    $cluster_i$ suffered LDoS attacks
  else
    $cluster_i$ did not suffer LDoS attacks
  end if
until M clusters are all analyzed

Experiments and results analysis

Experiments are carried out in NS2 and test-bed to evaluate the efficiency and performance of the detection method. For the purpose of further verifying the detection method's performance, experiments are also performed in the public dataset WIDE [22] from the perspective of evaluating the detection method's false positive rate.

Experiments in NS2 simulation platform

Experimental network environment

Experiments are constructed in NS2 simulation platform and Figure 5 shows the network topology. There are three routers in the simulation network, in which the link bandwidth between Router1 and Router2 is 100Mbps and the delay is 15ms; the link bandwidth between Router2 and Router3 is 10Mbps and the delay is 30ms, which is the bottleneck link in the network. There are thirty legitimate TCP links in the network, and TCP traffic is sent from node1 to node8 via three routers; there are twenty background TCP links, and TCP traffic is sent from node4 to node6 via Router1 and Router2; node2

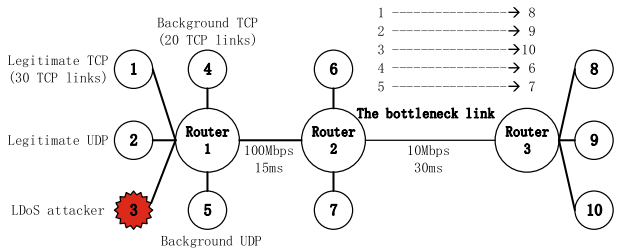


Fig. 5 Network topology in NS2 simulation platform

Table 1 Parameters setting of LDoS attacks in NS2 simulation platform

	T(s)	L(s)	R(Mbps)	Attack time period (s)
Training data	0	0	0	0
Testing data	1	0.1	15	1320–1500
	1	0.2	15	1620–1800
	2	0.1	15	1920–2100
	2	0.2	15	2220–2400
	1	0.1	25	2520–2700
	1	0.2	25	2820–3000
	2	0.1	25	3120–3300
	2	0.2	25	3420–3600

sends legitimate UDP traffic to node9 via three routers; node5 sends background UDP traffic to node7 via Router1 and Router2. Legitimate TCP traffic and legitimate UDP traffic pass through the bottleneck link and background TCP traffic and background UDP traffic do not pass through the bottleneck link.

Table 1 shows the parameters of LDoS attacks in NS2 simulation platform. The first group is training data that contain 7200 s data without LDoS attacks; the second group is testing data containing 3600 s data of which eight LDoS attacks occurred and each attack lasted 180 s. The length of the detection unit is set as 3 s.

Results analysis

In this subsection, the experimental results are presented and discussed. Network traffic passing through the bottleneck link is collected for analysis and detection. The normalized variance and normalized Shannon entropy are used to characterize TCP traffic. Figure 6 shows the distribution of testing data in two-dimensional space and data points of TCP traffic with similar characteristics are distribute closer. Data points representing the traffic under LDoS attacks are generally distributed in space's upper right corner, and data points representing the traffic without LDoS attacks are generally distributed in space's lower left corner, which is consistent with the analysis of network traffic characteristics in "Analysis of network traffic characteristics" section.

Figure 7 shows the UTR frequency distribution of training data, which approaches the normal distribution. The mean value of UTR (μ) is about 0.44664 and the standard deviation value of UTR (σ) is about 0.00754. The value of the z in Eq. 8 affects the

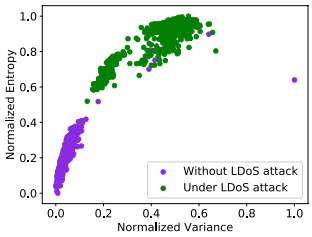


Fig. 6 Distribution of testing data in two-dimensional space

accuracy of the UTR analysis and z is determined to be three in this paper, which 3σ means a confidence interval of 99.7% [39]. According to Eq. 8, the threshold is about 0.46927. Figure 8 shows the results of BIRCH. Thirty sub-clusters are obtained and data points with similar characteristics are aggregated in the same cluster. Figure 9 shows the change of WMSE with the number of clusters. The optimal number M of clusters is determined to four according to the WMSE's "elbow point". Figure 10 shows the results of the aggregation hierarchical clustering algorithm which is also the results of two-step cluster analysis. Through two-step cluster analysis, the data points with closer distance in the space are divided into the same cluster, which means that data with similar characteristics are aggregated in the same cluster. Therefore, TCP traffic suffered LDoS attacks will be concentrated in the same cluster. Figure 11 shows the results of UTR analysis, with the red dashed line indicating the threshold. The UTRs of clusters 0, 2 and 3 are greater than the threshold, and the UTR of cluster 1 is less than the threshold, which indicates that cluster 0, cluster 2 and cluster 3 have suffered LDoS attacks. There are nine detection units falsely reported as being subjected to LDoS attacks. False positive detection units are 0, 500, 600, 700, 723,

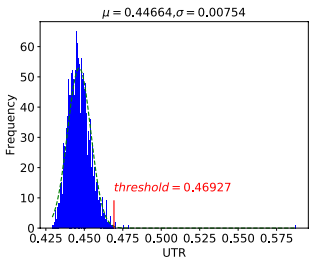


Fig. 7 UTR frequency distribution in training data

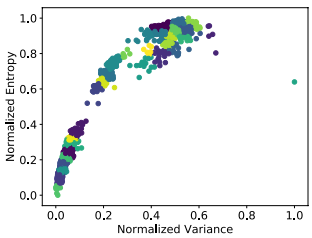


Fig. 8 Results of BIRCH

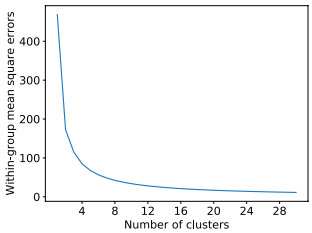


Fig. 9 The change of WMSE with the number of clusters

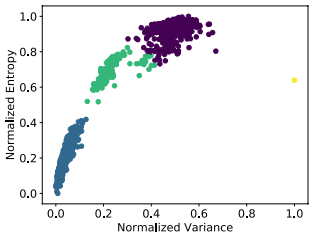


Fig. 10 Results of two-step cluster analysis

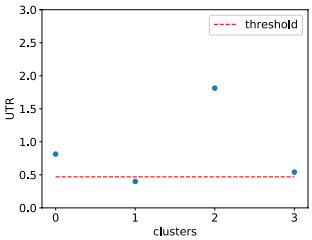
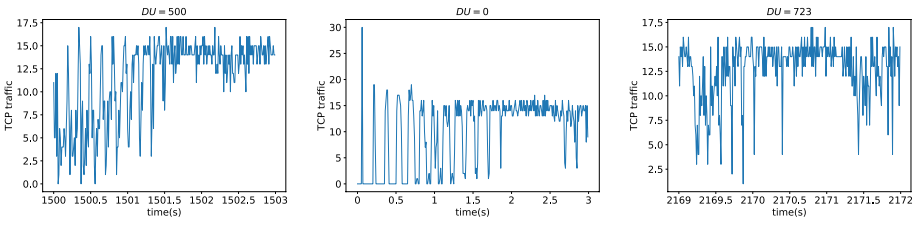


Fig. 11 Results of UTR analysis



a TCP traffic in DU500 **b** TCP traffic in DU0 **c** TCP traffic in DU723

Fig. 12 TCP traffic in false positive detection units

800, 900, 1000, 1100. Detection units 500, 600, 700, 800, 900, 1000, 1100 contain network traffic just after the LDoS attack and the network traffic characteristics are still affected by LDoS attacks. (a) in Fig. 12 shows the TCP traffic in detection unit 500, which indicates that TCP traffic is discrete and unstable. (b) in Fig. 12 shows the TCP

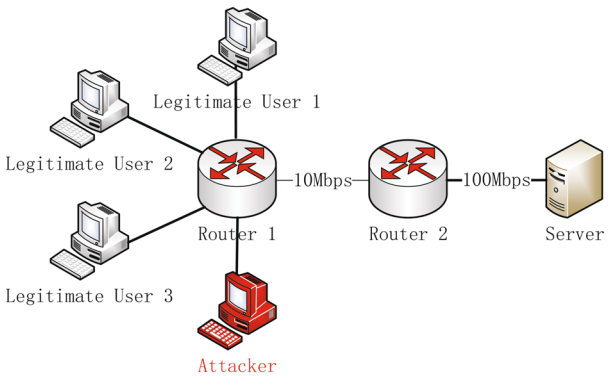


Fig. 13 Network topology in test-bed network environment

Table 2 Devices configuration in test-bed network environment

Device	Operate system	Software	Function
Legitimate User 1	Windows 7	Python socket	Legitimate user sending network traffic
Legitimate User 2	Windows 7	Python socket	Legitimate user sending network traffic
Legitimate User 3	Windows 7	Python socket	Legitimate user sending network traffic
Attacker	Windows 7	Python socket	LDoS attacker
Server	Windows 7	Python socket	Server receiving network traffic
Router1	–	Routing Protocol	Forwarding packets
Router2	–	Routing Protocol	Forwarding packets

traffic in detection unit 0. TCP traffic in this unit is discrete and unstable since the simulation just starts and TCP traffic is slowly increasing. (c) in Fig. 12 shows the TCP traffic in detection unit 723 and TCP traffic in this unit is also discrete and unstable. Because TCP traffic in these detection units is discrete and unstable, two-step cluster analysis divides them into the same cluster as the detection unit subjected to LDoS attacks. In "Results analysis" section, the detection performance in NS2 is compared with that in test-bed, which proves the method is effective in detecting LDoS attacks.

Experiments in test-bed network environment

Experimental network environment

In addition to the simulation carried out in NS2 platform, experiments are also conducted in test-bed network environment. Figure 13 shows the test-bed network environment topology which is a deformed dumbbell-like shape. It consists of two routers and five hosts, in which three hosts simulate a legitimate user sending TCP traffic and UDP traffic, one host simulates the LDoS attacker, and one host simulates the server that is configured as a victim. Traffic sent by legitimate users and attackers reaches the server via Router1 and Router2 whose link is the bottleneck link, and the bandwidth is 10Mbps.

Table 2 shows the detailed device configuration in test-bed network environment. The operating system of PCs is windows 7 and python socket is used to generate network traffic. Legitimate users are the hosts that send normal network traffic to the server. For each legitimate user, there are multiple python socket threads sending network traffic.

Table 3 Parameters setting in test-bed network environment

Experiments	Number of TCP connections
1	5
2	10
3	15

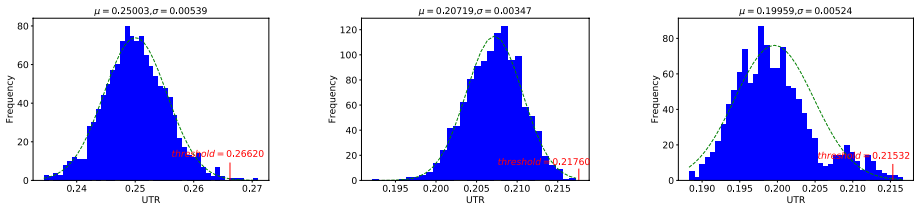
Table 4 Parameters setting of LDoS attacks in test-bed network environment

	T(s)	L(s)	R(Mbps)	Attack time period (s)
Training data	0	0	0	0
Testing data	1	0.2	15	660–750
	1	0.3	15	810–900
	2	0.2	15	960–1050
	2	0.3	15	1110–1200
	1	0.2	25	1260–1350
	1	0.3	25	1410–1500
	2	0.2	25	1560–1650
	2	0.3	25	1710–1800

The number of TCP socket threads in a host is the number of TCP connections from that host to the server. If a host has five python socket threads sending TCP traffic to the server, the host and server have five TCP connections. To comprehensively evaluate the proposed approach, three groups of experiments with a different number of TCP connections are carried out. Table 3 shows the experimental parameters. The difference between the three groups of experiments is that the number of TCP connections on each legitimate user is different. In experiment 1, the number of TCP connections for a legitimate user to the server is five and the number in experiments 2 and 3 is ten and fifteen, respectively. Table 4 shows the detail parameters of LDoS attacks in each group of experiments. In each group of experiments, the training data contains 3600 s data that is the network traffic without LDoS attacks and the testing data contains 1800 s data that is the network traffic suffered LDoS attacks in which eight LDoS attacks occurred and each attack lasted 90 s.

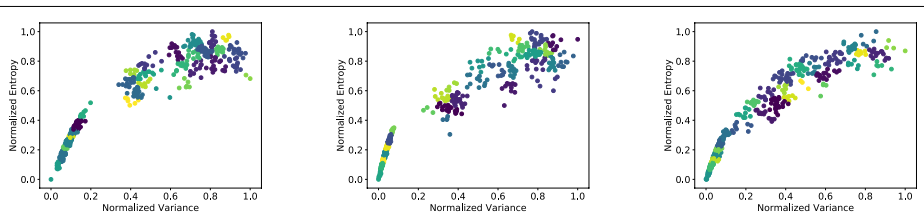
Results analysis

In this subsection, the experimental results are presented and discussed. Network traffic is collected on the server for LDoS attacks detection. Figure 14 shows the UTR frequency distribution of training data in three groups of experiments. As the figures show, the UTR frequency distribution conforms to a normal distribution. In three groups of experiments, the mean values of UTR (μ) are about 0.25003, 0.20719 and 0.19959, and the standard deviation values of UTR (σ) are about 0.00539, 0.00347 and 0.00524. According to Eq. 8, the thresholds in three groups of experiments are about 0.26620, 0.21760 and 0.21532. Figure 15 shows the results of BIRCH, with thirty sub-clusters obtained for each experiment. In three groups of experiments, data points of TCP traffic with similar characteristics are in the same cluster as these data points



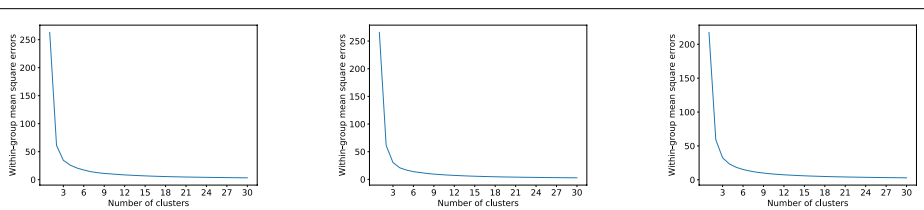
a Experiment 1 **b** Experiment 2 **c** Experiment 3

Fig. 14 UTR frequency distribution in training data



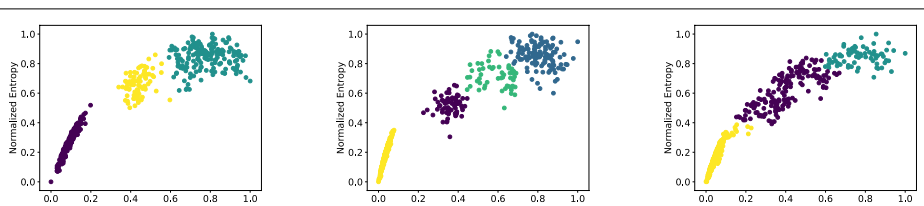
a Experiment 1 **b** Experiment 2 **c** Experiment 3

Fig. 15 Results of BIRCH



a Experiment 1 **b** Experiment 2 **c** Experiment 3

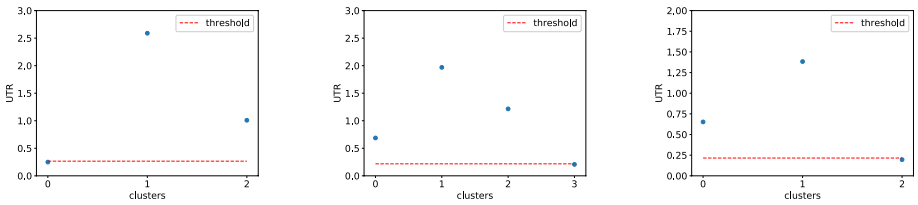
Fig. 16 The change of WMSE with the number of clusters



a Experiment 1 **b** Experiment 2 **c** Experiment 3

Fig. 17 Results of two-step cluster analysis

have closer distance in the space. Figure 16 shows the change of WMSE with the number of clusters. In three groups of experiments, the value of WMSE decreases with the number of clusters increase. According to the "elbow point", the optimal number M of clusters in the three groups of experiments are determined to be three, four and three, respectively. Figure 17 shows the results of two-step cluster analysis. In three groups of experiments, three, four and three clusters are obtained through two-step



a Experiment 1 **b** Experiment 2 **c** Experiment 3

Fig. 18 Results of UTR analysis

cluster analysis. Highly similar data points are divided into the same cluster since their distances are closer. Figure 18 shows the results of UTR analysis. If the UTR of a cluster is higher than the threshold represented by the red dashed line, it indicates that the cluster has suffered LDoS attacks. In experiment 1, clusters 1 and 2 are detected as suffered LDoS attacks; in experiment 2, clusters 0, 1 and 2 are detected as suffered LDoS attacks; in experiment 3, clusters 0 and 1 are detected as suffered LDoS attacks. In experiment 1, eight detection units are false positive reported and no detection unit is false negative reported, which is the same as the detection results in experiment 2. In experiment 3, eight detection units are false positive reported and eleven detection units are false negative reported.

For evaluating the experimental results, six evaluation metrics are computed, which is false negative rate (FNR), false positive rate (FPR), f1-score, accuracy, precision and recall measures.

- TP: if a detection unit with LDoS attacks is detected to have suffered LDoS attacks, it is accepted as TP.
- FP: if a detection unit without LDoS attacks is detected to have suffered LDoS attacks, it is accepted as FP.
- TN: if a detection unit without LDoS attacks is detected to have not suffered LDoS attacks, it is accepted as TN.
- FN: if a detection unit with LDoS attacks is detected to have not suffered LDoS attacks, it is accepted as FN.

$$FNR = \frac{FN}{TP + FP + TN + FN} \tag{9}$$

$$FPR = \frac{FP}{TP + FP + TN + FN} \tag{10}$$

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

Table 5 Comparison of different experimental platforms

Platform	Precision	Recall	F1-score	Accuracy	FPR	FNR
NS2 simulation platform	0.9816	1.0000	0.9907	0.9925	0.0075	0.0000
Test-bed network environment	0.9673	0.9847	0.9759	0.9806	0.0133	0.0061

Table 6 Comparison with different methods

Method	FPR	FNR
Multifractal	0.10	0.09
KPCA	0.02	0.008
Wavelet feature extraction	0.005	0.009
Adaptive *v*–SVR	0.0063	0.0062
Our method	0.0133	0.0061

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{13}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{14}$$

Table 5 shows the comparison of detection performance in the different experimental platforms. The detection results in NS2 are more accurate than those in test-bed, and the six evaluation metrics on both platforms are good. The comprehensive analysis of six evaluation metrics proves that the detection method can detect LDoS attacks with good performance.

To further and more comprehensively evaluate the performance and efficiency of the proposed method, the experimental results are also compared with that of other detection methods. Table 6 shows a comparison of different detection methods. Our method has an FPR of 0.0133 and an FNR of 0.0061. The FPR of our method is higher than that of the wavelet feature extraction method [37] and the adaptive *v*-SVR method [40], but lower than that of the multifractal method [19] and the KPCA method [30]. Although the FPR of our method is not the lowest of these five methods, the FNR of our method is the lowest, which proves that our detection method achieves good performance.

Experiments in public dataset WIDE

Since the experiments carried out in NS2 are based on simulation and the test-bed experiments are also performed under experimental conditions, the effectiveness of the proposed method needs to further verify. Since there is no suitable public database containing LDoS attack data, the detection method is evaluated by testing false positive rate from another perspective. WIDE dataset maintained by the MAWI Working Group is a traffic data repository and researchers can use it to evaluate their anomaly

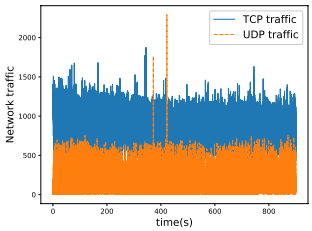


Fig. 19 Network traffic in WIDE dataset

detection methods. Network traffic used in the experiment is collected at the sample-point-G which traces from the main IX link of WIDE to DIX-IE. Four days of 2018.01.16, 2018.01.23, 2018.02.06, and 2018.02.16 are selected, and a total of 3600 s of network data traffic per day from 14:00 to 14:15 is used as training data. Two days of 2018.02.19 and 2018.03.10 are selected, and a total of 1800 s of network data traffic per day from 14:00 to 14:15 is used as testing data. Figure 19 shows the network traffic in 2018.01.16 from 14:00 to 14:15. The ratio of UDP traffic to TCP traffic is roughly maintained at a constant level. Although the WIDE dataset does not contain any LDoS attack data, it can still be used as a supplement to NS2 experiments and test-bed experiments for further verifying the detection method's effectiveness.

Figure 20 shows the UTR frequency distribution of the training data. The mean value of UTR (μ) is about 0.11235 and the standard deviation value (σ) of UTR is about 0.04031. According to Eq. 8, the threshold is about 0.23327. Figure 21 shows the results of BIRCH. A total of thirty sub-clusters are obtained through BIRCH and data points with closer distance are in the same cluster. Figure 22 shows the change of WMSE with

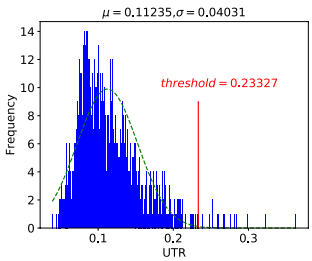


Fig. 20 UTR frequency distribution in training data

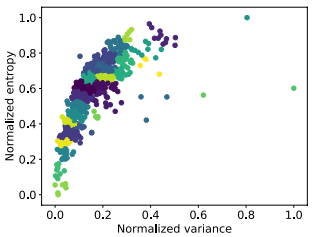


Fig. 21 Results of BIRCH

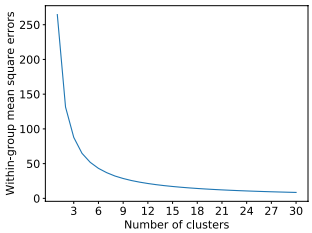


Fig. 22 The change of WMSE with the number of clusters

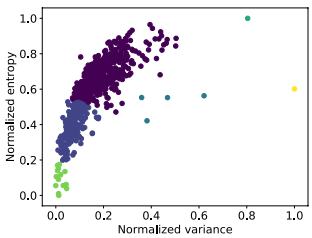


Fig. 23 Results of two-step cluster analysis

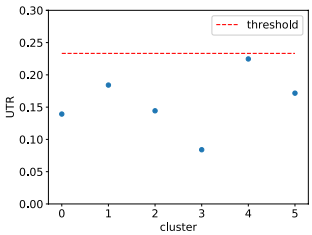


Fig. 24 Results of UTR analysis

the number of clusters. According to the "elbow point", the optimal number M of clusters is determined to be six. Figure 23 shows the results of the aggregation hierarchical clustering algorithm that are also the results of two-step cluster analysis. As the figure shows, data points in space are not very dispersed since the characteristics of TCP traffic without LDoS attacks are similar. Figure 24 shows the results of UTR analysis, in which the six clusters' UTRs are all larger than the threshold, that is, no cluster is detected suffered LDoS attacks, which is consistent with the true situation. The experimental results without detecting LDoS attacks show that the false positive rate of the detection method is low, which proves the good performance of the method.

Conclusion and future work

An LDoS attack detection method with good detection performance is proposed in this paper. Two-step cluster analysis is used to divide traffic with similar characteristics into the same cluster. Network traffic that has been subjected to LDoS attacks has similar characteristics and is therefore aggregated into the same cluster. When an LDoS attack occurs, the ratio of UDP traffic to TCP traffic changes, as LDoS attacks usually use the UDP protocol. Therefore, UTR analysis is then used to determine which clusters have suffered LDoS attacks.

NS2 simulation platform, test-bed network environment, and WIDE dataset are used to evaluate the effectiveness of the proposed detection method. However, more experiments are needed to further evaluate the detection method in the future, such as SDN based experiments. The proposed method can only detect UDP based LDoS attack vector, and the detection method against TCP based LDoS attack vector is worthy of being studied in the future.

Acknowledgements
The authors would like to thank the editors for handing the manuscript. The authors also thank the referees for their helpful comments in improving the contents of this paper.

Authors' contributions
DT, RD and LT were involved in proposing the system idea, performing the experiments, and writing the manuscript; XL helped to revise this paper. All authors read and approved the final manuscript.

Funding
This work was supported by National Natural Science Foundation of China (61772189), and Hunan Provincial Natural Science Foundation of China (2019JJ40037).

Availability of data and materials
Please contact authors for data requests.

Competing interests
The authors declare that they have no competing interests.

Author details
[1] College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. [2] School of Computer Science and Engineering, Hunan University of Science and Technology, Xiangtan, China.

Received: 17 May 2019 Accepted: 16 January 2020
Published online: 06 February 2020

References

1. David J, Thomas C (2019) Efficient ddos flood attack detection using dynamic thresholding on flow-based network traffic. Comput Secur 82:284–295
2. Aiello M, Cambiaso E, Scaglione S, Papaleo G (2013) A similarity based approach for application dos attacks detection. In: 2013 IEEE symposium on computers and communications (ISCC), IEEE, pp 000430–000435
3. Hodo E, Bellekens X, Hamilton A, Dubouilh P-L, Iorkyase E, Tachtatzis C, Atkinson R (2016) Threat analysis of iot networks using artificial neural network intrusion detection system. In: 2016 international symposium on networks, computers and communications (ISNCC), IEEE, pp 1–6
4. Baig ZA, Sanguanpong S, Firdous SN, Nguyen TG, So-In C et al (2020) Averaged dependence estimators for dos attack detection in iot networks. Future Gener Comput Syst 102:198–209
5. Dridi L, Zhani MF (2016) Sdn-guard: Dos attacks mitigation in sdn networks. In: 2016 5th IEEE international conference on cloud networking (Cloudnet), IEEE, pp 212–217
6. Imran M, Durad MH, Khan FA, Derhab A (2019) Reducing the effects of dos attacks in software defined networks using parallel flow installation. Hum Cent Comput Inf Sci 9(1):16
7. Masdari M, Jalali M (2016) A survey and taxonomy of dos attacks in cloud computing. Secur Commun Netw 9(16):3724–3751
8. Gupta B, Badve OP (2017) Taxonomy of dos and ddos attacks and desirable defense mechanism in a cloud computing environment. Neural Comput Appl 28(12):3655–3682
9. Zhang H, Qi Y, Wu J, Fu L, He L (2016) Dos attack energy management against remote state estimation. IEEE Trans Control Netw Syst 5(1):383–394
10. Luo J, Yang X, Wang J, Xu J, Sun J, Long K (2014) On a mathematical model for low-rate shrew ddos. IEEE Trans Inf Forensics Secur 9(7):1069–1083

11. Cambiaso E, Papaleo G, Chiola G, Aiello M (2013) Slow dos attacks: definition and categorisation. Int J Trust Manage Comput Commun 1(3–4):300–319
12. Mongelli M, Aiello M, Cambiaso E, Papaleo G (2015) Detection of dos attacks through fourier transform and mutual information. In: 2015 IEEE international conference on communications (ICC), IEEE, pp 7204–7209
13. Yue M, Wang M, Wu Z (2019) Low-high burst: a double potency varying-rtt based full-buffer shrew attack model. IEEE Trans Dependable Secure Comput. https://doi.org/10.1109/TDSC.2019.2948167
14. Maciá-Fernández G, Rodríguez-Gómez RA, Díaz-Verdejo JE (2010) Defense techniques for low-rate dos attacks against application servers. Comput Netw 54(15):2711–2727
15. Cambiaso E, Papaleo G, Chiola G, Aiello M (2015) Designing and modeling the slow next dos attack. In: Computational intelligence in security for information systems conference, Springer, pp 249–259
16. Cambiaso E, Papaleo G, Aiello M (2017) Slowcomm: design, development and performance evaluation of a new slow dos attack. J Inf Secur Appl 35:23–31
17. Cui T, Andrew LL, Zukerman M, Tan L (2006) Improving the fairness of fast tcp to new flows. IEEE Commun Lett 10(5):414–416
18. Chen Z, Yeo CK, Lee BS, Lau CT (2018) Power spectrum entropy based detection and mitigation of low-rate dos attacks. Comput Netw 136:80–94
19. Wu Z, Zhang L, Yue M (2015) Low-rate dos attacks detection based on network multifractal. IEEE Trans Dependable Secure Comput 13(5):559–567
20. Zhan S, Tang D, Man J, Dai R, Wang X (2020) Low-rate dos attacks detection based on maf-adm. Sensors 20(1):189
21. Tang D, Dai R, Tang L, Zhan S, Man J (2018) Low-rate dos attack detection based on two-step cluster analysis. In: International conference on information and communications security, Springer, pp 92–104
22. Fontugne R, Borgnat P, Abry P, Fukuda K (2010) Mawilab: combining diverse anomaly detectors for automated anomaly labeling and performance benchmarking. In: Proceedings of the 6th international conference, ACM, p 8
23. Şimşek M, Şentürk A (2018) Fast and lightweight detection and filtering method for low-rate tcp targeted distributed denial of service (lddos) attacks. Int J Commun Syst 31(18):3823
24. Cambiaso E, Chiola G, Aiello M (2019) Introducing the slowdrop attack. Comput Netw 150:234–249
25. Yue M, Wu Z, Wang J (2019) Detecting ldos attack bursts based on queue distribution. IET Inf Secur 13(3):285–292
26. Wu Z, Pan Q, Yue M, Liu L (2019) Sequence alignment detection of tcp-targeted synchronous low-rate dos attacks. Comput Netw 152:64–77
27. Yue M, Liu L, Wu Z, Wang M (2018) Identifying ldos attack traffic based on wavelet energy spectrum and combined neural network. Int J Commun Syst 31(2):3449
28. Sun H, Lui JC, Yau DK (2006) Distributed mechanism in detecting and defending against the low-rate tcp attack. Comput Netw 50(13):2312–2330
29. Cao Y, Han L, Zhao X, Pan X (2019) Accflow: Defending against the low-rate tcp dos attack in wireless sensor networks. arXiv preprint arXiv:1903.06394
30. Zhang X, Wu Z, Chen J, Yue M (2017) An adaptive kpca approach for detecting ldos attack. Int J Commun Syst 30(4):2993
31. Wu X, Tang D, Tang L, Man J, Zhan S, Liu Q (2018) A low-rate dos attack detection method based on hilbert spectrum and correlation. In: 2018 IEEE smartworld, ubiquitous intelligence & computing, advanced & trusted computing, scalable computing & communications, cloud & big data computing, internet of people and smart city innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), IEEE, pp 1358–1363
32. Agrawal N, Tapaswi S (2018) Low rate cloud ddos attack defense method based on power spectral density analysis. Inf Process Lett 138:44–50
33. Chen H, Meng C, Shan Z, Fu Z, Bhargava BK (2019) A novel low-rate denial of service attack detection approach in zigbee wireless sensor network by combining hilbert-huang transformation and trust evaluation. IEEE Access 7:32853–32866
34. Yan Y, Tang D, Zhan S, Dai R, Chen J, Zhu N (2019) Low-rate dos attack detection based on improved logistic regression. In: 2019 IEEE 21st international conference on high performance computing and communications; IEEE 17th international conference on smart city; IEEE 5th international conference on data science and systems (HPCC/SmartCity/DSS), IEEE, pp 468–476
35. Zhang D, Tang D, Tang L, Dai R, Chen J, Zhu N (2019) Pca-svm-based approach of detecting low-rate dos attack. In: 2019 IEEE 21st international conference on high performance computing and communications; IEEE 17th international conference on smart city; IEEE 5th international conference on data science and systems (HPCC/SmartCity/DSS), IEEE, pp 1163–1170
36. Tang D, Tang L, Dai R, Chen J, Li X, Rodrigues JJ (2020) Mf-adaboost: Ldos attack detection based on multi-features and improved adaboost. Future Gener Comput Syst. https://doi.org/10.1016/j.future.2019.12.034
37. He Y-X, Cao Q, Liu T, Han Y, Xiong Q (2009) A low-rate dos detection method based on feature extraction using wavelet transform. J Softw 20(4):930–941
38. Zhang T, Ramakrishnan R, Livny M (1996) Birch: an efficient data clustering method for very large databases. In: ACM sigmod record, vol 25. ACM, pp 103–114
39. Chen Y, Hwang K, Kwok Y-K (2005) Filtering of shrew ddos attacks in frequency domain. In: The IEEE conference on local computer networks 30th anniversary (LCN'05) L, IEEE, p 8
40. Zhang X, Wu Z, Zhang J, Chen J (2018) An adaptive network traffic prediction approach for ldos attacks detection. Int J Commun Syst 31(5):3505

Publisher's Note