A Framework for Classifying Denial of Service Attacks*

Alefiya Hussain John Heidemann Christos Papadopoulos
USC/Information Sciences Institute
{hussain,johnh,christos}@isi.edu

ABSTRACT

Launching a denial of service (DoS) attack is trivial, but detection and response is a painfully slow and often a manual process. Automatic classification of attacks as single- or multi-source can help focus a response, but current packet-header-based approaches are susceptible to spoofing. This paper introduces a framework for classifying DoS attacks based on header content, transient ramp-up behavior and novel techniques such as spectral analysis. Although headers are easily forged, we show that characteristics of attack ramp-up and attack spectrum are more difficult to spoof. To evaluate our framework we monitored access links of a regional ISP detecting 80 live attacks. Header analysis identified the number of attackers in 67 attacks, while the remaining 13 attacks were classified based on ramp-up and spectral analysis. We validate our results through monitoring at a second site, controlled experiments, and simulation. We use experiments and simulation to understand the underlying reasons for the characteristics observed. In addition to helping understand attack dynamics, classification mechanisms such as ours are important for the development of realistic models of DoS traffic, can be packaged as an automated tool to aid in rapid response to attacks, and can also be used to estimate the level of DoS activity on the Internet.

Categories and Subject Descriptors

C.2.0 [**COMPUTER-COMMUNICATION NETWORKS**]: General–*Security and Protection* G.3 [**PROBABILITY AND STATISTICS**]: Time series Analysis

General Terms

Measurement, Security

Keywords

Security, Measurement, Denial of Service Attacks, Time Series Analysis.

*This material is based upon work supported by DARPA via the Space and Naval Warfare Systems Center San Diego under Contract No. N66001-00-C-8066 ("SAMAN"), by NSF under grant number ANI-9986208 ("CONSER"), by DARPA via the Fault Tolerant Networks program under grant number N66001-01-1-8939 ("COSSACK") and by Los Alamos National Laboratory under grant number 53272-001.

SIGCOMM'03, August 25–29, 2003, Karlsruhe, Germany.

1. INTRODUCTION

The Internet connects hundreds of millions of computers across the world running on multiple hardware and software platforms. It serves uncountable personal and professional needs for people and corporations. However, this interconnectivity among computers also enables malicious users to misuse resources and mount denial of service (DoS) attacks against arbitrary sites.

In a denial of service attack, a malicious user exploits the connectivity of the Internet to cripple the services offered by a victim site, often simply by *flooding* a victim with many requests. A DoS attack can be either a *single-source* attack, originating at only one host, or a *multi-source*, where multiple hosts coordinate to flood the victim with a barrage of attack packets. The latter is called a distributed denial of service (DDoS) attack. Sophisticated attack tools that automate the procedure of compromising hosts and launching attacks are readily available on the Internet, and detailed instructions allow even an amateur to use them effectively.

Denial of service attacks cause significant financial damage every year, making it essential to devise techniques to detect and respond to attacks quickly. Development of effective response techniques requires intimate knowledge of attack dynamics, yet little information about attacks in the wild is published in the research community. Moore et al provide insight into the prevalence of DoS activity on the Internet [24], but their analysis is based on back-scatter packets and lacks the level of detail required to study attack dynamics or generate high-fidelity models needed for DoS research. Monitoring tools today can detect an attack and identify basic properties such as traffic rates and packet types. However, because attackers can forge most packet information, characterizing attacks as single- or multi-source and identifying the number of attackers is difficult.

In this paper, we develop a framework to classify attacks based on header analysis, ramp-up behavior and spectral analysis. First, we analyze the header content to get a rapid characterization of the attackers. Since headers can be forged by the attacker, we develop two new techniques to analyze packet stream dynamics using the ramp-up behavior and the spectral characteristics of the attack traffic. The absence of an initial ramp-up suggests a single attacker, whereas a slow ramp-up (several hundred milliseconds or more) suggests a multi-source attack. Since ramp-up is also easily spoofed, we identify spectral characteristics that distinguish single- from multi-source attacks and show that attackers cannot easily spoof spectral content without reducing attack effectiveness. We describe the algorithms used in our framework in Section 4 and discuss robustness to counter-measures in Section 7.

The contribution of this paper is an automated methodology for characterizing DoS attacks that adds new techniques of ramp-up and spectral analysis, building on the existing approach of header analysis. In addition to providing a better understanding of DoS attack dynamics, our work has several direct applications. This identification framework can be used as part of an automated DoS detection and response system. It can provide the classification component of a real-time attack analysis system to aid network ad-

ministrators in selecting an appropriate response depending on the type of ongoing DoS attack. For example, if an attack consists of only a single source using traceback to identify the culprit is trivial, but as the number of attackers increase traceback becomes rapidly intractable. Thus one application of our framework is to judiciously decide if activation of traceback is appropriate during a particular attack. This analysis can also be used to create and validate models of DoS and DDoS attacks for simulation and experimentation. Finally, long-term automated measurements of DoS attacks can be used to estimate the level of DoS attack activity in the Internet. We describe these applications in Section 8.

We evaluated our framework on traffic collected from two peering links at Los Nettos, a regional ISP in Los Angeles. Over a period of five months we observed and analyzed 80 attacks. We could classify 67 attacks as single- or multi-source with header analysis; the remaining 13 attacks were classified based on ramp-up and spectral behavior. We validate our algorithm and conclusions in three ways. First, we monitor a second site at University of Southern California and compare the observed attack dynamics. Second, to understand the spectral characteristics of attacks we conduct a series of experiments with synthetically generated attack traffic sent over a wide-area network and with real attack traffic generated using attack tools on an isolated testbed. Finally, we use simple numerical simulations to improve and confirm our understanding of the underlying causes for differences in spectral behavior. Our validation methodology is detailed in Section 6.

2. RELATED WORK

Denial of service attacks attempt to exhaust or disable access to resources at the victim. These resources are either network bandwidth, computing power, or operating system data structures. Research on denial of service attacks is primarily focused on attack detection and response mechanisms. Attack detection identifies an ongoing attack using either anomaly-detection [13, 25, 38] or signature-scan techniques [28, 30]. Most response mechanisms attempt to alleviate the damage caused by the attack by taking reactive measures like reducing the intensity of the attack by blocking attack packets [17, 21, 25], or localizing the source of the attack using traceback techniques [3, 8, 31, 32, 33, 35]. Besides the reactive techniques discussed above, some systems take proactive measures to discourage DoS activity. For example, distributed packet filtering [26] blocks spoofed packets using local routing information and SOS [19] uses overlay techniques with selective re-routing to prevent large flooding attacks . In this paper, we use a simple anomaly-detection technique to identify attacks and focus on a classification mechanism to understand attack dynamics.

Beside attack detection and response mechanisms, it is important to understand DoS attack prevalence and attack dynamics on the Internet. Moore et al used backscatter analysis and detected 12,805 attacks during a period of 3 weeks [24]. The backscatter technique allows detection of attacks that uniformly spoof source addresses in the complete IP address space. Many attack tools use reflection techniques, subnet spoofing, or do not spoof source addresses [14, 29]. The backscatter technique will not detect these attacks. In this paper, we develop an alternate approach where we extrapolate the attack activity observed at Los Nettos to the Internet (Section 8.3).

Signal processing technique have been used previously to analyze malicious network traffic and to detect ongoing attacks. Cheng et al use spectral analysis to detect high volume DoS attack due to change in periodicities in the aggregate traffic [9] while Barford et al use flow-level information to identify frequency characteristics of DoS attacks and other anomalous network traffic [2]. Further, wavelets and other signal processing techniques have been extensively used to analyze both wired and wireless network traffic [7, 27, 39]. In this paper we analyze the spectral behavior of the attack stream to provide information regarding the presence of multiple attackers.

3. ATTACK TAXONOMY

To launch a DDoS attack, a malicious user first compromises Internet hosts by exploiting security holes, many of which are openly disclosed by software vendors. Subsequently, the malicious user installs attack tools on the compromised host (also known as a *zombie*), making it available to attack any victim on command. With full control on the zombie the attacker can construct any packet including illegal packets, such as packets with incorrect checksums, incorrect header field values, or an invalid combination of flags.

The different types of denial of service attacks can be broadly classified into *software exploits* and *flooding attacks*. In software exploits the attacker sends a few packets to exercise specific software bugs within the target's OS or application, disabling the victim. In flooding attacks, one or more attackers sending incessant streams of packets aimed at overwhelming link bandwidth or computing resources at the victim. Although software-exploit attacks are important, this paper focuses on flooding attacks, since they cannot be addressed by software fixes.

Based on the location of the observation point we classify flooding attacks as single-source when a single zombie is observed flooding the victim and as direct multi-source when multiple zombies are observed, as shown in Figure 1(b). It is difficult to distinguish between single- and multi-source attacks by observing only source addresses since most attacks spoof the source address. In both cases there may be additional zombies present that are not discernible from our observation point. Therefore an attack classified as single-source may potentially contain multiple zombies when observed at the victim. Zombies are usually insecure machines that have been compromised by a malicious user. Multiple attackers may be summoned for an attack to increase firepower, or to evade detection.

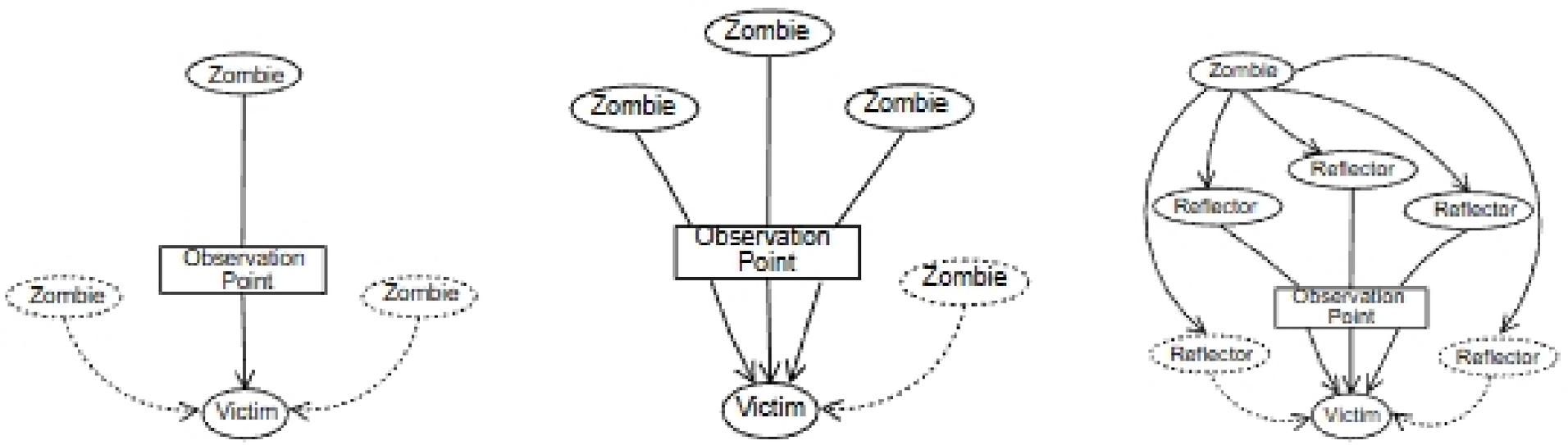
Reflector attacks (Figure 1(c)) are a special case of multi-source attacks. Such attacks are used to hide the identity of the attacker, or to amplify an attack [29]. A reflector is any host that responds to requests, for example a web server that responds to TCP SYN requests with a SYN-ACK reply, or any host that respond to ICMP echo requests with ICMP echo replies. Any host can be used as a reflector by spoofing the the victim's IP address in the source field of the request, tricking the reflector into directing its response to the victim. Reflectors can also be used as amplifiers by sending packets to the broadcast address on the reflector network, soliciting a response from every host on the LAN. Unlike zombies that represent improperly secured hosts, reflectors are typically legitimate hosts providing Internet services, making reflector attacks more difficult to eradicate.

4. ATTACK CLASSIFICATION

Our framework classifies attacks using header contents, transient ramp-up behavior, and spectral characteristics. This three-pronged approach is necessary to deal with an increasing level of difficulty in classifying attacks depending on the level of IP header spoofing present in an attack.

4.1 Header Contents

Most attacks spoof the source address concealing the number of attackers. However, other header fields, such as the fragment identification field (ID) and time-to-live field (TTL), can be indirectly interpreted to provide hints regarding the number of attackers. Such



(a) Single-source (b) Multi-source (c) Reflector

Figure 1: Flooding attacks are classified as (a) single-source, (b) multi-source, or (c) reflected based on the number of attackers and their location, with respect to the observation point and victim.

Let P ={attack packets}, $P_i \subset P, P = \bigcup_{i=2}^{n} P_i$
If $\forall\, p \in P$
    ID value increases monotonically **and**
    TTL value remains constant
    then Single-source
elseif $\forall\, p \in P_i$
        ID value increases monotonically **and**
        TTL value remains constant
        then Multi-source with n attackers
else **Unclassified**

Figure 2: Pseudo code to identify number of attackers based on header content.

techniques have been used before to identify multiple interfaces on routers [34] and count number of hosts behind a NAT box [4]. These techniques work because many operating systems sequentially increment the ID field for each successive packet. As a result, all packets generated by the same host will contain monotonically increasing ID values. In addition, assuming the routes remain relatively stable during the attack, the TTL value will remain constant for the same source-destination pair. Thus for attacks where the ID and TTL fields are not forged we use the algorithm outlined in Figure 2 to estimate the number of attackers and classify attacks as single- or multi-source.

We estimate the number of attackers by counting the number of distinct ID sequences present in the attack. Packets are classified as belonging to the same sequence if their ID values are separated by less than *idgap* (we use an *idgap* of 16) and the TTL value remains constant for all packets. We allow for some separation in idgap to tolerate moderate packet reordering. In high volume attacks the ID value typically wraps around within a second. Therefore using a small idgap also limits collisions during sequence identification. If a packet does not belong to an existing sequence, it forms the beginning of a new sequence. In most cases, attack packets arrive close to each other and have a idgap of one. An attack sequence must consist of at least 100 packets to identify a distinct attacker.

Some attacks have short silence periods during the attack. After a silence period, packets may form a new attack sequence that should be considered as a continuation of an old sequence, but would not be identified as such due to the strict idgap. To bridge these silence periods we coalesce such streams into one stream if they are within 500ms of each other. Finally, since many operating systems do not send the ID value in network byte order, we infer byte-order from the first 10 packets observed.

Many attack tools spoof the source IP address but allow the operating system to fill in default values for other fields [10]. These tools are susceptible to ID analysis. We are not aware of any attack tools that attempt to coordinate the ID field over a distributed set of attackers. In fact, differences in RTT and available bandwidth make it inherently difficult to coordinate packet streams from multiple hosts such that their ID fields consistently arrive in order without reducing the rate (and hence effectiveness) of the attack.

Some attack tools forge all header contents, including both the ID and the TTL field. For such attacks it is impossible to distinguish between a single or multiple sources based on header information alone, making it essential to use additional techniques.

4.2 Ramp-up Behavior

In a multi-source attack, a master typically activates a large number of zombies by sending a trigger message that either activates the zombies immediately or at some later time. When observed near the victim, this distributed activation of zombies results in a *ramp-up* of the attack intensity due to the variation in path latency between the master and the zombies and weak synchronization of local clocks at the zombies. In contrast, single-source attacks do not exhibit a ramp-up behavior and typically begin their attack at full strength. Thus, the presence of a ramp-up provides a hint as to whether the attack is single- or multi-source. This method cannot robustly identify single-source attacks since an intelligent attacker could create an artificial ramp-up from a single site. To our knowledge, current attack tools do not attempt to do so.

4.3 Spectral Analysis

A more robust method for classifying attacks as single- or multi-source is to consider their spectral characteristics. We observed attack streams have markedly different spectral content that varies depending on the number of attackers. In this section, we present our methodology for analyzing the spectral characteristics of an attack stream; in Section 5.5 we present several examples with intuition why it works.

Spectral analysis requires treating the packet trace as a time series. We divide the attack stream into 30 second segments, defining $x(t), 0 \leq t < 30,000$ as the number of attack packet arrivals in each 1ms interval. Since non-stationarity can taint spectral analysis, we discard segments that show initial ramp-up or abrupt changes (perhaps due to a change in number of attackers). We use