Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2018) 26: 1796 – 1811
© TÜBİTAK
doi:10.3906/elk-1712-3

Threshold-based distributed DDoS attack detection in ISP networks

Karanbir SINGH[1,*], **Kanwalvir Singh DHINDSA**[2], **Bharat BHUSHAN**[3]
[1]IKG Punjab Technical University, Kapurthala, Punjab, India
[2]Department of Computer Science & Engineering, Baba Banda Singh Bahadur Engineering College, Fatehgarh Sahib, Punjab, India
[3]Department of Computer Science, Guru Nanak Khalsa College, Yamunanagar, Haryana, India

Received: 01.12.2017 • **Accepted/Published Online:** 03.04.2018 • **Final Version:** 27.07.2018

Abstract: The purpose of this paper is to propose a more efficient and accurate distributed denial of service (DDoS) attack detection mechanism that detects DDoS attacks by monitoring the incoming traffic on the edge routers of ISP networks. It can be implemented as a module or agent function on the machine that is responsible for processing router traffic. The detection algorithm works by monitoring the traffic passing through the edge routers and identifying the occurrence of DDoS attacks or flash events. The algorithm calculates different values like the normalized router entropy, packet rate, and entropy rate and compares them against the preidentified threshold values to detect the happening of a DDoS attack or flash event. The threshold values used in the algorithm are evaluated offline by taking the sample attack and the legitimate traffic flows. The proposed detection mechanism can be implemented on the edge routers of the ISP networks. ISPs are selected for the deployment of attack detection because the customer networks are directly connected with them. The effectiveness of the algorithms can be validated mathematically using a sample test bed containing realistic internet topology. The results clearly indicate that the proposed detection mechanism does effective detection with a high detection rate and fewer false positives.

Key words: DDoS attacks, DDoS defense, entropy, threshold, internet service providers

1. Introduction

Denial of service attacks can prevent legitimate users from using a particular network, service, or resource. The distributed denial of service (DDoS) attack works by flooding a particular network or web server with a large number of attack packets. The DDoS attack is a more organized large-scale attack on a particular web server or network and can make it unavailable to its intended users [1,2]. This attack can be performed indirectly with the help of many intermediate compromised machines on the Internet. Recently there is an additional type of Internet traffic, together with DDoS attacks, that is getting popular among people focusing on security research: a flash event. A flash event is similar to a DDoS attack wherein thousands of legitimate users try to access a particular network resource such as a web server simultaneously [3]. In some situations, DDoS attacks and flash events occur simultaneously and it becomes difficult to distinguish between the two.

The proposed attack detection method is an extension of the work on a collaborative agent-based model for a distributed defense model against DDoS attacks in ISP networks proposed by us in [4]. The attack detection method proposed here uses entropy and threshold values to identify between legitimate traffic, attack traffic, and flash traffic. The entropy of flow remains stable in the absence of a DDoS attack or a flash event, but

[*]Correspondence: karan_nehra@yahoo.co.in

when a DDoS attack or flash event happens, the flow of entropy drops because the attack flow or flash traffic dominates the traffic on the router. The defense system can take advantage of this feature to raise the alarm as soon as the flow entropy decreases significantly. Attack detection is normally the initial step of any DDoS defense mechanism. The proposed threshold-based attack detection method can detect DDoS attacks or flash events on the edge routers of the source networks. The main advantage of the detection method is that it can be implemented as a part of an agent and can be used for the characterization of incoming traffic. An agent is a software program, which can work on behalf of Internet service providers and can monitor the traffic heading towards a particular destination [5]. The agent can be put on the edge router of ISP networks. The edge routers of ISP networks are chosen for the deployment of agents because the early detection of attack traffic will help to reduce collateral damage. The attack detection mechanism analyzes the incoming traffic on the edge routers and identifies suspicious traffic. The identified suspicious traffic can later be confirmed as being a DDoS attack or not. The traffic that is confirmed as a DDoS attack can be dropped so as to avoid it reaching its target. The proposed attack detection algorithm is the first, to the best of the authors' knowledge, that can be used as an agent-based distributed attack detection and mitigation system. This method can help ISPs secure their customer networks and provide them with uninterrupted service.

2. Related work and background

Much research on DDoS defense mechanisms has been done in the literature and a variety of defense mechanisms have been developed. Entropy has been widely used as a metric for measuring the randomness of distributions and DDoS attack detection. Initially, some methods from the literature that use entropy as a feature for attack detection are summarized and how they help in attack detection will be discussed.

2.1. Related work

Liu et al. [6] suggested a flow and reputation-based defense system that can be put on the border router of an organization network. The authors extracted the essential features of the latest DDoS attack traffic and identified the targets that are under attack, then collected the credit for each flow from microlevel views. The probabilistic drop technique can be used to filter the DDoS traffic.

Yu and Zhou [7] suggested entropy-based methods for collaborative detection of the DDoS attacks on community networks. The authors used the entropy feature to measure the randomness of flow in a given router. Qin et al. [8] identified a new approach for entropy-based DDoS attack detection. This method is divided into two parts. The first part builds the normal models using the clustering algorithm after having selected some representative features of DDoS, and the second part identifies abnormalities in the monitored traffic based on the normal traffic models.

Lee and Xiang [9] suggested the use of different information theoretic measures such as entropy, conditional entropy, relative conditional entropy, information gain, and information cost for anomaly detection. These measures can be used to describe the characteristics of an audit dataset and suggest the appropriate anomaly detection models. Bhandari et al. [10] proposed a destination address entropy-based detection and trace back mechanism against DDoS attacks. It calculates entropy on the basis of the destination address of the flows. The six-sigma method is used for computing the threshold values. The suggested technique does not enforce an extra burden on the routers. Additionally, the trace back will not put any marking overhead on the routers. The technique can be deployed as a separate module on the different routers of the network. However, this method is unable to identify and trace back DDoS attacks that are isotropic in nature. The concept of generalized entropy described in [11] is used to discover the connection between the port numbers and the source/destination IP addresses. It was identified that there is a high correlation between the source/destination port numbers in legitimate traffic, but no strong correlation among the source/destination IP addresses and the ports in the attack traffic was found. The authors used several real traffic traces and NetFlow data to certify their detection method.

Nychis et al. [12] used IP packet header and behavioral features to identify anomalies in traffic flows. They observed that port and address distributions are highly correlated with each other. This correlation happens due to the nature of traffic patterns. The anomalies detected by port and address distributions overlap significantly. These observations have significant implications for entropy-based analysis. The behavior-based anomaly detection method identified in [13] was used to detect traffic anomalies by comparing the current traffic features with baseline behavior. It uses maximum entropy and relative entropy to identify traffic anomalies. Behal and Kumar [14] proposed a novel set of information theory metrics called entropy and divergence metrics for detecting DDoS attacks and flash events.

David and Thomas [15] proposed an improvement in the detection of DDoS attacks based on a fast entropy method using flow-based analysis. It is an adaptive threshold-based algorithm that uses both network activities and user behavior. The network traffic is observed and the fast entropy of requests per flow is calculated. A DDoS attack is identified when the difference between the entropy of flow count at each instant and mean value of entropy in that time interval is greater than the threshold value. The threshold value is updated adaptively based on the traffic pattern condition to improve detection accuracy.

2.2. Preliminaries of information theory

There are many definitions of flow available in the literature [16,17] but we follow the IP Flow Information Export (IPFIX) working group of the IETF [18]. As per IPFIX, a flow can be defined as a set of IP packets passing through a monitoring point in a network during a particular time interval. All the packets belonging to a specific flow have a set of properties. The common sets of properties are source and destination address, source and destination port, and IP protocol.

Entropy [19] is an information theoretic concept that is used to compute the randomness of the packets in a flow. It can be used in different ways for the detection of anomalies in traffic features. It is a useful tool for inspecting the similarity and distribution of a given flow at a router for a particular time interval. When an attack occurs in the observation window, the entropy of that flow will drop noticeably and we can identify that situation as a DDoS attack. The value of sample entropy lies in the range [0, $\log_n$]. The entropy shows its minimum value of 0 when all the flows (IP address or port) are the same and its maximum value of $\log_n$ when all the flows are different. The maximum value of n could be 2^{32} if one needs to measure the entropy of unique source and destination IP addresses [20]. Similarly, there could be a maximum n number of ports, and the maximum value of ports for each protocol could be 65,535. The various values used in the algorithm are calculated as per the following equations.

2.2.1. Flow entropy

The flow entropy of a random variable X with possible values $\{x_1, x_2, x_3 x_n\}$ can be calculated as:

$$F\,(xi) = -P(xi) \times Log_2(P(xi)), \tag{1}$$

where $P(xi)$ is the probability of packets belonging to flow (xi).

2.2.2. Router entropy

The router entropy at a particular flow can be calculated by combining the entropies of distinct flows (source IP, dest IP) during time window Δt:

$$H = \sum_{1}^{n} F(xi), \tag{2}$$

where n is the total number of flows.

2.2.3. Normalized router entropy

The normalized router entropy (NRE) calculates the overall probability distribution of the captured flow for time window Δt. It can be calculated as:

$$NRE = \frac{H}{Log_2 n}, \tag{3}$$

where H is the sum of router entropies during Δt, and n is the total number of distinct flows at the router.

2.2.4. Packet rate

The packet rate belongs to an average number of packets sent by the source to the destination. The packet rate for a particular flow in a given time window can be calculated as:

$$Prt(xi) = \frac{\text{Total number of packets belonging to}\, flow\,(xi)}{n}, \tag{4}$$

where n is the total number of captured flows in time window Δt.

2.2.5. Entropy rate

The entropy rate is used to measure the growth rate of the entropy of any random process. For a series of n random variables, the entropy rate E_R of a stochastic process (xi) is calculated as:

$$E_R(x) = \frac{\sum_{1}^{n} H(xi)}{n}, \tag{5}$$

where n = total number of captured flows in the time window Δt

The flow entropy can be calculated by collecting sample packets over a specific time window passing through an edge router. The normalized router entropy, calculated using flow and router entropy, plays a vital role in capturing any suspicious activity on the edge router. The packet rate helps in identification of suspicious flows when there is more than one flow passing through the edge router. Finally, the entropy rate gives us a confirmation about whether a DDoS attack is happening or not.

3. Proposed attack detection algorithm

The detection algorithm can be deployed on the edge routers of the stub networks. The algorithm running on edge routers will monitor the flow of incoming traffic using flow entropy as a metric. The flow entropy remains stable in the absence of an attack, but when an attack happens, the entropy drops radically because of the domination of a flow on the router. Some assumptions are made in advance and directly used in the design of algorithm. Initially, the threshold values to be used in the detection algorithm are identified. The values can be determined by running extensive simulations on sample flows and monitoring the results in the presence and absence of attacks. The values should be dynamic, and they will be changed as Internet traffic changes so that detection accuracy can be improved. The working of the attack detection algorithm along with the algorithms used to calculate threshold values is described below.

3.1. Assumptions

The following assumptions are made during the design of the threshold and the attack detection algorithm:

i. The attack packets enter the network via a minimum of two routers (edge routers), and their traffic will merge at the gateway router.

ii. The threshold values calculated using Algorithms 1, 2, and 3 are directly used in the detection algorithm.

iii. The values of the sampling period (T) and the time window size (Δt) are taken as 10 s and 1 s, respectively.

iv. The cooperation between the edge and gateway routers is necessary to perform attack detection.

v. The detection algorithm can be implemented as a part of the defense agents on the edge and gateway routers.

3.2. Threshold identification

The threshold values to be used in the detection algorithms need to be identified first. The threshold value T1 is used to detect a suspicious flow on the edge router. It can be calculated by constructing a sample flow of legitimate traffic and calculating the normalized router entropy. The value of normalized router entropy is recorded for the specific time window. The same process is also repeated for attack traffic. There is a difference in the values of normalized router entropy for legitimate and attack traffic.

Algorithm 1 Identification of threshold value T1.

```
Step 1: Initialize sampling time T and time window $\Delta$t
Step 2: Construct sample flows of legitimate traffic after every $\Delta$t s on edge router
Step 3: Calculate router entropy and normalized entropy for time window $\Delta$t s
Step 4: Repeat Steps 2 and 3 for attack traffic also
Step 5: Let En(xi) be the normalized entropy for attack traffic and Ea(xi) for attack traffic
Step 6: Record the values of En(xi) and Ea(xi) for different time intervals
Step 7: Create the range of values for attack as well as legitimate traffic
        ■ Let En (a1, a2) and Ea (b1, b2) be the range of values for legitimate and attack traffic
        Where a1 and a2 are the lowest and highest values of the normalized router entropy for legitimate
        traffic, b1 and b2 are the lowest and highest values of normalized router entropy for attack traffic
Step 8: Choose appropriate threshold T1 so that it is sufficiently less than a1 and greater than b2
```

Step 7 of Algorithm 1 shows that a range of highest and lowest values is created for legitimate and attack traffic. We choose an appropriate value of T1, which is sufficiently larger than the largest value found during the attack traffic and smaller than the smallest value found during the legitimate traffic.

Threshold T1 identifies only the happening of suspicious activity but not the suspicious flow. The threshold value T2 is used to identify the suspicious flow among the flow passing through the router. It can be calculated by calculating the average packet rate of legitimate and attack traffic during specific time windows. A range is created by taking average packet rate values for all legitimate and attack traffic. Step 7 of Algorithm 2 helps us decide the appropriate value of threshold T2.

Algorithm 2 Identification of threshold value T2.

```
Step 1: Initialize sampling time T and time window $\Delta$t
Step 2: Construct sample flows of normal traffic after every $\Delta$t s on one edge router
Step 3: For each flow, calculate packet rate; let it be prt1, prt2,.... prtn
Step 4: Calculate average packet rate; let it be $avg1$ $avg1 = \frac{prt1+prt2+.....prtn}{\Delta t}$ Here $\Delta$t is time window
Step 5: Record average packet rate (avg1) values for different time intervals; let this be range1
Step 6: Repeat Steps 2, 3, 4, and 5 for attack traffic and let the average packet rate be avg2 and range be
        range2
Step 7: Choose appropriate threshold T2 so that it is sufficiently larger the largest value of range1 and
        smaller than the smallest value of range2
```

After the successful identification of suspicious flow using T2, the next step is to identify threshold value T3, which is used to confirm whether the suspicious flow is part of a flash event or an attack flow, i.e. a DDoS attack. T3 can be identified by calculating the entropy rate for the suspicious flow on the edge router. The downstream gateway router is also called to check the entropy rate for the same suspicious flow for other upstream edge routers. The difference of entropy rates on the edge and gateway routers is calculated and recorded for different time intervals. Threshold value T3 could be chosen such that it is sufficiently larger than the largest value of range3.

Algorithm 3 Identification of threshold value T3.

```
Step 1: Initialize sampling time T and time window $\Delta$t
Step 2: Construct sample flows of attack traffic after every $\Delta$t s on two edge Routers, say E1 and E2
Step 3: Calculate entropy rate for attack flow on E1 and E2 using time window $\Delta$t
Step 4: Calculate difference between entropy rate at E1 and E2; let it be Diff, $Diff = |E1 - E2|$
Step 5: Record different values of Diff using different time intervals; let this be range3
Step 6: Choose appropriate threshold T3 so that it is sufficiently larger than the largest value of range3
```

The threshold values play a vital role in the detection algorithm. After their identification, the next step is to design and discuss the attack detection algorithm. The main task of the detection algorithm is the identification of suspicious flow and later confirmation of whether the suspicious flow is a DDoS attack or a flash event on edge routers.

3.3. Detection algorithm

The detection algorithm works by executing the various steps mentioned in Algorithm 4. Initially, the various flows passing through the edge router are collected and their flow entropy, router entropy, and normalized router entropy are calculated for every Δt s. Then the value of normalized router entropy is compared against threshold T1 to identify the occurrence of a suspicious event. If the value becomes less than T1, it indicates that a suspicious flow is passing through the edge router. Step 9 will identify the suspicious flow by calculating and comparing the packet rate of all flows against T2. After the successful identification of suspicious flow, the next step is to identify whether the suspicious flow is an attack flow or just a flash event. Step 10 will calculate the entropy rate for the suspicious flow on the current edge router. Steps 11 and 12 will be performed on the gateway router where the entropy rate for the suspicious flow is calculated for the neighboring edge routers. Steps 13 and 14 will do the confirmation of the DDoS attack by comparing the difference between entropy rates against threshold T3.

Algorithm 4 DDoS attack detection.

```
 Step 1:  Initialize various parameters like time window $\Delta$t, sampling time T, detection thresholds T1, T2,
          and T3
 Step 2:  Collect traffic after every $\Delta$t s on edge router
 Step 3:  Construct flows from the collected traffic in $\Delta$t s
          | Flow1 | Flow2 | Flow3 | Flow4 | $\cdots$ | Flow$_n$ |
 Step 4:  For each flow, calculate flow entropy using Eq. (1)
 Step 5:  Calculate router entropy using Eq. (2)
 Step 6:  Calculate normalized router entropy (NRE) using Eq. (3)
 Step 7:  Compare NRE against threshold T1
 Step 8:  If the value of NRE is $\geq$T1 then treat all flows as legitimate flow; otherwise, if NRE is $<$T1, then
          there is a suspicious flow
 Step 9:  Identify the suspicious flow and follow the following steps: a) For each flow x(i), calculate packet rate
          using Eq. (4) b) If Prt(xi) $\leq$ T2, then the flow is not suspicious; otherwise, the flow is suspicious
Step 10:  To further confirm whether the suspicious flow (xi) is legitimate or an attack, calculate its entropy
          rate, E1, using Eq. (5), for the current edge router
Step 11:  Message the downstream gateway router to check the entropy rate of the suspected flow (xi) for
          other edge routers
Step 12:  Calculate the entropy rate of the suspected flow (xi) from other edge routers; let it be E2
Step 13:  Calculate the difference between E1 and E2 $Diff = |E1 - E2|$
Step 14:  Compare the value of $\mathit{Diff}$ with threshold T3, If $\mathit{Diff} >$ T3, then flow = legitimate flow (flash
          event) Else flow = attack flow (DDoS attack)
```

4. Experimentation

To evaluate the performance of the proposed detection mechanism, a sample test bed reflecting packet movements between an ISP and a server has been designed. The test bed consists of six source nodes, two destination nodes, four edge routers, two gateway routers. and two core routers. Source nodes A1, A2, B1, B2, C1, and C2 are connected with edge routers R1, R2, and R3 as shown in Figure 1. Nodes A1, B1, and C1 will act as attacking nodes in the presence of an attack. The two web servers S1 and S2 are used as destination nodes connected to edge router R4. The traffic originating from source networks will pass through at least one edge router before it reaches victim web server S1. The detection algorithm will be deployed on the edge routers where it monitors and extracts attack traffic from the legitimate traffic.

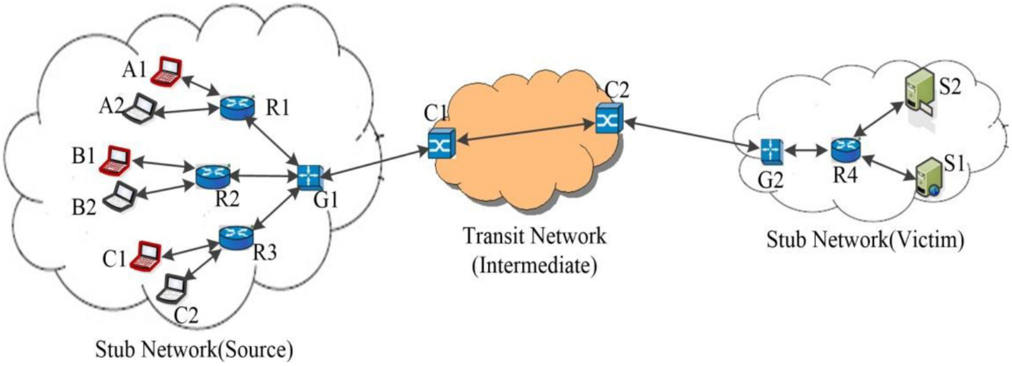

Figure 1. Topology reflecting transit–stub Internet model for attack detection algorithm.

Figure 1 shows the example topology chosen to prove the effectiveness of the detection algorithm. The attack traffic originating from nodes A1, B1, and C1 will pass through edge routers R1, R2, and R3 and merge at gateway router G1, so the defense agents holding the detection algorithm can be put on R1, R2, and R3. Gateway node G1 will do the confirmation of the DDoS attack.

4.1. Simulation environment

OMNeT++ [21] along with the INET framework (https://omnetpp.org/doc/inet/api-current/inet-manual-draft.pdf) is used to test the detection algorithm on the example topology given in Figure 1. ReaSE [22] is an extension of the INET framework, which permits us to produce realistic simulation topologies keeping multiple aspects of a real-world network such as nodes, traffic patterns, link bandwidth, and attack traffic. Table 1 depicts the various parameters used in the simulation to test the effectiveness of the proposed attack detection algorithm.

Figure 2 shows the screenshot of the example topology (as discussed in Figure 1) chosen to prove the effectiveness of the detection algorithm. The host1(A2), host2(B2), and host3(C2) will act as web clients and generate HTTP traffic towards the protected web server, server1(S1). Nodes DDoSZombie1(A1), DDoSZombie2(B1), and DDoSZombie3(C1) are used to flood web server server1 with TCP SYN attack packets. The detection algorithm can be put on R1, R2, and R3 as they pass most of the attack traffic. Gateway node G1 will do the confirmation of the DDoS attack. The simulation runs for 10 s and the traffic generated from different nodes is collected in scalar and vector files, which are later converted in the form of datasets.

Table 1. Basic simulation parameters.

Parameter	Value
Attacker nodes	3
Legitimate nodes	3
Web servers	2
Edge routers (source network)	3
Edge routers (victim network)	1
Gateway routers (source network)	1
Gateway routers (victim network)	1
Sampling time (T)	10 s
Time window (ΔT)	1 s

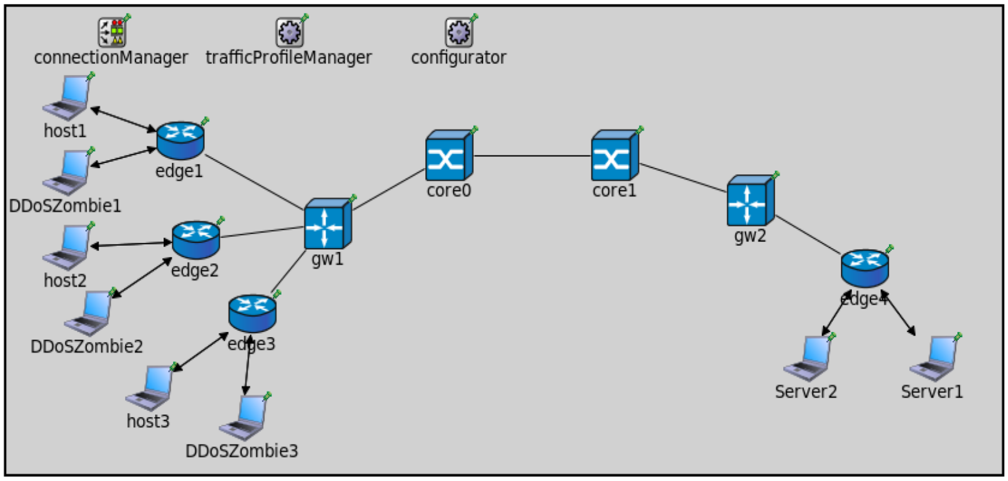

Figure 2. Screenshot of example topology for testing detection algorithm.

4.2. Deciding threshold values

The threshold values used in the detection algorithm are identified as per Algorithms 1, 2, and 3. These values are calculated by measuring flow entropy, router entropy, normalized router entropy, and packet rate using sample legitimate and attack traffic generated with the help of a simulation.

- Threshold value T1 = 0.90 is a numeric value used to identify the happening of a suspicious flow.

- Threshold value T2 = 100 is the number of packets used to identify the suspicious flows among the flows passing through router.

- Threshold value T3 = 0.10 is a numeric value used to confirm whether the identified suspicious flow is a DDoS attack or flash event.

4.3. Simulation scenarios

4.3.1. Scenario 1 (DDoS attacks)

In this scenario, we will perform a DDoS attack on destination S1. Nodes A1, B1, and C1 are used to flood the victim with TCP SYN attack packets. The attack will start at the 3 s and continued for the next 6 s. The simulation lasts for 10 s. The data generated by the OMNeT++ simulation are collected in the form of scalar and vector values. Table 2 shows the legitimate and attack flows (collected from .vec and .sca files, generated through simulation) and mathematically calculated values of normalized router entropy per flow for routers R1, R2, and R3.

Table 2. Router entropy of different edge routers during DDoS attacks.

Time (s)	Edge router R1			Edge router R2			Edge router R3		
	Flow 2 A2→S1	Flow 1 A1→S1	NRE	Flow 4 B2→S2	Flow 3 B1→S1	NRE	Flow 6 C2→S2	Flow 5 C1→S1	NRE
1	40	50	0.99	30	50	0.96	45	40	0.99
2	20	30	0.97	40	45	0.99	55	65	0.99
3	60	300	0.65	50	325	0.56	60	350	0.60
4	70	350	0.64	40	340	0.50	55	380	0.54
5	75	380	0.73	45	370	0.49	80	420	0.63
6	80	400	0.66	55	410	0.52	70	450	0.58
7	70	390	0.65	70	400	0.61	50	400	0.51
8	90	450	0.65	60	430	0.55	60	430	0.53
9	30	40	0.98	80	50	0.97	70	40	0.95
10	40	60	0.97	40	60	0.97	40	30	0.99

NRE: Normalized router entropy.

Since the number of flows per edge router is 2, the value of $\log_2 2$ will become 1. Hence, as per Eq. (3), the value of normalized router entropy for each flow will become equal to the value of router entropy. At 3 s the detection algorithm running on edge router R1 observes that the value of normalized router entropy will become less than threshold T1, so there is a suspicious flow, which causes the decrease in the normalized entropy value. Now, to identify the suspicious flow, we have to calculate the packet rate of each flow and compare it against threshold T2:

$$\text{Packet rate of flow 2 (A2} \rightarrow \text{S1)} = 60,$$

$$\text{Packet rate of flow 1 (A1} \rightarrow \text{S1)} = 300.$$

The packet rate at t = 3 of flow 1 becomes greater than threshold T2; hence, it will be treated as a suspicious flow. The flow A1→S1 (flow 1) is treated as a suspicious flow but to confirm further whether it is an attack flow, its entropy rate is calculated as:

$$\text{Entropy rate for the flow A1} \rightarrow \text{S1 (flow 1) at R1, Ea} = 0.22.$$

The defense agent at R1 asks gateway router G1 to compute the entropy rate for the flow heading towards protected destination S1 for other upstream edge routers, i.e. R2 and R3. The entropy rate for flow B1→S1 (flow 3) at router R2 and C1→S1 (flow 5) at router R3 is calculated as:

Entropy rate for flow B1→S1 (flow 3) at R2, Eb = 0.18,

Entropy rate for flow C1→S1 (flow 5) at R2, Ec = 0.19.

Now we calculate the difference between entropy rates for flow 1 at router R1 and flow 3 at router R2. This is calculated as:

$$\text{Diff} = \text{Ea} - \text{Eb} = |0.22 - 0.18| = 0.04$$

Similarly, the difference between entropy rates of flow 1 at router R1 and flow 5 at router R3 is calculated as:

$$\text{Diff} = \text{Ea} - \text{Ec} = |0.22 - 0.19| = 0.03.$$

Since the value of Diff in both cases is less than threshold T3, flow A1→S1 (flow 1) is confirmed as an attack flow (DDoS attack).

4.3.2. Scenario 2 (flash events)

DDoS attack detection becomes more difficult when a very similar situation called a flash event occurs. A flash event is a situation in which a large number of legitimate users simultaneously access a destination. Flash events are very similar to DDoS attacks in traffic characteristics. The discrimination between a DDoS attack and a flash event is very important for any DDoS defense mechanism. In this scenario, we assume all the nodes (as given in Figure 1) are legitimate users, but there is surge access (similar to a DDoS attack) during the 4th to 6th seconds at router R1. Table 3 shows the calculated values of router entropy per flow for the traced data after simulations for routers R1, R2, and R3.

Table 3. Router entropy of different edge routers during flash event.

Time (s)	Edge router R1			Edge router R2			Edge router R3		
	Flow A2→S1	Flow 1 A1→S1	NRE	Flow 4 B2→S2	Flow 3 B1→S1	NRE	Flow 6 C2→S2	Flow 5 C1→S1	NRE
1	40	50	0.99	30	50	0.95	45	40	0.99
2	20	30	0.97	40	45	1.00	55	65	0.99
3	30	50	0.96	70	40	0.95	20	30	0.97
4	70	350	0.64	55	70	0.99	70	40	0.95
5	75	380	0.73	70	55	0.99	30	50	0.96
6	80	400	0.66	60	45	0.99	40	60	0.97
7	70	40	0.95	30	50	0.95	30	40	0.98
8	30	40	0.97	70	40	0.95	40	45	0.99
9	30	40	0.98	80	50	0.96	70	40	0.95
10	40	60	0.97	40	60	0.97	40	30	0.99

NRE: Normalized router entropy.

The value of the normalized router entropy on router R3 remains greater than the threshold value T1 during the first seconds, but it will get decreased during the 4th to 6th seconds. At this stage, the detection algorithm detects flow A1→S1 (flow 1) as a suspicious flow and it will be further investigated whether it is an attack flow (DDoS attack) or legitimate flow (flash event). This can be done by calculating the entropy rate for flow A1→S1 (flow 1) on router R1 and correspondingly on routers R2 and R3:

$$\text{Entropy rate for flow A1}\rightarrow\text{S1 (flow 1) at R1, let Ea} = 0.22,$$

$$\text{Entropy rate for flow B1}\rightarrow\text{S1 (flow 3) at R2, let Eb} = 0.47,$$

$$\text{Entropy rate for flow C1}\rightarrow\text{S1 (flow 5) at R3, let Ec} = 0.53.$$

Now calculate the difference between entropy rates of flow one at router R1 and flow three at router R2. This is calculated as:

$$\text{Diff} = \text{Ea} - \text{Eb} = |0.22 - 0.47| = 0.25.$$

Similarly, the difference between entropy rates of flow one at router R1 and flow five at router R3 is calculated as:

$$\text{Diff} = \text{Ea} - \text{Ec} = |0.22 - 0.53| = 0.31.$$

Since the value of Diff in both cases is greater than threshold T3, flow A1→S1 (flow 1) is confirmed as a legitimate flow (flash event).

5. Performance analysis

An attack detection system can detect 100% of DDoS attacks only if it considers a good threshold value. The identification of a good threshold value to be used in the detection algorithm is one of the challenging tasks for the researchers. We have already identified the different threshold values to be used in the detection process. If the T1 value is 0.90, then it will be good for the identification of attacks, but it can produce false positive alerts as we increase it from 0.95 to 0.99. The effect of the threshold value on the performance of attack detection is discussed here.

5.1. Effect of detection algorithm on attack packets

Consider Table 2, showing the status of the attack and the legitimate packets from source nodes A1 and A2. The attack started at the 3 s from node A1 and continued for the next 6 s. The effect of the attack detection method and the threshold value on the attack and legitimate packets passed by router R1 is discussed here.

Figure 3 shows the effect of the detection algorithm and threshold value on the attack packets detected and dropped during the attack on router R1. Router R1 passes most of the attack packets in the absence of the detection algorithm. In the presence of the detection algorithm, the legitimate packets coming from node A2 will be allowed to pass, but the packets coming from node A1 are identified as attack packets (during the attack interval, i.e. 3rd to 8th second) as per the threshold value. An appropriate threshold value can thus identify and drop most of the attack packets.

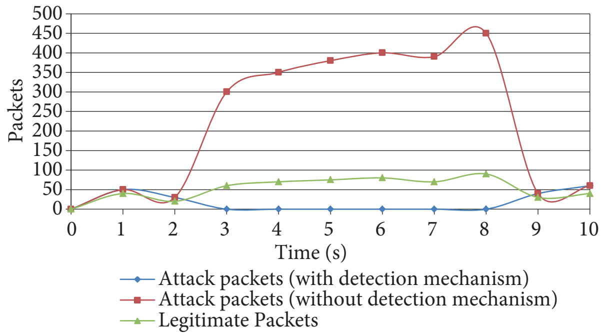

Figure 3. Status of packets during attack at router R1.

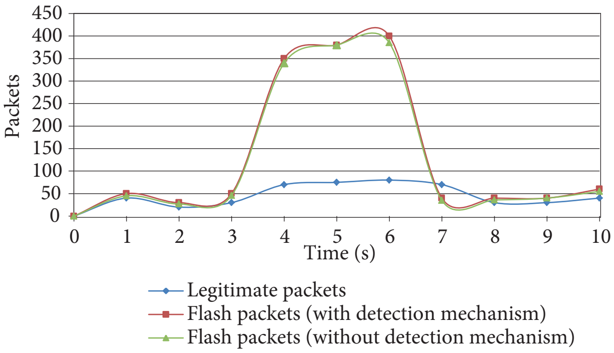

Figure 4. Status of packets during flash event at router R1.

5.2. Effect of detection algorithm on flash events

Table 3 illustrates the happening of a flash event on routers R1 and R2 for 3 s (i.e. during the 4th to 6th seconds). Figure 4 shows the effect of the detection algorithm on the number of packets passed by router R1 in the absence and presence of the detection algorithm.

In the absence of the detection algorithm, the router will pass all the packets, but when the detection algorithm is activated, it detects the happening of abnormal activity during the 4th to 6th seconds and identifies it as a flash event. The packets belonging to the flash event are also considered as normal packets and router R1 allows them to pass. Threshold value T3 helps in the accurate identification of the attack and flash packets.

5.3. Detection rate

The detection rate (Rd) is the ratio of the number of attack packets detected by the detection algorithm to the total number of attack packets generated from different sources. It is measured as:

$$Rd = \frac{D}{N}, \tag{6}$$

where D is the number of attack packets detected and N is the total number of attack packets generated.

Ideally, the value of the detection rate should be one. The detection rate highly depends on the threshold values. If the value is too low, then it will increase the false negatives, i.e. attack packets will be mistakenly considered as legitimate packets. If the value is too high, then it will increase the false negatives, i.e. the legitimate packets will be considered as attack packets. The purpose of the detection algorithm is to keep the detection rate as high as possible.

Figure 5 shows that the detection rate varies with the varying threshold value for edge router R1 as per Table 2 when an attack is happening from node A1. The detection rate will remain zero if we set threshold T1 as <60 but if we increase T1 further then the detection rate will also increase. In the current scenario, a detection threshold T1 of 0.75 is sufficient to detect 100% of the attacks. A detection threshold T1 of 0.90, chosen by us, is appropriate for the attack detection algorithm.

5.4. False positive rate

The false positive rate (Rfp) is the ratio of the number of legitimate packets, which are mistakenly detected as attack packets to the total number of legitimate packets. It is measured as:

$$Rfp = \frac{P}{M}, \tag{7}$$

where P is the number of packets detected as attack packets and M is the total number of legitimate packets.

Based on observation of Table 3 for the legitimate traffic at router R2, if we set T1 in the range of 0 to 0.94 then the false positive rate is 0, but if we set T1 in the range of 0.94 to 0.99 then the false positive rate starts increasing, and for T = 1, the false positive rate will become 100%. Thus, the threshold T1 of value 0.90 chosen by us is appropriate for accurate attack detection. Figure 6 shows the effect of threshold values on the percentage of legitimate packets detected as attack packets.

The accuracy of the attack detection algorithm will highly depend on the threshold value. Inappropriate threshold values can cause high false positive and false negative rates. If the value is too high, it produces high false positives, and if the value is too low, it creates high false negatives. Thus, the threshold values should be chosen such that they keep false positive and false negative rates as low as possible.

6. Conclusion and future work

The proposed threshold-based attack detection mechanism detects DDoS attacks by monitoring flows on edge routers. The flow entropy, router entropy, and normalized router entropy are calculated for each flow passing through the edge router of source networks. If during a particular moment the value of normalized router entropy for a particular flow goes below a specific threshold, then the flow will be considered as suspicious. The suspicious flow is then further tested to confirm whether it is an attack flow or legitimate flow. This can be done by calculating the entropy rate for the suspicious flow on the existing edge router and the gateway router (for other upstream edge routers). If the difference between entropy rates is less than a given threshold then the flow is confirmed as a DDoS attack; otherwise, it will be a legitimate flow. The algorithm is tested for varying threshold values and the results show that the detection algorithms manage to detect more than 97% of attack packets during the attack. The false positive rate could be reduced to zero and the detection rate can be raised to 100% under perfect threshold values.

This paper presents only the first part of the defense model, which detects DDoS attacks in single ISP networks. Presently, we are exploring a variety of DDoS attacks and modifying this mechanism to suit different attack scenarios. In the future, we hope to extend this model to prevent DDoS attacks in multiple ISP networks. That involves the use of a hierarchy, where each ISP can have one controller and several agents. The agents will

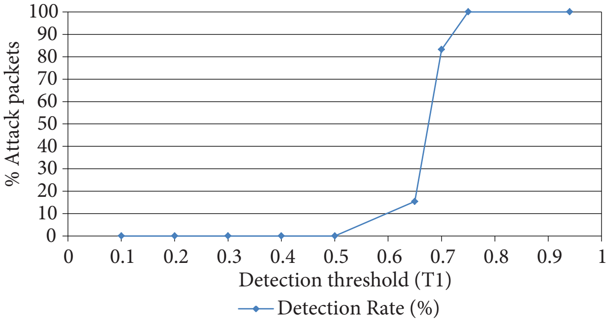

Figure 5. Effect of threshold value on detection rate.

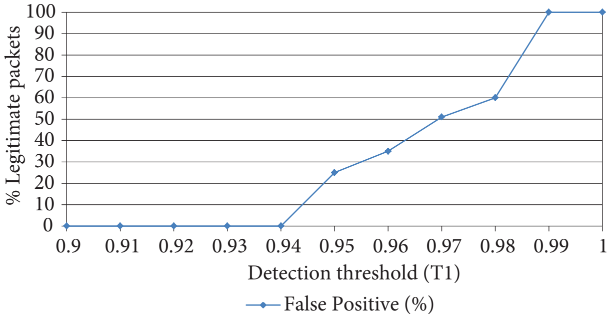

Figure 6. Effect of threshold value on false positive rate.

hold this detection algorithm and help in the detection and mitigation of DDoS attacks. The agents further share attack-related information with their controllers, which further share this information with neighboring controllers to carry out effective distributed defense.

References

[1] Douligeris C, Mitrokotsa A. DDoS attacks and defense mechanisms: classification and state-of-the-art. Comput Netw 2004; 44: 643-666.

[2] Mirkovic J, Rehier P. A taxonomy of DDoS attack and DDoS defense mechanisms. ACM SIGCOMM Comp Comm 2004; 34: 39-53.

[3] Bhandari A, Sangal A, and Kumar K. Characterizing flash events and distributed denial-of-service attacks: an empirical investigation. Secur Commun Netw 2016; 9: 2222-2239.

[4] Singh K, Dhindsa K, Bhushan B. Collaborative agent-based model for distributed defense against DDoS attacks in ISP networks. Int J Secur Appl 2017; 11: 1-12.

[5] Kassa M, Libsie M. A synchronized distributed denial of service prevention system. Comp Sci Inform Tech 2012; 2: 9-23.

[6] Liu H, Sun Y, Valgenti VC, Kim MS. TrustGuard: A flow-level reputation-based DDoS defense system. In: IEEE Consumer Communications and Networking Conference; 2011; Las Vegas, NV, USA. New York, NY, USA: IEEE. pp. 287-291.

[7] Yu S, Zhou W. Entropy-based collaborative detection of DDOS attacks on community networks. In: IEEE Sixth Annual International Conference on Pervasive Computing and Communications; 2008; Hong Kong. New York, NY, USA: IEEE. pp. 566-571.

[8] Qin X, Xu T, Wang C. DDoS attack detection using flow entropy and clustering technique. In: Proceedings of 11th International Conference on Computational Intelligence and Security; 2015; Shenzhen, China. pp. 412-415.

[9] Lee W, Xiang D. Information-theoretic measures for anomaly detection. In: IEEE Symposium on Security and Privacy; 2001; California, USA. New York, NY, USA: IEEE. pp. 130-143.

[10] Bhandari A, Sangal A, Kumar K. Destination address entropy based detection and traceback approach against distributed denial of service attacks. Int J Comput Netw Inform Secur 2015; 7: 9-20.

[11] Tellenbach B, Burkhart M, Schatzmann D, Gugelmann D, Sornette D. Accurate network anomaly classification with generalized entropy metrics. Comput Netw 2011; 55: 3485-3502.

[12] Nychis G, Sekar V, Andersen D, Kim H, Zhang H. An empirical evaluation of entropy-based traffic anomaly detection. In: Proceedings of 8th ACM SIGCOMM Conference on Internet Measurement; 2008; Vouliagmeni, Greece. New York, NY, USA: ACM. pp. 151-156.

[13] Gu Y, McCallum A, Towsley D. Detecting anomalies in network traffic using maximum entropy estimation. In: Proceedings of the 5th ACM SIGCOMM Conference on Internet Measurement; 2005; Berkeley, CA, USA. New York, NY, USA: ACM. pp. 1-6.

[14] Behal S, Kumar K. Detection of DDoS attacks and flash events using novel information theory metrics. Comput Netw 2017; 116: 96-110.

[15] David J, Thomas C. DDoS attack detection using fast entropy approach on flow based network traffic. In: Proceedings of 2nd International Symposium on Big Data and Cloud Computing; 2015; Chennai, India. pp. 30-36.

[16] Fioreze T, Wolbers MO, Meent R, Pras A. Finding elephant flows for optical networks. In: IEEE/IFIP 10th International Symposium on Integrated Network Management; 2007; Munich, Germany. New York, NY, USA: IEEE. pp. 627-640.

[17] Claise B. Cisco Systems NetFlow Services Export Version 9. Request for Comments 3954. Fremont, CA, USA: IETF, 2004.

[18] Quittek J, Zseby T, Claise B, Zander, S. Requirements for IP Flow Information Export (IPFIX). Request for Comments 3917. Fremont, CA, USA: IETF, 2004.

[19] Cover T, Thomas J. Elements of Information Theory. 2nd ed. New York, NY, USA: John Wiley, 2007.

[20] Romana D, Mushashi Y. Entropy based analysis of DNS query traffic in the campus network. Journal of Systemics, Cybernetics and Informatics 2008; 6: 42-44.

[21] Varga A, Horing R. An overview of the OMNeT++ simulation environment. In: Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops; 2008; Marseille, France. New York, NY, USA: ACM. pp. 60:1-60:10.

[22] Gamer T, Scharf M. Realistic simulation environment for IP-based networks. In: Proceedings of 1st International Conference on Simulation Tools and Techniques for Communication and Systems & Workshops; 2008; Marseille, France. New York, NY, USA: ACM. pp. 83:1-83:7.