A Framework for Classifying Denial of Service Attacks *

Alefiya Hussain, John Heidemann, and Christos Papadopoulos
ISI-TR-2003-569
Date: 25 Feb 2003
{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 response, but current packet-header-based approaches are susceptible to spoofing. This paper introduces a framework for classifying DoS attacks based on header content, ramp-up behavior, and novel techniques based on spectral analysis. Although headers are easily forged, we show that characteristics of ramp-up and the attack spectrum are much more difficult to spoof. To evaluate our framework we monitored access links of a regional ISP over a period of five months, 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 over the Internet, and simulation. We use experiment 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, and can be packaged as an automated tool to aid in rapid response to attacks. Finally, we use our attack observations to estimate the level of DoS activity on the Internet.

1. INTRODUCTION

The Internet connects hundreds of millions of computers across the world running on multiple hardware and software platforms [28]. It serves uncountable personal and professional needs for millions of people and corporations. However, 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 currently published in the research community. Moore et al provide insight into the prevalence of DoS activity on the Internet, but their analysis is based on back-scatter packets and lacks the level of detail required to generate high-fidelity models needed for DoS research [27]. Monitoring tools today can detect an attack and identify basic properties of an attack, 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, as others have done, we analyze header content to get a rapid characterization of attackers. Since headers can be forged by the attacker, we develop new techniques to analyze packet stream dynamics using 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 (but does not prove) 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 in Section 7.

The contribution of this paper is an automated methodology for analyzing DoS attacks that is based on ramp-up and spectral analysis to build upon existing approaches of header analysis. In addition to providing a better understanding of DDoS attack dynamics, our work has several direct applications. This identification framework can be used as part of an automated DDoS detection and response system. It can provide the classification component of a real-time attack analysis system to aid network administrators in selecting appropriate responses depending on the type of ongoing DoS 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 amount of DoS attack activity in the Internet. We describe these applications and our estimate of attack activity in Section 8.

We tested our algorithms on traffic collected from two peering links at a moderate size regional ISP. Over a period of five months we observed 80 attacks, which were analyzed to develop our framework. Header analysis provided significant insight allowing us to classify 67 attacks as either single- or multi-source. Using ramp-up

behavior and spectral analysis, we developed spectral signatures for the classified attacks, which were subsequently used to classify the remaining 13 attacks. We validate our algorithm and conclusions in three ways. First, we monitor a second site at a major university and compare attack observed there. Second, to understand the spectral characteristics of attacks, we analyze synthetically generated attack traffic sent over a wide-area network and compare it to traffic from real attack tools on a testbed. Finally, we use simple numerical simulations to improve and confirm our understanding of the physical causes for differences in attack characteristics.

2. RELATED WORK

Denial of service attacks attempt to exhaust the resources at the victim. These resources are either network bandwidth, computing power, or operating system data structures. Research on denial of service attacks is focused on either attack detection mechanisms to identify an ongoing attack [10, 14, 30, 38, 43] or response mechanisms that attempt to alleviate the damage caused by the attack. Response mechanism usually take two approaches; localizing the source of the attack using traceback techniques [8, 16, 34, 35], or reducing the intensity of the attack [22, 18, 44] by blocking attack packets. Besides the reactive techniques discussed above, some systems take proactive measures to discourage DoS activity. Both, CenterTrack [38] and SOS [20] use overlay techniques with selective rerouting to prevent large flooding attacks. This paper presents a unique framework to identify single- and multi-source attacks based on spectral content of the attack and does not have the coordination and infrastructure requirements imposed by other techniques.

Many techniques have been proposed to detect an ongoing DoS attack. Cisco routers provide support for attack detection via RMON [40] and Netflow [39] data, that can be processed offline to detect an attack. Multops exploits the correlation of incoming and outgoing packet rates at different level of subnet prefix aggregation to identify attacks [14]. Wang provides a rigorous statistical model to detect abrupt changes in the number of TCP SYN packets as compared to the TCP SYN ACK packets [43]. Bro, an intrusion detection system uses change in (statistical) normal behavior of applications and protocols to detect attacks [30] while Cheng use spectral analysis to detect high volume DoS attack due to change in periodicities in the aggregate traffic [10]. All the above techniques are based on *anomaly-detection* which is faster than static *signature-scan* techniques used by Snort [32]. Snort has one main disadvantage; new attacks that do not have well-defined signatures may go undetected until the signature is defined. In this paper, we use a simple anomaly-detection technique that tracks the number of source connecting to a single destination. Traffic is flagged as an attack if there is an abnormally high number of source addresses connecting to a single destination address.

Response to an attack consists of localizing the attackers and reducing the intensity of the attack. The SPIE system can traceback individual packets within a domain using packet digests [35]. On the other hand, Burch and Cheswick propose a technique to traceback to the source by flooding routes to the victim and observing change in the attack rates [8]. IP Traceback [34, 11, 36] and ICMP traceback [4] provide mechanisms to identify the source of the attack using packet marking at routers. Most of these mechanisms require large scale deployment over the Internet to be effective and as the number of attackers increase, the number of packets and computational time required to identify the attacker increases drastically (SPIE is the exception). In this paper we propose a framework to identify the presence of single- or multi-sources in an attack based on local attack stream information. If an attack
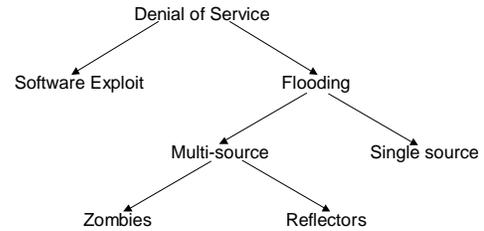
Figure 1: Classification of DoS attacks based on volume of packets and number of attackers. In this paper we analyze flooding attacks.

consists of only a single attackers, using traceback to identify the culprits is trivial, but as the number of attackers increase traceback becomes rapidly intractable. Thus the additional information provided by our framework can be used to judiciously decide the response mechanism.

To reduce the intensity of an attack, Mahajan et al propose an aggregate congestion control and pushback technique to identify and throttle the attack flows [22]. Pushback is a cooperative technique that allows routers to block an aggregate upstream. On the other hand, D-WARD uses TCP-based rate control at the first hop to prevent attackers from participating in an attack [18]. Packet filters are the best line of defense during an attack [44]. Filtering decisions are typically based on source and destination addresses, port numbers or packet contents. Once an attack is classified as single or multi-source using the proposed framework, network operators can strategically deploy packet filters to block the attack packets.

Beside attack detection and response mechanisms, it is important to understand DoS attack prevalence and the attack dynamics on the Internet. Moore et al used backscatter analysis and detected 12,805 attacks during a period of 3 weeks [27]. The backscatter technique allows detection of attacks that uniformly spoof source addresses in the complete IP addresses space. Many attack tools use reflection techniques, subnet spoofing, or do not spoof source addresses [15, 31]. The backscatter technique will not detect these attacks. Much work needs to be done to understand DoS dynamics to formulate correct DoS models for simulation and testbed experiments. Barford et al use flow-level information to identify frequency characteristics of DoS attacks and other anomalous network traffic [3]. They develop a network anomaly detection mechanism based on time-series and wavelet analysis. In this paper we attempt to characterize the behavior of different attacks based on their header content, transient behavior and spectral content. We discuss applications of our classification framework later in the paper.

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. The malicious user then installs attack tools on the compromised host (also known as a *zombie*), that now becomes available to attack any victim on command. With full control of 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.
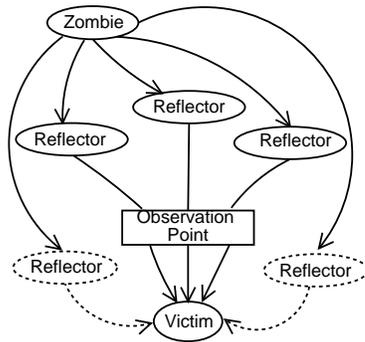
The different types of denial of service attacks can be broadly classified into *software exploits* and *flooding attacks*. Flooding attacks can be further classified into single- and multi-source attacks based on the number of attackers. This classification is depicted in Figure 1 and explained next.
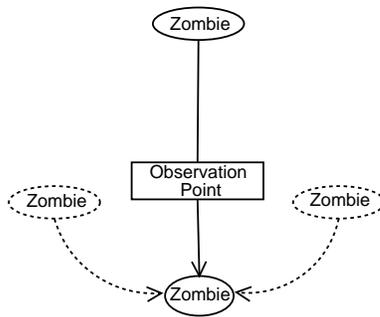
(a) Single-source
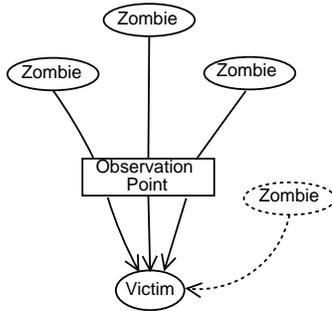
(b) Multi-source

(c) Reflector

Figure 2: 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.

3.1 Software Exploits

These attacks exploit specific software bugs in the operating system or an application, and can potentially disable the victim machine with a single or a few packets. A well known example is the *ping of death*, that causes the operating system to crash by sending a single large ICMP echo packet. Similarly, the *land attack* sends a single TCP SYN packet containing the victim's IP address in both the source and destination address fields, resulting in an endless loop in the protocol stack. Such attacks can only be prevented by diligently applying software updates. While important,

such attacks are beyond the scope of this paper.

3.2 Flooding attacks

Flooding attacks are the result of one or more attackers sending incessant streams of packets aimed at overwhelming link bandwidth or computing resources at the victim. Based on the location of the observation point, we classify flooding attacks as single-source attacks when a single zombie is observed flooding the victim (although there may be more zombies), and as multi-source when multiple zombies are observed, as shown in Figure 2(a) and Figure 2(b) respectively. Multiple attackers may be summoned for an attack to increase firepower, or to evade detection. In both attack classes, the master can installs attack tools on the host machine that can generate illegal packets. Examples include the TCP NULL attack that generates packets with no flags set, the Xmas attack that has all TCP flags set, and attacks that use packets with a non-existent IP protocol number [2]. Several canned attack tools are available on the Internet, such as Stacheldraht, Trinoo, Tribal Flood Network 2000, and Mstream that generate flooding attacks using a combination of TCP, UDP, and ICMP packets [12]

A significant percentage of captured attacks consist of a single source. Moore et al detected 14% of all DoS attacks were directed toward home machines using either dial-up or broadband access [27]. CERT also reports most DoS attacks on the Internet are from a single source to a single victim [15]. Thus, a single high bandwidth zombie can potentially generate enough packets to overwhelm a victim.

The third type of attack is the *reflector* attack (Figure 2(c)). Such attacks are used to hide the identity of the attacker and/or to amplify an attack [31]. A reflector is any host that responds to requests, such as web servers or ftp servers, that respond to TCP SYN requests with a TCP SYN-ACK packets, or hosts that respond to ICMP echo requests with ICMP echo replies. Servers may be used as reflectors by spoofing the victim's IP address in the source field of the request, tricking the reflector into directing its response to the victim. Unlike direct zombie attacks, reflector attacks require well-formed packets to solicit a reply. If many reflector machines are employed, such an attack can easily overwhelm the victim without adversely affecting the reflectors or triggering the local IDS. 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 which represent improperly secured hosts, reflectors are often hosts intentionally providing Internet services, and so reflector attacks may be more difficult to prevent.

4. ATTACK CLASSIFICATION

Our framework classifies attacks using (a) header contents, (b) transient ramp-up behavior, and (c) 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. If the source address in the attack packets is not spoofed, classifying an attack as single- or multi-source becomes a simple matter of counting the distinct sources present in the attack stream. When the source address is spoofed, we must look at other header fields (such as ID and TTL) for clues. Finally, when the entire IP header is spoofed, we resort to ramp-up and spectral analysis for classification. Next, we describe these stages in more detail.

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

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

interpreted to provide hints regarding the number of attackers. Such techniques have been used before to identify multiple interfaces on routers [37] and count number of hosts behind a NAT box [5]. This technique works 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, the TTL value provides further hints because it remains 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 3 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 gap 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.

Current attack tools like Stacheldraht and variants of TFN2K spoof the source IP address but allow the operating system to fill in its default values for other fields [12]. Such 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, it is inherently difficult to coordinate packet streams from multiple hosts such that their ID fields consistently arrive in order without reducing the 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 the techniques described next.

4.2 Ramp-up Behavior

In a multi-source attack, a master typically activates a large num-
ber of zombies by sending a trigger message that either activates the zombies immediately or at some later time. When observed near the victim, both activation processes will result in a ramp-up of the attack intensity due to the variation in path latency between the master and the zombies, coupled with 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 a single- or multi-source attack. This method, however, cannot robustly identify single-source attacks, since an intelligent attacker could create an artificial ramp-up. 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 have observed attack streams exhibit markedly different signatures that vary 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.

Spectral analysis requires treating the packet trace as a time series. We divide the attack stream into 30 second intervals and define $x(t), 0 \leq t < 30$ as the number of attack packet arrivals in each 1ms interval. To avoid initial ramp-up and abrupt changes within the attack stream (e.g. due to a change in number of attackers), we use linear least-square regression to compute the slope and verify that the difference between the slope and zero is statistically insignificant within a 95% confidence interval [23]. Further, we condition $x(t)$ by subtracting the mean arrival rate before proceeding with spectral analysis. The mean value results in a large DC component in the spectrum that does not provide any useful information for our classification framework.

We use the Bartlett window method to compute the frequency spectrum by performing a discrete-time Fourier transform on the autocorrelation function (ACF) of the attack stream. The autocorrelation of an attack stream is a measure of how similar the attack is to itself shifted in time by offset k [6, 7]. When $k = 0$ we compare the attack stream to itself, and the autocorrelation is maximum and equal to the variance of the attack stream. When $k > 0$ we compare the attack stream with a version of itself shifted by lag k. The autocorrelation sequence $r(k)$ at lag k is

$$c(k) = 1/N \sum_{t=0}^{N-k} (x(t) - \bar{x})(x(t+k) - \bar{x}); \qquad (1)$$

$$r(k) = c(k)/c(0) \qquad (2)$$

where $\bar{x}$ is the expected value and N is the length of the attack stream $x(t)$. The spectrum $S(f)$ of attack obtained by the discrete-time Fourier transform of the autocorrelation sequence of length M as given below:

$$S(f) = \sum_{k=0}^{M} r(k)e^{-\imath f k} \qquad (3)$$

The highest frequency observable in the spectrum $S(f)$ is 500Hz (the Fourier transform is a symmetric function). Intuitively, the spectrum captures the *power* the attack stream contains at a particular frequency.

Once we generate the spectrum we need a technique to compare the spectral characteristics of different attacks. Therefore, for each attack we define the cumulative spectrum $P(f)$ as the amount of
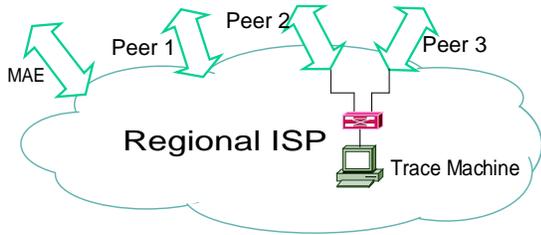
Figure 4: The trace machine monitors two of the four peering links at Los Nettos.

power in the range 0 to f. We then normalize this value by the total power to get the normalized cumulative spectrum (NCS), $C(f)$ [9, 7]. Finally, we define $F(p)$ as the frequency at which the NCS captures $p\%$ of the power. Formally:

$$P(f) = \sum_{i=0}^{f-1} \frac{(S(i) + S(i+1))}{2}; \qquad (4)$$

$$C(f) = \frac{P(f)}{P(f_{max})}; \qquad (5)$$

$$F(p) = \min_{C(f) \geq p} f \qquad (6)$$

Attacks are classified as single- or multi-source based on $F(60\%)$. The sixty percent cut-off provides an effective technique to detect localization of power in lower frequencies. Our observations indicate single-source attacks have a linear cumulative spectrum due to dominant frequencies spread across the spectrum. This causes $F(60\%)$ to be in the range of 240–296Hz. In contrast, multi-source attacks have localization of power in lower frequencies resulting a $F(60\%)$ in the range of 142–210Hz. In Section 5.5 we show how spectral analysis can be used to robustly classify attacks whose headers are completely forged.

5. EXPERIMENTAL EVALUATION

In this section we present our trace collection infrastructure and our experimental analysis based on attack captured at Los Nettos. Validation results are presented in the next section.

5.1 Packet Trace Infrastructure

We tested our framework (described in Section 4) using attacks captured by our trace infrastructure installed at Los Nettos in Los Angeles [21]. We captured 80 large-scale attacks over a period of five months, from July 2002 to Nov 2003.

Los Nettos has four major peering links with commercial providers. Due to technical reasons, we were able to monitor only two of those links, as shown in Figure 4. Los Nettos has a diverse clientele including academic and commercial customers. The trace machine is an off-the-shelf Intel P4 1.8Ghz, with 1GB of RAM running FreeBSD 4.5. We use a Netgear GA620 1000BT-SX NIC (Tigon II chipset) with a modified driver that supports partial packets transfer from the NIC card to the kernel. Typical daytime observed load is 140Mbps with a mean of 38Kpps. Measurement drops (as reported by tcpdump) were usually below 0.04% during normal operation, rising to 0.6% during attacks that reached 100Kpps.

We continuously capture packet headers using tcpdump [17], creating a trace file every two minutes. Each trace is then post-processed and flagged as containing a potential attack if either of

Attack Class	# Attacks	Range in pps	Range in Kbps
Single-source	37	680–1360	640–2600
Multi-source	10	16600–84000	13000–46000
Reflected	20	1300–3700	1700–3000
Unclassified	13	550–33500	1600-16000

Table 1: Number of attacks in each class based on header analysis

Protocol	Packet Type	Attack Class			
		S	M	R	U
TCP	SYN	2	3 (2)	-	7 (5)
	ACK	5	2 (2)	-	3 (2)
	SYN-ACK	9	-	4	-
	no flags	15	1 (1)	-	-
	unusual	5	1	-	-
	state exploit	2	-	-	-
ICMP	echo request	5	-	-	-
	echo reply	1	-	16 (3)	-
	invalid	-	-	-	1 (1)
UDP	all	6 (1)	-	-	5 (4)
Other	ip-proto 0	5	-	-	-
	ip-proto 255	-	3	-	-
	fragmented	1	-	-	3 (3)

Table 2: Detailed analysis of packet headers. S indicates single-source, M indicates multi-source, R indicates distributed reflectors, and U indicates unclassified attacks. The number in parenthesis indicates attacks terminating within our ISP while the first number indicates total attacks.

two thresholds are reached: (a) the number sources that talk to the same destination within one second exceeds 60, or (b) the traffic exceeds 40Kpps. Traces that are not flagged are discarded. We identify and ignore known servers that would trigger these thresholds. Finally, we manually verify each flagged trace to confirm the presence of an attack. The automated thresholding works reasonably well but provides a false positive rate of 25–35%. Ongoing attacks that do not trigger our detection mechanism are not identified. We thus miss many small DoS attacks, including some attacks that would incapacitate a dial-up line.

We monitor both inbound and outbound traffic. For attacks terminating in Los Nettos, we capture most of the attack traffic, missing only portions from peering links we do not monitor and from attackers within Los Nettos. For attacks transiting through Los Nettos, our monitoring point may not be exposed to the full intensity of the attack since there may be attackers outside Los Nettos and we do not monitor all external links of Los Nettos. The distinction between transient and terminating attacks becomes more important in Section 8.3

5.2 Classifying Attacks based on Packet Headers

First, we classify attacks based on packet header information alone. As shown in Table 1, we classified all but 13 attacks using this method. Table 2 shows a more detailed breakdown of attacks based on manual analysis with tcpdump [17] and tcpshow [33]. The categories listed in the table are not mutually exclusive since some attack streams carry multiple packet types.

From header analysis we can make several observations about the prevalence of attack techniques in the wild. First, 87% of the

zombie attacks use illegal packet formats or randomize fields, indicating the presence of root access. Use of TCP protocol was most common, with reflection attacks typically exploiting web servers (port 80) and FTP servers (port 21). In Table 2, TCP *no flags* refers to pure data packets with no flags set, while *unusual* refers to attacks that use non-standard (but not always invalid) combinations of TCP flags, such as setting all the flags. *State exploit* refers to attacks that exhaust OS data-structures based on the TCP-state diagram, (e.g. ESTABLISHED, FIN-WAIT1 states) [1]. Even though TCP-SYN attacks belong to this class, we list them separately since they are common. The most aggressive attack was a TCP attack that reached a peak of 98Kpps and generated packets with a combination of TCP flags and options. Many attacks set the type of service (TOS) bits to minimize delay, maximize throughput and increase reliability.

ICMP is the next protocol of choice. The echo reply attack was the most popular reflector attack, since most Internet hosts respond to an echo request packet allowing the attacker to choose from the large number of possible reflectors. One attack consists of 1262 distinct reflectors (based on the source IP address) visible at our observation point. The other ICMP attacks use echo request packet or an *invalid* ICMP code. Several reflector attacks share the same attack signature (for example, identical ICMP sequence number, ID and checksum fields), indicating the use of the same attack tool. Finally, we detected five attacks that use a combination of protocols, such as TCP, ICMP, UDP, and IP proto-0. UDP and other invalid protocols were less frequently used in the attacks.

## 5.3 Classifying Attacks based on Packet Arrival Rate

This section investigates the relation between attack rate and attacker population. We captured attacks with rates ranging from 133pps to 98Kpps. Figure 5 shows the correlation between the attack classes (defined earlier) and attack rate. In Figure 5(a) we show measured attack rates in Kbps and pps in logarithmic scale for each attack. Not surprisingly, single-source attacks are clustered toward lower packet rates whereas multi-source attacks exhibit higher rates, most likely due to aggregation from multiple zombies. In reflection attacks, many reflectors are typically employed to generate high attack aggregates without overloading the reflectors. The captured reflection attacks have a much lower intensity than multi-source attacks since the observation point might not be exposed to the complete intensity of the attack.

To statistically confirm attack rates of single-source, multi-source, and reflected attack have different means, we performed Kruskal-Wallis one-way ANOVA test [23]. We consider the null hypothesis, H_0; there is no relation between the attack rates and attack class. The alternative hypothesis, H_a states there there is a relation between attack rate and class. If H_0 is true, the variance estimate based on within-group variability should be approximately the same as the variance due to between-group variability. This test defines a *F ratio* that evaluates the two variance estimates; if the F ratio is significantly greater than 1, the test is statistically significant, and we can conclude that the means for the three groups are different from each other and reject H_0. It also defines a *p*-value, the probability of observing the sample result assuming H_0 true. Hence a smaller *p*-value provides higher confidence in rejecting H_0. For the data in Figure 5(a), the F ratio is 37.42, indicating a strong relation between the attack rates and the attack classes. Further, the *p*-value is 1.7×10^{-11}, indicating a very low probability of H_0 being correct.

The box plot in Figure 5(b) provides graphical representation of the means of different classes. The lower and upper lines of the box
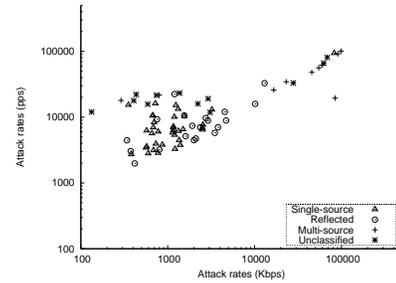
(a) Attack intensity in pps and Kbps
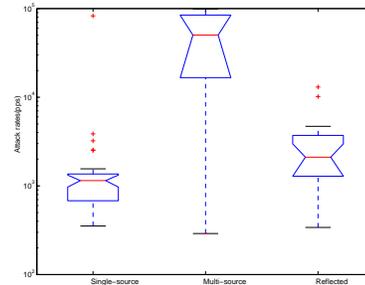
(b) Attack intensity in pps for each class

Figure 5: Correlation of attack rates and number of attackers

are the 25th and 75th percentiles of the sample. The distance between the top and bottom of the box is the interquartile range. The line in the middle of the box is the sample median. The "whiskers" (lines extending above and below the box) show the range of the rest of the sample (unless there are outliers). Single-source attacks have the lowest median while the median and range of the multi-source attacks is significantly higher than single-source and reflected attacks.

Figure 6 shows the cumulative distribution of the attack duration and peak attack rates in packets and Kbps. The peak attack rates vary from about 133pps to 98Kpps. In some attacks packet rates vary over the attack duration. An increase in the attack rate is usually due to addition of new machines or the addition of new type of attack. We also observed a reduction in attack rate, possibly due to withdrawal of sources or installation of filters by network operators.

5.4 Classifying Attacks based on Ramp-up Behavior

To identify the presence of multiple sources when the header is forged we measure the attack's ramp-up behavior (changes in the traffic volume of the attack as a function of time). Single-source attacks typically exhibit no ramp-up, while all multi-source attacks showed ramp-up behavior, ranging from 200ms to 14s.

Figure 7 illustrates the attack ramp-up for two observed attacks. Figure 7(a) shows an attack where packet headers were not forged, and thus the attacker population was visible. The graph shows a three second ramp-up at about 27s as the number of attackers gradually increase to six. The attack reaches a peak rate of 78Kpps with 14 active sources. We observe a total of 40 unique IP ad-
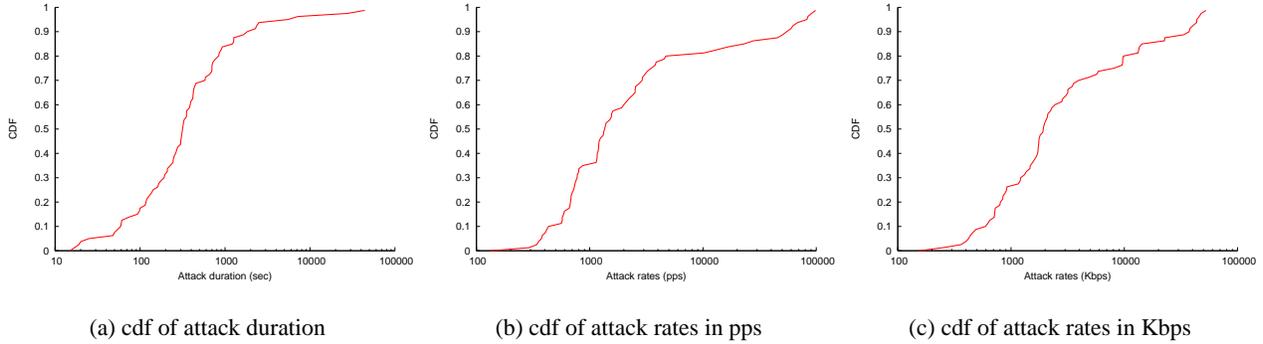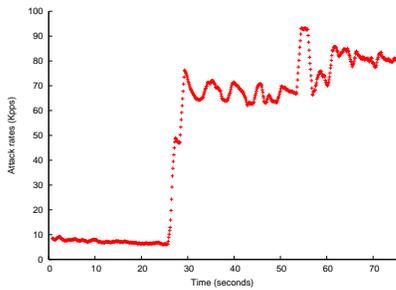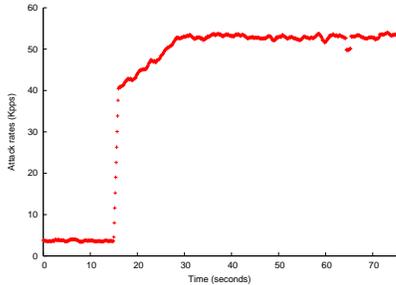
(a) cdf of attack duration   (b) cdf of attack rates in pps   (c) cdf of attack rates in Kbps

Figure 6: The cumulative distribution of (a) attack duration, and attack rates both in (b) pps and (c) Kbps for 80 attacks

(a) Multiple source addresses observed in attack

(b) Subnet spoofed source addresses

Figure 7: Due to lack of synchronization among the zombies, multi-source attacks exhibit initial ramp-up behavior

dresses during the attack, with sources steadily arriving and departing. Figure 7(b) shows an attack where the last eight bits of the source address are forged. The attack is classified as a multi-source attack since it exhibits a ramp-up, rising from 36Kpps to 50Kpps in 14 seconds. In this attack the source addresses and ID field is spoofed, and all packets have the same TTL value, making it difficult to classify the attack based on header content. The presence of transient ramp-up behavior in the first few seconds of the attack strongly suggests the presence of multiple sources. We also verified it is a multi-source attack via spectral analysis.

5.5 Classifying Attacks based on Spectral Analysis

In this section we demonstrate that spectral analysis of the attack time-series (described in Section 4.3) can distinguish between single- and multi-source attacks, even if all headers are spoofed. Because the traffic spectrum is influenced by OS and network behavior we argue that it will be difficult for attackers to easily conceal their spectrum without reducing attack effectiveness. We review this claim more carefully in Sections 6 and 7, for now we present example spectra.

We analyzed the spectral content of all 67 attacks previously classified by header analysis. Based on observations from these known classes, we conclude that single- and multi-source attacks can be distinguished by their spectra:

- Single-source attacks include dominant high frequencies creating a linear trend in the normalized cumulative spectrum.

- Multi-source attacks have dominant low frequencies with a normalized cumulative spectrum that sharply rises at lower frequencies.

Figure 8(a) shows an example of the spectrum of a single-source attack. In this case, the attacker that generates TCP no flag packets at a rate of 1100pps. The source addresses are spoofed, but the ID and TTL values clearly indicate a single-source attack (using analysis from Section 4.1). There are noticeable peaks at higher frequencies in the spectrum and the NCS is linear.

By contrast, Figure 8(b) shows a reflected attack using echo reply packets. Since the source address in reflected attacks is not spoofed, we can count 145 different reflectors located in countries such as Brazil, Japan, Korea, Singapore, and United States. The attack rate is 4300pps. Here we observe concentration of power in lower frequencies creating a corresponding shift in the NCS.

We consider the physical explanations for this shift in spectra in Section 6.3. The intuition behind the result requires consideration of a single attack source and then the interaction of multiple attackers. We suggest that a single attacker sending at full rate will always have high frequency components in the attack traffic because any computer and network interface has a maximum possible transmission rate due to hardware or operating system limits. This rate gives that attacker a basic frequency and harmonics at multiples of that frequency, consistent with our single-source observation. Now consider a collaborative, distributed attack with multiple attackers, each sending as fast as possible Each attacker will have its own maximum rate and corresponding spectra, but in the aggregate,
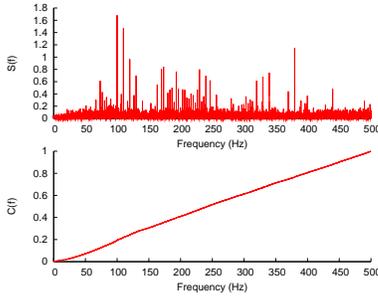
(a) Single-source
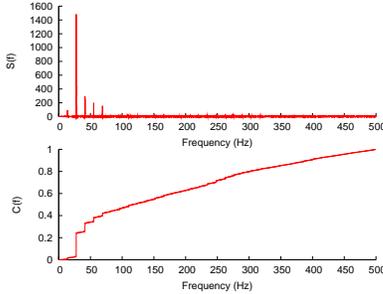
(b) Multi-source

Figure 8: The power spectrum(top) and NCS(bottom) for two example attacks

their traffic will "blur together", loosing dominant high frequency components, because the attackers operate independently at different rates and frequencies, and because each attacker experiences noise from different levels of cross-traffic. In fact, we suggest that it is inherently difficult to coordinate high-rate attackers. We expand on this intuition in several steps: through experiments in Section 6.2, simple simulations in Section 6.3, and discussion about robustness in Section 7.

Since it is difficult to compare the graphical NCS across attacks, we will use the $F(60\%)$ value (from Equation 6) for each attack to detect if the power is concentrated in lower frequencies. Figure 9 plots $F(60\%)$ against the attack rates in pps (log-scale). Single-source attacks are concentrated in the middle frequencies because their linear normalized cumulative spectrum results in mid-range $F(60\%)$ values. Multi-source attacks are concentrated in the lower frequency band, due to the accumulation of power in lower frequencies. The two classes of attacks also have a significant difference in first order statistics: single-source attacks have a mean 268Hz and a 95% confidence interval between 240–295Hz, while multi-source attacks have a mean of 172Hz, and a 95% confidence interval between 142–210Hz. We performed the Wilcoxon rank sum test [23] to verify that the two classes have different $F(60\%)$ ranges. The test strongly rejects the null hypothesis, that single- and multi-source attacks have identical dominant frequencies, with a p-value of 7.7×10^{-5}. We also visually verified it using a box plot.

We use the spectral analysis described above to classify the remaining 13 unclassified attacks. The spectrum of five attacks match spectral characteristics of single-source attacks, with a $F(60\%)$ located above 240Hz. The remaining eight attacks have spectral
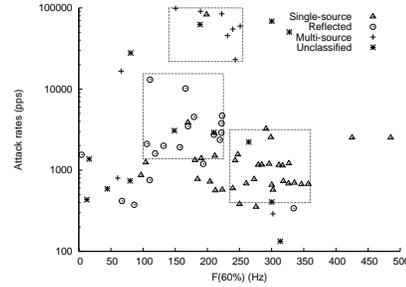
Figure 9: Comparison of $F(60\%)$ against attack rate for each attack class

Protocols	Los Nettos	USC
TCP	84.2%	95.6%
UDP	13.8%	4.10%
ICMP	1.21%	0.118%
other	0.894%	0.175%

Table 3: Percentage of packets observed for each protocol at the two sites

characteristics similar to multi-source attacks. These attacks also exhibit an initial ramp-up lasting from 300ms to 14 seconds that corroborates the presence of multiple attackers.

6. VALIDATION

We use three techniques to validate our classification algorithms and understand the nature of our observations. We measure DoS attacks from a second site to confirm that the numbers and kinds of attacks we identified were not unique to our original observation point. To understand the physical explanations behind our classification techniques we conduct controlled experiments and use simple numerical simulations. These experiments also suggest how robust our methods would be to knowledgeable attackers.

6.1 Observations from an Alternate Site

We deployed a second trace machine at USC's connection to Internet2. Typical daytime load is 112Mbps with a mean of 25Kpps. The traffic mix on the Internet2 link is fairly different than observed at Los Nettos; see Tables 3 for a breakdown of traffic at each site by protocol. Los Nettos shows much more DNS traffic (due to the presence of a root nameserver) and web traffic, while USC shows more "other" traffic due to gaming, files sharing, and research that use atypical or ephemeral ports.

We observed 18 attacks at USC in the months of October and November 2003. Because of differences in monitoring duration and traffic quantity, it is difficult to compare the absolute number of attacks to our observations at Los Nettos, but we observe about the same ratio of attacks in each attack class.

Table 4 lists attacks by class as determined from header content. Three attacks were classified as unknown since they completely randomize the ID value. Table 5 shows a more detailed manual analysis of packet headers. Again, it is difficult to directly compare it to Table 2, but we observe a similar set of attacks. Packet types TCP SYN-ACK, TCP unusual and ICMP illegal were not seen at USC, however some of these attacks were not very frequent at our primary location either.

Ramp-up and spectral analysis of attacks at USC were similar to attacks observed at our original site, and hence we do not reproduce

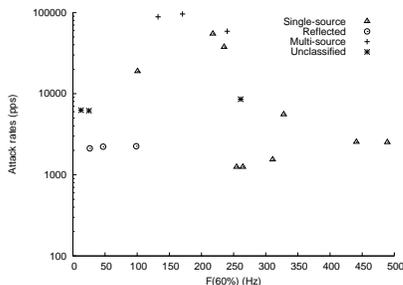| Attack Class	# Attacks	Range in pps	Range in Kbps
Single-source	9	1250–54000	1100–10000
Multi-source	3	58700–95000	28000–72000
Reflected	3	2120-2250	1641–2142
Unclassified	3	6170–8500	2600–6500

Table 4: Number of attacks in each class based on header analysis at USC.

Protocol	Packet Type	Attack Class			
		S	M	R	U
TCP	SYN	-	-	-	2
	ACK	3 (1)	-	-	-
	no flag	5	-	-	-
	unusual	3	-	-	-
	state exploit	-	-	-	1
ICMP	echo request	4	-	-	-
	echo reply	-	-	3	-
UDP	all	5	2 (2)	-	-
Other	ip-proto 0	4	-	-	-
	ip-proto 255	1	1 (1)	-	-
	fragmented	1	-	-	-
	routing	1	-	-	-

Table 5: Detailed analysis of packet headers at USC.

Location	CPU (Mhz)	Hop Count	RTT (ms)
UCSB	1800	9	5
UCSD	500	10	7
UCLA	900	11	2
ISIE	900	15	74
UMass	600	16	90
USC1	1800	6	1
USC2	1800	6	1
USC3	1000	6	1
USC4	900	6	1
USC5	500	6	1

Table 6: The WAN testbed. The top block of hosts are used for distributed experiments while the bottom block of hosts are used for clustered experiments.

spectra of individual attacks here. Figure 10 plots $F(60\%)$ against the attack rate (in log-scale) for each attack class. As expected, $F(60\%)$ is located in the middle frequency band for single-source attacks, and in the low frequency band for multi-source attacks. The two classes of attacks also have first-order statistics similar to the Los Nettos. The mean for single-source attacks is 292Hz and a 95% confidence interval between 202–382Hz, while multi-source attacks have a mean of 120Hz and a 95% confidence interval between 35Hz–202Hz. One unknown attack is most likely a single-source attack since besides the absence of a ramp-up, $F(60\%)$ is 2Hz. The other two unknown attacks are similar to each other in many aspects. They exhibit a small ramp-up of 120ms and have low $F(60\%)$ of 12Hz, indicating multiple attackers.

The tendency of multi-source attacks to localize power in lower frequencies is distinctly visible in the summary of $F(60\%)$ frequencies for both sites, Los Nettos in Figure 9 and USC in Figure 10. Based on these observations, we conclude that our results are not distorted by unusual traffic characteristics at our original site and that our techniques apply to at least some other traffic mixes.

Figure 10: Comparison of $F(60\%)$ and attack rate by attack class for USC attacks.

6.2 Experimental Confirmation

To understand the effect of network topology and number of sources on attack traffic we varied both these parameters in controlled experiments over the Internet. We placed synthetic attackers at six universities and research labs on both coasts of the United States (at ISI East, UCLA, UCSB, UCSD, UMass, and USC). We measured traffic at a target while varying the number of sources from 1–5 considering two topologies: a *clustered* attack, where all attackers reside on the same LAN segment and are well connected to the target via a high bandwidth, low latency link, and a *distributed* topology where attackers are widely distributed (with attackers on both coasts). We repeated these experiments multiple times during heavy and light network utilization, during peak weekday hours and early morning/weekend times, respectively (as measured local to the target), although it is obviously not possible to control Internet traffic. The victim and the observation point were located on the same Ethernet segment, connected via a hub. The traffic traces are collected at the the observation point using tcpdump [17]. Each synthetic DoS attacker is an Iperf [41] UDP sources sending 50 byte packets at a rates of 1Mbps and all experiment was run for 100 seconds. The hosts in the experiments have different operating speeds and all run variants of Linux. Table 6 provides a complete list of all the hosts, their operating speeds and the number of hops and RTT from the victim.
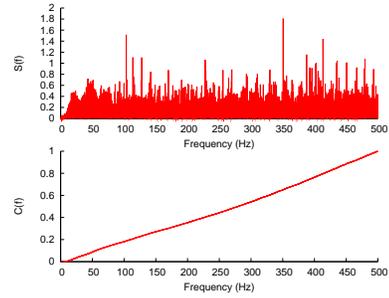
Figure 11(a) shows the clustered topology with only one sender. We see strong peaks in the high frequency ranges. This behavior is an inherent characteristic of a host sending at a rapid pace. All computers run at certain frequencies due to clocks in the CPU, the network card, and the operating system. We therefore believe that this pattern will be present in any host is sending as rapidly as possible. It could be masked by sending at slower rates, but that would reduce the effectiveness of the attack.

Looking across Figure 11 we see how the spectrum changes as the number of sources increase from 1 to 3, with all sources on the same Ethernet segment. The dominant spectral characteristics tend to shift toward low frequencies as the number of sources increase, with $F(60\%)$ at 300Hz, 150Hz, and 21Hz for 1, 2 and 3 sources respectively. In Section 6.3 we examine this effect more closely to show that it is due to multiple attackers operating out of phase with each other.
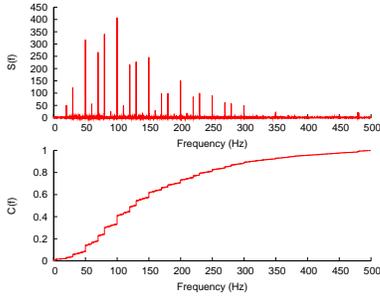
To examine the effect of network topology we repeated this experiment with each source at different locations around the Internet. Figure 12(a) shows the spectrum of a single attacker at UMass. The spectrum lacks the distinct peaks of Figure 11(a) which we believe this is due to a larger amount of cross traffic and more variation
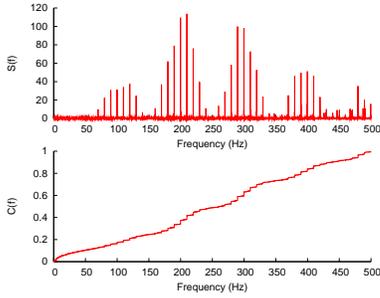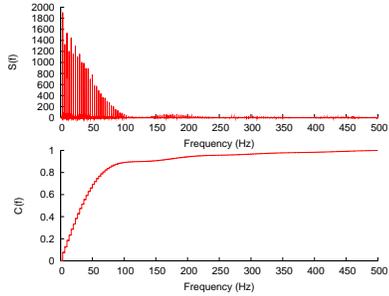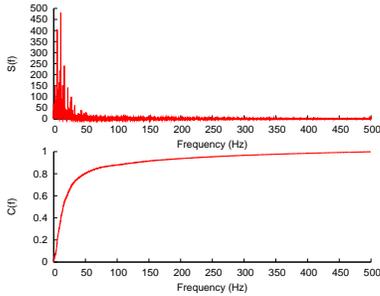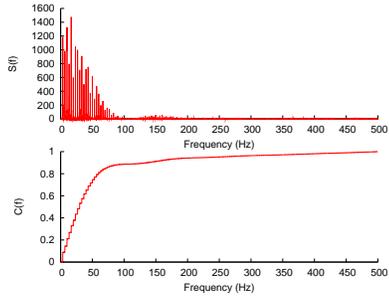
(a) One Source

(b) Two Sources

(c) Three Sources

Figure 11: WAN experiments using the clustered topology.

(a) One Source

(b) Two Sources

(c) Three Sources

Figure 12: WAN experiments using the distributed topology.

in transit time than with a single attacker in the clustered topology. The normalized cumulative spectrum is robust to this affect though, with both single-source attacks showing nearly linear trends.

Comparing Figure 12(a) to Figures 12(b) and 12(c), we see a shift in the spectrum to lower frequencies, with $F(60\%)$ at 43Hz and 35Hz for 2 and 3 sources, as compared to 328Hz for a single-source. Again, we suggest this is due to the presence of multiple, unsynchronized sources.

Figure 13 summarizes the $F(60\%)$ results for 30 WAN experiments conducted at different times of the day. The experiments show a localization of power in lower frequencies as the number of sources increase from 1–5 in both the clustered and the distributed topology. As seen in Figure 13 the $F(60\%)$ is close to 300Hz during single-source experiments, but reduces to 100Hz when more sources are added. This indicates that although the absolute value of $F(60\%)$ differs from one experiment to the next, the multi-

source attacks always have a much lower $F(60\%)$ in both topologies, qualitatively confirming our observations at Los Nettos.

To confirm the above results are not due to characteristics unique to Iperf, we conducted experiments with real DoS attack tools on a dumbbell-like topology consisting of 12 host machines, four hubs and two Cisco routers (we could not deploy the attack tools on the WAN topologies due to signature-based IDS monitoring tools). The testbed provides a low latency (¡1ms), high bandwidth (100Mbps) connection between the attackers and the victim. We generated attack traffic using three DoS tools: punk, stream, and synful, and web-based background traffic with WebStone [42]. Figure 14 shows all three attack tools produced spectral characteristics similar to Figure 13, both in the single- and multi-source experiments. We observe the single-source spectra (Figure 14(a), (b), and (c)) created by attack tools show strong characteristic high frequencies and linear normalized cumulative spectra, while the power in the
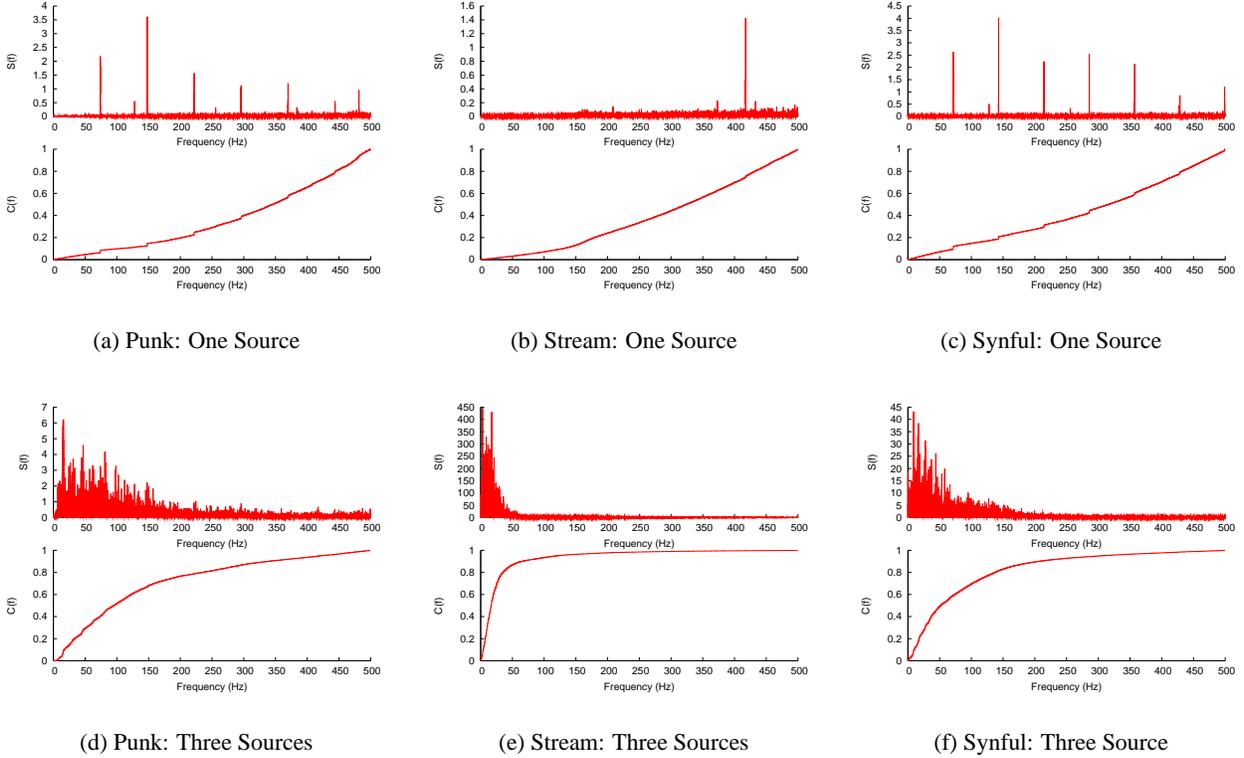
10

(a) Punk: One Source

(b) Stream: One Source

(c) Synful: One Source

(d) Punk: Three Sources

(e) Stream: Three Sources

(f) Synful: Three Source

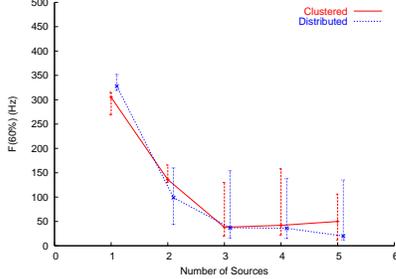**Figure 14: Testbed experiments using real attack tools.**

Figure 13: Localization of power as the number of sources increase in both clustered and distributed topologies.

higher frequencies start to reduce and the NCS shows a localization of power in lower frequencies as more attackers are added (Figure 14(c), (d) and (e)).

These experiments confirm the presence of multiple attackers changes the attack spectrum, and that the NCS and $F(60\%)$ are a reasonably robust discriminators between single- and multi-source attacks. They do not completely explain the reasons why multiple attackers shift the spectrum; we consider that next.

6.3 Understanding Multiple Source Effects

Although Section 6.2 confirms the validity of the use of spectral analysis to discriminate between single- and multiple-sources, it does not explain *why* spectral content is a good discriminator. To understand the physical meaning behind the shift in $F(60\%)$ to lower frequencies, we considered three hypotheses for its cause:

1. Aggregation of multiple sources at either slightly, or very different rates,

2. Bunching of traffic due to queuing behavior (analogous to ACK compression [26], but in the data direction),

3. Aggregation of multiple sources at different phases

To explore these hypotheses we perform simple numerical simulation. To test Hypothesis 1, we aggregate a *scaled* attack trace with the original attack trace to simulate aggregation of multiple attackers at different rates. If $a(t)$ represents the packet arrival sequence in the original trace, we multiply the time-stamp by a *scaling factor* s, jittered by ϵ, to generate a scaled trace. Therefore the aggregate trace is given by:

$$a_1(t) = a(t) + a((s+\epsilon)t) \qquad (7)$$

We use the packet trace from the single-source LAN experiment (Figure 11(a)) and vary the scaling factor from 0.5 to 2 representing attackers with rates varying from twice to half the original attack rate respectively (ϵ is uniformly distributed between 1–5μs). The scaled trace is then aggregated with the original attack trace using the approach defined by Kamath et al. [19]. If Hypothesis 1 is true, then a change in the attack rate should cause a corresponding change in $F(60\%)$. Figure 15 plots the scaling factor s against $F(60\%)$. We observe $F(60\%)$ remains nearly constant even when aggregated with an attacker with dissimilar attack rates. Hence we reject Hypothesis 1.

To test Hypothesis 2 we capture a packet arrival sequence on the attacker host and filter the arrival sequence to delay transmission until p packets, p varies from 5–15, have arrived, sending out all
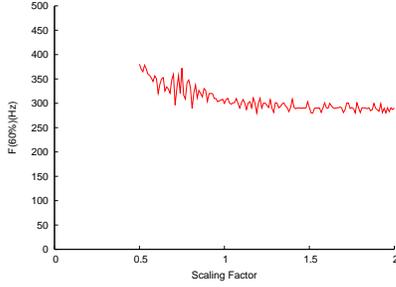
Figure 15: The effect of aggregation of two sources at different rates
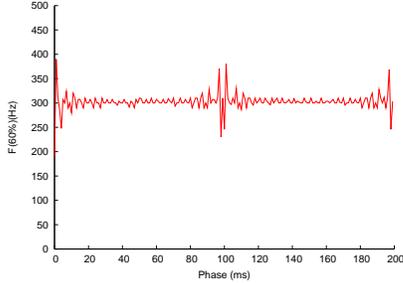
Figure 16: The effect of aggregation of two sources at different phases.

Figure 17: The effect of aggregation of multiple sources at different phases.

the packets at once. The trace passed through this process produces very different spectra. We observe a cluster of prominent frequencies around 320Hz with very little power (less than 15%) in the lower frequency band. The normalized cumulative spectrum has a sharp rise between 300–320Hz which is unlike spectra we have observed earlier. We therefore discarded Hypothesis 2.

To test Hypothesis 3, we aggregate a *shifted* attack trace with the original attack trace to simulate aggregation of attackers at different phases. If $a(t)$ represents the packet arrival sequence in the original trace, we add a *phase ϕ*, jittered by ϵ, to generate a shifted trace. Therefore the aggregate trace is given by:

$$a_3(t) = a(t) + a(t + \phi + \epsilon) \qquad (8)$$

We vary the phase from 1–200ms, representing the difficulty of attackers to start and remain synchronized. If Hypothesis 3 is true, then changes in attacker phase should cause a corresponding change in $F(60\%)$. Figure 16 plots the phase ϕ against $F(60\%)$. We observe $F(60\%)$ remains nearly constant even when aggregated with an attacker with dissimilar phase demonstrating that phase alone (Hypothesis 3) does not cause the shift.

Finally we considered a variation on Hypothesis 3. Suppose we aggregate multiple streams, each slightly out of phase. To test it we aggregate shifted attack traces with the original attack trace to simulate aggregation of multiple attackers at different phases. If $a(t)$ represents the packet arrival sequence in the original trace, we generate the shifted trace by:

$$a_{3b}(t) = \sum_{i=2}^{n} a(t + i\phi) \qquad (9)$$

We vary the number of attackers n from 2–15 with a 1ms phase shift between each attacker. If Hypothesis is true, then we should observe a drop is $F(60\%)$ as the number of attacker increase. Figure 17 plots the number of sources against $F(60\%)$ when using
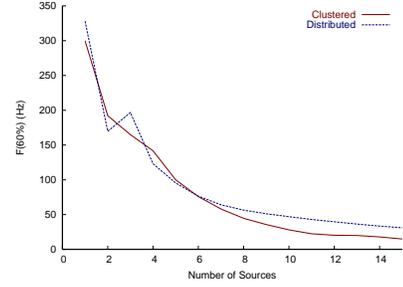
packet traces from both Figures 11(a) and 12(a). In both case we observe a drop in the $F(60\%)$ as the number of sources increase demonstrating phase along with aggregation of multiple sources causes localization of power in the lower frequencies. This results in consistent with the attack traffic observed at Los Nettos and USC.

The three hypothesis and simulation experiments provide insight into the attack stream dynamics that have been used by our framework to classify attacks robustly.

These experiments support our claims that: (a) sharp peaks in a high-rate, single-source attack are inherent. (b) Peaks cannot be maintained in a high-rate distributed attack because it is impossible to keep distributed sources in tight synchrony. (c) The effects of topological distance decrease the prominence of individual frequencies, but do not change its character. While these results apply to high-rate attackers, it is possible for attackers to affect their spectral characteristics by changing their attack rate. We examine this issue in the next section.

7. SENSITIVITY OF TECHNIQUES TO COUNTERMEASURES

Network security is an arms race; both attack tools and defenses evolve in relation to each other. Thus an important consideration of our framework is how robust they are to improved attack tools. In fact, our ramp-up and spectral analysis techniques were motivated by limitations of header analysis in the face of packet spoofing.

Header analysis was successful at classifying 83% of the 80 attacks we observed. We expect this percentage to drop as more sophisticated tools become more widely used. Even though source addresses are forged, currently most attack tools neglect randomizing the ID field. However it is easy for attackers to spoof this field, and as well as many operating systems are randomizing the ID field when the packet is not fragmented (to discourage OS fingerprinting [13]). Use of TTL is somewhat more robust, since attack packets with too low TTL values will fail to reach the victim. Statistical analysis of TTL values may be helpful at determining attacker distance in spite of spoofing. Unfortunately usefulness of this approach will be limited because a distance of a few hops encompasses much of the Internet. We expect evolution of attack tools to increase dependence on more advanced classification techniques based on spectral content.

Even though none of the observed single-source attacks exhibit an initial ramp-up, it can be easily generated by an attacker that gradually increases the attack rate. On the other hand, in large multi-source attacks we believe an initial ramp-up is an inherent part of the attack dynamics. The duration of the ramp-up may vary based on the zombie clock skew and differences in the zombie-

victim network distance, but masking the ramp-up by accounting for both sources of variability would require great sophistication.

Spectral analysis is much more robust to attacker manipulation than header analysis. We believe the characteristics of high-rate attack traffic are inherent; they cannot be avoided by single- or multi-source attackers sending at maximum rate. Further, it is not practical for a multi-source attacker to synchronize geographically distributed attackers to to create spectral characteristics similar to single-source attacks. To accomplish comparable levels of synchronization requires not only tight time synchronization between attacking hosts (perhaps using NTP [25]), but also measurement and accounting for the varying propagation and queuing delay between each attacker and the victim.

It may be possible for a single-source attacker to masquerade as a multi-source attack if it is willing to reduce its attack rate. A single-source can generate packets in bursty, on-off patterns by introducing a delay between packets, creating dominant low frequency contents in its spectrum.

8. APPLICATIONS

The focus of this paper is classification and understanding of DoS attack traffic. There are several applications of our results, including automating attack detection, providing synthetic models of attack traffic for simulation or testbed use, and inferring the amount of DoS attack activity in the Internet as a whole. Although details of these applications are outside the scope of this paper, next we briefly identify each.

8.1 Automating attack detection

A robust automatic attack detection tool is useful in guiding manual or automated response systems in installing filters [18, 29] or, in case of flash crowds, for use in aggregate congestion control [22]. Discrimination between single- and multi-source attacks is useful in selecting the appropriate response mechanism, since some mechanisms are much more expensive when dealing with multiple attackers compared to single attackers (for example, traceback [34]). We have already created an automated tool that given an attack trace will carry out the spectral analysis, demonstrating the feasibility of such a tool, but work remains to integrate this tool with other detection systems. A different kind of automation would be to develop spectral signatures for use in attack detection systems such as Snort [32].

8.2 Modeling

Many simulation studies of DoS attacks and responses use fairly simple traffic models such as constant bit-rate sources with fixed size packets, yet such models fail to capture the nuances of attack traffic. While real attack tools are easy to obtain and can be used in a testbed, there remain questions about how to support large numbers of attack machines and how to configure a testbed to reproduce attacks similar to those in the wild. Thus to create synthetic DoS traffic, both in simulation and testbeds, we need a better understanding of DoS attack traffic.

To our knowledge there have been no published studies of detailed characterization or models of DoS attacks. Studies based on back-scatter observe attacks indirectly, and thus do not capture fine-grained details of the attack dynamics [27]. Given the many modes of failure an attack can cause (hardware failures, exploitation of software glitches and misconfiguration, etc.), it is important to create faithful reproductions of real attacks. Although not the focus of this paper, we include some statistics about the kinds of attacks we see in the wild. Future work may use our tools as part of a broader study to better characterize DoS attacks, laying down
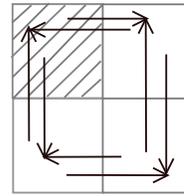
Figure 18: Limitations when extrapolating Los Nettos DoS activity to the Internet.

the groundwork for the creation of more realistic attack models.

8.3 Inferring DoS Activity in the Internet

Using our detection tools, we captured 80 DoS attacks in Los Nettos over five months. If we consider these attacks to be a sample of DoS activity in the Internet as a whole, we can project this activity to the public Internet. Such a projection should be considered *extremely* rough because of the small fraction of the Internet, the relatively small number of attacks we observed, and because such an estimation requires several assumptions about characteristics of the component information. Due to the limitations about these assumptions, we consider our estimates to have an error of at least a factor of 2. However, we suggest that the methodology proposed below coupled with a larger future monitoring effort can provide a reasonable Internet-wide estimate of attacks.

To provide a rough projection, we first compare the size of the monitored address space to the Internet. We monitored about 0.105% of the advertised Internet address space, determined by comparing the size of the routing table advertised by Los Nettos to the size of the advertised Internet address space as reported by Route Views [24] on 15 December 2002. We assume that both Los Nettos and the Internet allocates address space equally uniformly. (At small granularities, we know that addresses are not allocated uniformly, and ISP policy strongly influences address allocation density.) Given these assumptions, we can scale our observations to the Internet accordingly (by a factor of 950).

We observe both DoS attacks that transit and terminate in Los Nettos. Since distributed DoS attacks depart from many sources to attack a single victim, we would expect that counts of transiting attacks to be more prevalent than terminating attacks alone. For example, in Figure 18 we monitor a quarter of the address space. If we measure unique victims from the shaded area we observe 3 attacks and project 12, overestimating by a factor of three. If instead we observe only terminating attacks, we get an accurate estimate of 4. In general, projections from transit traffic identify a loose upper-bound on the number of attacks, since it may overestimate by the minimum of the scale-up factor or the number distributed attackers. Unfortunately, the number of attacks that terminate in our monitored address space is quite low, and so the accuracy of our projections is limited.

Since we project based on the number of attacks that terminate in our monitored area, our projections assume that the monitored area draws about the same number of attacks as a typical part of the Internet. We only monitored two of the four external connections of Los Nettos (see Figure 4) and so we increase our projection by a factor of two.

Our automated monitoring tool discards traces without major attacks (see Section 5.1), so we miss low-rate attacks that use few addresses. It is difficult to quantify this false negative rate, but believe we capture most large attacks and miss many small attacks. If these assumptions are true, the number of attacks observed can be

Month	In Los Nettos		In the Internet
	transiting	terminating	terminating (projected)
July	18	6	11400
Aug	12	1	1800
Sept	10	5	9500
Oct	10	0	0*
Nov	9	6	11400

Table 7: Extrapolating Los Nettos DoS activity to the Internet.

considered a lower bound.

Based on these assumptions, Table 7 projects our observations to the Internet as a whole. Clearly these projections are tentative, since it is very unlikely, for example that there were no attacks anywhere in the Internet in October. One point of comparison is the work of Moore et al. where they observe backscatter from 12,805 attacks in 3 weeks [27]. Direct comparison between their observation and ours is extremely difficult since the methodology and classes of counted attacks are very different, but it is somewhat reassuring that both their observation and our estimate are roughly the same order of magnitude.

Although the observations are very rough and require many assumptions, we believe this methodology will be useful at approximating attack prevalence if we can increase the size and duration of the monitored region. We are working on doing both.

9. CONCLUSION AND FUTURE WORK

In this paper we present a framework to classify attacks based on the number of sources present in the attack stream. The identification of single- and multi-sources attacks is based only on local information readily available in the attack stream and uses header content, initial ramp-up transients, and spectral analysis.

We test our framework on 80 attacks collected from two peering links at a moderate size commercial ISP. We validate our framework with attacks captured at a second monitoring site and experimentally, using synthetically generated attack traffic on a wide-area network and with real attack tools on a testbed. We use experiments and simulations to explain the physical causes for the difference in attack characteristics.

DoS attacks are constantly evolving, and currently there is a dearth of detailed information regarding attack dynamics. An empirical study, as presented in this paper, enables fine-grained analysis of attack patterns and topologies, that can be used to validate defense mechanisms and increase confidence in the solutions. The attack database created during this study can be used for a number of purposes; developing an automatic detection and response system based on number of attackers, developing high-fidelity models of attack dynamics, and inferring global DoS activity.

This work adopts a unique approach in analysis of attacks based on local attack information. Currently we have two observation points to capture attacks. When large attacks occur, like the recent root-server attack, additional detection sites would aid by providing multiple vantage points for the same attack, strengthening our classification framework. Additional detection sites would also provide more insight when projecting the prevalence of DoS activity on the Internet.

Acknowledgments

10. REFERENCES

[1] M. Allman, V. Paxson, and W. Stevens. TCP congestion control. RFC 2581, Internet Request For Comments, April 1999.

[2] Incident Detection Analysis and Response. http://ki.sei.cmu.edu/idar.

[3] Paul Barford, Jeffery Kline, David Plonka, and Ron Amos. A signal analysis of network traffic anomalies. In *Proceedings of the ACM SIGCOMM Internet Measurement Workshop*, Marseilles, France, November 2002.

[4] Steven Bellovin. ICMP traceback messages. Work in Progress: draft-bellovin-itrace-00.txt.

[5] Steven Bellovin. A technique for counting nated hosts. In *Proceedings of the ACM SIGCOMM Internet Measurement Workshop*, pages 112–208, Marseille, France, November 2002.

[6] George Box, Gwilym Jenkins, and Gregory Reinsel. *Time series analysis: forecasting and control*. Prentice-Hall, Upper Saddle River, New Jersey, 1994.

[7] Ronald Bracewell. *The Fourier Transform and Its Applications*. Series in Electrical Engineering. McGraw-Hill, New York, NY, 1986.

[8] Hal Burch and Bill Cheswick. Tracing anonymous packets to their approximate source. In *Proceedings of the USENIX Large Installation Systems Administration Conference*, pages 319–327, New Orleans, USA, Decemeber 2000. USENIX.

[9] Chris Chatfield. *The Analysis of Time Series: An Introduction*. Chapman and Hall texts in Statistical science series. Chapman & Hall, University of Bath, UK, 1996.

[10] Chen-Mou Cheng, H.T. Kung, and Koan-Sin Tan. Use of spectral analysis in defense against dos attacks. In *Proceedings of the IEEE GLOBECOM*, Taipei, Taiwan, 2002.

[11] Drew Dean, Matt Franklin, and Adam Stubblefield. An algebraic approach to IP traceback. In *In Proceedings of Network and Distributed Systems Security Symposium*, San Diego, CA, February 2001.

[12] David Dittirch. Distributed denial of service (DDoS) Attacks/tools. http://staff.washington.edu/dittrich/misc/ddos.

[13] Fyodor. Remote OS detection via TCP/IP stack fingerprinting. http://www.insecure.org/nmap/, October 1998.

[14] Thomer M. Gil and Massimiliano Poletto. MULTOPS: A Data-Structure for bandwidth attack detection. In *Proceedings of the USENIX Security Symposium*, pages 23–38, Washington, DC, USA, July 2001. USENIX.

[15] Hevin Houle and George Weaver. Trends in denial of service technology. CERT Coordination Center at Carnegie-Mellon University, October 2001.

[16] John Ioannidis and Steven M. Bellovin. Implementing pushback: Router-based defense against DDoS attacks. In *Proceedings of Network and Distributed System Security Symposium*, San Diego, CA, February 2002. The Internet Society.

[17] Van Jacobson, Craig Leres, and Steven McCanne. tcpdump - the protocol packet cpature and dumper program. http://www.tcpdmp.org.

[18] Peter Reiher Jelena Mirkovic, Greg Prier. Attacking DDoS at the source. In *Proceedings of the IEEE International Conference on Network Protocols10*, Paris, France, November 2002.

[19] Purushotham Kamath, Kun chan Lan, John Heidemann, Joe Bannister, and Joe Touch. Generation of high bandwidth network traffic traces. In *Proceedings of the International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, pages 401–410, Fort Worth, Texas, USA, October 2002. USC/Information Sciences Institute, IEEE.

[20] Angelos D. Keromytis, Vishal. Misra, and Dan. Rubenstein. Sos: Secure overlay services. In *Proceedings of ACM SIGCOMM 2002*, August 2002.

[21] http://www.ln.net.

[22] Ratul Mahajan, Steven M. Bellovin, Sally Floyd, John Ioannidis, Vern Paxson, and Scott Shenker. Controlling high bandwidth aggregates in the network. In *ACM Computer Communication Review*, July 2001.

[23] Robert L. Mason, Richard F. Gunst, and James L. Hess. *Statistical Design and Analysis of Experiments: with Applications to Engineering and Science*. Wiley series in probability and mathematical statistics. John Wiley & Sons, New York, NY, 1989.

[24] D. Meyer. University of oregon Route Views Project. Advanced Network Technology Center web site, http://www.antc.uoregon.edu/route-views.

[25] David Mills. Internet time synchronization: The network time protocol. *IEEE Transactions on Computers*, 39(10):1482–1493, October 1991.

[26] Jeffrey C. Mogul. Observing TCP dynamics in real networks. Technical Report 92.2, DEC Western Research Laboratory, April 1992.

[27] David Moore, Geoffrey Voelker, and Stefan Savage. Inferring Internet denial of service activity. In *Proceedings of the USENIX Security Symposium*, Washington, DC, USA, August 2001. USENIX.

[28] Number of addresses announced on the Internet. http://www.apnic.net/info/reports/index.html.

[29] Christos Papadopoulos, Robert Lindell, John Mehringer, Alefiya Hussain, and Ramesh Govindan. COSSACK: Coordinated Suppression of Simultaneous Attacks. To Appear in Proceeding of Discex III.

[30] Vern Paxson. Bro: A system for detecting network intruders in real-time. *Computer Networks*, 31(23–24):2435–2463, Decemeber 1999. (revised web page version).

[31] Vern Paxson. An analysis of using reflectors for distributed denial-of-service attacks. *ACM Computer Communications Review (CCR)*, 31(3), July 2001.

[32] Martin Roesch. Snort - lightweight intrusion detection for networks. http://www.snort.org.

[33] Mike Ryan. tcpshow - a tool to decode a tcpdump binary file. http://www.trionetworks.com.

[34] Stefan Savage, David Wetherall, Anna Karlin, and Tom Anderson. Practical network support for IP traceback. In *Proceedings of the ACM SIGCOMM Conference*, pages 295–306, Stockholm, Sweeden, August 2000. ACM.

[35] Alex C. Snoeren, Craig Partridge, Luis A. Sanchez, Christine E. Jones, Fabrice Tchakountio Stephen T. Kent, and W. Timothy Strayer. Hash-based ip traceback. In *Proceedings of the ACM SIGCOMM*, pages 3–14, San Deigo CA, August 2001. ACM.

[36] Dawn X. Song and Adrian Perrig. Advanced and authenticated marking schemes for IP traceback. In *Proceedings IEEE Infocomm*, Anchorage, Alaska, April 2001.

[37] Neil Spring, Ratul Mahajan, and David Wetherall. Measuring ISP topologies with rocketfuel. In *Proceedings of ACM/SIGCOMM '02*, August 2002.

[38] Robert Stone. Centertrack: An IP overlay network for tracking DoS floods. In *Proceedings of the USENIX Security Symposium*, pages 199–212, Denver, CO, USA, July 2000. USENIX.

[39] Cisco Systems. Netflow services and applications. http://www.cisco.com/warp/public/732/netflow.

[40] Cisco Systems. Rmon. http://www.cisco.com/warp/public/614/4.html.

[41] Ajay Tirumala, Feng Qin, Jon Dugan, Jim Ferguson, and Kevin Gibbs. Iperf Version 1.6.5. http://dast.nlanr.net/Projects/Iperf/.

[42] Gene Trent and Mark Sake. WebSTONE: The first generation in HTTP server benchmarking. http://www.sgi.com/Products/WebFORCE/WebStone/paper.html, February 1995.

[43] Haining Wang, Danlu Zhang, and Kang Shin. Detecting SYN flooding attacks. In *Proceedings of the IEEE Infocom*, pages 000–001, New York, NY, June 2002. IEEE.

[44] E. Zwicky, S. Cooper, D. Chapman, and D.Ru. *Building Internet Firewalls*. 2nd Edition. O'Reilly and Associates, 2000.