

DETECTING DISTRIBUTED DENIAL OF SERVICE ATTACK TRAFFIC AT THE AGENT MACHINES

Vicky Laurens and Abdulmotaleb El Saddik
*Multimedia Communications Research
Laboratory
University of Ottawa*
email: {vicky, abed}@mcrmlab.uottawa.ca

Pulak Dhar and Vineet Srivastava
*Cistech Limited,
30 Concourse Gate, Unit 35
Ottawa, ON K2E 7V*
email: {pulak, vinnet}@cistech.ca

Abstract

Due to financial losses caused by Distributed Denial of Service (DDoS) attacks, most defence mechanisms have been deployed at the network where the target server is located. We believe this paradigm should change in order to tackle the DDoS threat in its basis: thwart agent machines participation in DDoS attacks. Our proposal consists of developing an agent to monitor the packet traffic rate (outgoing packets / incoming packets). Our first deployment is based upon characterizing TCP connections; normal TCP connections can be characterized by the ratio of the sent packets to the received packets from a given destination [1]. Preliminary results have shown that the traffic ratio values usually present larger values at the beginning of the run when there are not enough packets to make a decision on whether or not traffic is legitimate. A low value for threshold allows for faster attack detection, but it also increases the number of false-positives.

Keywords: Internet security; DDoS; Traffic monitoring.

1. Introduction

Internet distributed and non-distributed DoS attacks render a server offline causing legitimate users to be unable to access the targeted server. A DDoS attack is a coordinated DoS attack coming from several different machines, usually called zombies or agents (other names include daemons and slaves; in this paper we adopt the term agent). DDoS attacks are typically accomplished by overwhelming the targeted server with several bogus service requests so that the server engages its resources in processing false requests and therefore denying its services to legitimate users; this type of attack is known as a flooding attack. The difference between DoS and DDoS flooding attacks is that in the former, attack packets come from a single source whereas in the latter, attack packets come from multiple sources. This is why DDoS attacks are also referred to as multiple source DoS attacks. A typical example of a DDoS flooding attack is the TCP SYN Flooding attack, which is based on breaking out the three-way handshake when opening a TCP connection. The key factor of a flooding attack is to send enough packets out in order to consume the victim's resources (CPU cycles, network bandwidth, memory, and so forth).

Another method used to perpetrate DDoS attacks is focused on exploiting one or more vulnerabilities existing in the targeted system; this DDoS attack is called a vulnerability attack. Frequently the vulnerability is a bug in an application program, a protocol, or an operating system. By sending an incomplete packet, for instance, the application could reboot, crash, or slow down its performance. The main advantage of vulnerability attacks is that only a few packets can cause great damage. Nevertheless, these two types of attacks are in fact similar, and thus, an attack can be classified as belonging to both categories. Regardless of the type of DDoS attack, intruders usually recruit and control multiple computers in order to form an army of attack agent machines. A DDoS attack thus comprises two main phases: the recruitment phase (where agent machines are compromised), and the attack itself on the main victim (where services are denied to legitimate users).

For an attacker, success is comprised of not only denying services to legitimate clients, but also of not being caught in the act. A key factor that contributes to achieving the attacker's goals is the ease of the pre-attack phase or recruitment phase. In early DDoS attacks, recruitment was done manually; current attack trends include the use of blended threats and Internet worms in order to scan for vulnerable machines, compromise them, and then install the attack tool (if any). An example of Internet worms used for DDoS purposes are the Code-Red versions 1 and 2, CRv1 and CRv2 respectively. These worms were programmed to spread themselves out during days 1 to 19 of every month, and then, during days 20-27, they launched a DDoS attack on www1.whitehouse.gov [2]. CRv2 infected more than 359,000 machines in only fourteen hours, and this was accomplished by introducing a small change in the code of CRv1.

Due to the fast development of DDoS techniques as well as the increasing number of new vulnerabilities (which make it almost impossible to keep up to date), there is a need to deploy a tool which, does not require updates in order to detect ongoing attacks. Our approach to tackle the DDoS attacks is to develop a software tool for detecting outgoing attack traffic at the agent machines. This paper is organized as follows. Section 2 covers related work, Section 3 describes our proposal and

presents preliminary results as well, and finally Section 4 gives an overview of future work and concludes the paper.

2. Related Work

DDoS defence mechanisms can be classified in several ways, the two most commonly used classifications are: a) based on the defence mechanism's goal, that is, preventive versus reactive; and b) based on the deployment location, that is: the victim network, the intermediate network, and the source network. We will adopt the deployment location classification since our concern is solutions deployed at the source network where the attack packets are generated. Historically, source-end defence mechanisms have lacked deployment motivations primarily because the main victim suffers the most damage caused by DDoS attacks. In the past, agent machines could experience a reduced speed in their performance when the attacker was using their bandwidths but still damages were not severe. However, with the new DDoS attack trends, agent machines are now in great danger as well because sensitive data could be gathered after the intrusion (Symantec reported 54% of identified threats are able to expose confidential information [3]). Handler and agent machines have always been second, third, and nth layer victims; nevertheless, DDoS defence mechanisms are deployed primarily at the main victim network. We believe this paradigm should change in order to tackle the DDoS threat in its basis: thwart agent machines' participation in DDoS attacks.

The available source-end defence mechanisms can be divided into three categories: corporate level, academic research level, and personal level. At this time, we are not aware of any solution aimed at detecting attack traffic at each particular agent machine in any of the three categories. However, two similar approaches that have been developed are the Reverse Firewall offered by Cs3 [4], and D-WARD (DDoS Network Attack Recognition and Defense) [1], at the corporate and academic research level respectively. Both of these solutions are implemented at the edge of networks in order to have access to all incoming and outgoing network traffic.

Traditional firewalls protect networks from incoming malicious traffic whereas the Reverse Firewall protects the external network from flooding DDoS attacks [4]. By monitoring two-way connections, packets are filtered out and rate-limiting policies are applied if anomalies are detected. The Reverse Firewall is a hardware-based solution targeted for ISPs, Universities, and Corporations. In the case of the Reverse Firewall, we share the principle of monitoring outgoing traffic, but technically our solution cannot be classified as a firewall since we are not implementing any packet filtering policy. Another advantage of the Reverse Firewall is that there is no need to update for detecting new type of attacks. On the other hand, D-WARD detects and discards DDoS attack traffic originating at the network where the mechanism is deployed. D-WARD works by collecting two-way traffic statistics from the border router and comparing these statistics with previously

built legitimate traffic models [1]. When an anomaly is detected, rate-limiting policies are applied. D-WARD distinguishes between the flow traffic and the connection traffic such that legitimate traffic is favoured during an attack. We share two general principles with D-WARD: monitoring two-way connections in order to detect traffic anomalies, and our TCP model is somewhat similar to D-WARD's TCP model which is based on monitoring traffic ratios.

3. Detecting DDoS Attack Traffic

As most Internet traffic is two-way communications (TCP is about 90% of Internet packets [1]), our first step focuses on TCP traffic. TCP provides reliable two-way communication between hosts; communication reliability is accomplished by requesting acknowledgments from the receiver-end, which also controls the amount of data sent by the sender-end so that buffers do not overflow [5]. Due to the TCP flow control mechanism, the difference between received packets and sent packets is expected to be relatively small (since packets are acknowledged). A TCP connection is defined by the source and destination port numbers and by the IP addresses of the source and destination machines [5]; normal TCP connections can be characterized by the ratio of sent packets to and received packets from a given destination [1].

Our approach consists of keeping a table for monitoring TCP traffic ratios in order to detect potential attack traffic. In normal conditions, the lack of acknowledgments could be attributed to network congestions or other related problems; however, during a DDoS attack, no acknowledgments will be sent because the victim will be overwhelmed. In the design phase, our first intention consisted of monitoring each individual TCP connection by keeping a table of TCP traffic ratios in order to detect potential attack traffic. Nevertheless, in the first experimental phase, our approach consisted of monitoring the traffic ratios overall TCP connections. More precisely, we monitored the total number of outgoing packets and incoming traffic regardless of which TCP connection the packets belonged to. By adopting this approach, we are reducing the processing time by each packet, which then will be less costly in terms of computation performance.

During a DDoS attack, the traffic ratio is higher than usual because the number of outgoing packets is greater than the number of incoming packets. Determining a threshold to distinguish legitimate traffic from attack traffic is a key factor in order to avoid a large number of false-positive attack detections. As the distribution of traffic is unknown, raw data is captured (by our first agent prototype) to empirically determine the value of the traffic threshold in which attack detection will be based. This data was complemented by data provided by Cistech Limited; by using Nakina Systems IP Solutions, diverse traffic statistics (from five different sites) were examined; traffic patterns were studied on a monthly, a weekly and a daily basis. Due to space constraints, the raw data is not included in this paper; nevertheless, as an example:

figure 1 depicts http traffic in a week (traffic ratios are calculated per connection).

The acquired raw data also showed that traffic ratio values usually present larger values at the beginning of the run when there are not enough packets to make a decision on whether or not traffic is legitimate. Based on these results, a transient period was introduced in the algorithm in order to not only save time for not computing the traffic ratio, but also to reduce the consumption of resources by classifying traffic during this period. After the transient time is consumed, traffic ratios are calculated every COMPUTE_RATIO time, and if at a given moment, the current traffic ratio is greater than the value of the threshold, an attack might be ongoing and the user will be notified. It should be noted that the transient period is based on the number of total packets instead of time because the time needed to reach the number of packets, to avoid false-positives at the beginning of the run (when the total number of packets is not sufficient to detect whether or not traffic is legitimate), depends on the user's behaviour..

In general, testing defence mechanisms to prevent DDoS attacks in a realistic environment is difficult to achieve. For the purposes of this research, detecting DDoS attack packets in the agent machines, a large-scale network is not necessary because results could be measured in terms of the number of attack packets needed to be sent by each agent causing the traffic ratio to exceed the value of the predetermined threshold. Consequently, an attack tool was developed to generate DDoS attack packets at different rates. The next section presents the results.

3.1. Preliminary Results

As briefly mentioned in the previous section, raw data was captured by a first agent prototype in order to calculate the average traffic ratio overall connections. Based on the raw data acquired to measure traffic ratios, the value of the threshold should be chosen from 1.5 to 2.5; for illustration purposes, figures 2 shows the traffic ratios per connections, and figure 3 illustrates the overall traffic ratio per total number of packets. Different threshold values were tested versus distinct attack rates. Table 1 summarizes the most relevant experiment settings and its results.

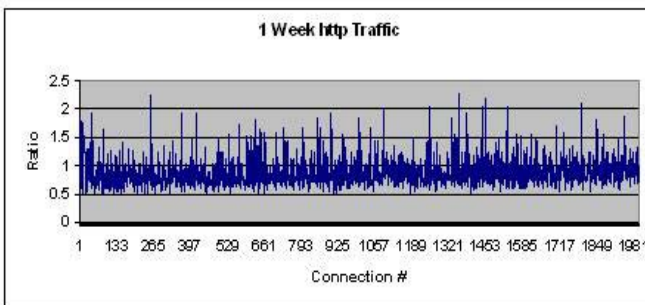


Figure 1. HTTP Traffic Ratio over 1 Week

Threshold	Attack Rate [p/s]	Attack Duration [s]	Attack Detection [time, s]
2.0	100	30	-
1.5	100	30	11.423
1.3	100	30	8.4635

Table 1. Experiment Settings and Results [Average]

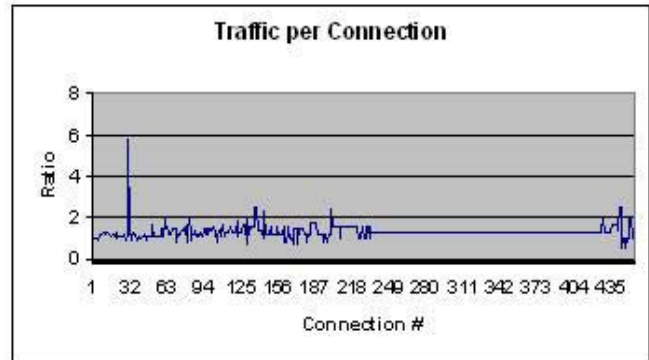


Figure 2. Traffic Ratio per Connection

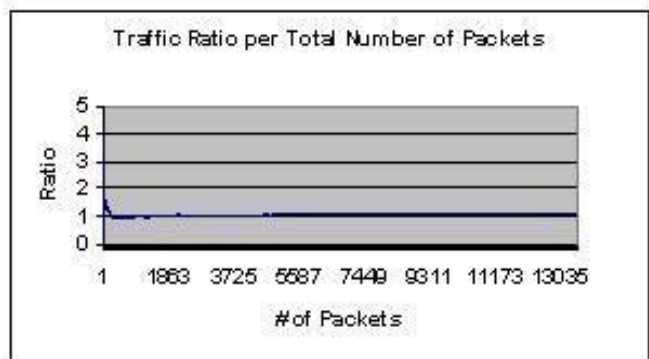


Figure 3. Overall Traffic Ratio per Total Number of Packets

Attack detection is possible depending on the duration of the attack as well as its packet rate; a higher attack rate will allow for faster detection, but higher attack rates were not tested because attacks with low rate transmission packets are the main challenge when detecting attack traffic at the agent machines. While a lower threshold will allow earlier attack detection, it will also increase the amount of false-positives. An alternative to improve attack detection will be to determine the value of a threshold based on a user's behaviour. To prevent an attacker from training the agent, attack detection will not only be based on the threshold determined by the agent, but will also be based on an upper bound threshold established by previously built traffic models.

On the other hand, if the user does not take action after the detection of an attack, subsequent traffic will still be classified as an attack until the long-term average absorbs the effect of the attack on the traffic ratio. For example, figure 4 depicts the traffic ratio with a DDoS attack (starting at packet

4798 - ending at packet 9024) and with no action taken by the user. Our next goal is to include intelligence in the agent such that the value of threshold is adjusted based on the user's behaviour.

4. Conclusions and Future Work

Although results are promising, more research must be conducted. It is necessary to develop a more flexible attack tool to test the agent under several different attack scenarios such as pulsing attacks and higher rate attacks with a short duration. DDoS attacks with low rate transmission packets are probably the biggest challenges in detecting attack traffic at the agent machines. To improve detection of these type of attacks, traffic ratios for different types of protocols must be examined in greater detail.

[5] Internet Task Force, "Transmission Control Protocol, RFC793," [online] 1981, <http://www.ietf.org/rfc/rfc0793.txt?number=793> (Accessed: 15 June 2005)

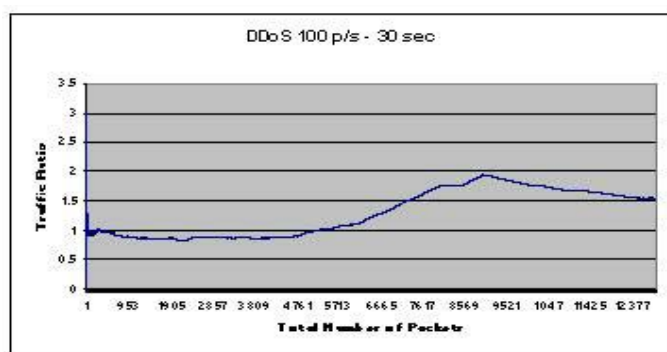


Figure 4. Overall Traffic Ratio per Total Number of Packets

Acknowledgements

The authors wish to thank Brian J. Stacey, Cistel Technology Inc., for the valuable comments and feedback during the first experimental phase of this project.

References

- [1] J. Mirkovic, "D-WARD: Source-End Defense Against Distributed Denial-of-Service Attacks," Ph.D. dissertation, University of California, Los Angeles, 2003. <http://www.lasr.cs.ucla.edu/ddos/dward-thesis.pdf>
- [2] David Moore and Colleen Shannon, "The Spread of the Code-Red worm (CRv2)," (Cooperative Association for Internet Data Analysis (CAIDA): analysis: security: code red), [online] 30 July 2001, http://www.caida.org/analysis/security/code-red/coderedv2_analysis.xml (Accessed: 03 June 2005)
- [3] Symantec, "Symantec Internet Security Threat Report, Trends for July 04–December 04 Volume VII," [online] March 2005, http://www.symantec.com/region/se/sepress/download/istr_no7.pdf (Accessed: 05 June 2005)
- [4] The Reverse Firewall: Defeating DDoS Attacks, Cs3, [online] 2003, <http://www.cs3-inc.com/rfw.html> (Accessed: 09 May 2005)