A Proactive Approach to Distributed DoS Attack Prevention using Route-Based Packet Filtering*

Kihong Park† Heejo Lee‡
Network Systems Lab
Department of Computer Sciences
Purdue University
West Lafayette, IN 47907
{park,hlee}@cs.purdue.edu

CSD-TR 00-017
December 3, 2000

Abstract

Denial-of-service (DoS) attack on the Internet has become a pressing problem. In this paper, we describe, analyze and evaluate route-based distributed packet filtering (DPF), a novel approach to distributed DoS (DDoS) attack prevention. We show that there is an intimate relationship between the effectiveness of DPF at mitigating DDoS attacks and power-law network topology. We evaluate performance using Internet autonomous system and artificially generated topologies.

The salient features of this work are two-fold. First, we show that DPF is able to proactively filter out a significant fraction of spoofed packet flows and prevent attack packets from reaching their targets in the first place. The IP flows that cannot be proactively curtailed are extremely sparse such that their origin can be localized—i.e., IP traceback—to within a small, constant number of candidate sites. We show that the two proactive and reactive performance effects can be achieved by implementing route-based filtering on less than 20% of Internet autonomous system (AS) sites. Second, we show that the two complementary performance measures are dependent on the properties of the underlying AS graph topology. In particular, we show that the power-law structure of Internet AS topology leads to connectivity properties which are crucial in facilitating the observed performance effects.

As a DDoS prevention architecture, DPF is able to emulate the IP traceback prowess of probabilistic packet marking, while alleviating the latter's three principal weaknesses: (i) need to inscribe link information in the IP packet header, (ii) reactiveness—traceback occurs after the impact of DoS attack has been felt—and (iii) scalability where the effort needed to achieve IP traceback grows proportionally with the number of attack hosts engaged in a DDoS attack.

*This work was supported in part by NSF grant EIA-9972883.
†Additionally supported by NSF grants ANI-9714707, ANI-9875789 (CAREER), ESS-9806741, ANI-0082861 (ITR), and grants from the Purdue Research Foundation, Santa Fe Institute, Sprint, and Xerox.
‡Additionally supported by the Center for Education and Research in Information Assurance and Security (CERIAS).

1 Introduction

1.1 Background

Denial of service (DoS) is a pressing problem on the Internet as evidenced by recent attacks on commercial servers and ISPs and their consequent disruption of services [8]. DoS attacks [4, 11, 17, 30] consume resources associated with various network elements—e.g., Web servers, routers, firewalls, and end hosts—which impedes the efficient functioning and provisioning of services in accordance with their intended purpose. Their impact is more pronounced than network congestion due to the concentrated and targeted nature of resource depletion and clogging, which not only impacts quality of service (QoS) but can affect the very availability of services. When the attack is distributed—e.g., affected by multiple compromised hosts on the Internet—then its impact is proportionally severe.

Susceptibility to DoS is an intrinsic problem of any service provisioning system where, at a minimum, the occurrence of a potentially valid event (e.g., service request, TCP SYN packet) must be processed to ascertain its validity. Even though the resource expenditure associated with processing a single event may be negligible, when this is multiplied by the large factors enabled by the high bandwidth of modern broadband networks, its impact can be significant no matter how small the individual processing overhead. As with prank telephone calls or ringing of door bells in days gone by, an effective means of preventing DoS attacks from occurring in the first place—also the only fundamental solution given the intrinsic susceptibility of service provisioning systems to DoS—lies in identification of the attacker which admits assigning commensurate costs (e.g., legal or economical) to the perpetrating entity. Even if the attack was instituted from compromised hosts intruded by an attacker, if the physical source of DoS traffic can be identified, then at the very least the invaded network element can be isolated or shut down, and in some instances, the attacker's identity can be further traced back by state information remnant on the compromised system.

In this paper we address two complementary problems and goals: (1) source identification (i.e., IP traceback) of spoofed IP flows, and (2) prevention of spoofed IP packets from reaching their destination. We describe a novel approach to proactive/reactive distributed DoS (DDoS) attack prevention called route-based distributed packet filtering and evaluate its efficacy in Internet autonomous system (AS) topologies.

1.2 New Contributions

Route-based distributed packet filtering (DPF) uses routing information to determine if a packet arriving at a router—e.g., border router at an AS—is valid with respect to its inscribed source/destination addresses, given the reachability constraints imposed by routing and network topology. A single AS—there were 4872 autonomous systems on the Internet in 1999—can only exert a limited impact with respect to identifying and discarding forged IP flows. At the other extreme, if all autonomous systems and their routers implement router-based packet filtering, then no spoofed IP flows can escape, but its ultimate effect is not much different from that achievable by perfect ingress filtering.

Route-based DPF's main strength lies in the fact that with *partial* coverage or deployment—about 18% in Internet AS topologies—a synergistic filtering effect can be achieved that *proactively* prevents spoofed IP flows from reaching other autonomous systems in the first place. Proactive protection, due to intrinsic connectivity properties of Internet topology, cannot be perfect. It is, however, strong enough such that those IP flows that cannot be prevented from penetrating are sufficiently sparse and, as a consequence, their origin can be localized to within 5 possible sites. Thus, as with probabilistic packet marking, effective IP traceback can serve as a deterrent as well as facilitate responsive, albeit reactive recovery upon attack. Another important proactive advantage of route-based DPF over PPM is its robustness with respect to *distributed* DoS. Unlike PPM whose attack site localization deteriorates almost-proportionally with the number of attack hosts [20]—thus necessitating commensurate time and effort by the victim to recover—in route-based DPF the fraction of AS from which spoofed IP flows can reach other AS is a small subpopulation (less than 12%) which makes harnessing attack sites (e.g., by intrusion) in a DDoS attack more difficult for the attacker. From an implementation perspective, DPF does not require expending IP header fields to encode stamped link information as PPM does. However, computing appropriate filtering tables alongside existing inter-domain routing protocols (e.g., BGP) is a nontrivial problem due to the destination-based structure of Internet routing protocols (inter-domain and intra-domain).

We submit that few fundamental solutions exist for preventing distributed DoS attacks, and this paper's main contribution lies in advancing a scalable architecture for DDoS attack prevention that is effective for Internet AS topology and is, in principle, implementable in IP internetworks if global routing information is made available at border routers. We define relevant performance measures that capture pertinent filtering performance—both proactive and reactive—and show their intimate dependence on power-law structure of Internet AS topology. We demonstrate the efficacy of route-based DPF using comprehensive benchmarking with both Internet AS and artificially generated network topologies. Finding efficient implementations and evaluating the costs associated with deployment and router overhead vis-à-vis the demonstrated performance benefits is a major challenge in itself and a task for future work.

The rest of the paper is organized as follows. In the next section, we give a summary of related works. In Section 3, we define the key notions, measurement, and performance variables for route-based DPF. We also discuss the core issues surrounding performance evaluation. In Section 4, we present performance results based on benchmark experiments with both real and artificial network topologies. We conclude with a discussion of our results.

2 Related Work

Several types of DoS attacks have been identified [8, 17, 30] with the most basic DoS attack demanding more resources than the target system or network can supply. Resources may be network bandwidth, file system space, processes, or network connections [17]. While host-based DoS attacks are more easily traced and managed, network-based DoS attacks which exploit weaknesses of the TCP/IP protocol suite [15], represent a more subtle and challenging threat [17, 23]. Network-based DoS attacks, by default, employ spoofing to forge the source address of DoS packets, and thereby hide the identity of the physical source [4]. Previous works have focused on detecting DoS attacks and mitigating their detrimental impact upon the victim [1, 13, 24, 27]. This approach does not eliminate the problem, nor does it deter potential attackers.

A number of recent works have studied source identification—also called IP traceback [23]—which span a range of techniques with their individual pros and cons. In link testing, the physical source of an attack is identified by tracing it back hop-by-hop through the network [29]. Traceback is typically performed manually and recursively repeated at the upstream router until the originating host is reached. The drawbacks of link testing include multiple branch points, slow traceback during an attack, communication overhead due to message exchange, and administrative constraints between network operators [29]. The audit trail approach facilitates tracing via traffic logs at routers and gateways [22, 28]. This method is conducive to off-line traceback of DoS attacks. A principal weakness, however, is the high storage and processing overhead incurred at routers which can exert a significant burden. In behavioral monitoring [17], the likely behavior of an attacker during a DoS attack is monitored to identify the source. For example, an attacker may perform DNS requests to resolve the name of the target host which may not be resident in its local name server's cache. During a DoS attack, an attacker may try to gauge the impact of the attack using various service requests including Web and ICMP echo requests. Thus, logging of such events and activities can reveal information about the attacker's source.

In packet-based traceback, packets are marked with the addresses of intermediate routers, in some sense, an inverse operation of source routing and similar to the IP Record Route option [21]. The victim uses information inscribed in packets to trace the attack back to its source. A related method is generating information packets—separate from data packets—that convey analogous path information as ICMP traceback messages to the victim [2]. In both methods, overhead in the form of variable-length marking fields that depend on path length or traffic overhead due to extra messaging packets are incurred. Probabilistic packet marking [3, 23] has been proposed for achieving the best of both worlds—space efficiency in the form of constant marking field and processing efficiency in the form of minimal router support—at the expense of introducing uncertainty due to probabilistic sampling of a flow's path. The effectiveness of probabilistic packet marking was analyzed when considering the intrinsic vulnerability of marking field spoofing [20] and shown that the attacker's location can be localized to within 5 equally likely sites on the Internet under single-source attack. Improved marking schemes including for authentication were studied in [26]. In spite of its efficiency properties, PPM has several drawbacks: (i) spoofed packets are allowed to exert their debilitating influence on server resources before being reactively curtailed; (ii) bits in the IP header must be expended to inscribe link information; and (iii) uncertainty of IP traceback amplifies proportionally with the number of hosts partaking in the distributed DoS attack. We show that route-based distributed packet filtering, in addition to matching the IP traceback prowess of PPM, solves its three weaknesses.

Packet filtering is a network security mechanism for controlling what data can flow to and from a network affected routers or firewalls [33]. Filtering decisions, typically, are made based on packet content including source/destination addresses and port numbers. As a means of preventing network-based DoS attacks, ingress filtering in border gateways has been proposed for limiting IP source address spoofing [5, 7, 25, 31]. Ingress filtering requires a prolonged period to be broadly deployed on the Internet, and even then, it is subject to attacks from AS that are not compliant (see Section 4.5 for a discussion of its performance effects).

3 Route-based Distributed Packet Filtering

3.1 Route-based Detection of Spoofed IP Packets

Consider the AS topology shown in Figure 3.1 which depicts the routes from node 2 to all other nodes (solid arrows). Assume a host belonging to AS 7 is attempting a DoS attack targeted at a server residing in AS 4 by using a forged source IP address belonging to AS 2. A border router belonging to AS 6 at the peering point with AS 7—if cognizant of the route topology—would recognize that a packet originating from AS 2 destined to AS 4 would not enter through link $(7, 6)$ implying that its source address must be spoofed. Such packets could be discarded at AS 6 thus proactively protecting AS 4 from the DoS attack. Note that in this specific instance AS 6 only need inspect the source IP address to determine that no packet from AS 2—irrespective of destination IP address—can arrive on link $(7, 6)$. This example serves to illustrate the potential opportunities available by exploiting routing information to identify and filter spoofed packets at forwarding points in the system. We remark that

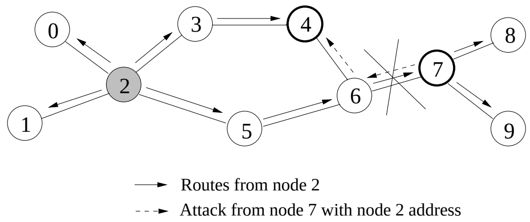


Figure 3.1: Illustration of route-based packet filtering executed at node 6. Note 7 uses IP address belonging to node 2 when attacking node 4.

the above description—from an inter-domain IP routing perspective—is imprecise. First, an edge in the AS graph between a pair of nodes is in general a set of peering point connections and all corresponding border routers must carry out the specified filtering tasks. Second, two or more IP prefixes belonging to the same destination AS may lead to different paths on an AS topology. This is incorporated in our AS model by allowing multi-path routing. Third, we ignore potentially relevant classification of AS nodes into stub, multi-homed, and transit AS where only the latter may engage in routing proper (i.e., in the sense of packet forwarding). When we speak of an AS node performing route-based filtering, it must be understood that the finer resolution picture is more complex, although logical consistency between the two descriptions is achieved.

Consider the case where the attack host residing in AS 7 uses an IP address belonging to AS 8 when attacking servers in AS 4. The gateway at AS 6 cannot unambiguously decide that the IP packet with source address in AS 8 is spoofed since it may indeed have come from AS 8 (and forwarded by AS 7). This demonstrates that performing route-based filtering at a *single* site can achieve only so much. Route-based distributed packet filtering aims to achieve a synergistic, proactive filtering effect through the collective action of a small number of AS nodes. The key objectives of DPF can be summarized as follows: (i) maximize proactive filtering of spoofed IP packets; (ii) for bogus packets that do get through, minimize the number of sites that could have sent the packets (IP traceback); achieve objectives (i) and (ii) while minimizing the number of sites at which route-based filtering is carried out; (iv) in tandem with (iii), find the optimum sites where filtering is to be performed.

3.2 Maximal and Semi-maximal Filters

Let $G = (V, E)$ be an undirected graph representing Internet AS topology. We remark that our framework and conclusions may be carried over to router topologies *within* an AS, however, presently little is known about the internal structure of large, commercial autonomous systems[1], and testing needs to await further measurement studies. Let $\mathcal{L}(u, v)$ denote the set of all loop-free paths from u to v where $u, v \in V$. A routing algorithm and its computed routes lead to a subset $R(u, v) \subseteq \mathcal{L}(u, v)$. An IP packet $M(s, t)$ with source IP address s and destination IP address t is routed through the network according to $R(s, t)$. If $|R(s, t)| > 1$, we assume a separate network mechanism resolves selection among multiple paths. Performance results for multi-path routing are discussed in Section 5.7.

A *filter* $F_e : V^2 \to \{0, 1\}$ is a function defined for link $e = (u, v) \in E$ where this is interpreted to mean that a gateway router in v acting as a peering point inspects an IP packet $M(s, t)$ arriving on e, then decides whether to forward the packet ($F_e(s, t) = 0$), or filter—i.e., discard—the packet ($F_e(s, t) = 1$). We call F_e a *route-based packet filter with respect to* R if

$$F_e(s, t) = \begin{cases} 0, & \text{if } e \in R(s, t); \\ 1, & \text{otherwise.} \end{cases}$$

To avoid cluttering, with a slight abuse of notation, we use "$e \in R(s, t)$" to mean that link e is on some path belonging to $R(s, t)$. Similarly for a node v. A route-based filter is *maximal* if it satisfies $F_e(s, t) = 0$ if, and only if, there exists a path in $R(s, t)$ with e as one of its links. Thus a maximal route-based filter carries out all the filtering of spoofed IP traffic that is possible without adversely affecting routing of non-spoofed IP packets as determined by R. If a set of route-based filters collectively were "perfect" in the sense that no spoofed datagram is allowed to reach its destination, then this may be

[1] Router topologies may obey power-law connectivity structure similar to AS topologies [6, 18]. There are, however, semantic differences between AS and router topologies—e.g., geographical distance between two nodes in an AS graph may not be meaningful—which have to be taken into consideration when advancing interpretations.

viewed as providing a form of authentication service. Computing a maximal route-based filter—e.g., represented as a table—is straightforward, but it requires in general $O(n^2)$ space ($n = |V|$) which is an overwhelming burden to place on routers that are expected to perform fast table look-up.

A *semi-maximal* filter is a maximal filter which uses only the source IP address of a packet to carry out its filtering (i.e., a projection of F_e). In other words, $\hat{F}_e(s,t)$ is a *semi-maximal filter with respect to R* if

$$\hat{F}_e(s,t) = \begin{cases} 0, & \text{if } e \in R(s,v) \text{ for some } v \in V; \\ 1, & \text{otherwise.} \end{cases}$$

Hence, its filtering capability is, in general, less than that of its maximal counterpart, i.e., $\hat{F}_e(s,t) \leq F_e(s,t)$. Although we lose in potential filtering power—by not much as shown in Section 5—a semi-maximal filter can be represented by a filtering table in linear space which brings it in the domain of feasibility if not practicality. As with routing and techniques for speed-up of routing table look-up, further optimizations will be needed to affect practical implementations.

3.3 A Remark on Implementability

The most important implementation concern in the context of IP internetworks is not space requirement—many issues are shared with routing table look-up—but the computation of semi-maximal filters at routers belonging to participating AS nodes $T \subseteq V$ given a routing algorithm R. The main difficulty arises from the fact that IP routing—inter- and intra-domain—follows a destination-based approach where routing table update exchanges convey information about destination reachability but not "source reachability." In BGP, for example, an update message contains AS-PATH which is a sequence of AS numbers that identify the autonomous systems—starting with the AS that originated the advertisement of reachability for an IP prefix—from which, if assuming bi-directionality and bounce-back of an "ACK-to-Update" message, source information may be extracted. At a minimum this would require augmentation to BGP, or introduction of a new protocol that interacts with BGP. In addition, some form of compression and encoding may be needed to keep the return messages small—including merger of several ACK-to-Update messages at branch points—given their tendency to grow in number and size the closer they reach the target AS (the essential overhead associated with breaking routing asymmetry). Other challenges arise when trying to construct an accurate global AS routing map.

We do not have an answer to the efficient implementability question for IP internets. This may, perhaps, be route-based DPF's Achilles' heel. We view the contribution of this paper to lie in the definition and evaluation of a scalable DDoS prevention architecture as part of a set of fundamental solutions to the denial-of-service attack problem (of which there are few), and Internet specificity is injected with respect to showing how filtering performance depends on topological properties of Internet AS. We believe that the performance results for route-based DPF are encouraging and suggest that investigation of how to implement route-based DPF so as to minimize overhead and cost for Internet deployment may be worthwhile and should be the focus of future effort. However, a cost-benefit analysis of the gains vis-à-vis the associated costs is a matter of debate and further examination.

3.4 Performance Measures for Distributed Packet Filtering

3.4.1 Filtering Effect: Attacker and Victim Perspectives

A (semi) maximal filter is *distributed* if it is executed at more than one node in V. We will use T to denote a subset $T \subseteq V$ of nodes where filtering is performed. We call $\gamma = |T|/|V|$ the *coverage ratio*. To quantify and measure the collective filtering effect of route-based DPF—including IP traceback—we define a set of performance metrics that is used in the rest of the paper. First, we define two families of variables $S_{a,t}$ and $C_{s,t}$ $(a, s, t \in V)$ which are then used to define other high-level measures of more direct interest and relevance to quantifying DDoS mitigation.

$S_{a,t}$ denotes the set of nodes—more precisely, the set of IP addresses belonging to an AS node in $S_{a,t}$—that an attacker at AS a can use as spoofed source IP addresses to reach t without being cut-off by filters executed at autonomous systems in T. By definition, $a \in S_{a,t}$ for all $a, t \in V$. The larger the set $S_{a,t}$, the more options an attacker at a has in terms of forging the IP source address field with a bogus address which will go undetected and unhindered with respect to R at filters in T. Whereas $S_{a,t}$ is defined from the attacker's perspective, $C_{s,t}$ captures the victim's perspective and denotes the set of nodes that could have sent an IP packet $M(s,t)$ with spoofed source IP address s and destination address t which did not get filtered on its way. We allow $s \in C_{s,t}$ for all $s, t \in V$ in the definition. The larger $C_{s,t}$, the more uncertain the victim at t is upon receiving spoofed packet $M(s,t)$ with respect to its true origin. If $|C_{s,t}| = 1$, then this means that IP address s cannot be used by any attacker $a \in V$ (outside of s itself) to mount a spoofed DoS attack aimed at t. Figure 3.2 illustrates the impact of route-based distributed filtering on curtailing the attacker's ability to engage in spoofing. Without

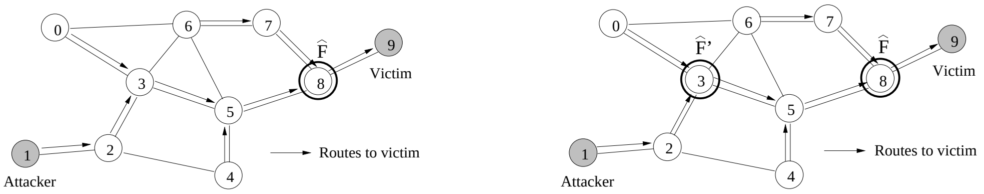


Figure 3.2: Left: With route-based filtering executed at node 8, the spoofable address range at attack site 1 is reduced from $S_{1,9} = \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$ to $\{0, 1, 2, 3, 4, 5\}$. Right: Distributed filtering with filter $\hat{F}$ at AS 3, the spoofable range further reduces to $S_{1,9} = \{1, 2\}$.

route-based filtering, an attacker residing at AS 1 can disguise himself with undetectable spoofed IP addresses belonging to AS 0–8, i.e., $S_{1,9} = \{0, 1, \ldots, 8\}$, when attacking a server in AS 9. With route-based filtering at AS 8, the spoofable address range shrinks to $\{0, 1, \ldots, 5\}$. With distributed filtering at AS 8 and AS 3, $S_{1,9} = \{1, 2\}$.

3.4.2 Proactive Filtering Measures

The most immediate—but also practically useless—proactive filtering effect is captured by $\Phi_1(\tau)$ which is defined as $\Phi_1(\tau) = |\{t : \forall\, a \in V,\ |S_{a,t}| \leq \tau\}|\, n^{-1}$. The range of τ is $\tau \geq 1$. Thus, $0 \leq \Phi_1(1) \leq 1$

denotes the fraction of AS's that cannot be reached by spoofed packets from anywhere. The closer $\Phi_1(1)$ is to 1, the fewer the number of AS's exposed to DoS attack. For $\tau \geq 2$, $\Phi_1(\tau)$ has a less relevant and directly interpretable meaning. In spite of its appealing semantic relevance, we will show that $\Phi_1(1)$ is near zero for Internet AS topologies when the coverage ratio γ is not near 1, and thus of little import as a performance measure.

A more subtle, but practically relevant, proactive performance measure is given by

$$\Phi_2(\tau) \quad = \quad \frac{|\{a : \forall\, t \in V,\ |S_{a,t}| \leq \tau\}|}{n}.$$

$\Phi_2(1)$ measures the fraction of attack sites from which sending spoofed IP packets targeted at other AS is futile since they will be filtered by nodes in T. Thus, if $\Phi_2(1) = 0.8$, then an attacker wishing to engage in DDoS attack cannot make productive use of attack hosts residing in 80% of all autonomous systems. This imposes an upper bound on the distributedness of DDoS attack achievable by any attacker, severely limiting the latter the closer $\Phi_2(1)$ is to 1. Policy-wise, it is also possible for other AS to be "on guard" with respect to traffic emanating from AS where mounting an attack is feasible. As with $\Phi_1(\tau)$, $\Phi_2(\tau)$ does not have directly relevant semantics for $\tau \geq 2$. Unlike $\Phi_1(1)$, however, $\Phi_2(1)$ achieves large values for Internet AS topologies. It is our principal proactive performance measure.

$\Phi_3(\tau)$, Θ, and $\Psi_2(\tau)$ are auxiliary metrics capturing proactive filtering with less sharply delineated semantics which are defined as $\Phi_3(\tau) = |\{(a,t) : |S_{a,t}| \leq \tau\}|/n(n-1)$, $\Psi_2(\tau) = |\{s : \forall\, t \in V,\ |C_{s,t}| \leq \tau\}|/\,n$, and $\Theta = |\{(a,s,t) : s \in S_{a,t}\}|/n(n-1)^2 = |\{(a,s,t) : a \in C_{s,t}\}|/n(n-1)^2$. $\Phi_3(1)$ denotes the fraction of all attacker-victim AS pairs (out of a total of $n^2 - n$) where the attacker cannot reach the victim with spoofed IP packets. Thus an attacker whose aim is to wreck general havoc on the Internet via DoS attack without specific interest in a particular victim may choose random attack site-victim pairs to do so. The larger $\Phi_3(1)$, the less impact such random DDoS attacks will have. Θ captures the reduced attack volume—ratio of unfilterable packets—when, in addition, attacks are mounted using randomly inscribed source IP addresses. $\Psi_2(\tau)$, viewed from the attacker's perspective, represents the fraction of all (spoofable) IP addresses whose use would allow the victim to localize the attack site to within τ locations.

3.4.3 Reactive Filtering Measure: IP Traceback

The performance measures defined in the previous section are proactive in nature in that they capture how effectively spoofed IP packets are prevented from reaching their destination in the first place by filters in T. Perfect proactivity, as captured by $\Phi_1(1) \gg 0$, however, is intrinsically difficult to achieve in Internet topologies due to their connectivity structure unless the coverage ratio $\gamma \approx 1$. Significant, albeit imperfect, proactive filtering is captured by the quantitative values of $\Phi_2(1)$, $\Phi_3(1)$, Θ, and $\Psi_2(\tau)$, where $\Phi_2(1)$ plays the most relevant role.

Since not all spoofed IP packets can be effectively filtered, complementing the proactive performance measures is the reactive metric $\Psi_1(\tau)$ which captures the IP traceback (or source identification) effect:

$$\Psi_1(\tau) \quad = \quad \frac{|\{t : \forall\ s \in V,\ |C_{s,t}| \leq \tau\}|}{n}.$$

For example, $\Psi_1(5)$ represents the fraction of (target) autonomous systems which, when attacked with an arbitrary spoofed IP packet, can resolve the attack location to within 5 possible attack sites. The parameter $\tau \geq 1$—meaningful for values greater than 1—represents the uncertainty associated with IP traceback localization[2]. If $\Psi_1(\tau) = 1$ for τ a small constant, then those spoofed IP flows that cannot be prevented from penetrating the "filter net" spanned by nodes in T can be effectively localized with respect to their true attack origin to within τ candidate sites, i.e., we achieve IP traceback. This can act as a deterrent as well as allow responsive on-line counter measures to be undertaken against the attacking party including isolation of offending flows at appropriate peering points. $\Psi_3(\tau)$ is analogously defined as $\Phi_3(\tau)$ with $C_{s,t}$ in place of $S_{a,t}$ but does not have relevant semantics and is omitted from further consideration.

4 Performance Evaluation and Benchmarking Issues

4.1 Overall Objectives

Formally a route-based (semi) maximal distributed filter $\mathcal{F}$ is given by a triple $\mathcal{F} = \langle G, T, R\rangle$ where $G = (V, E)$ is the AS graph, $T \subseteq V$ the subset of AS where route-based filtering is performed, and R is the routing algorithm. For two route-based DPF's $\mathcal{F} = \langle G, T, R\rangle$ and $\mathcal{F} = \langle G, T', R\rangle$ with $T \subseteq T'$, it can be checked that $|S_{a,t}| \leq |S'_{a,t}|$ and $|C_{s,t}| \leq |C'_{s,t}|$ for all $a, s, t \in V$. This, in turn, implies $\Phi_2(1) \leq \Phi'_2(1)$ and $\Psi_1(\tau) \leq \Psi'_1(\tau)$ for all $\tau \geq 1$. Similar monotonicity properties hold for the other performance metrics. Moreover, $\Phi'_2(1) = \Psi'_1(1) = 1$ if $T' = V$ (i.e., there is a trivial lower bound). Evaluating the effectiveness of $\mathcal{F}$ with respect to the proactive and reactive performance measures entails studying its dependence on topology G, the size of the filter net T, its structure, and routing R. Our goal is to uncover the above relationships and find feasible means to economize T when trying to achieve a target performance.

4.2 Influence of Topology

Empirical evidence shows that Internet AS topology exhibits power-law connectivity [6, 10] which may also hold for router topologies [18]. Power-law graph structure induces "centers" where connectivity is concentrated on a few nodes, with most vertices possessing sparse connectivity (e.g., comprised of AS stubs and non-transit multi-homed AS). A key aspect of our DDoS benchmark evaluation is to ascertain if, and how, topology affects proactive and reactive performance of DPF. We employ 1997–1999 Internet AS topologies taken from NLANR [16] which have been used in other studies aimed at

[2]See [20] for a discussion of IP traceback localization issues—also called uncertainty factor—under probabilistic packet marking.

understanding the connectivity structure of Internet topology, especially with respect to its recently discovered power-law property [6]. In addition to actual Internet AS topologies, we use artificial Internet topology generators [12, 14] and random graphs to perform comparative benchmarking. An (unintended) side effect of our study is the reverse evaluation of artificial topology generators with respect to capturing relevant graph properties—above-and-beyond power-law relations—in the context of DPF.

4.3 Filter Placement

In addition to the influence of the *size* of the filter net T on DPF performance, for a given coverage ratio $\gamma = |T|/n$, the selection of the nodes in T is a key performance variable. We consider the effect of choosing T randomly—we sample from V uniformly randomly until a target coverage size $|T|$ is reached—and by more customized design rules, in particular, the case when T forms a vertex cover[3]. It can be checked that T being a vertex cover (VC) is neither a sufficient nor necessary condition for $\Phi'_2(1) = \Psi'_1(\tau) = 1$. However, since a VC forms a cover of all edges in the graph—being VC implies that on any path, at least every other vertex on the path belongs to T—it may be expected that the VC property is conducive to enhancing the performance of DPF. In tandem, the presence of "centers" in power-law graphs leads one to expect that a small coverage ratio γ may be achievable.

Finding a minimal VC in a graph is an NP-complete problem [9]. We use two approximation algorithms—one with a constant factor guarantee and the other a heuristic—for finding small VCs. The first algorithm is a well-known constant-factor approximation scheme whose output is guaranteed to be at most twice as large as an optimal VC [19]. There is a randomization component, and the approximation scheme is run multiple (in our evaluations 10) times with the smallest VC constituting the final output. The second algorithm is a heuristic: little is known rigorously about its behavior although, in practice, it oftentimes outperforms the constant-factor approximation scheme. The heuristic—greedy algorithm—iteratively grows a VC by picking a node which covers the most remaining uncovered edges. The presence of centers in power-law graphs makes it more conducive for the heuristic to find small VCs which is verified in our performance results. We use the minimum VC found by the two algorithms as our T.

We also consider a rank-based placement strategy to isolate VC's tendency to pick "large centers" vis-à-vis its complete edge covering property. We call this algorithm *Rank*, and it simply orders vertices by their degree (in decreasing order) and selects the top γ percentage as elements of T. We show that *Rank*—even with larger T—affects a diminished performance effect than VC.

4.4 Maximal vs. Semi-maximal Filters

Our performance results are for semi-maximal filters which are, in general, less powerful than maximal filters. In comparative evaluations we show that replacing semi-maximal with maximal filters

[3] T is a *vertex cover* of G if every edge in E is incident on some node in T.

results only in an incremental improvement in proactive/reactive filtering performance. The marginal performance difference—e.g., maximal filtering can localize to within 4 sites for IP traceback instead of 5—justifies the use of semi-maximal filters when performing route-based DPF in addition to its consideration of efficiency.

4.5 Ingress Filtering

Consider the case when the nodes in T were to perform ingress filtering only. Then for coverage ratio $\gamma = |T|/n$, the DDoS prevention performance effect as captured by $\Phi_2(1)$ and $\Psi_1(\tau)$ would be: $\Phi_2(1) = \gamma$, $\Psi_1(\tau) = 0$ for $\tau < n - |T|$ and $\Psi_1(\tau) = 1$ for $\tau \geq n - |T|$. Thus, ingress filtering, unless carried out almost everywhere, is an ineffective DDoS prevention strategy. Even when $\gamma = 0.95$, for the 1999 Internet AS topology where $n = 4872$, IP traceback capability as captured by Ψ_1 incurs a non-constant uncertainty factor of 243 (the trivial number of possible attack sites to investigate when trying to pin down the true attack location). There is little compelling reason for a group of AS in the global Internet to form trusted security partnerships based on mandatory ingress filtering since the collective performance effect is low. In contrast, we show that when AS in T implement route-based DPF, then with $\gamma < 0.2$ coverage $\Phi_2(1) > 0.88$ and $\Psi_1(5) = 1$ for 1997–1999 Internet AS topologies.

4.6 Routing

The set of feasible routes is influenced by topology but, in addition, we consider the impact of having multiple paths from source to destination. Note that $R' \subseteq R$ implies $\Phi_2(1) \leq \Phi'_2(1)$ and $\Psi_1(\tau) \leq \Psi'_1(\tau)$ for all $\tau \geq 1$. Consider routing policies that allow R to have up to m separate paths—not necessarily disjoint—between two nodes. This allows us to evaluate the influence, other things being equal, that the more paths are permitted when routing a packet from source to destination, the more easily the packet can elude route-based filtering when using spoofed source IP addresses. The latter is due to the attack site's spoofable IP address space $S_{a,t}$ having increased. When multi-path routing is performed between two nodes a and t with $|R(a,t)| = m$, we select m shortest paths from $\mathcal{L}(a,t)$. In the case where two or more candidates have the same path length, we choose the path coming first in the canonical (i.e., lexicographic) order. We give special names to two extreme forms of R: *loose* and *tight*. "R=*loose*" means that all possible paths among two nodes can be used for routing, i.e., $R(a,t) = \mathcal{L}(a,t)$. When R allows only a single routing path ($m = 1$), we choose a shortest path between a and t, and denote this case as "R=*tight*."

5 Performance Results

5.1 Set-up

We have built a performance evaluation tool called *dpf* which implements the benchmarking set-up described in Section 4. *dpf* consists of three core modules: `cover`, `dpf`, and `stats`. `cover` handles the generation of T with various input specifications including random selection, VC, and rank ordering. `dpf` is the main module which computes $S_{a,t}$ and $C_{s,t}$; its input specification include the filter type, T, and routing algorithm. `stats` takes the output of `dpf` and computes the various performance measures including $\Phi(1)$ and $\Psi(\tau)$. We use topology generators *Inet* [12] and *Brite* [14] to generate benchmark graphs which are included in the test suite.

5.2 Proactive Filtering Effect

5.2.1 Limitations to Perfect Proactivity

$\Phi_1(1)$ measures the fraction of AS which are immune from DoS attack—i.e., no spoofed IP packet can reach them—distributed or single-source. Figures 5.1 (left) and (middle) show $\Phi_1(\tau)$ as a function of τ for different coverage and routing combinations for 1997 Internet AS topology ($|V| = 3015$ and $|E| = 5230$). In Figure 5.1 (middle), $\Phi_1(\tau) = 0$ up to $\tau = 4$. That is, perfect proactivity where there exists at least one AS that is immune from DoS attack from anywhere is unachievable at 18.9% coverage ratio under the best of circumstances. The two graphs show that, overall, $R = \mathit{tight}$ gives better performance than $R = \mathit{loose}$ and, other thing being equal, T being VC—the size of the 1997 Internet AS vertex cover is 18.9%—is more effective than T being random even with higher coverages Rnd30 ($\gamma = 0.3$) and Rnd50 ($\gamma = 0.5$). These plots depict a general trend but are not otherwise very useful since for performance evaluation purposes only $\Phi_1(1)$ has direct relevance.

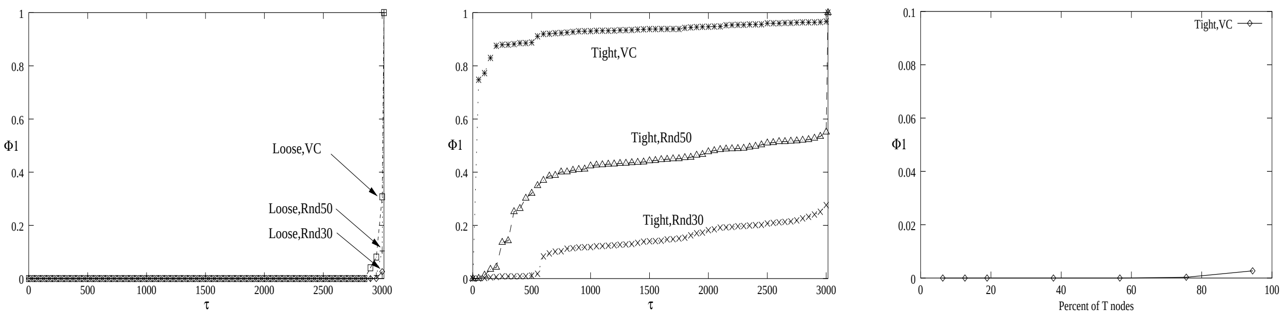


Figure 5.1: 1997 Internet AS topology. Left: $\Phi_1(\tau)$ for $R = \mathit{loose}$. Middle: $\Phi_1(\tau)$ for $R = \mathit{tight}$. Right: $\Phi_1(1)$ as a function of $|T|$ while maintaining VC property.

Figure 5.1 (right) shows that the limitation to achieving perfect proactivity does not change when the VC is grown to larger sizes up to 100%. Although eventually $\Phi_1(1)$ becomes positive when coverage is above 90%, its value is negligible to warrant the high cost of almost full coverage. Perfect proactivity as captured by $\Phi_1(1)$ is intrinsically difficult to attain, and should not be construed as a viable performance goal.

5.2.2 DDoS and Proactive Filtering

$\Phi_2(1)$ measures the fraction of AS from which DoS attacks cannot be launched since all spoofed packets—whoever their target—will be detected and filtered. $\Phi_2(1)$ thus puts an upper bound on the distributedness of DDoS attacks. Figure 5.2 (left) and (middle) show $\Phi_2(\tau)$ as a function of τ for $R = loose$ and $R = tight$. As with Φ_1, $\Phi_2(\tau)$ for $\tau \geq 2$ does not have a concrete, relevant meaning and are shown to depict the general trend. Figure 5.2 (right) is the more relevant plot which shows $\Phi_2(1)$

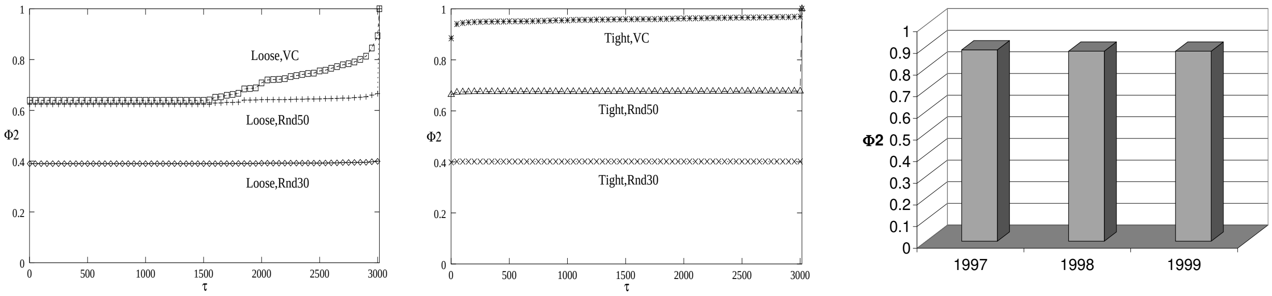


Figure 5.2: 1997 Internet AS topology. Left: $\Phi_2(\tau)$ as a function of τ for $R = loose$. Middle: Corresponding graph for $R = tight$. Right: $\Phi_2(1)$ for 1997–1999 Internet AS topologies.

for Internet AS topologies during 1997–1999. $\Phi_2(1)$ achieves a value of around 88% during the three years which implies that only 12% of all autonomous systems can be used by attackers to launch DoS attacks. Since the number of AS's has grown from 3015 in 1997 to 3878 in 1998 to 4872 in 1999, the absolute number of possible attack sites has grown commensurately. However, as a percentage, viable attack sites have remained well-behaved at 12%.

Figure 5.3 (left) shows $\Phi_3(\tau)$ as a function of τ with $\Phi_3(1) = 0.96$. That is, only 4% of all source-destination AS pairs are feasible attack AS/victim AS combinations from the attacker's perspective, where spoofed packets emitted from the attack AS can reach the victim AS. For example, an attacker who tries to enlist attack hosts in a DDoS attack by intruding these hosts will waste 96% of its effort if the source-destination AS's are chosen randomly. Thus proactive filtering erects barriers in terms of effort and cost to mounting effective DoS attacks which, in turn, can act as a deterrent in addition to

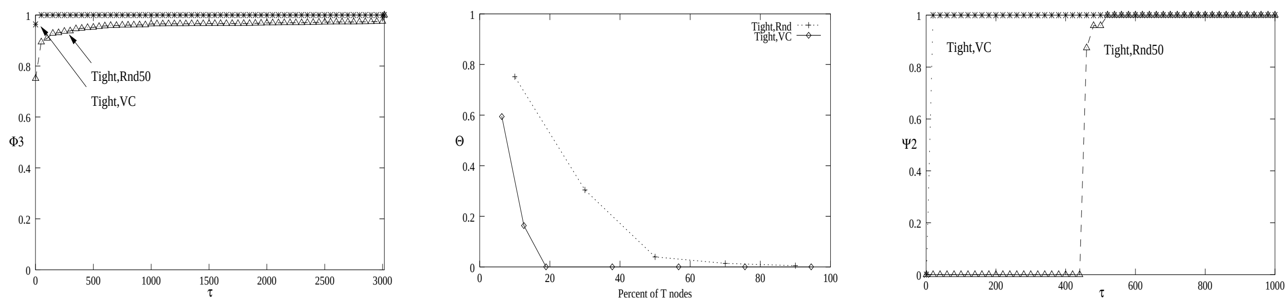


Figure 5.3: 1997 Internet AS topology. Left: $\Phi_3(\tau)$. Middle: Θ as a function of $|T|$. Right: $\Psi_2(\tau)$.

its primary curtailing effect. Figure 5.3 (middle) shows Θ, the coarsest measure, which represents the fraction of source, destination, and spoof address triples (a, t, s) where a host residing at AS a is able to send an IP packet to target AS t with spoofed source IP address s. We observe that for coverage above 20%, the fraction of forgeable triplets shrinks to near 0. This means that if, in addition to a and t, the spoof address s is randomly generated, then the spoofed IP packet has almost zero chance of reaching its target. Figure 5.3 (right) shows $\Psi_2(\tau)$ as a function of τ. For an uncertainty factor of $\tau = 20$, the fraction of traceable—i.e., to within τ sites—IP addresses is nearly 1. Collectively, these results show that the attacker's effort, resources, and sophistication needed to launch a successful DDoS attack is significant and brought about by route-based DPF's proactive filtering effect.

5.3 Reactive Filtering Effect: IP Traceback

As shown in the previous section, eliminating *all* spoofable IP flows is an unrealistic goal given its intrinsic difficulty with respect to Internet AS connectivity. A different consequence of proactive filtering is the more subtle, complementary effect where spoofed IP flows that cannot be prevented from penetrating the network system can be localized to within a few possible sites. This is affected by DPF filtering sufficiently many flows such that the remaining spoofable IP flows form a sparse subset which, in turn, facilitates source identification, i.e., IP traceback.

Figure 5.4 shows $\Psi_1(\tau)$ as a function of τ for $R =$ *loose*, *tight*, and $T =$ VC, Rnd30, Rnd50. The general trend shows that $\Psi_1(\tau)$ undergoes a sharp transition at some τ value, especially for $T =$ VC and $R =$ *tight*. Figure 5.5 (left) shows $\Psi_1(\tau)$ for 1997–1999 Internet AS topologies for $1 \leq \tau \leq 10$.

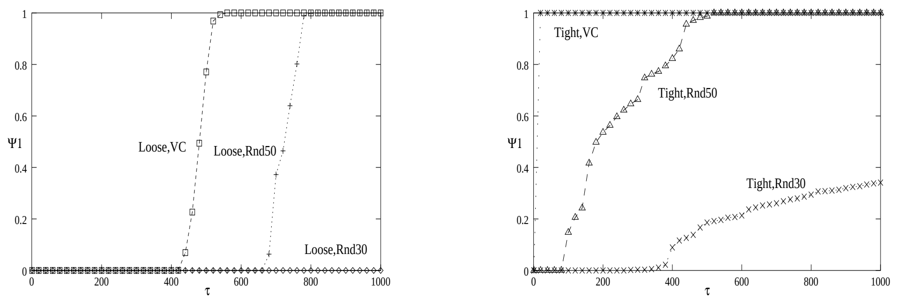


Figure 5.4: 1997 Internet AS topology. Left: $\Psi_1(\tau)$ for $R =$ *loose*. Right: $\Psi_1(\tau)$ for $R =$ *tight*.

We observe that across 1997, 1998, and 1999, $\Psi_1(5)$ is preserved—i.e., every attack can be localized to within 5 candidate sites—and the only performance difference occurs for $\tau < 5$ where $\Psi_1(\tau) < 1$. IP traceback is achieved "instantly" and thus allows speedy on-line response by the attacked site with respect to actions against the perpetrating attack site. Compared to probabilistic packet marking, route-based DPF is proactive even with respect to IP traceback since a *single* spoofed IP packet suffices to reveal the attacker's AS location to within a small constant number of locations. In PPM, a sufficient number of DoS attack packets must be received before the attack path can be reconstructed by the probabilistically inscribed link values in the IP datagram [20, 23]. Thus, not only can route-based

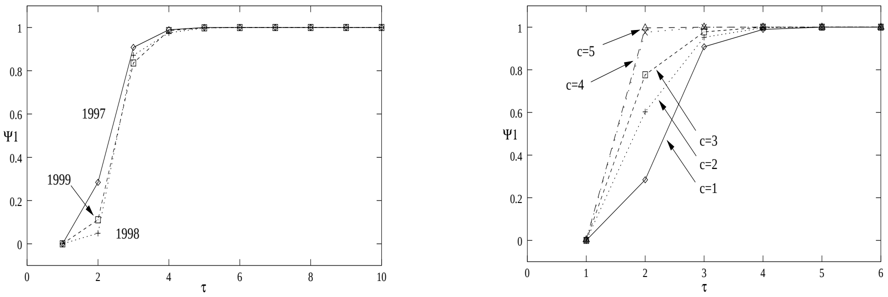


Figure 5.5: Left: $\Psi_1(\tau)$ for 1997–1999 Internet AS topologies. Right: Shape of $\Psi_1(\tau)$ for $|T| = c \cdot |VC|$ with dilation factor $c = 1, 2, \ldots, 5$.

DPF emulate the IP traceback prowess of PPM, it can do so more efficiently and speedily than PPM. Figure 5.5 (right) shows the marginal benefit of increasing the number of nodes in T after achieving T =VC. We observe that increasing the size of the vertex cover as represented by the dilation factor $c = |T|/|VC|$ has only an incremental effect. This shows that much of the IP traceback effect is attained at the smaller vertex cover size (18.9%) which facilitates economy of coverage and deployment.

5.4 Maximal Filters vs. Semi-maximal Filters

All the results reported in this paper are, by default, based on semi-maximal filters. To ascertain the potential performance loss due to not using maximal filters, we compare filtering performance with respect to $\Psi_1(\tau)$ and $\Phi_2(1)$. Figure 5.6 (left) shows $\Psi_1(\tau)$ for 1997 Internet AS topology as a function of τ when performing route-based DPF with maximal versus semi-maximal filters under $R = tight$ and T being VC. We observe that the performance difference in IP traceback capability as captured by $\Psi_1(\tau)$ is minimal. For example, for $\tau = 5$, there is no performance difference. Figure 5.6 (right) compares $\Phi_2(1)$ for maximal and semi-maximal filters which, in fact, are equal. Thus the marginal performance difference coupled with space efficiency warrants the use of semi-maximal filters when implementing route-based DPF.

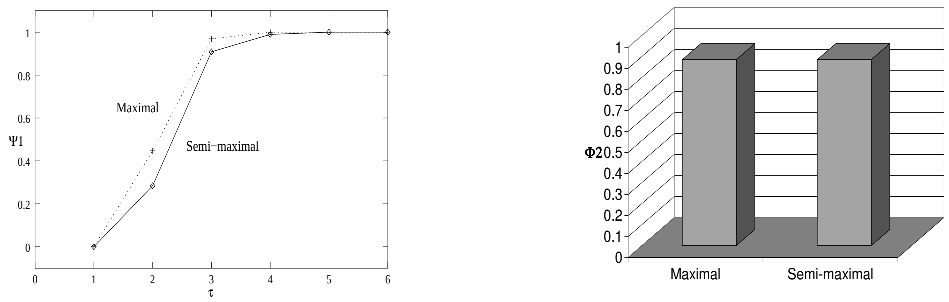


Figure 5.6: 1997 Internet AS topology. Left: Comparison of $\Psi_1(\tau)$ for maximal and semi-maximal filters. Right: Corresponding comparison of $\Phi_2(1)$ for maximal and semi-maximal filters.

5.5 Impact of Network Topology

5.5.1 Internet AS Topology

Figure 5.7 shows the vertex cover sizes (expressed as a percentage) and $\Psi_1(5)$ values for 1997–1999 Internet AS topologies. We observe that $|\mathrm{VC}|/n$—as well as $\Psi_1(5)$ and $\Phi_2(1)$—remain invariant over 1997–1999. In the rest of this section we focus on $\Psi_1(\tau)$ and discuss the results for $\Phi_2(1)$ when their performance is qualitatively different. The size of the vertex cover plays an important role as an

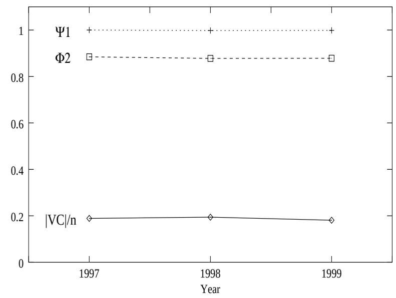


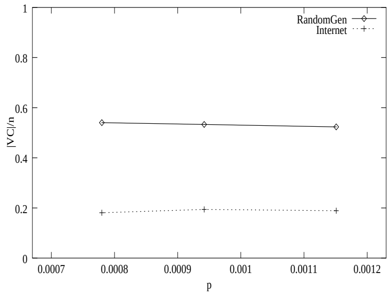
Figure 5.7: Vertex cover size $|\mathrm{VC}|/n$ and $\Psi_1(5)$, $\Phi_2(1)$ for 1997–1999 Internet AS topologies.

intermediate indicator and facilitator of filtering performance. In fact, the smaller the VC, the better the filtering performance in spite of the small coverage ratio—note that $\gamma = |\mathrm{VC}|/n$—which indicates that the VC property and its relative size is a useful indicator of connectivity property relevant to DPF performance.

5.5.2 Random Topology

We generate p-random graphs by connecting two nodes with link probability p. For a given Internet AS graph, we generate its corresponding random graph by setting $p = \frac{2e}{n(n-1)}$ where $e = |E|$. The specification and p values for 1997–1999 Internet AS topologies are shown in Figure 5.8 (left). The two families of graphs differ only in their connectivity pattern. Figure 5.8 (middle) shows vertex cover size of the generated random graphs and corresponding Internet AS topologies. On average, the VC sizes of

Year	n	e	p
1997	3015	5230	0.001151
1998	3878	7080	0.000942
1999	4872	9254	0.000780



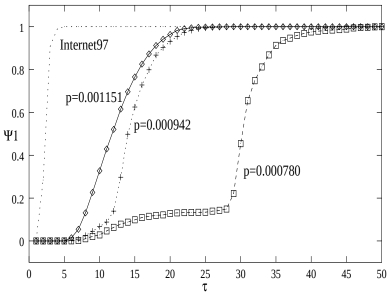


Figure 5.8: Left: Link probabilities for random graphs corresponding to 1997–1999 Internet AS topologies. Middle: $|\mathrm{VC}|/n$ as a function of p and comparison with Internet AS. Right: $\Psi_1(\tau)$ plot.

the random graphs are 2.5 larger than their Internet AS counterparts. Figure 5.8 (right) shows $\Psi_1(\tau)$ as a function of τ for different topologies. In spite of engaging more nodes when performing filtering, the performance as captured by $\Psi_1(\tau)$ is significantly less than that of Internet AS topology. Moreover, the performance difference amplifies as the size of the graph increases. Recall that the performance values for 1997–1999 Internet AS topologies (see Figure 5.5 (left)) stayed invariant.

5.5.3 Inet Topology Generator

We use Inet 2.0 [12][4], a network topology generator, for generating artificial topologies closer to the Internet in their connectivity structure than random graphs. Inet is designed to generate graph topologies with connectivity properties similar to Internet AS in terms of power-law structure. Figure 5.9 (left) shows the VC sizes of Inet generated graphs and their Internet AS counterparts for 1997–1999. We observe that the VC sizes of Inet graphs are about 50% larger than corresponding Internet AS graphs. Figure 5.9 (middle) shows $\Psi_1(\tau)$ as a function of τ for Inet, Internet AS, and random graphs. We observe, as expected, that filtering performance for Inet graphs is closer to that of Internet AS than random graphs.

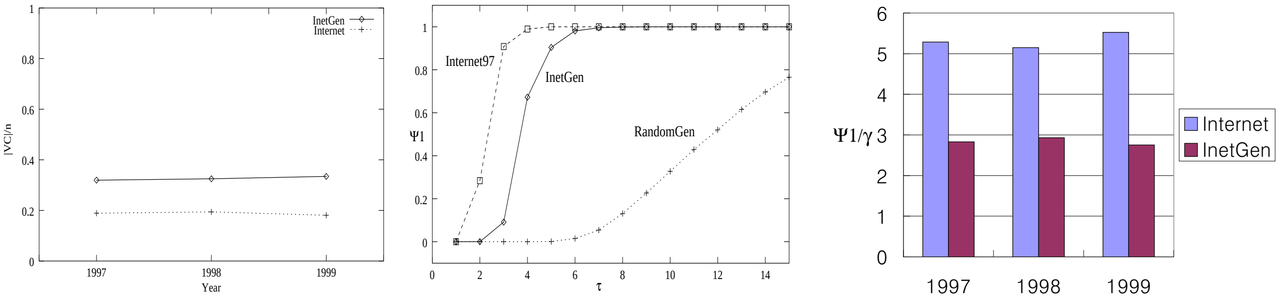
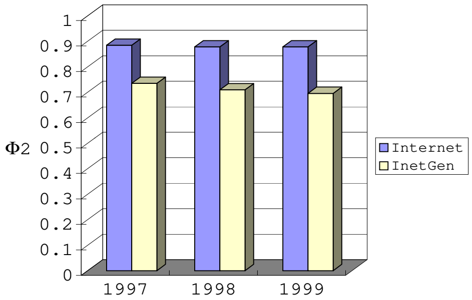


Figure 5.9: Left: VC sizes for Inet graphs and corresponding Internet AS graphs. Middle: Comparison of $\Psi_1(\tau)$ of Inet graph with Internet AS and random graphs. Right: Performance difference between Inet and Internet AS graphs normalized by VC size: $\Psi_1(5)/(|\mathrm{VC}|/n)$.

Figure 5.9 (right) shows *normalized* filtering performance $\overline{\Psi}_1(\tau) = \Psi_1(\tau)/\gamma$ for $\tau = 5$ where the relative size of the filter set is incorporated. Since $\Psi_1(1) = 1$ if $T = V$ no matter what the structure of the underlying topology, $\overline{\Psi}_1$ measures filtering performance per filter node (relative to $|V|$) which is a more accurate metric for comparative evaluation. Figure 5.9 (right) shows that there is significant difference in DPF performance between Inet and Internet AS topologies stemming, in part, from VC size difference. Inet is a topology generator whose primary feature is that of emulating power-law relations for vertex degrees as observed in [6]. The fact that the well-known VC graph property exhibits nontrivial gaps between Internet AS and Inet topologies indicates that more refined structure

[4]We also tested with benchmark graphs generated by Inet2.1—it was conveyed to us recently that Inet2.0 had a bug when generating large graphs of size 30K—with similar results. The Inet2.1 graphs resulted in a marginally smaller VC size—less than 2% difference—for graph sizes corresponding to 1997–1999 Internet AS topologies.



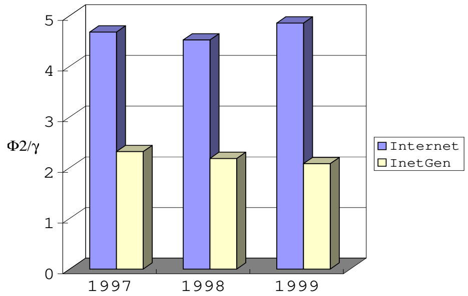


Figure 5.10: Left: Comparison of $\Phi_2(1)$ for Inet and 1997–1999 Internet AS topologies. Right: Corresponding comparison of normalized $\bar{\Phi}_2(1) = \Phi_2(1)/\gamma$.

may need to be uncovered within the family of power-law graphs to accurately capture the Internet's topological properties.

Figure 5.10 (left) shows a comparison of $\Phi_2(1)$ for Inet and 1997–1999 Internet AS topologies. As with $\Psi_1(\tau)$, we observe a persistent gap in reactive filtering performance which is consistent with the corresponding gap in the VC size. Figure 5.10 (right) shows normalized IP traceback performance $\bar{\Phi}_2(1) = \Phi_2(1)/\gamma$ for the same benchmark set-up which incorporates the size of the filter net in the performance measure. As expected, performance difference is further amplified.

5.5.4 Brite Topology Generator

Brite [14] is a network topology generator that, in addition to capturing power-law connectivity structure, seeks to inject spatial proximity in the constructive process. Brite specifies seven parameters: size of higher plane (HS), size of lower plane (LS), number of nodes (n), number of edges added for each new node (m), node placement (NP), preferential connectivity[5] (PC), and incremental growth (IG). When PC=0, a new node is connected to node i with Waxman's probability density [32], $p_i = \alpha e^{-d/(\beta L)}$, where $0 < \alpha, \beta \leq 1$, d is the Euclidean distance between two nodes, and L is the maximum distance between any two nodes. When PC=1, a new node connects to node i with probability $\frac{d_i}{\sum_{j \in C} d_j}$ where d_i is the degree of node i and C is the set of candidate neighbor nodes. With PC=2, the probability of connecting to node i is given by $\frac{p_i d_i}{\sum_{j \in C} p_j d_j}$. Thus PC=0 considers spatial proximity only, PC=1 focuses on power-law structure as captured by node degree distribution, and PC=2 is a hybrid.

Using HS=1000, LS=10, IG=1, and n=3015, test graphs were generated with the three PC options. The specification and results for VC size are shown in Figure 5.11 (left). Figure 5.11 (middle) and (right) show the performance effects with respect to $\Psi_1(\tau)$ and $\Phi_2(1)$, respectively. When PC=0, we observe that the graph generated—in addition to not being power-law—has too small VC (3.6%). Its performance with respect to $\Psi_1(\tau)$ and $\Phi_2(1)$ is closer than that of PC=1 and 2, however, the performance gap from the corresponding Internet AS topology for $\Phi_2(1)$ is significant, being worse

[5] The Brite generator [14] had a small bug with respect to option PC=2 which was fixed.

Topology	n	e	$\lvert\mathrm{VC}\rvert/n$
Internet	3015	5230	18.9%
Brite (PC=0)	3029	5978	3.6%
Brite (PC=1)	3006	5935	42.7%
Brite (PC=2)	3002	5908	44.1%

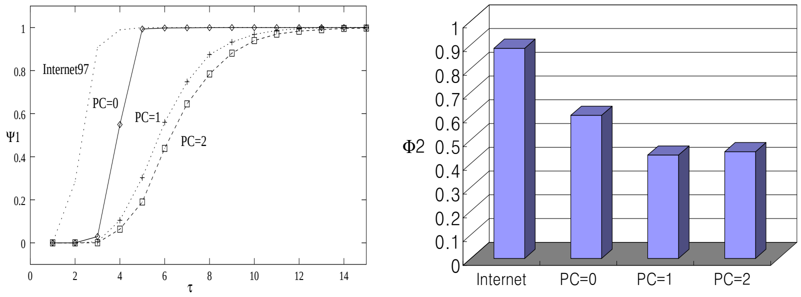


Figure 5.11: Left: VC sizes for Brite graphs with PC=0, 1, and 2. Middle: $\Psi_1(\tau)$ as a function of τ for PC=0, 1, and 2. Right: Corresponding $\Phi_2(1)$ plot.

than that of the Inet generator. For PC=1 and 2, the VC sizes are too large, and performance for both $\Psi_1(\tau)$ and $\Phi_2(1)$ significantly worse than Internet AS (and Inet). We have tried the Brite generator with other parameter specifications but were unsuccessful in getting topologies that resemble Internet AS both from the VC size and filtering performance perspectives. We have also tried extending option PC=2 by using the weighting $\alpha p_i + (1-\alpha)\frac{d_i}{\sum_{j \in C} d_j}$ to inject both spatial and degree sensitivity in a more controlled fashion. As α increases |VC| monotonically decreases, and for $\alpha = 0.13$ the VC size can be approximated to that of Internet AS with $\Psi_1(\tau)$ close to its Internet AS value. However, the corresponding $\Phi_2(1)$ performance is dismal (about 20%) when compared to Internet AS.

5.5.5 Rank-based Filter Placement

Figure 5.12 shows filtering performance of rank-based filter placement for a range of coverage ratios. Compared to VC-based filtering which has coverage ratio $\gamma = 0.189$, we observe that both $\Psi_1(\tau)$ and $\Phi_2(1)$ show diminished performance even when their coverage ratio is higher. Note that the greedy algorithm for vertex cover selects vertices that are able to cover the maximum number of remaining edges which is not the same as picking maximum degree nodes. The performance gap indicates that selection of high degree vertices as filtering sites is an important—but not the only—effect of VC-based filter selection on DPF performance in Internet AS topology. Indeed, rank-based filter placement can

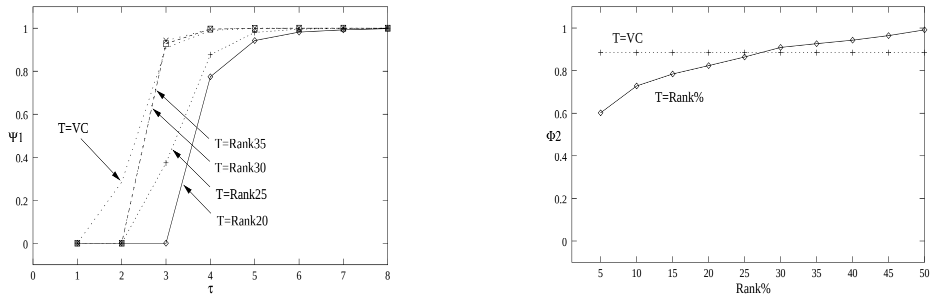


Figure 5.12: 1997 Internet AS topology. Left: Comparison of $\Psi_1(\tau)$ for rank-based filter placement with $\gamma = 0.20, 0.25, 0.30, 0.35$ versus VC-based T. Right: Corresponding comparison for $\Phi_2(1)$.

lead to inefficiencies by selecting high-degree nodes that are incident on edges that are already covered. This suggests that optimal filter placement may be an NP-hard problem[6].

5.6 Ingress Filtering

Section 4.5 showed that ingress filtering is not a viable strategy for achieving proactive and reactive filtering performance for DDoS attack prevention. Since AS belonging to T represent "trusted" domains where route-based DPF is guaranteed to be executed at its border routers, ingress filtering was assumed to be carried out by AS belonging to T. It is, however, conceivable that AS in T implement route-based DPF but do not assure ingress filtering. That is, they seek to protect themselves from external DoS attack flows while allowing DoS attacks to occur within their domain including those targeted at other domains. Figure 5.13 shows proactive and reactive filtering performance when AS in T perform route-based DPF but do not perform ingress filtering. Figure 5.13 (left) shows $\Psi_1(\tau)$ for 1997–1999 Internet AS topologies. We observe that there is a performance penalty such that $\Psi_1(5) \neq 1$. On

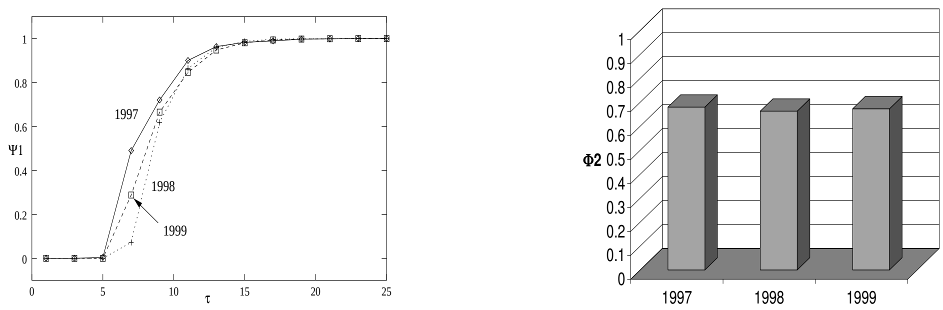


Figure 5.13: Route-based DPF without ingress filtering. Left: $\Psi_1(\tau)$ as a function of τ for 1997–1999 Internet AS topologies. Right: Corresponding $\Phi_2(1)$ values.

the other hand, $\Psi_1(20) = 1$ for all three years. That is, IP traceback can localize the attack site to within 20 locations. This is worse than 5—the number achievable with ingress filtering—however, considering that there were in the range 3000–5000 autonomous systems during 1997–1999, 20 is still a small constant, and thus managable number. Figure 5.13 (right) shows the corresponding $\Phi_2(1)$ values. $\Phi_2(1)$ drops from around 90% to 70% which is still significantly higher than the 20% proactive effect achievable with ingress filtering alone. Interestingly, the performance gap of 20% roughly corresponds to the coverage ratio $\gamma = |T|/n$ for VC in the Internet AS topologies.

5.7 Multi-path Routing

If multiple paths are permitted when routing packets from source to destination, the more easily packets can elude route-based filtering when using spoofed source IP addresses. Figure 5.14 shows the impact of multi-path routing on filtering performance. Figure 5.14 (left) and (middle) show that traceback

[6] This is an interesting problem to explore in future work. Generally, it may be fruitful to investigate whether network resource allocation problems can be more accurately solved—NP-hardness will remain—for a family of power-law graphs.

capability as captured by $\Psi_1(\tau)$ decreases gradually as the number of multi-paths allowed is increased. A similar result holds for $\Phi_2(1)$ and is shown in Figure 5.14 (right). Collectively, these performance plots show that presence of multi-paths—a more common phenomenon in Internet AS topologies than in router topologies—has a graded effect and does not significantly impact the effectiveness of route-based DPF.

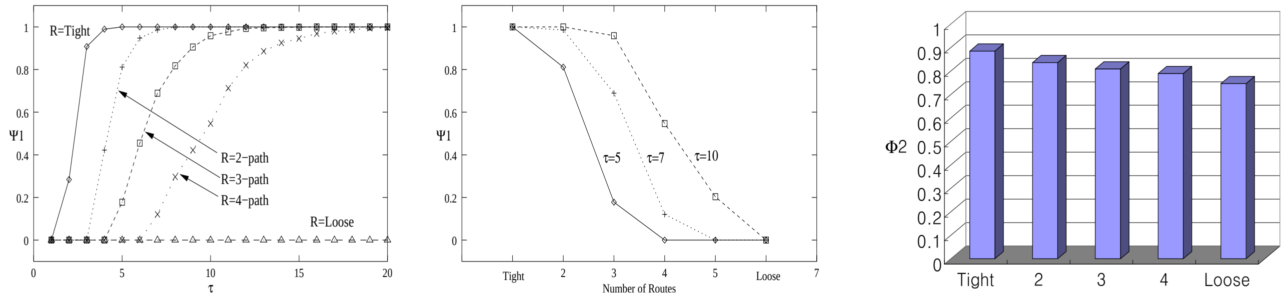


Figure 5.14: Effect of multi-path routing for 1997 Internet AS topology. Left: $\Psi_1(\tau)$. Middle: Distribution of $\Psi_1(5)$ and $\Psi_1(10)$ over the number of multi-paths. Right: $\Phi_2(1)$ for R from *tight* to *loose*.

6 Conclusion

We have described a proactive/reactive approach to distributed DoS attack prevention based on route-based distributed packet filtering. We have shown route-based DPF's efficacy at proactively curtailing spoofed IP flows from reaching their intended targets, including the drastically reduced Internet AS sites from which such attacks can be launched. We have shown that perfect proactivity—no spoofed IP flow can penetrate—is intrinsically difficult to achieve in Internet AS topologies while maintaining sparse (e.g., 20% or less AS sites deploying route-based DPF) coverage. However, this is mitigated by the fact that those spoofed IP flows that can penetrate the filter net can be localized to within 5 candidate sites which facilitates efficient IP traceback. Compared to probabilistic packet marking, we have shown that route-based DPF is able to solve PPM's three key weaknesses (see Section 2). We have also shown that the filtering effect achieved by route-based DPF is sensitive to the underlying Internet AS connectivity structure. In particular, we have shown that power-law structure of Internet AS topology plays an important role in facilitating proactive/reactive filtering. Finding efficient implementations for computing semi-maximal filters and evaluating the costs associated with deployment and router overhead vis-à-vis the performance benefits of route-based DPF is a major challenge and a task for future work.

References

[1] G. Banga, P. Druschel, and J. Mogul. Resource containers: A new facility for resource management in server systems. In *Proc. of the third USENIX/ACM Symp. on Operating Systems Design and Implementation (OSDI'99)*, pages 45–58, February 1999.

[2] S. Bellovin. ICMP traceback messages, March 2000. Internet Draft: draft-bellovin-itrace-00.txt (expires September 2000).

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

[4] Computer Emergency Response Team (CERT). CERT Advisory CA-2000-01 Denial-of-service developments, January 2000. http://www.cert.org/advisories/CA-2000-01.html.

[5] CERT/CC, SANS Institute, and CERIAS. Consensus roadmap for defeating distributed denial of service attacks, February 2000. A Project of the Partnership for Critical Infrastructure Security, http://www.sans.org/ddos_roadmap.htm.

[6] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the Internet topology. In *Proc. of ACM SIGCOMM*, pages 251–262, 1999.

[7] P. Ferguson and D. Senie. Network ingress filtering: Defeating denial of service attacks which employ IP source address spoofing, May 2000. RFC 2827.

[8] Lee Garber. Denial-of-service attacks rip the Internet. *Computer*, pages 12–17, April 2000.

[9] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979.

[10] R. Govindan and A. Reddy. An analysis of Internet inter-domain topology and route stability. In *Proc. IEEE INFOCOM '97*, 1997.

[11] John Howard. *An Analysis of Security Incidents on the Internet*. PhD thesis, Carnegie Mellon Univerisity, August 1998.

[12] C. Jin, Q. Chen, and S. Jamin. Inet: Internet Topology Generator. Technical Report CSE-TR-443-00, Department of EECS, University of Michigan, 2000.

[13] C. Meadows. A formal framework and evaluation method for network denial of service. In *Proc. of the 1999 IEEE Computer Security Foundations Workshop*, June 1999.

[14] A. Medina and I. Matta. Brite: A flexible generator of Internet topologies. Technical Report BU-CS-TR-2000-005, Boston University, January 2000.

[15] Robert Morris. A weakness in the 4.2BSD Unix TCP/IP software. Technical Report Computer Science #117, AT&T Bell Labs, February 1985.

[16] National Laboratory for Applied Network Research. Routing data, 2000. Supported by NFS, http://moat.nlanr.net/Routing/rawdata/.

[17] NightAxis and Rain Forest Puppy. Purgatory 101: Learning to cope with the SYNs of the Internet, 2000. Some practical approaches to introducing accountability and responsibility on the public internet, http://packetstorm.securify.com/papers/contest/RFP.doc.

[18] J. Pansiot and D. Grad. On routes and multicast trees in the Internet. *Computer Communication Review*, 28(1):41–50, 1995.

[19] C. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity.* Prentice Hall, Inc., 1982.

[20] K. Park and H. Lee. On the effectiveness of probabilistic packet marking for IP traceback under denial of service attack. Technical Report CSD-TR 00-013, Department of Computer Sciences, Purdue University, June 2000. To appear in IEEE INFOCOM 2001.

[21] Jon Postel. Internet protocol, September 1981. RFC 791.

[22] Glenn Sager. Security fun with OCxmon and cflowd, November 1998. Presentation at the Internet 2 Working Group.

[23] S. Savage, D. Wetherall, A. Karlin, and T. Anderson. Practical network support for IP traceback. In *Proc. of ACM SIGCOMM*, pages 295–306, August 2000.

[24] C. Schuba, I. Krsul, M. Kuhn, E. Spafford, A. Sundaram, and D. Zamboni. Analysis of a denial of service attack on TCP. In *Proc. of the 1997 IEEE Symp. on Security and Privacy*, pages 208–223, May 1997.

[25] Daniel Senie. Changing the default for directed broadcasts in routers, August 1999. RFC 2644.

[26] D. Song and A. Perrig. Advanced and authenticated marking schemes for IP traceback. Technical Report UCB/CSD-00-1107, Computer Science Department, University of California, Berkeley, 2000. To appear in IEEE INFOCOM 2001.

[27] O. Spatscheck and L. Peterson. Defending against denial of service attacks in Scout. In *Proc. of the third USENIX/ACM Symp. on Operating Systems Design and Implementation (OSDI'99)*, pages 59–72, February 1999.

[28] Robert Stone. Centertrack: An IP overlay network for tracking DoS floods. In *Proc. of 9th USENIX Security Symposium*, August 2000.

[29] Cisco Systems. Characterizing and tracing packet floods using Cisco routers, Aug 1999.

[30] Computer Emergency Response Team. Denial of service, February 1999. Tech Tips, http://www.cert.org/tech_tips/denial_of_service.html, 2nd revision.

[31] Computer Emergency Response Team. Results of the distributed-systems intruder tools workshop, November 1999. http://www.cert.org/reports/dsit_workshop.pdf.

[32] B. Waxman. Routing of multipoint connections. *IEEE Journal of Selected Areas in Communications*, pages 6(9):1617–1622, December 1988.

[33] E. Zwicky, S. Cooper, D. Chapman, and D. Ru. *Building Internet Firewalls.* O'Reilly & Associates, Inc., 2nd edition, 2000.