Computer Communications 35 (2012) 1380–1391

Contents lists available at SciVerse ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

DDoS flooding attack detection scheme based on F-divergence

Hamza Rahmani *, Nabil Sahli, Farouk Kamoun

CRISTAL Lab., National School for Computer Sciences of Tunis, University Campus Manouba, 2010 Manouba, Tunisia

ARTICLE INFO

Article history:
Received 29 August 2011
Received in revised form 10 February 2012
Accepted 3 April 2012
Available online 13 April 2012

Keywords:
DDoS
Flash crowd
F-divergence
Total variation distance
Entropy

ABSTRACT

The nature of the threats carried by Distributed Denial of Service (DDoS) attack requires effective detection as well as efficient response methods. However, feature-based schemes are unsuitable for real-time detection due to their complicated calculations and most of the statistical-based schemes do not distinguish DDoS attacks from legitimate changes. Besides, it is impossible to set a threshold that takes into account both false positives and false negatives. A hard threshold reduces the risk of false negatives but significantly increases the rate of false positives. In contrast, a soft threshold can easily be exploited by attackers to insert a malicious traffic that respects the conduct of good flow. To avoid these defects, we suggest a two-stage approach based on the detection of breaks in the distribution of connections size. A connection is defined as the aggregate traffic between two IP addresses, where one address belongs to the police address set, and the other is a foreign address. The connection size is measured in number of packets. To achieve our goal, we employ Total Variation Distance (TVD) to measure horizontal and vertical similarity among flows. We investigate a class of intelligent denial of service attacks which, unlike high-rate attacks, are difficult for other's schemes to detect. The experimental results indicate that our scheme can detect DDoS flooding attacks accurately. The effectiveness of our approach, even against intelligent attacks, is around 90%.

1. Introduction

Denial-of-service is not simply another weak spot in the Internet, but rather a slip that can be mended with slight protocol changes or by the deployment of sophisticated defenses at potential target sites. The origin of denial-of-service lies in the very core of the Internet architecture. Design decisions reached several decades ago, that brought us connectivity and information wealth beyond our wildest dreams, carry within their key concepts the root of the DDoS threat. Due to the Internet vulnerability, it is easy for hackers to spoof source IP addresses of attack packets [1], verifying the attack flow pattern [2,3], etc. Generally, DDoS detection schemes include a change- point detection [4,5], activity profiling [6,7], information distance detector [8], wavelet analysis [9,10], and so on and so forth. All these methods are based on the features or fingerprints of specific DDoS attacks. Unfortunately, it is accessible for hackers to mimic these features and fool the user detection methods. Due to the open architecture of the Internet, hackers can also spoof the source IP addresses of the attack packets according to the real Internet IP address distribution. To act against the source address distribution based detection algorithms [11,12], hackers can change the TTL value of the attack packets according to the real hop distance between zombies and victim respectively in order to counter hop-count detection methods [12,13]. However, all these events are sometimes hard to be detected in real time from an observed traffic. This affair becomes harder, if the observed network carries a large amount of traffic, which makes malicious one inconspicuous [14]. So, how to make malicious traffic exposed and detected accurately is a pending problem.

To fly under the radar, attackers may also mimic the behaviors of flash crowds [4,15,16], a sudden increase of legitimate traffic, e.g. many fans will access the official website when an important match is ongoing; many people will check CNN website when breaking news comes. DDoS attacks and flash crowds share similar behaviors, and we must differentiate them effectively, otherwise, we may raise false alarms. In fact, it is a big challenge for defenders to distinguish DDoS flooding attacks from flash events [4,15,16], and the consequences are serious, in case we fail to do so. On the one hand, attackers can mimic the traffic features of flash crowds to disable our detectors, i.e. occurrence of false negative. On the other hand, our detectors may treat the legitimate flash crowds as DDoS attacks, i.e. occurrence of false positive.

Information theory-based measurements have been successfully used to detect specific types of malicious traffic. Research [16] tried to use three dimensions to leave out flash crowds from DDoS attacks: traffic patterns, client characteristics and file reference characteristics. The entropy detector mentioned in the survey [15] came from reference [6], which can raise the alarm for a crowd

* Corresponding author. Tel.: +(00216) 97560593; fax: +(00216) 71430570.
E-mail addresses: hamza.rahmani@cristal.rnu.tn (H. Rahmani), n.sahli@ansi.tn (N. Sahli), frk.kamoun@planet.tn (F. Kamoun).

0140-3664/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.comcom.2012.04.002

access. However, it cannot identify DDoS attacks from the surge of legitimate accesses, e.g. flash crowds. Reference [4] tried to separate flash crowds from DDoS flows using the change-point detection method, but this method can be cheated easily. Zombies, for instance, can increase the number of attack packets very slowly, which will probably disable the change point detectors. In Research [17] a distributed approach has been suggested to detect DDoS attacks in ISP domain. Entropy on four traffic features (source IP, destination IP, source port and destination port) is calculated at every PoPs (Points of Presence) of ISP. In Research [18], entropy has also been used to characterize distributions of packet features (such as IP addresses and ports) observed in OD (Origin-Destination) flows traces, and then a multi-way subspace method is recommended to detect a wide range of anomalies. These two methods work at the cost of a hard statistical work. When traffic records are huge, the time spent to extract traffic features is a bottleneck. Kolmogorov Complexity is closely related to entropy, which can be applied to detect DDoS attacks [19] or scan traffic [20] with high accuracy based on distributional changes in the traffic features. These methods specialize respectively in a certain type of anomaly, but they don't analyze and detect anomalies in general.

Distance measurement of traffic flows is an effective way to single out DDoS attack flows from flash crowds. As we know, zombies use pre-built programs to pump attack packets to the victim. Consequently, the similarity between attack flows is much higher than the similarity between random legitimated flash crowds. Some researches have been done to solve the similarity problem using stochastic methods in frequency-domain [22,23]. Cheng et al. [22] mapped DDoS attacks from the time domain to the frequency domain, and then transformed it to power spectral density to identify the DDoS attacks. Spectral analysis [21] employed digital signal processing method to reveal the hidden mischievous DDoS attacking packets. Ref. [23] used data mining technology to dig the DDoS attack information, but it is costly in terms of computing and delay. Ref. [24] explored the similarity methodology preliminarily, and the effectiveness of the proposed method is confirmed. Ref. [8] used Hellinger distance to detect voIP floods in peer-to-peer networks. Finally, Sengar et al., introduced distance gauge between various traffic features in the traffic data streams for anomaly detection [25].

For the construction of accurate online traffic profiles, we developed an online statistical anomaly detection framework that generates alerts based on abnormal variations in the connections size distribution. It does so by viewing collections of related packet streams as evolving probability distributions and measuring abnormal variations in their relationships based on the Total Variation Distance. The number of false positives and negatives due to behavioral alias can be significantly reduced. Moreover, to capture the essence of the dynamics of the used network traffic, a moving window mechanism is applied to have normal traffic profile. We evaluate the accuracy and performance using live traffic traces ranging from points in the core of the internet collected in 2010 at trans-Pacific line [26] to those inside an edge network collected in 2001 at UCLA CSD [27]. The experimental results indicate that the scheme can detect DDoS attacks accurately in time and space. The major contributions of this paper are as follows:

- The originality of our approach lies in the double applying of the TVD to study the traffic features variations: horizontally, for suspect flows detection and vertically to distinguish DDoS attacks from legitimate variations.
- A pre-treatment is applied before the phase of detection to extract the useful information and decrease the rate of false-positives.
- The suggested strategy is scalable and practical. It can be applied at the source-level, in the core of Internet or at the victim-level.
- Our suggestion needs to only inspect the IP header fields of every packet. This makes it simpler and more practical for real-time implementation even on high speed links.
- Throughout our analysis, we found out that our scheme is capable of detecting not only high-rate attacks but also low-rate attacks that go far beyond the volume-based and entropy-based schemes ability to detect.
- The proposed method is independent of any specific DDoS flooding attack tools. Therefore, it can actively detect any new form of forthcoming attacks.
- The proposed algorithms are tested by real datasets, and we can differentiate DDoS flooding attacks from flash crowds with lots of accuracy.

The remainder of this paper is structured as follows: In Section 2 we describe the detection design, including Total Variation Distance. In section 3 we analyze the first phase of our detection scheme after which a flow will be stated as normal or suspicious. In section 4, we study the differentiation phase of our detection scheme which allows us take a decision on the legitimacy of a suspect flow. In section 5, we evaluate the performance of the suggested scheme. Eventually, in section 6, we draw a conclusion on our research.

2. Detection design

In this section, we describe the design of our detection algorithms that measure the statistical properties of specific fields in the IP packet's headers. Four steps are basically to be followed from observing a traffic stream to raising an alarm as shown in Fig. 1. The first step is to gather information from streaming traffic. At a particular link, the traffic monitoring module observes packets IP headers and collects statistics. The second step is to measure any variation observed in the distribution of inflow connections size over time. This is the horizontal distance measurement process where the profiles of packet features are compared, and their similarity is given a numerical value between 0 and 1. A threshold of measured Total Variation Distance that can differentiate network anomalies from normal behaviors is set. After this phase, a flow is said to be suspicious or normal. The third step is applied only on suspicious flows to distinguish DDoS attacks from flash crowds by vertically applying the Total Variation Distance between the inflow and outflow connections size distribution. Finally, the fourth step is to gather all the information and classify the attack alerts with the possibility of identifying the sources causing change. Now we describe all the steps in detail, including the architecture of the on-line detection system and its packet's features collection mechanism.

2.1. Data collection

It is very hard to obtain an anomaly causing data, mainly when these flows are sensible and susceptible to be used in real attacks as is the case of DoS attack. Most of the work parts that deal with DoS attacks use flows achieved in laboratories by traffic generators or by DDoS tools. Throughout this work, we use a variety of real Internet traffic traces collected from 2001 to 2010. The normal traffic traces are taken from:

- Trans-Pacific line packet trace collected in 2010 at The MAWI working group that focuses on traffic measurement analysis. In particular, the focus of the working group is a long-term measurement on wide-area, global internet [26].
- UCLA Computer Science Department Packet Traces collected in 2001 [27].

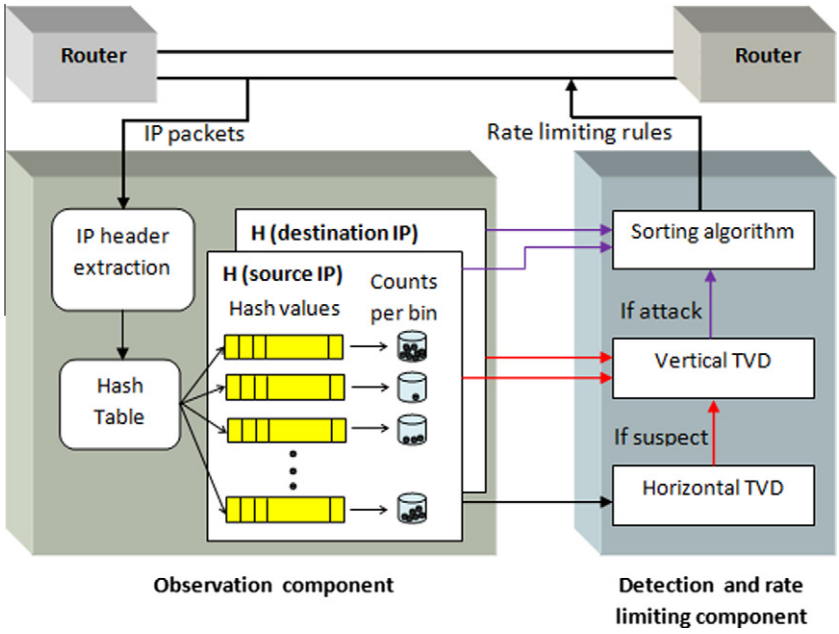

Fig. 1. Detection design.

The DDoS traces are extracted from:

- Some Generated Attack Traces from UCLA Computer Science Department Packet Traces [27].
- The CAIDA "DDoS Attack 2007" Dataset: This dataset contains approximately one hour of anonymized traffic traces from a DDoS attack on August 4, 2007 (20:50:08 UTC to 21:56:16 UTC). This type of denial-of-service attack attempts to block access to the targeted server by consuming computing resources on the server and by consuming all the bandwidth of the network connecting the server to the Internet [28].

2.2. Traffic monitoring

The observation component monitors the traffic at the connection granularity in order to detect difficulties in communication that could be a sign of a DDoS attack. A flow is defined as the aggregate traffic between the police address set and a foreign host. Additionally, it monitors two-way traffic at the connection level, attempting to identify legitimate connections that should receive good service in case the associated flow becomes rate-limited. A connection is defined as the aggregate traffic between two IP addresses, where one address belongs to the police address set, and the other is a foreign address. Fig. 1 shows the observation component which captures every packet at a particular Internet link. The packet attributes (source/destination IP addresses) are extracted from the packet header and hashed to unique value. These unique hash values of selected features correspond to hash bins, with an initial counter value of 1. Each time, with the arrival of a next incoming packet, if the hashed value of its traffic feature points to an already existing hash bin, the traffic monitoring module just increments the counter value. Otherwise, a new hash bin with an initial counter value of 1 is created. The creation of hash bins and incrementing counter values occur in each observation window. Our approach only requires access to the IP header of each packet whose sources or destinations IP address belong to all monitored IP addresses. This makes it practical and easy to implement online even at high speed links.

2.3. F-divergence

How to effectively describe the change on connections size distribution in a manner that provides the necessary information for attack detection is the key question. Generally, a network provides the customers with a limited number of services (web, mail, download etc.) and every Internet service corresponds to a specific Internet application. Knowing that the sizes of connections vary substantially in accordance with the type of the used Internet application (FTP, HTTP, DNS etc.), the connection size distribution depends primarily on the number of clients using each service. It is then reasonable to expect that the connections size distribution is generally short term stable. So, the majority of active connections at the nth time window will be featured in the next time window as shown in the Fig. 2. Our idea is to find a gauge that allows exploiting this property of IP addresses to detect abnormal variations in the inflow connections size, while highlighting the new IP addresses that should be the source of a possible attack. After conducting observations, we found out that F-divergence [29], which describes the degree of similarity between two distributions, is an excellent gauge for extracting the properties of connections size distribution in a manner that is appropriate for DDoS attack detection. The experiments that we conducted show that the Total Variation Distance provides better results than other F-divergence measures such as Kullback–Leibler, Hellinger, $\chi 2$-divergence, etc. [18]. Our choice of TVD is also due to its desirable properties: it is based on the proportional abundance, has the range of similarity values between 0 and 1 and takes into account the probability 0. TVD presents an intrinsic way to estimate the distances between probability measures independently of the parameters.

To explain this, let P and Q be two probability distributions on a finite sample space Ω, where P and Q on Ω are N-tuples $\{p_1, p_2, \ldots, p_N\}$ and $\{q_1, q_2, \ldots, q_N\}$; respectively, satisfying (in)equalities $p_i \geqslant 0$, $q_i \geqslant 0$, $\sum p_i = 1$ and $\sum q_i = 1$. Then the TVD between P and Q is defined as

$$d = \frac{1}{2}\sum_{i=1}^{N}|p_i - q_i| \tag{1}$$

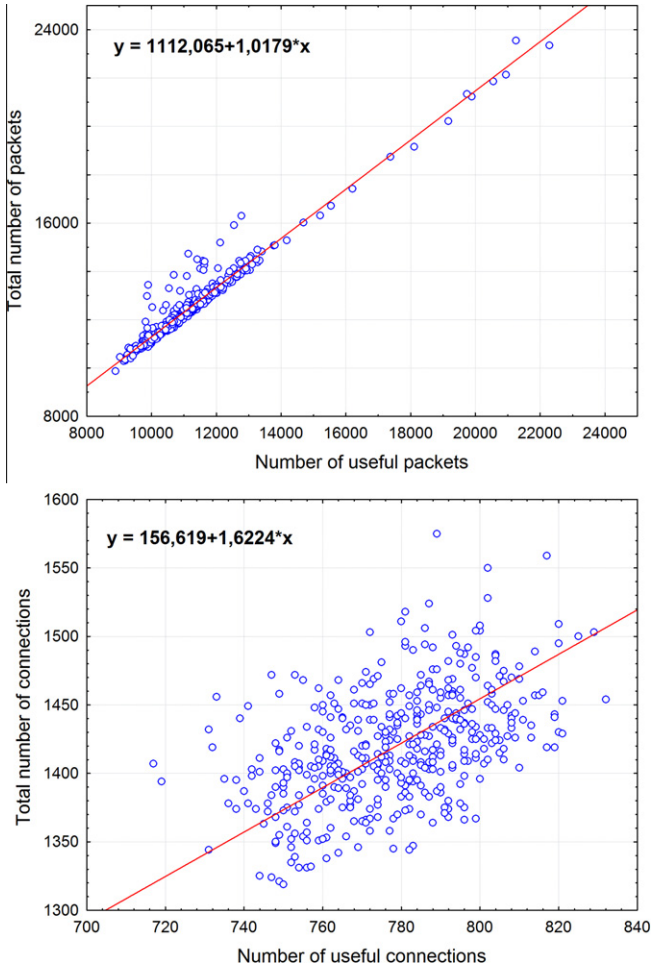

Fig. 2. Useful flow extraction mechanism.

The TVD satisfies the inequality $0 \leqslant d \leqslant 1$ and $d = 0$ when $P = Q$. Disjoint P and Q shows the maximum distance of one. Further details on TVD can be found in [30].

2.4. Detection strategy

In statistical approaches for DDoS attacks detection, the occurrence of attacks is reported when a preset threshold of normal traffic is reached. The statistical properties of the Internet flow present a large variability depending on time. The threshold value of normal flow must take into account the changing nature of Internet flows. A firm threshold reduces the risk of false negatives but significantly increases the rate of false positives. In contrast, a soft threshold can easily be exploited by attackers to insert a malicious traffic that respects the conduct of legitimate traffic. To avoid these defects, we suggest a two-stage approach based on the detection of breaks in the flow statistics. To detect suspicious flows, we use the TVD to measure the changes in the inflow connections size distribution over time. At the end of this step, the aggregate flow is declared normal when the calculated TVD is greater than a predefined threshold. Subsequently, to Figure DDoS attacks from legitimate increases, TVD is used to measure the similarity between inflow and outflow connections size distribution for each observation interval T. Lastly, let us announce that the first phase of our approach is already partially studied in the literature related to the DDoS attack detection [8,25]. However, we have improved and added an intermediate stage to extract only useful connections, which can be the object of a DDoS flooding attack. The differentiation phase is an innovation within the scope of works connected to the DDoS attack detection. A comprehensive approach of detection and differentiation based on the connections size distribution is thus recommended.

3. DDoS attack detection

3.1. Limitations of the entropy-based schemes

We mainly offer this work to correct defects in DDoS attack detection approaches based entropy. Entropy was the most previously used DDoS detection gauge [6,15,18]. Entropy in its basic form is a measure of uncertainty rather than a measure of information. Specifically, the entropy of a random variable describes the degree of dispersal or concentration of a distribution. When the entropy of a random variable is large, uncertainty as to the value of that random variable is large, and vice versa. In other words, the entropy can detect and distinguish DDoS attacks from legitimate flow when the flow maintains a target level of coherence between the size of aggregate flow and the number of active connections. We note two major flaws in the use of entropy. First, the calculated value of entropy depends only on the values taken by each variable distribution (i.e. proportion of each connection), but does not depend on the variable itself (i.e. couple of IP address that constitutes each connection). This prevents the entropy to follow the natural evolution of the flow which appears in the renewal of IP addresses. Secondly, the values of the sizes of the legitimate connections can be very different in the sense that the size of few connections may be greater than the sum of the sizes of other connections. In this case, coherence between the aggregate flow size and the number of active connections is impossible to quantify using a single gauge as it is the case for entropy or F-divergence. In section 6 we will show that the use of an F-divergence gives best results. These characteristics of entropy-based approaches can be easily exploited by hackers to construct a stream that follows the distribution of normal flow. The task of the hackers is facilitated by the fact that the preset threshold of normal flow must take into account the sudden changes frequently reported in the Internet streams. Lastly, in this work we used only the distribution of the number of packets per connection as a feature. all existing entropy-based schemes used many other features such as the number of bytes, the number of ports, etc. Our goal is to show that our F-divergence-based approach gives better results. An extension for other types of features or a combination of features is then possible.

Assume that P is an array of normalized frequencies of source address bins size distribution (i.e., $p_i = n_i/\sum n_i$, $i = 1,2,\ldots,N$) collected over the training window, i.e., nth time windows data set. The entropy is defined as:

$$E = -\sum_{i=1}^{N} p_i Log_2(p_i)$$

3.2. Useful flow extraction

In the field of Internet traffic classification, large connections will be sometimes referred to as elephant and small connections as mice. For several reasons, this dichotomy is largely used in the literature, e.g. see the discussion in Papagiannaki et al. [31]. Internet flow is mainly made of mice connections whose size is of a few packets [32]. Despite that these connections represent the majority of active connections; they only transfer a small proportion of the total number of packets. These connections cannot be the origin of a DDoS attack. Thus, they are useless within the scope of this work. Taking into account only the useful information, we have considered the common IP connections which are active in both

the training window and the observation window (i.e. two successive windows). To prove our hypothesis, we used two scatter plot figures. In the Fig. 2 (a), the *x*-axis represents the number of packets transferred by common connections and the *y*-axis represents the number of packets transferred by the active connections during the observation window. In the Fig. 2 (b), the *x*-axis represents the number of common connections and the *y*-axis represents the number of active connections during the observation window. We note that almost half of the number of connections available during the observation window transfers only a very small percentage of packets (i.e. slope almost equal to 1) and the common connections that represent the other half transfer almost 90% of the total number of packets. The TVD value, as shown in Equation (1), depends only on the size of each IP connection, this offers several advantages. It gives accurate results that make it easy to identify the nature of variation efficiently, ignores the peaks with very low period that are very often the sources of false-positives and applies the entire distance gauge even those that do not take into account the probability 0.

3.3. Detection algorithm

This first phase involves using the TVD to indicate the occurrence of abnormal disruption in statistics for the inflow connections size distribution. It enables us to quantify the distribution of the variation over the time of the useful connections size. By observing the time series of the TVD between two successive time windows, we can expose the similarity between two connection size distributions and detect suspicious attack points. Section 6 shows that at this stage of detection, our approach is better than entropy-based detection. Our calculation algorithm of the TVD between inflow connections size distribution over two successive time windows is the following:

Assume that P is an array of normalized frequencies of source address bins size distribution (i.e., $p_i = n_i/\sum n_i,\ i = 1,2,\ldots,N$) collected over the training window. Q is an array of the same feature bins size distribution (i.e., $q_i = m_i/\sum m_i,\ i = 1,2,\ldots,M$) collected during the observation window, i.e., at the $(n+1)$th time window. We consider only the K useful connections, i.e., common connections in the training window and the observation window. The horizontal TVD between P and Q is defined as:

$$H_{TVD} = \frac{1}{2}\sum_{i=1}^{K}|p_i - q_i|$$

3.4. Discussion

Fig. 3 shows that the occurrence of DDoS attack leads to a significant decrease in the horizontal TVD values. We note that the increase in the number of packets is accompanied by an increase in the number of connections. The number of connections has increased from 1500 to 2500, while the number of incoming packets has increased from 12000 to 40000. This Figure shows a typical example of DDoS attack based on IP address spoofing. However, the distribution of the size of connections from the attack is different from that of legitimate flow, and our model could detect it. Attack's connections represent the highest values of P during the training window and highest values of Q during the observation window. When calculating TVD values, large probability values will cancel each other to give a distance value even lower than the proportion of DDoS flows is larger. Figs. 4 and 5 show, however, that during a normal behavior the increase in the number of packets is accompanied by a proportional increase in the number of active connections. The values of probabilities representing the new connections look like those of a normal traffic and TVD value undergoes a small change which depends essentially on the

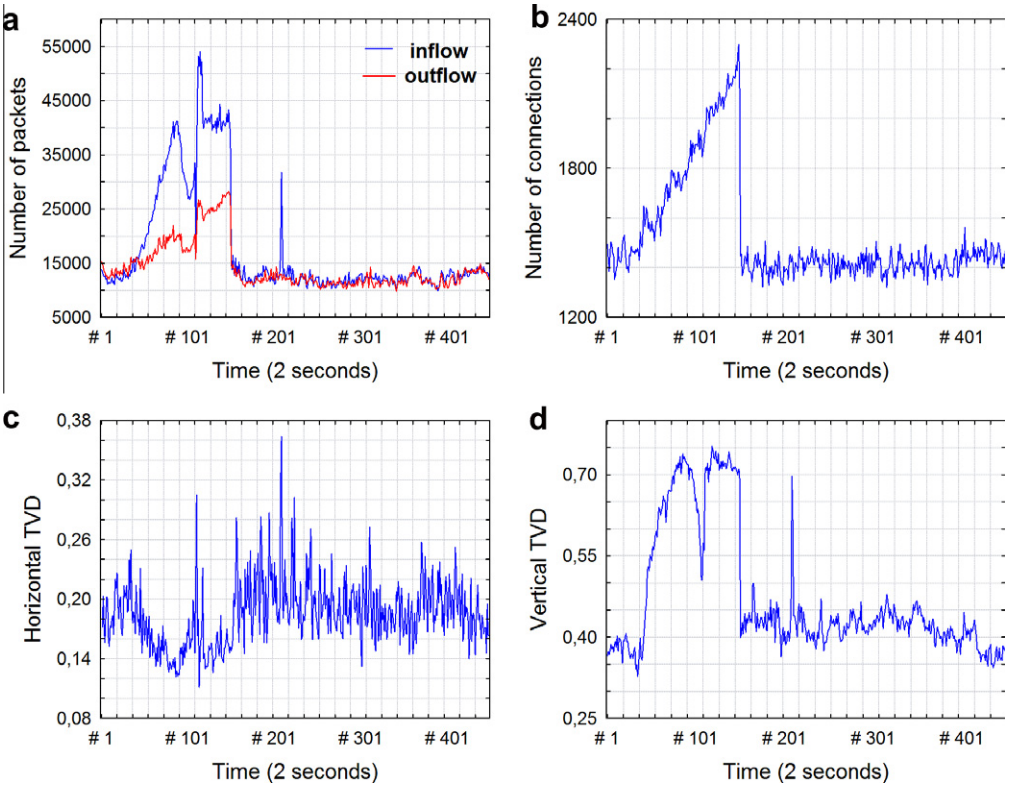

Fig. 3. DDoS attack flow: (a) inflow and outflow sizes; (b) number of connections; (c) Horizontal TVD; (d) Vertical TVD.

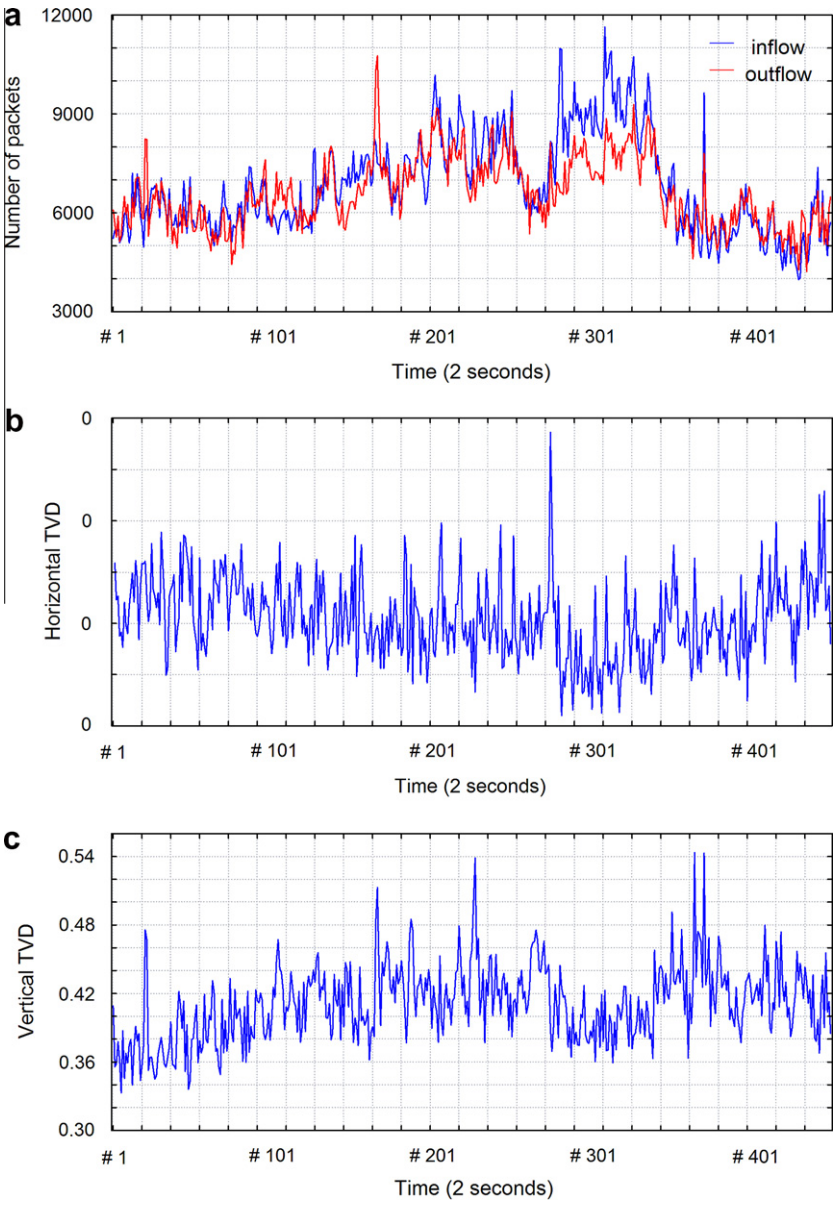

Fig. 4. Normal flow: (a) inflow and outflow sizes; (b) horizontal TVD; (c) vertical TVD.

number of active connections. We can easily deduce from equation 1 that the the attack is distributed the lower the chance of detection is. We must opt for a low-threshold value during the detection phase and add a second phase for differentiation. Finally, in Fig. 3, at the second 209–210, the very strong peak reported in the number of packets caused no decrease in the horizontal TVD curve. Knowing that the traffic contains only common connections, this short period peak has been ignored during the extraction of useful information. This allows us to reduce the rate of false positives and facilitates the selection of a reliable threshold.

4. Differentiating DDoS flows from flash crowds

In this section, we are going to describe the details of the differentiation algorithms. This step is imposed by the fact that it is impossible to find a simple gauge which allows taking into account the legitimate and illegitimate changes. It is the most important element that will decide on the flow legitimacy. When a flow reaches at the differentiation module, we are not sure whether it is a DDoS attack or not. However, we name the surge flows as suspicious flows, for the moment. The differentiation component will activate the differentiation algorithm to take the decision later. A DDoS attack is characterized by an explicit attempt by attackers to prevent legitimate users of a service from using that service. There are two general forms of DDoS attacks: those that crash services and those that flood services. We are interested in the following three types of flooding attacks: ICMP flood, TCP SYN flood and UDP flood. During a flood attack, the attacker recruits as many agents and each agent sends more packets to saturate as soon as possible the victim. This kind of attack is the most common and leads to the consumption of the victim's target resource.

As shown in Fig. 3, a flood attack always leads to an increase in the distance between the curves defining the size of inflow and outflow. Our differentiation approach tries to measure this discrepancy by vertically applying TVD to calculate the distance between the inflow and the outflow connections size distribution.

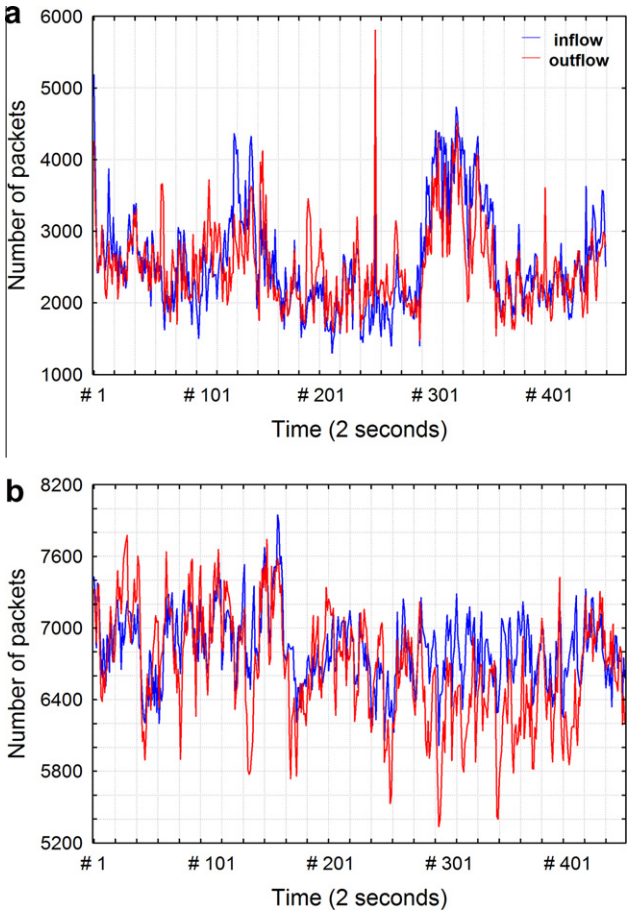

Fig. 5. Inflow and outflow sizes comparison for: (a) TCP traffic; (b) UDP traffic.

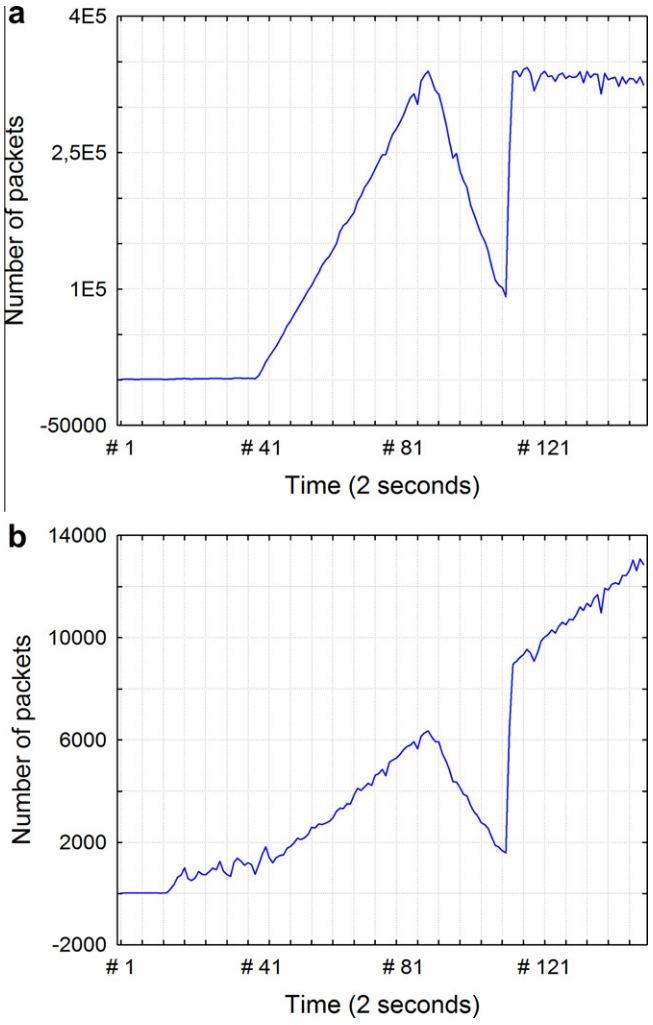

Fig. 6. High-rate DDoS attack: (a) victim inflow size; (b) victim outflow size.

4.1. Congestion

4.1.1. TCP connections

TCP connections represent the most part of Internet connections. TCP uses a two-way communication paradigm to achieve consistent delivery. During a TCP connection, the data from the source to the destination is controlled by a constant flow of acknowledgments in the reverse direction. If the flow of acknowledgments subsides, it is regarded as a sign of abnormal behavior such as congestion [33]. Therefore, the sending rate from the victim's network is promptly reduced. Thus, normal TCP communication can be modeled by the relationships between the numbers of packets sent-to and received-from a specific destination. Ideally, this ratio should be one, but network congestion and different TCP implementations using delayed and selective acknowledgments push it to slightly close values [34]. The experiments we undertook show that the ratio can be higher, almost equal, or even lower than one depending on the services provided by the network being monitored. More importantly, we show that the two curves defining respectively the number of packets sent-to and received-from a specific destination always keep the same shape during DDoS attack as shown in the Fig. 3. However, the distance between the two curves indicates an abnormal increase during abnormal behavior like congestion or DDoS attack. Our goal is to quantify this discrepancy and show that above a certain value of the TVD, the chance of the occurrence of an anomaly, mainly DDoS attack, is considerable.

4.1.2. UDP connections

The UDP protocol is used for unreliable message delivery and in many cases; it does not require any reverse packets for its proper operation. Continuing its growth in traffic, connectivity and complexity, the current Internet is full of applications with rapidly changing characteristics. Although it is still an accepted assumption that most Internet traffic is transmitted via the TCP protocol [35], we expect the rise of new streaming applications [36] (e.g. IPTV such as PPStream, PPLive) and new P2P protocols (μTP) to increase the usage of UDP as a transport protocol. The use of UDP as transport protocol has rapidly increased from 2002 to 2009, although TCP sessions are still responsible for most packets and bytes. However, in terms of flows, UDP turned out to be the dominant transport protocol. For traces from 2002 to 2003, around 40% of UDP flows run on ports below 1024, including DNS (port 53), NTP (port 123) and NetBios traffic (port 137). Since 2003, the usage of ephemeral ports (>1024) has increased considerably. Besides DNS, NTP and NetBios ports, the most-used ports in terms of UDP flows are those normally used by P2P applications, such as 4672 and 4665 (eDonkey), 6881 (BitTorrent), 6346 (Gnutella) and 6257 (WinMX). A port-based analysis suggests that the recent increase in UDP flows on the traces analyzed stems mainly from P2P applications using UDP for their overlay signaling traffic [37]. In addition, for an ideal implementation of a P2P application,

everyone using this service should have a ratio of 1:1. There are thus as much data sent as received, in such a way that the remote loadings are fast and effective for all. The experiments we have carried out confirm all these assumptions. As shown in Fig. 5, the number of packets curves sent to and received from specific destinations keep the same shape for both UDP flow and TCP flow. As is the case for TCP, during a UDP flood, we expect a disproportion between the number of packets sent to and received from the victim.

4.1.3. ICMP connections

The ICMP protocol specifies various kinds of message. The "timestamp", "information request" and "echo" messages should be paired with the corresponding reply. The frequency of other ICMP messages, such as "destination unreachable", "source quench", "redirect", etc., is expected to be so small. During a normal behavior, the number of ICMP connections is low and any abrupt variation in the ICMP statistics corresponds to a legitimate or an illegitimate anomaly. As shown at the beginning of Fig. 6, the occurrence of an ICMP attack leads to a disproportion between the ICMP inflow and outflow curves even during low-rate attack. In this Fig. we also saw that no response was reported from the victim to the ICMP echo requests sent by the DDoS agents. This is because the network security procedures of the victim block ICMP echo packets.

4.2. Computing vertical distance

The occurrence of a DDoS attack causes discrepancy between the number of packets sent-to and received-from a specific destination seen as a sign of congestion during bandwidth attack. Discrepancy also occurs when the firewall blocks the flow of attack as is the case in the TCP SYN and ICMP echo request attacks. To measure this disparity, we use Total Variation Distance between inflow and outflow connections size distributions for each observation window. A false positive occurs when the size of one or several legitimate connections is very large compared to the average size of all active connections. Accordingly, it is very difficult to distinguish a flash crowd from DDoS attack when the variation in the size of the aggregate incoming flow is not accompanied by a proportional change in the number of active connections. During a legitimate traffic, the variation of the aggregate traffic size from the source to the destination is accompanied by a proportional change in the opposite direction. However, when DDoS attack occurs, this dependence disappears and a disproportion is announced at the level of the flow exchanged between the attacker and the victim [33,34]. This disproportion is a sign of the beginning of congestion caused by DDoS attack.

The calculation of a differentiation distance (Vertical $_{TVD}$) involves source IP address and destination IP address traffic features. Assuming that P is an array of normalized frequencies of source address bins size distribution (i.e., $p_i = n_i/\sum n_i$, $i = 1,2,\ldots,N$) collected over the observation window. Q is an array of normalized frequencies of the destination address bins size distribution (i.e., $q_i = m_i/\sum m_i$, $i = 1,2,\ldots,N$) collected over the same observation window. We assume that the ith bin corresponds to the same couple of IP addresses for P and Q and that the bin size may be equal to zero.

$$V_{TVD} = \frac{1}{2}\sum_{i=1}^{N}|p_i - q_i|$$

Our scheme defines the maximum allowed TVD value between incoming and outgoing traffic connections size respectively. The flow is classified as an attack if its TVD value is above the threshold.

4.3. Discussion

Fig. 5 shows very clearly that during normal comportment the curves of the number of packets sent to and received from a set of networks keep the same shape but have different scales. The position of the two curves varies with the type of services provided by the target networks. The occurrence of an attack causes a significant dissimilarity between these two curves. In Fig. 3, we show that a sharp increase in the number of legitimate packets, even if it is accompanied by an increase in the number of connections, causes disturbance in the horizontal TVD values. However, it is almost impossible to confirm the legitimacy of the flow through this single gauge. The application of vertical TVD values, however, allows very significant results and vertical TVD values greatly exceed the threshold of normal traffic. During normal behavior, in Fig. 4, the target network has responded with a proportional amount of traffic. The vertical application of TVD has removed the doubt and confirmed on the legitimacy of the flow even if the amount of traffic has doubled.

In Fig. 7, the histograms show a huge disproportion between the inflow, and the outflow connections size distribution. The grater the disproportion is, the more violent the attack is. So, over 98% of connections during normal behavior have a size between 1 and 20 packets. However, during the attack almost 60% of connections have a size between 50 and 100 packets and almost 7% have a size between 100 and 200 packets. Fig. 6 shows a real case of a DDoS attack issued from CAIDA and based on an ICMP echo request. We note that the attack can be divided into three phases.
- Preparation: the number of malicious packets is low. This phase

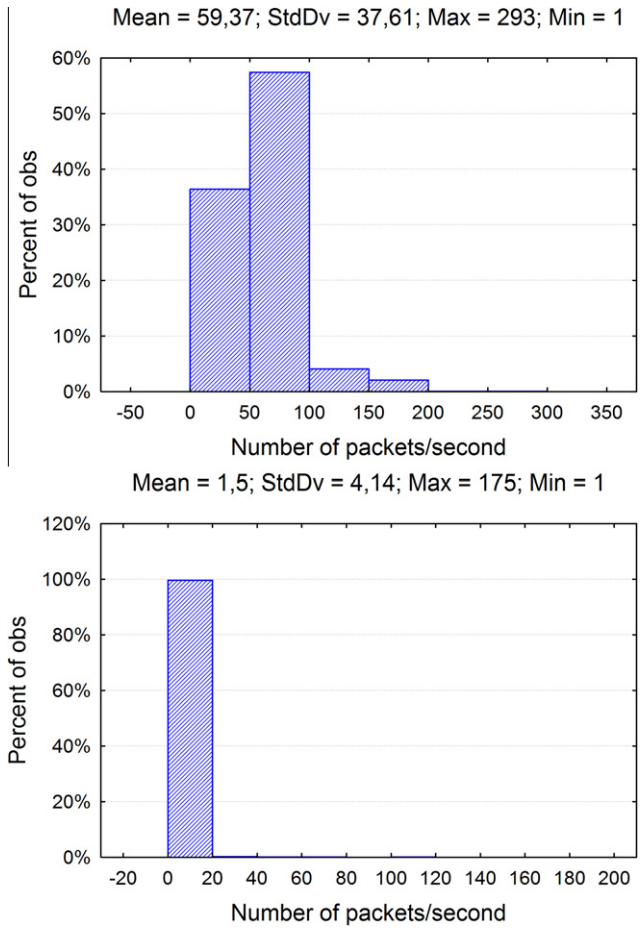

Fig. 7. Histograms of connections size: (a) victim inflow; (b) victim outflow.

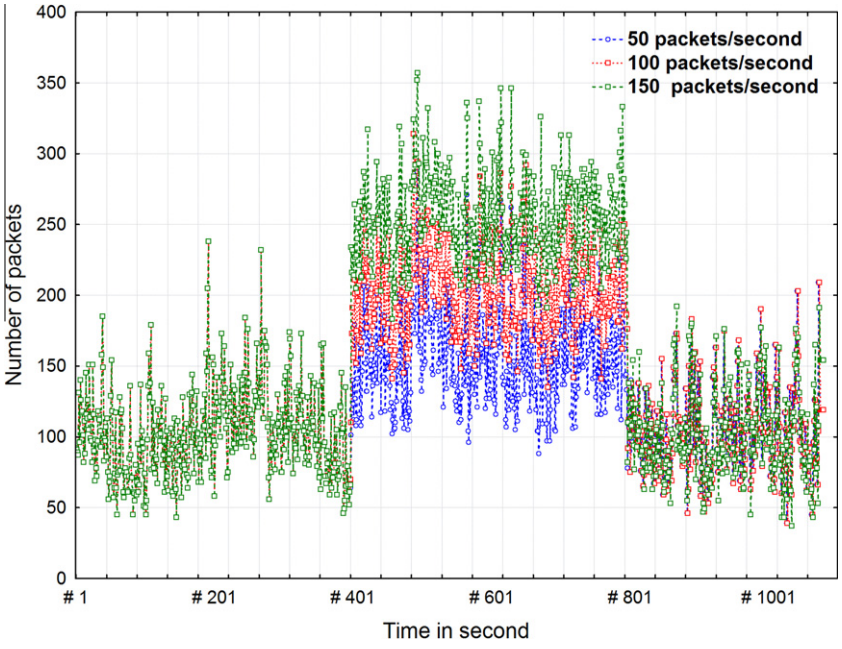

Fig. 8. Generated DDoS attacks using UDP packets.

lasted almost 18 minutes in our case. In the first phase, the attack is low-scale, but no response is reported to the ICMP echo requests. This is because the firewall or protocol implementations block this type of packet. Indeed, during this phase, we see a great similarity between the inflow and outflow size of TCP and UDP traffics. - Increase: In the second phase which lasted 80 seconds, the number of agents, and therefore, the number of packets, increased gradually until it reached a maximum value. We can see that the similarity in the TCP and UDP streams has disappeared. This is due to the fact that the attacked network resources are saturated and were unable to respond to the attack. - Stabilization: In the third phase, the number of packets and agents stabilized around the maximum value. The victim was unable to meet the requests from the attack agents and legitimate users.

5. Performances evaluation

To evaluate the effectiveness and performances of the proposed behavioral distance-based anomaly-detection mechanism, a software prototype is implemented to measure the horizontal and vertical distances of the selected traffic features and has been tested with real traffic traces. The effectiveness of TVD is evaluated in terms of its ability to detect and distinguish attack flow from legitimate flow. Of course, our detection scheme cannot detect all attacks completely, and actually no other scheme can do it. In a volume and entropy-based scheme, a false-negative case will occur for a low-rate DDoS attack that does not cause detectable disruption in traffic volume or in connections size distribution. For a feature-based scheme, a false-negative case will occur for a new kind of attack that has not been described in the database of attack features. Compared with the traditional entropy-based schemes, TVD-based scheme can significantly reduce the rate of false positive and negative. Moreover, because the TVD-based scheme is not based on the database of existing features attack, it can detect the new kind of attacks. Our validation approach is an answer to two questions. (1) How does the attack traffic rate affect the method's performance? (2) How can our approach withstand a sophisticated attack using multiple agents? To answer these two questions, we injected the attack traffic of different rates and number of agents into the real traffic data sets as shown in Fig. 8. In all these cases, the total flow was uniformly distributed between the agents. For example,

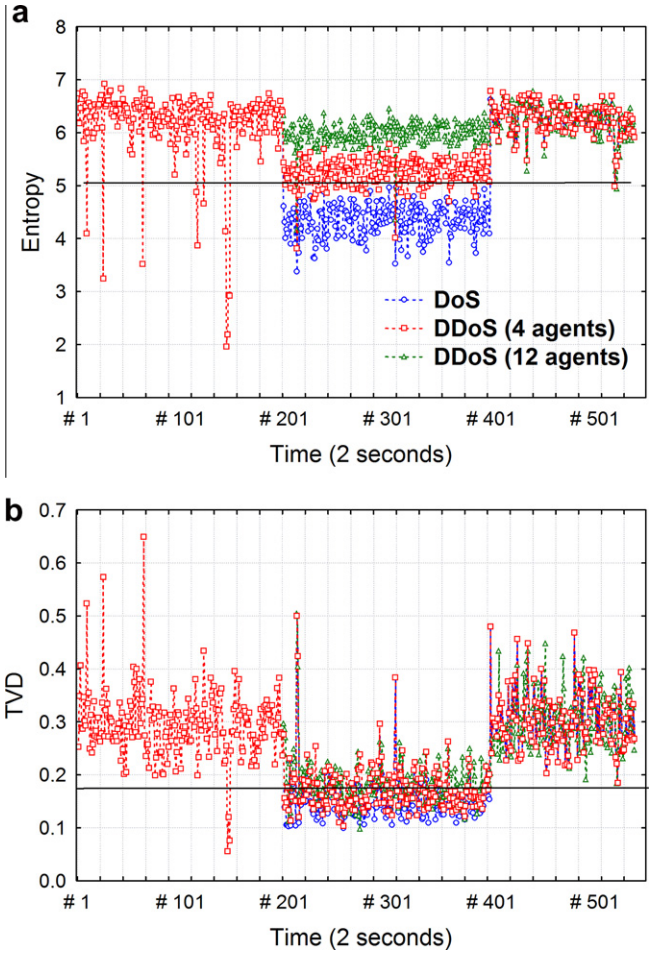

Fig. 9. 150 packets/second attack: (a) entropy; (b) horizontal TVD.

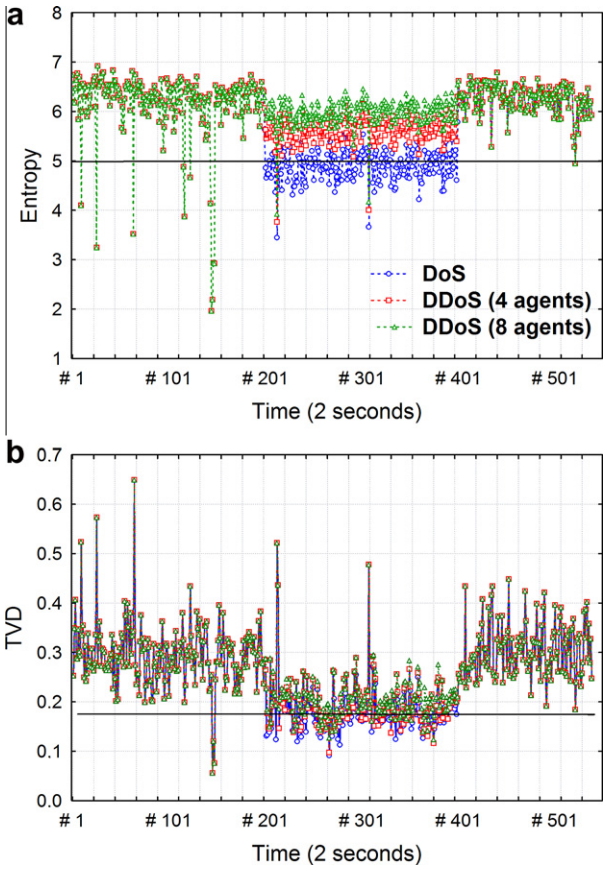

Fig. 10. 100 packets/second attack: (a) entropy; (b) horizontal TVD.

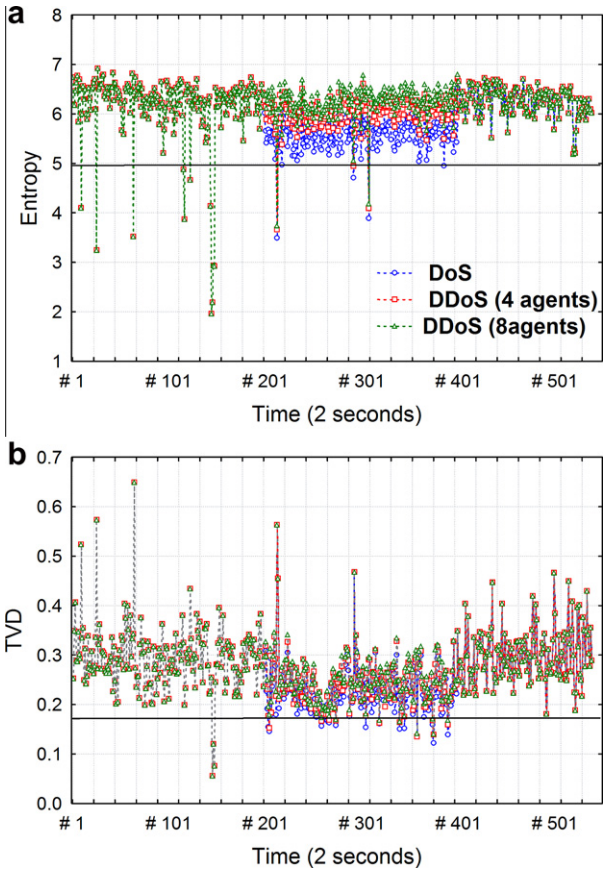

Fig. 11. 50 packets/second attack: (a) entropy; (b) horizontal TVD.

when the total flow was 100 packets per second, and the number of agents was equal to 4, each agent sent 25 packets per second. Finally, we note that our work will be completed by a rate-limiting procedure based on the connections size sorting. When an attack is detected, IP connections which have the largest $|p_{in}-p_{out}|$ are rate-limited. This procedure is repeated until the vertical TVD value drops below the predefined threshold of normal traffic.

5.1. Robustness against false negatives

In this section, we study the robustness of our detection scheme against risks of false negatives. The greater the size of bandwidth attach flow is, the lower the risk of false negative is. Figs. 9 and 10 show that during a large-scale attack, the entropy values tend to zero and our approach is 100% effective. A false-negative case will occur when a wily attacker knows the detection scheme. He can change his strategy by generating volume traffic carefully distributed that looks like a normal traffic. The risk of false negative also occurs for low scale attacks not detectable by other schemes. To prove the effectiveness of our TVD-based detection against such attacks, we used a traffic generator tool to insert gradually attack stream. The size of the normal traffic varies around 150 packets/second. The size of the inserted traffic varies between 50 and 150 packets per second, and the number of agents varies between 1 and 12. Traffic is reported suspicious if the values of TVD are less than 0.5 or if the values of the entropy are less than 1.8. One can notice that a significant change in the number of agents causes a large change in the entropy values, but TVD values keep a resistance to this change. Horizontal TVD gives better results against distributed attacks. The results in Figs. 9 and 10 clearly show that our detection phase offers more resistance to distributed attacks. This result reinforces our choice of using an F-divergence instead of entropy. Fig. 11 shows that the chances of detecting such attacks using horizontal TVD are slightly higher than those using an entropy-based scheme. Even for low rate attack, the calculated TVD values may be quite distant from the predefined threshold of normal traffic despite that the entropy values are above the threshold. Theoretically, this is explained by the fact that unlike entropy-based approaches, our two proposed applications of TVD take into account not only the probability distribution but also the active IP address distribution. In the first 100 seconds, Fig. 3 shows that the application of the vertical distance can determine the legitimacy of a stream with high accuracy even for low level attack. For comparison, our detection phase runs the same types of risks as entropy-based schemes, but the attackers will find more difficulties to create intelligent attacks that meet the predefined threshold.

5.2. Robustness against false positives

In this section, we study in detail the effectiveness of our detection system with false positives. False positives are most often reported in the case of a legitimate peak not accompanied by a large increase in the number of connections. Since it is impossible to define a threshold that allows taking into account low-rate attacks (i.e. risk of false negatives) and high-rate attacks (i.e. risk of false positives), we propose a second phase which allows deciding on the nature of suspicious flows by vertically applying the F-divergence. Since TVD yields very high detection probabilities, vertical TVD scheme has excellent performance against low rate variation as shown in Fig. 3. This phase allows choosing a soft threshold during suspicious flow detection phase and also greatly reducing the

Table 1
Entropy vs. horizontal TVD: effectiveness comparison.

Number of agents	50 packets/second		100 packets/second		150 packets/second	
	Entropy	H.TVD	Entropy	H.TVD	Entropy	H.TVD
DoS	Yes	Yes	Yes	Yes	Yes	Yes
4 agents	No	Yes	No	Yes	Yes	Yes
8 agents	No	Yes	No	Yes	No	Yes
12 agents	No	No	No	Yes	No	Yes

rate of false positives. The first major risk of false positives occurs during strong peaks due to a small number of large-size connections, which exactly resemble DDoS flows. These elephant connections, which are due to peer-to-peer connections or download, appear very often on the Internet flows. In the presence of such connections, the threshold of normal traffic is often crossed. It is impossible in this case to decide on the flow legitimacy by horizontally applying entropy, F-divergence, or any gauge. Luckily, as the traffic is legitimate, the target network responds to the received traffic with an equivalent amount of traffic in the opposite direction as shown in Figs. 4 and 5. By applying Vertical TVD, it is possible to decide on the legitimacy of the traffic. The second major risk of false positives occurs during strong peaks of short period not accompanied by traffic in the opposite direction. These peaks are always reported as attacks while they often correspond to other network problems. Fig. 10 clearly shows that these peaks always correspond to an overtaking of the threshold and therefore, to false negatives. Conversely, the horizontal TVD values are above the threshold for the majority of isolated peaks. In the Fig. 10, only three false positives are reported at the horizontal TVD curve. However, more than ten false positives are reported at the entropy curve. This result confirms that the useful connections' extraction mechanism facilitates the selection of a soft threshold for the detection phase.

5.3. Robustness against intelligent attacks

To assess the performance of our TVD scheme against such attacks, we consider two scenarios: single flooding source and multiple flooding sources. Firstly, for single flooding sources, the performances of the entropy-based and the horizontal TVD-based approaches are almost identical. In this case increasing the size of traffic is not accompanied by any increase in the number of active connections. Coherence between the number of connections and the total traffic size is not respected. The results provided by the application of the horizontal TVD are in this case excellent. Secondly, for multiple flooding sources, we show that our approach has enhanced the excellent results obtained by entropy-based scheme. Figs. 9, 10 and 11 clearly show that to escape our approach detection, it is necessary to use a very large number of agents to make the task of the intruder more difficult to achieve. Table1 shows that for the same amount of attack traffic, the results obtained using the simple entropy with N agents are similar to those obtained using the horizontal TVD only when the number of agents used in the attack is doubled. We can also see that for intelligent attacks, the effectiveness has increased from 33% for an entropy-based approach to 91% for a horizontal TVD-based approach.

On the other hand, when one or more agents' machine sent randomized spoofed IP traffic (e.g. syn flood attack) or spam, the number of connections becomes very large. The distribution of incoming traffic is drastically changed since the average value of the IP connections sizes decrease radically. This will generate a large statistical break between normal traffic and attack traffic. As for the differentiation phase during a bandwidth attack, a large number of requests lead to a saturation of buffer memory. The victim will struggle to respond to requests. Moreover, for security reasons, network administrators typically define a threshold for various types of packets such as TCP syn (the number of half-open connections), and ICMP echo. Thus, disproportion between the inflow and outflow sizes is inevitable. Fig. 3 shows that the vertical application of TVD is very effective against flooding attacks even in the presence of IP address spoofing. Secondly, we observe in the histogram of Fig. 7 that the sizes of the majority of attack connections are between 25 and 200 packets. IP addresses involved in the attack, even if they are stolen, have a greater life than the interval T. Thus, these connections will be considered by our approach even in the presence of the useful packet's extraction mechanism.

6. Conclusions and futures works

All statistical approach-based DDoS attacks detection present a risk of false positives and false negatives, mainly to sophisticated attacks because of the great variability of Internet traffic and the high frequency of legitimate peaks. We conclude that it is impossible to define a threshold that can take into account both false-positives and false-negatives. In other terms, it is impossible to find a gauge which can detect the DDoS attacks and be able to single them out from legitimate traffic. To cure all these defects, we proposed a new approach of detection and differentiation based on the Total Variation Distance. This approach is divided into two phases, an initial phase tending to suspect flow detection based on connections size distribution evolution over time and a second phase for the differentiation based on estimating the degree of congestion. The results show first that our approach is better than the entropy-based approaches and is more accurate according to sophisticated and low rate attacks. Our approach requires only access to the IP header of each packet and is practical for real-time implementation even on high speed links. Our approach is also practical since it can be applied near the source, on the core of the Internet or near the victim. On the other hand, we'd like to announce the difficulty of getting real traffic traces containing DDoS attacks. The CAIDA's 2007 DDoS Attack Dataset used in this paper contains the attack flow and the response of the victim to this flow, but does not contain any legitimate flow. These traces are thus not useful for an adequate and complete study of the DDoS problem. This prevented us from suitably studying the relationship between DDoS attack and congestion for real attacks. In our future we will try works to study this relationship taking into account the effectiveness of vertical application of F-divergence according to the various factors intervening in DDoS attacks.

References

[1] T. Peng, C. Leckie, K. Ramamohanarao, Survey of network- based defense mechanisms countering the DoS and DDoS problems, ACM Computing Surveys 39 (1) (2007) 1–42.
[2] M. Guirguis, A. Bestavros, I. Matta, Y. Zhang, Reduction of quality (RoQ) attacks on internet end systems, Proceedings of IEEE INFOCOM 2 (March 2005) 1362–1372.
[3] Y. Chen, K. Hwang, Spectral Analysis of TCP flows for defense against reduction-of-quality attacks, in: The 2007 IEEE International Conference on Communications (ICC'07), June 2007, pp. 1203–1210.

[4] Y. Chen, K. Hwang, Collaborative change detection of DDoS attacks on community and ISP networks", in: The IEEE International Symposium on Collaborative Technologies and Systems (CTS 2006), May 2006, pp. 401–410.
[5] Hongli Zhang, Zhimin Gu, Caixia Liu, Tang Jie, Detecting VoIP-specific denial-of-service using change-point method, in: 11th International Conf. on Feb. 2009, pp. 1059–1064.
[6] L. Feinstein, D. Schnackenberg, R. Balupari, D. Kindred, Statistical approaches to DDoS attack detection and response, in: Proceedings of DARPA Information Survivability Conf. and Exposition, vol. 1, IEEE CS Press, Washington, DC, USA, 22–24 April 2003, pp. 303–314.
[7] D. Moore, G.M. Voelker, S. Savage, Inferring internet denial of- service activity, in: Proceedings of 2001 USENIX Security Symposium, vol. 10, USENIX Association, August 2001, pp. 115–139.
[8] Yu, Shui, Thapngam, Theerasak, Liu, Jianwen, Wei, Su and Zhou, Wanlei 2009, "Discriminating DDoS flows from flash crowds using information distance", Proc. of the third International Conf. on Network and System Security, IEEE, Piscataway, N. J., 2009, pp. 351-356.
[9] Mohamed. Hamdi, Noureddine. Boudriga, Detecting denial-of-service attacks using the wavelet transform, Computer Communications 30 (16) (2007) 3203–3213.
[10] G. Carl, R.R. Brooks, S. Rai, Wavelet based denial-of-service detection, Computers and Security 25 (8) (2006) 600–615.
[11] Z. Duan, X. Yuan, J. Chandrashekar, Controlling IP Spoofing through Inter domain Packet Filters, in: IEEE Trans. on Dependable and Secure, Computing, 5(1), January-March 2008, pp. 22–36.
[12] F. Yi, S. Yu, W. Zhou, J. Hai, A. Bonti, Source-based filtering algorithm against DDOS attacks, International Journal of Database Theory and Application 1 (1) (December 2008) 9–20.
[13] H. Wang, C. Jin, K.G. Shin, Defense against spoofed IP traffic using hop-count filtering, in: IEEE/ACM Trans.s on Networking, vol. 15, No. 1, February 2007, pp. 40–53.
[14] Scott Fowler, Sherali Zeadally, Naveen Chilamkurti, Impact of denial of service solutions on network quality of service, security and communication networks article first published, online 26 JUL 2010, http://dx.doi.org/10.1002/sec.219.
[15] G. Carl, G. Kesidis, R.R. Brooks, S. Rai, Denial-of-service attack detection techniques, IEEE Internet Computing 10 (1) (2006) 82–89.
[16] J. Jung, B. Krishnamurthy, M. Rabinovich, Flash crowds and denial-of-service attacks: characterization and implications for CDNs and web sites, in: Proceedings of the International World Wide Web Conference, ACM Press, New York, 2002, pp. 293–304.
[17] Krishan Kumar, R.C. Joshil, Kuldip Singh, A Distributed Approach using Entropy to Detect DDoS Attacks, in: ISP Domain", IEEE-ICSCN 2007, MIT Campus, Anna University, Chennai, India, 2007, pp.331–337.
[18] Anukool Lakhina, Mark Crovella, Christophe Diot, Mining anomalies using traffic feature distributions, SIGCOMM, Philadelphia, USA, 2005. pp. 217–228.
[19] Amit. Kulkarni, Stephen. Bush, Detecting distributed denial-of-service attacks using kolmogorov complexity gauges, Journal of Network and Systems Management 14 (1) (March 2006) 69–80.
[20] E. Earl Eiland D, Lorie M. Liebrock, An application of information theory to intrusion detection, in: Fourth IEEE International Workshop on Information Assurance, 2006, pp 119–135.
[21] Y. Chen, K. Hwang, Collaborative detection and filtering of shrew DDoS attacks using spectral analysis, Journal of Parallel and Distributed Computing 66 (9) (2006) 1137–1151.
[22] C.M. Cheng, H.T. Kung, K.S. Tan, Use of spectral analysis in defense against DoS attacks", in: IEEE Global Communications Conference, 2002, pp. 2143–2148.
[23] K. Lu, D. Wu, J. Fan, S. Todorovic, A. Nucci, Robust and efficient detection of DDoS attacks for large-scale internet, Computer Networks 51 (September 2007) 5036–5056.
[24] S. Yu, W. Zhou, R. Doss, Information theory based detection against network behavior mimicking DDoS attack, IEEE Communications Letters 12 (4) (April 2008) 319–321.
[25] H. Sengar, H. Wang, D. Wijesekera, S. Jajodia, Detecting VoIP floods using the Hellinger Distance, in: IEEE Transactions on Parallel and Distributed Systems, vol. 19, No. 6, June 2008. pp. 794–805.
[26] http://tracer.csl.sony.co.jp/mawi/, accessed November 05, 2010.
[27] http://www.lasr.cs.ucla.edu/ddos/traces/, accessed November 05, 2010.
[28] http://www.caida.org/data/passive/ddos-20070804_dataset.xml.
[29] M. Basseville, Distance measures for signal processing and pattern recognition, Signal Processing 18 (4) (1989) 349–369.
[30] T.M. Cover, J.A. Thomas, Elements of Information Theory, 2nd edition., Wiley-Interscience, June, 2006.
[31] H. Papagiannaki K., Taft N., Bhattacharyya S., Thiran P., Salamatian K., Diot C., A pragmatic definition of elephants in internet backbone traffic, in: Internet Measurement Workshop. ACM, 2002, pp. 175–176.
[32] Sumantra R. Kundu, Sourav Pal, Kalyan Basu, Sajal K. Das: Fast Classification and Estimation of Internet Traffic Flows, PAM 2007: 155–164.
[33] T. M. Gil, M. Poletto. MULTOPS: a data-structure for bandwidth attack detection. in Proceedings. of 10th Usenix Security, Symposium, August 2001. pp. 23–38.
[34] J. Mirkovic, P. Reiher, D-WARD: a source-end defense against flooding denial-of-service attacks, in: IEEE Transactions on Dependable and Secure, Computing, 2(3), July-Sept. 2005, pp. 216–232.
[35] W. John, S. Tafvelin. Analysis of Internet backbone traf?c and header anomalies observed, in: IMC '07: Proceedings of the Seventh ACM SIGCOMM conference on Internet measurement, New York, NY, USA, pp. 111–116.
[36] P. Pan, Y. Cui, B. Liu, A measurement study on video acceleration service; in: Sixth Annual IEEE Consumer Communication & Networking Conference (CCNC), Las Vegas, NV, January, 2009.
[37] M. Zhang, M. Dusi, W. John, C. Chen. Analysis of UDP Traffic Usage on Internet Backbone Links , in: Proceedings of the Nineth Annual International Symposium on Applications and the Internet (SAINT 2009, Student Workshop), Seattle, USA, Jul. 2009.