CRYPTOGRAPHIC MITIGATION OF DDOS ATTACKS BY PI (PATH IDENTIFICATION) MECHANISM

Ms.Kanika Malik [1], Ms.Shikha Goyal [2]

[1] Assistant Professor, NIIT, Najibabad (India)
[2] Assistant Professor, RVIT, Bijnor (India)

ABSTRACT

Distributed denial of service (DDoS) attack has been identified as one of the most serious problems on the Internet. While much of the current research focus on DDoS countermeasures, little attention has been paid on DDoS modelling, which is one of the important aspects that can help provide better solutions against DDoS attacks. This paper proposes an analytical model for the interactions between DDoS attack party and defense party, which allows us to have a deep insight of the interactions between the attack and defense parties.
Distributed Denial of Service (DDoS) attacks continues to plague the Internet. Defence against these attacks is complicated by spoofed source IP addresses, which make it difficult to determine a packet's true origin. In this paper a Pi (short for Path Identifier) mechanism, a new packet marking approach has been used .In this approach a path fingerprint is embedded in each packet, enabling a victim to identify packets traversing the same paths through the Internet on a per packet basis, regardless of source IP address spoofing.

Pi features many unique properties. It is a per packet deterministic mechanism each packet travelling along the same path carries the same identifier. This allows the victim to take a proactive role in defending against a DDoS attack by using the Pi mark to filter out packets matching the attackers' identifiers on a per packet basis. The Pi scheme performs well under large-scale DDoS attacks consisting of thousands of attackers, and is effective even when only half the routers in the Internet participate in packet marking. Pi marking and filtering are both extremely light-weight and require negligible state.
Trace route maps of real Internet topologies have been used to simulate DDoS attacks and validate the design.

Keywords -- Denial-Of-Service, Dos,, Ddos Defense, Packet Marking, Path Identifier

I.INTRODUCTION

Distributed denial of service (DDoS) attacks continue to plague the Internet. In a typical DDoS attack, attackers compromise multiple machines and use them to send large numbers of packets to a single victim server to overwhelm its capacity. For example, on October 21, 2002, an attacker flooded the root DNS servers with traffic in an effort to deprive the Internet of the DNS name lookup service (which would have paralyzed the majority of Internet applications.

Only five out of thirteen root servers were able to withstand the attack [23]. Previously, DDoS attacks had shut down several large Internet sites, such as Yahoo! and eBay. As an increasing number of businesses and services depend on the Internet, safeguarding them against attacks is a priority. Some critical infrastructures—for example, emergency telephone response (911)— increasingly rely on the Internet for communication and coordination [2]. Clearly, critical services demand effective countermeasures against DDoS attacks.

One challenge in defending against DDoS attacks is that attackers often use spoofed source IP addresses (also referred as *spoofed IP addresses*) which make it difficult to identify and block their packets under the current Internet infrastructure. Because of the importance and urgency of the DDoS problem, many researchers have studied countermeasures. A common solution in proposed systems is a *traceback* mechanism that has routers mark information on packets en-route to the victim, who can then use that information to reconstruct path that the packets take from the attacker through the Internet ,despite IP address spoofing. The path information obtained by the trace back mechanism can then be used to install network filters upstream from the victim to block attack traffic. The common assumption in these mechanisms is the need to reconstruct the exact path (or a path prefix)to the attacker in order to defend the victim. Most of these mechanisms (with the exception of [36]) also assume that the victim only also assume that the victim only initiates the trace back or passively receives trace back information, but does not otherwise actively participate in packet filtering. These assumptions face the following shortcomings:

_ The victim must receive large numbers of packets before it is able to reconstruct the path that they are taking.

_ Routers and/or victims need to perform non-trivial operations in marking packets or in reconstructing paths.

_ Network filtering is done on a per-flow or per-network basis using coarse identification criteria, rather than on a per-packet basis.

_ The victim has to rely on upstream routers to perform packet filtering, even once the attack paths have been identified.

In this paper, a new approach for defending against DDoS attacks that does not rely on these assumptions is being presented. It has been observed that reconstructing the exact path to the attacker is not necessary in defending against a DDoS attack

— One only needs to get an indication of the particular path that attack packets take. In addition, because our approach transmits path information in each packet, the victim can filter packets itself, based on its knowledge of the path information carried by a single prior attack packet.

Our approach embeds in each packet an identifier based on the router path that a packet traverses. The victim need only classify a single packet as malicious to be able to filter out all subsequent packets with the same marking. What makes this possible is that the packet marking is deterministic— all packets traversing the same path carry the same marking. All previous marking schemes that all are aware of are probabilistic in nature, in which the victim needs to collect a large number of packets to reconstruct the path. In this approach , a path identifier fits within a single packet so the victim can immediately filter traffic after receiving just one attack packet. This scheme is extremely light-weight, both on the routers for marking, and on the victims for decoding and filtering. The router marking in the scheme is also robust to the presence of legacy routers and shows strong incremental deployment properties.

II.METHODOLOGY USED

2.1 Overview

A common approach for DDoS defense is to provide the victim of a DDoS attack with the IP addresses of the routers along the path of the attack packets. With this information, the victim can request that upstream ISPs deploy packet filters to drop packets originating from the attacking networks, destined for the victim. Figure 1, shows that the average path length in the Internet, and thus, the average number of router IP addresses that must be transmitted to the victim, is roughly 15 (taken from Burch and Cheswick's Internet Mapping Project [4, 14] and from CAIDA's Skitter Map [6]). Assuming no compression, the lower bound on the amount of data the victim needs to reconstruct a single attack path of average length is 60 bytes. There are many proposed solutions for how to transmit this information to the victim.

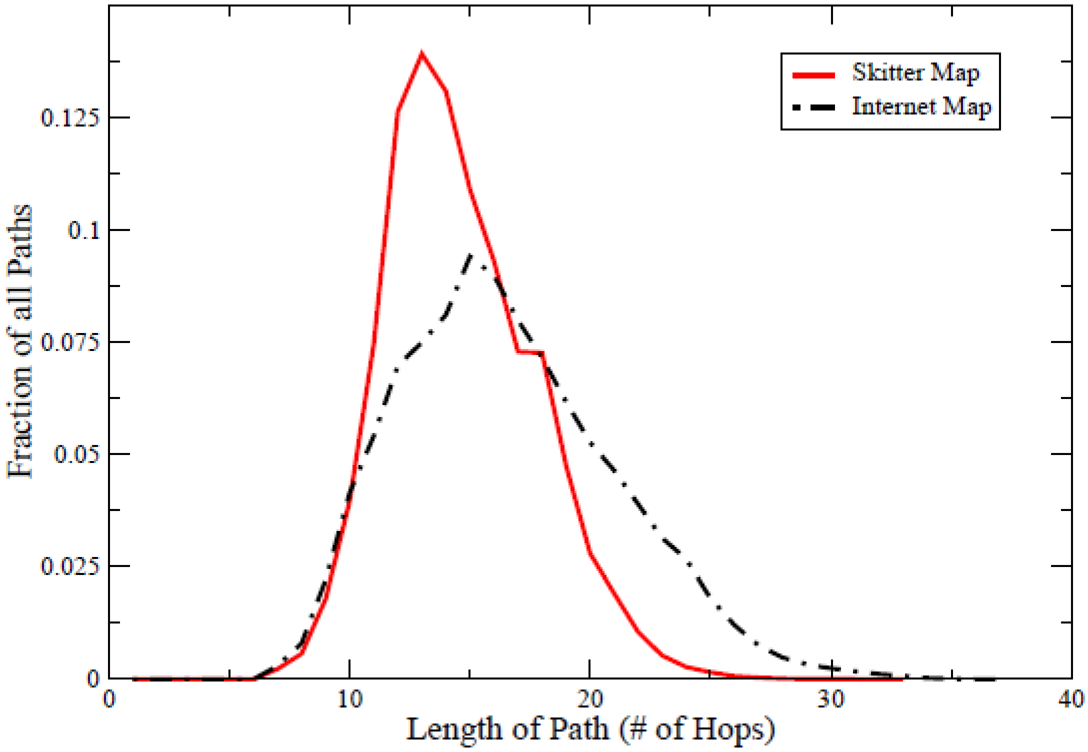


Figure 1. Distribution of Internet path lengths using the Skitter Map and the Internet Map

We propose a new approach for dealing with the DDoS attack problem. Since DDoS attacks often involve compromised machines, co-opted by a group of hackers exploiting other security vulnerabilities, there is little incentive for the victim to identify the path to specific attacker machines other than the need to provide information to help upstream.ISPs deploy packet filters as effectively—and with as little effect on legitimate traffic — as possible. However, if all packets arriving at the victim have some *distinctive marking*, then the victim need only note the markings that correspond to attack packets and then drop all incoming packets matching those markings.

To illustrate what we mean by the term *distinctive marking* we take the case of the Internet modelled as a complete binary tree, rooted at the victim server, with n nodes at the leaves. Using the estimated current size of the Internet [15] as n, we get dlog2(162; 128; 493)e = 28 bits to uniquely represent each path from the victim server to an end host (with a 0-bit representing a left branch and a1-bit representing a right branch). Although this model is an exceedingly simple representation of the Internet, we use it to show that path information need not be exclusively constructed of router IP addresses. We propose to construct a *path identifier*, akin to the binary tree path representation, to be embedded by routers in the IP Identification field 1 of every packet they forward. The path identifier will act as the distinctive marking which the victim can use to filter incoming packets.

Unfortunately, the 28 bit path identifier in our binary tree example is still 12 bits larger than the 16 bits that are available in the IP Identification field. In the Pi scheme, we limit ourselves to 16 bit path identifiers so that each packet carries all the marking information of the scheme. The router markings in Pi are also deterministic such that every packet traversing a particular path is marked with the same path identifier, which is generated piecemeal by the routers along the path from end-host to victim. Because each router has

only local knowledge (last-hop, next-hop) of a particular path, the marking for an entire path in Pi is not guaranteed to be globally unique. However, we show that a globally unique identifier is not necessary in providing strong DDoS protection and that the benefits of having a single-packet,

deterministic marking, allow the victim to develop rapidly responsive packet filters to protect itself during such attacks.

2.2 Basic Pi Marking Scheme

In its simplest form, we propose an n-bit scheme where a router marks the last n bits of its IP address in the IP Identification field of the packets it forwards. To determine the location within the field to mark the bits, we break the field into b16=nc different marking sections, and use the value of the packet's TTL, modulo b16=nc as an index into the section of the field mark. Figure 2 shows the C code for the Pi basic marking scheme where the marking bits function simply returns the IP address that is passed to it. Figure 3

shows an example marking scenario, using 1-bit marking

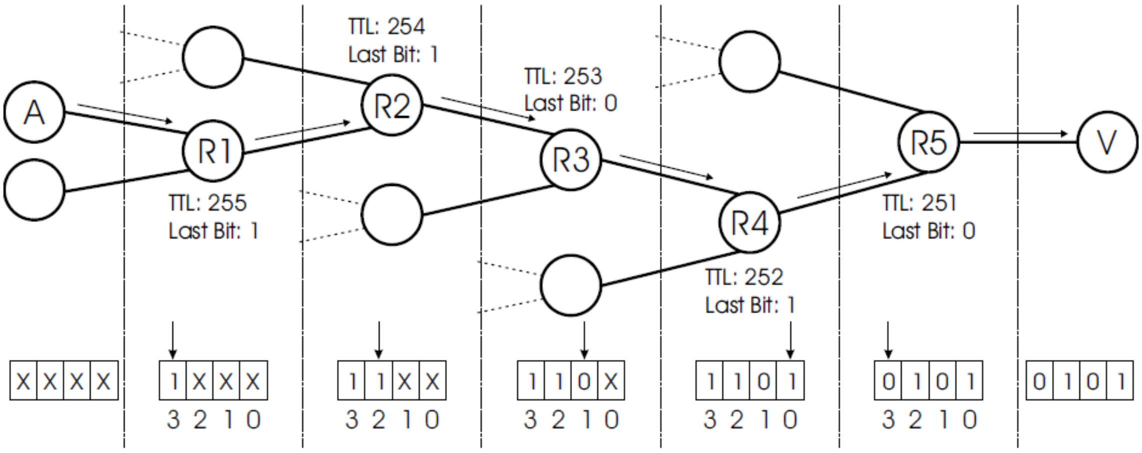


Figure 2. Example of our initial marking scheme. The packet travels from the attacker A to the victim V across the routers R1 to R5. Each router uses the TTL value of the packet to index into the IP identification field to insert its marking. a In this example we show 1bit Marking in a 4 bit field for simplicity

P = Pi mark of the packet
n = number of bits each router marks
Pimark(P, *TTL*, Curr IP, n*)*
f
m = 2n □ 1*;*
b = markingbits(Curr IP) &m;
bitpos = (*TTL* modP = Pi mark of the packet
n = number of bits each router marks

```
Pimark(P, TTL, Curr IP, n)
f
m = 2n □ 1;
b = markingbits(Curr IP) &m;
bitpos = (TTL mod (16/n)n)
b << bitpos;
m << bitpos;
return( (P & _m) j b );
}
```

THE PI MARKING ALGORITHM

III FILTRATION TECHNIQUE

This section describes how the victim can make use of the Pi marks to filter incoming packets during a DDoS attack.

In Section 3.1 a basic, simple filter strategy has been presented. In Section 3.2, we discuss an attack that an intelligent adversary can execute on a victim using this filter, and present a countermeasure called *TTL Unwrapping* to defend against it. In another section a more sophisticated filter based on the concept of threshold and the design space of possible filter algorithms is quite large..

3.1 The Basic Filter Scheme

The most basic filter a victim can apply to packets with Pi markings is to record the markings of identified attack packets and drop subsequent incoming packets matching any of those markings. Although this filter provides little flexibility to the victim, it has a very fast attack reaction time, since all the victim has to do is classify a single packet as an attack packet before being able to filter out all subsequent packets sent by that attacker. This filter also requires few memory resources, as it can be implemented in as little as 8Kbytes with a bit vector of length 2^{16} where the i’th bit of the vector is 0 if packets with a Pi mark of i are to be accepted and a 1 if packets with a Pi mark of i are to be dropped.

IV. EXPERIMENTAL PERFORMANCE

In this section, we evaluate Pi’s performance under DDoS attack. In order to evaluate Pi, we first describe our sample Internet data sets in the following section. We then explain the specific parameters that we choose for the design variables of our Pi scheme. We next present our DDoS attack model and the performance metrics that we measure. Finally, we present the results of our experiments and apply the experiments to incremental deployment scenarios.

4.1 Internet Data Sets

In our experiments, we use two Internet topologies: Burch and Cheswick’s Internet Map [4, 14] and CAIDA’s Skitter Map [6]. Both topologies were created by using a single host send trace routes to hosts throughout the Internet and recording the paths as the IP addresses of the routers along each route. We filtered the data sets to remove all incomplete routes and duplicate routes (although multiple routes to the same end-host were not removed). We also removed all routes of path length shorter than seven hops.

In our experiments, we take the trace route source of each map as the victim of our DDoS attack, and the end hosts on the trace route paths as our legitimate users and attackers

4.2 Experiment Design and Performance Metrics

For our experiments, we choose 5000 paths at random from one of our Internet data sets to act as legitimate users. We choose our attackers in the same way, but with the constraint that attackers and users are disjoint.

Each end-host at a path, whether user or attacker, sends three packets to the victim server in phase one of the attack, and three packets in phase two of the attack. We choose a three packet learning phase to illustrate how quickly Pi filters can react to DDoS attacks. A longer learning phase (which would almost certainly be the case in a real deployment scenario) would only improve performance further, because the victim would have more packet markings on which to base its filtering decisions. As our performance metric, we calculate the ratio of the number of attack packets accepted by the victim to the total number of attack packets sent (the attacker packet acceptance ratio) as well as the ratio of the number of user packets accepted by the victim to the total number of user packets sent.4 In some of our results we reach the victim easily. This limitation would make the effects of *marking saturation* more severe than they otherwise would be, so we eliminated n = 3 as an option as well. The number of bits per router marking n, must be a globally imposed constant in a deployed Pi system

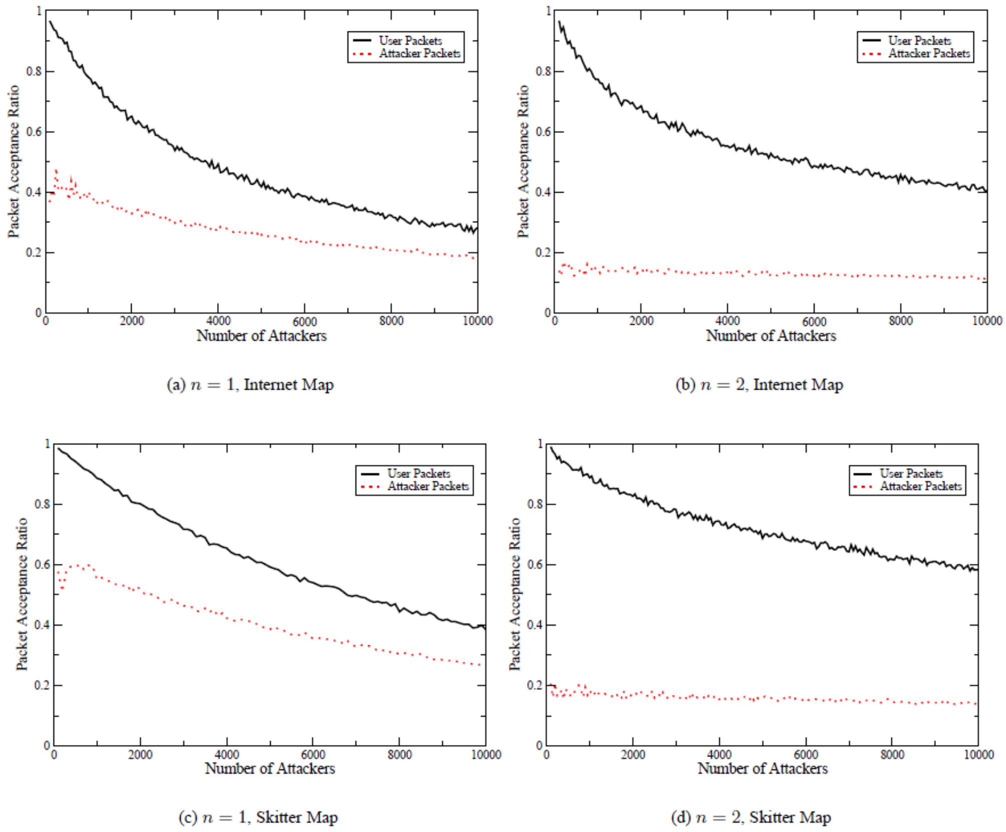
V. RESULTS AND DISCUSSIONS

In Fig 3 we see the n = 1 bit and n = 2 bit schemes with the basic Pi filter (equivalent to a threshold of T = 0).These curves represent the strictest possible filtering in the Pi scheme: a single attack packet with a particular marking received during the learning phase of the DDoS attack causes all packets with that marking to be dropped during the attack phase. The attack packet acceptance ratio is due to attackers located near enough to the victim that the random data that they initialize into the IP Identification field of their packets is not completely overwritten, allowing them to alternate to markings that were not recorded by the victim in the learning phase of the attack. Because the n = 1 bit scheme requires twice the number of marking routers as the n = 2 bit scheme to overwrite such random data, its attacker acceptance ratio is larger.

The downward slope exhibited for the user acceptance ratio, in both schemes, is due to the increasing number of attacker markings that collide with user markings, causing them to be dropped. This is an example of the *marking saturation* effect which we discuss in Section *saturation* effect which we discuss in Section 5.3. Surprisingly, marking saturation also affects attackers as well as legitimate users, as exhibited by the downward slope of the attacker acceptance ratios in Fig 3a and 3c. With a larger number of attackers, attack packets begin interfering with each other, in the sense that an attacker a may shift between four markings, two of which another attacker, b, is also shifting between. Because both a and b send packets in the learning phase, it is more likely that the overlapping markings will be received by the victim and added to the attacker markings list than it would be if only one of the attackers is present. The downward slope is minimized for the n = 2 bit scheme in Figures 3b and 3d because there are fewer attackers that are close enough to the victim to shift between markings.

In another fig. we show the effect of increasing the threshold value to combat the marking saturation effect. In this experiment, we set the threshold value to 50%, where more than half of the packets arriving with a particular marking must be attack packets before the victim begins dropping all packets with that marking. Of course, increasing the threshold value increases the overall number of packets accepted, which is reflected in the higher acceptance ratios for both the users and attackers

;



(a) $n = 1$, Internet Map

(b) $n = 2$, Internet Map

(c) $n = 1$, Skitter Map

(d) $n = 2$, Skitter Map

Fig 3: Pi filtering with 0 % threshold

VI. RELATED WORK

First general papers on network DoS are discussed. Moore, Voelker, and Savage use *backscatter* packets (the unsolicited responses that a DoS victim sends to the spoofed IP address that it receives in the attack packet) to gauge the level of Internet DoS activity [24]. Jung, Krishnamurthy, and Rabinovich attempt to answer the question of how a site can differentiate between a DoS attack and a simple high load condition by analyzing client request rates. Many approaches for securing against DoS and DDoS attacks are present in the literature. Early methods focused on detecting the ingress and egress points of DoS traffic within a single network administration. Ferguson and Senie propose to deploy network ingress filtering to limit spoofing of the source IP address [13]. A more recent and functional approach to ingress filtering s proposed by Li et al. in [20]. Their protocol, called SAVE, has routers construct tables of valid source addresses per incoming interface, in much the same way that they construct routing tables of destination addresses per interface. A packet whose source address is out of the proper range is easily identified and dropped. Stone proposes the Centre Track mechanism, which uses routers capable of input debugging (the ability to identify through which router interface a particular packet was received) that would be virtually connected through IP tunnels to all border routers on a network [35]. When a node in the network comes under attack, the overlay network is activated, and all border routers channel traffic through the overlay routers. These routers would use input debugging

VII .CONCLUSION

In this paper, Pi, a novel approach to defend against DDoS attacks is being proposed. Our proposal draws from elements of IP Trace back methods but is not concerned with reconstructing a path from a victim to an attacker, rather, it is concerned with marking paths with unique markings.

This gives the victim of a DDoS attack the ability to filter, on a per-packet basis, any incoming packets that match known attacker marks. We have shown how to increase entropy of the Pi marking by utilizing several improvements, specifically: IP address hashing to obtain a uniform distribution of packet marks per node; node omission based on the presence of intra-AS routes to increase the number of distant routers whose markings arrive at the victim; and edge marking to lower the probability of collisions of different paths. Marking method has been secured against attacker modified TTL values by utilizing TTL Unwrapping, which uses the TTL value at the victim to rotate the bits of a packet’s marking to a standard position, irrespective of the initial TTL. Finally, we demonstrate that the Pi marking scheme has strong incremental deployment properties, such that a victim is still able to filter incoming packets even when 50% of routers in our topology do not participate in the marking. We believe that Pi marking is the most general, flexible, and powerful of the packet marking schemes to date, and shows significant potential in reducing or eliminating the DDos threats.

REFERENCES

[1] M. Adler. Tradeoffs in probabilistic packet marking for IP traceback. In Proceedings of 34th ACM Symposium on Theory of Computing (STOC), 2002.

[2] Associated Press. Internet attack was much worse than anticipated. foxnews.com, Jan. 2003. http://www. foxnews.com/story/0,2933,76804,00.html.

[3] S. Bellovin, M. Leech, and T. Taylor. The ICMP trace back message. Internet-Draft, draft-ietf-itrace- 01.txt, Oct. 2001. Work in progress, available at ftp://ftp.ietf.org/internet-drafts/ draft-ietf-itrace-01.txt.

[4] H. Burch and B. Cheswick. Internet watch: Mapping the Internet. Computer, 32(4):97–98, Apr. 1999.

[5] H. Burch and B. Cheswick. Tracing anonymous packets to their approximate source. Unpublished paper, December 1999.

[6] Caida. Skitter. http://www.caida.org/tools/measurement/skitter/, 2000.

[7] Computer Emergency Response Team (CERT). TCP SYN flooding and IP spoofing attacks. Technical Report CA- 96:21, Carnegie Mellon University Pittsburgh, PA, Sept.1996.

[8] D. Dean, M. Franklin, and A. Stubblefield. An algebraic approach to IP traceback. In Network and Distributed System

Security Symposium (NDSS ’01), February 2001.

[9] D. Dean, M. Franklin, and A. Stubblefield. An algebraic approach to IP traceback. ACM Transactions on Information and System Security, May 2002.

[10] D. Dean and A. Stubblefield. Using client puzzles to protect TLS. In Proceedings of the 10th USENIX Security Symposium,Aug. 2001.

[11] N. G. Duffield and M. Grossglauser. Trajectory sampling for direct traffic observation. In Proceedings of the 2000 ACM SIGCOMM Conference, Aug. 2000. [12] C.Dwork and M. Naor. Pricing via processing or combating junk mail. In E. F. Brickell, editor, Advances in Cryptology – Crypto ’92, pages 139–147, 1992.

[13] P. Ferguson and D. Senie. Network ingress filtering: Defeating denial of service attacks which employ IP source address spoofing. RFC 2267, January 1998.

[14] Internet mapping. http://research.lumeta.com/ ches/map/, 2002.

[15] Internet Software Consortium. Internet domain survey host count. http://www.isc.org/ds/hosts.html, July 2002.

[16] J. Ioannidis and S. M. Bellovin. Implementing Pushback: Router-based defense against DDoS attacks. In Proceedings of the Symposium on Network and Distributed Systems Security(NDSS 2002), Feb. 2002.

[17] ICMP trace back (itrace). IETF working group, http://www.ietf.org/html.charters/itrace-charter.html.

[18] A. Juels and J. Brainard. Client puzzles: A cryptographic defense against connection depletion attacks. In Proceedings of the 1999 Network and Distributed System Security Symposium (NDSS ’99), pages 151–165, 1999.

[19] J. Jung, B. Krishnamurthy, and M. Rabinovich. Flash crowds and denial of service attacks: Characterization and implications for CDNs and web sites. In The Eleventh International World Wide Web Conference (WWW 11), May 2002.

[20] J. Li, J. Mirkovic, M.Wang, P. Reiher, and L. Zhang. SAVE: Source address validity enforcement protocol. In Proceedings of IEEE INFOCOMM 2001, Apr. 2001.

[21] R. Mahajan, S. M. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker. Controlling high bandwidth aggregates in the network. CCR, 32(3):62–73, July 2002.

[22] A. Mankin, D. Massey, C.Wu, S.Wu, and L. Zhang. On design and evaluation of intention-driven ICMP trace back. In Proceedings of the IEEE International Conference on Computer Communications and Networks, Oct. 2001.

[23] D. McGuire and B. Krebs. Attack on internet called largest ever. washingtonpost.com, Oct.2002.http://www.washingtonpost.com/wp-dyn/articles/A828-2002Oct22.html.

[24] D. Moore, G. Voelker, and S. Savage. Inferring internet denial of service activity. In Proceedings of the 10th USENIX Security Symposium, Aug. 2001.

[25] V. Paxson. An analysis of using reflectors for distributed denial-of-service attacks. Computer Communication Review,31(3), 2001.

[26] A. Perrig, D. Song, and A. Yaar. Pi: A new defense mechanism against IP spoofing and DDoS attacks. Technical Report CMU-CS-02-207, Carnegie Mellon University, School of Computer Science, Dec. 2002.

[27] R. L. Rivest. The MD5 message digest algorithm. RFC1321, Internet Activities Board, Internet Privacy Task Force, Apr. 1992.

[28] S. Savage, D.Wetherall, A. Karlin, and T. Anderson. Practical network support for IP trace back. In Proceedings of the

2000 ACM SIGCOMM Conference, August 2000.

[29] S. Savage, D. Wetherall, A. Karlin, and T. Anderson. Network support for IP trace back. ACM/IEEE Transactions on Networking, 9(3), June 2001.

[30] C. L. Schuba, I. V. Krsul, M. G. Kuhn, E. H. Spafford,A. Sundaram, and D. Zamboni. Analysis of a denial of service attack on TCP. In Proceedings of the IEEE Symposium on Research in Security and Privacy, May 1997.

[31] A. C. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, S. T. Kent, and W. T. Strayer. Hash-based IP traceback. In Proceedings of the ACM SIGCOMM 2001 Conference, pages 3–14, Aug. 2001.

[32] A. C. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, B. Schwartz, S. T. Kent, and W. T. Strayer. Single-packet IP trace back. IEEE/ACM Transactions on Networking (ToN), 10(6), Dec. 2002.

[33] D. X. Song and A. Peril. Advanced and authenticated
marking schemes for IP trace back. In Proceedings of IEEE INFOCOMM 2001, April 2001.

[34] I. Stoica and H. Zhang. Providing guaranteed services without per flow management. In Proceedings of the 1999 ACM SIGCOMM Conference, Apr. 1999.

[35] R. Stone. Center Track: An IP overlay network for tracking DoS floods. In Proceedings of the 9th USENIX Security Symposium, Denver, Colorado, Aug. 2000. USENIX.

[36] M. Sung and J. Xu. IP trace back-based intelligent packet filtering: A novel technique for defending against internet DDoS attacks. In Proceedings of the 10th IEEE International Conference on Network Protocols (ICNP’02), Nov. 2002.