On the Effectiveness of Probabilistic Packet Marking for IP Traceback under Denial of Service Attack*

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

CSD-TR 00-013
June 10, 2000

Abstract

Effective mitigation of denial of service (DoS) attack is a pressing problem on the Internet. In many instances, DoS attacks can be prevented if the spoofed source IP address is traced back to its origin which allows assigning penalties to the offending party or isolating the compromised hosts and domains from the rest of the network. Recently IP traceback mechanisms based on probabilistic packet marking (PPM) have been proposed for achieving traceback of DoS attacks. In this paper, we show that probabilistic packet marking—of interest due to its efficiency and implementability vis-à-vis deterministic packet marking and logging or messaging based schemes—suffers under spoofing of the marking field in the IP header by the attacker which can impede traceback by the victim. We show that there is a trade-off between the ability of the victim to localize the attacker and the severity of the DoS attack, which is represented as a function of the marking probability, path length, and traffic volume. The optimal decision problem—the victim can choose the marking probability whereas the attacker can choose the spoofed marking value, source address, and attack volume—can be expressed as a constrained minimax optimization problem, where the victim chooses the marking probability such that the number of forgeable attack paths is minimized. We show that the attacker's ability to hide his location is curtailed by increasing the marking probability, however, the latter is upper-bounded due to sampling constraints. In typical IP internets, the attacker's address can be localized to within 2–5 equally likely sites which renders PPM effective against single source attacks. Under distributed DoS attacks, the uncertainty achievable by the attacker can be amplified, which diminishes the effectiveness of PPM.

*This work was supported in part by NSF grant EIA-9972883.

†Additionally supported by NSF grants ANI-9714707, ANI-9875789 (CAREER), ESS-9806741, and grants from the Purdue Research Foundation, Santa Fe Institute, and Sprint.

‡Additionally supported by 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 [6, 9, 10, 15, 25, 27] 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. Susceptibility to DoS is an intrinsic problem of any service provisioning system—albeit amplified in the networked digital environment due to speed and automation—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. Firewalls and filters running at gateway routers can shield a network system from outside DoS flows, but if their function includes selectively admitting valid client flows resident outside a guarded domain, then this very filtering service can be impeded by DoS attacks targeted at the gateway. 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 available on the compromised system. In this paper, we address the source identification problem and analyze its properties from a probabilistic packet marking approach motivated by its appealing feature with respect to efficiency and implementability.

1.2 A Case for Probabilistic Packet Marking

A "simple" way of identifying the physical source of DoS traffic is by elimination of IP address spoofing. If all ISPs were to implement mechanisms for preventing IP source address spoofing—

which is, technically, easy to do—then source identification (also called IP traceback in [18]), would be solved. A less drastic measure, based on packet marking, would allow spoofed packets to pass through, however, with the corrected source IP address overwriting the spoofed source IP address. For various practical reasons, this may be difficult to achieve or require a prolonged period to be broadly deployed on the Internet. Thus, there is a need for incrementally deployable techniques that may not completely eliminate the DoS problem, but reduce it to a "manageable" level.

A number of recent works have studied the problem of tracing the physical source of a DoS attack [15, 18, 2, 24, 17, 23]. In *deterministic packet marking* [3], the source of a traffic flow is recovered by employing tracing information inscribed in the packet. Packet marking can be viewed as a form of "stateless logging" which emulates the capability of path recovery by router based information logging [17, 23], without incurring the latter's statefulness and associated space overhead. A related method is messaging based path recovery [2] which uses control messages emitted from routers conveying path information to destination nodes. Thus (router) statelessness is achieved, however, at the cost of message overhead. Packet marking—and, to some extent, messaging—follows the end-to-end paradigm [5] where complexity of path recovery is pushed to the edge while imposing a minimal footprint with respect to per-hop network support requirements.

A significant drawback of deterministic packet marking (DPM) is the increasing packet header size requirement which grows linearly with hop count. In addition to amplifying packet size—a form of communication complexity—dynamically variable packet sizes complicate router processing which can impart nontrivial overhead to achieving terabit-per-second switching speeds. In *probabilistic packet marking* [18], each router probabilistically inscribes its local path information onto a traversing packet so that the destination node (i.e., victim of an attack) can reconstruct, with high probability, the complete path traversed by inspecting the markings on the received packets, assuming the attack volume is sufficiently high. This corresponds to probabilistically "sampling" the route undertaken by an attack using *constant space* in the packet header independent of hop count, which provides the key advantage over deterministic packet marking. In probabilistic marking, when a router decides to mark based on a coin toss with marking probability p, it overwrites the information contained in the marking field, thus erasing any possible markings by upstream routers. Thus, for PPM to work, it is necessary that $p < 1$. By the same token, with some positive probability, a packet will arrive at the destination *without having been marked by any of the intermediate routers*. This reveals—above and beyond the need for requiring a set of packets to recover the attack path—a potentially serious weakness of PPM since the marking field may contain a value, inscribed by the attacker, whose aim is to confuse or impede the victim's ability to traceback. In this paper, we give a comprehensive treatment of the *spoofed marking field problem*.

1.3 New Contributions

We analyze the effectiveness of probabilistic packet marking for IP traceback under DoS attack. Our technical contributions are two-fold.

First, we define the source identification problem in the framework of probabilistic packet marking (PPM) and present a comprehensive analysis of its properties. We show that PPM is vulnerable to spoofing of the marking field in the IP header by the attacker which can impede traceback by the victim. We show that there is a trade-off relation between the ability of the victim to localize the attacker and the severity of the DoS attack, which is a function of the marking probability, path length, and traffic volume. The optimal decision problem—the victim can choose the marking probability and the attacker chooses the spoofed marking value, source address, and attack volume—can be expressed as a constrained minimax optimization problem: the victim selects the marking probability such that the number of forgeable attack paths is minimized and the attacker chooses the traffic volume and marking value to maximize uncertainty. We show that the attacker's ability to hide his location is curtailed by increasing the marking probability, however, the degree to which the victim can delimit the attacker's injection of uncertainty is bounded by sampling constraints. In particular, the attacker, by choosing a minimal attack traffic volume, can amplify the number of equally likely forged attack paths to $d - 1$, independent of the victim's choice of marking probability, where $d \geq 2$ is the path length. In IP internetworks with hop count 25 or less (as is the case on the Internet) and attack volume in the thousands of packets—to qualify as a DoS attack, the victim's resources must be nontrivially taxed—we show that the attacker's address can be localized to within 2–5 equally likely sites which renders PPM effective against single source attacks.

Second, we analyze the consequences of the attacker mounting distributed DoS attacks where each partaking attack host transmits a minimal traffic volume to maximize anonymity, and attack volume amplification is achieved by engaging a large number of sources. We show that for a given attack volume, by mounting a distributed denial of service attack, the uncertainty injected into IP traceback can be amplified above and beyond the the effect afforded by distributedness. Thus PPM, while effective against single-source attacks, is potentially vulnerable when subject to distributed DoS attacks.

The rest of the paper is organized as follows. In the next section, we give a summary of related works. In Section 3, we discuss the core issues surrounding source identification and define the IP traceback problem in the framework of PPM. In Section 4, we present the analysis of single-source DoS attack which is complemented by numerical evaluations using Internet related parameters. In Section 5 we study the distributed DoS case and show its detrimental consequences on PPM. We conclude with a discussion of our results.

2 Related Work

Several types of DoS attacks have been identified [8, 9, 15, 25], 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 [15]. While host-based DoS attacks are more easily traced and managed, network-based DoS attacks which exploit weaknesses of the TCP/IP protocol suite [14], represent a more subtle and challenging threat [15, 18]. Network-based DoS attacks, by default, employ spoofing to forge the source address of DoS packets to hide the identity of the physical source [27]. Previous works have focused on detecting DoS attacks and mitigating their detrimental impact upon the victim [1, 11, 19, 21]. This approach does not eliminate the problem, nor does it deter potential attackers. As a means of preventing network-based DoS attacks, edge filtering in border gateways has been proposed for limiting IP source address spoofing [4, 7, 20]. The filtering rules can affect dropping of forged packets using egress filtering in user organizations and ingress filtering in ISPs [8, 26].

A number of recent works have studied source identification (also called IP traceback in [18]) 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 [24]. 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 including legal issues [24]. The audit trail approach facilitates tracing via traffic logs at routers and gateways [12, 17, 23]. 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 are expected to switch at Tbps rates—which can exert a significant burden. In behavioral monitoring, the likely behavior of an attacker during a DoS attack is monitored to identify the source [15]. 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 [16]. 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.

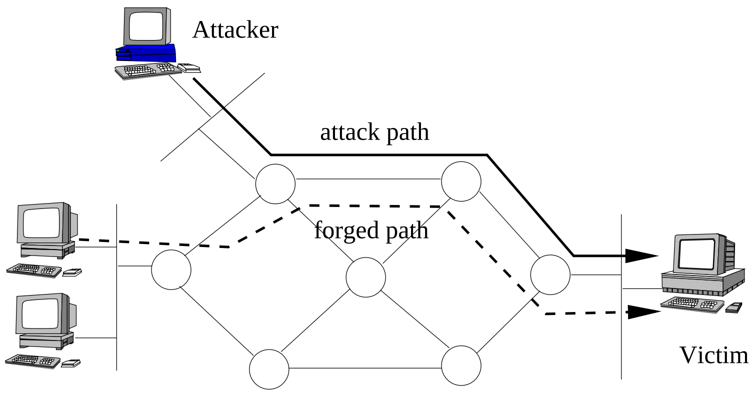

Figure 2.1: In probabilistic packet marking, an attacker can forge a path that is equally likely as the true attack path by transmitting corrupted packets that reach the victim untouched (i.e., unmarked).

Probabilistic packet marking [3, 18] achieves 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 latter has two important, and opposing, effects: (a) discovery of correct path information by sampling which aids the victim's objective of traceback, and (b) injection of corrupted information by the attacker. In the latter, with a certain probability a packet—however formatted by the attacker—will travel through untouched, which can impede the victim's ability to identify the true attack path. This is illustrated in Figure 2.1. More generally, the number of forgeable paths that are from an information-theoretic point-of-view indistinguishable with respect to their validity as the true attack path can further render source identification difficult if their numbers are large. In [18], issue (a) was analyzed yielding a partial and, perhaps, overly optimistic evaluation of probabilistic packet marking as a DoS prevention method. The principal contribution of [18] lies in the investigation of coding issues aimed at further reducing the (constant) marking bits needed in the IP header via fragmentation. In this paper, we study the critical issue (b)—the attacker's ability to inject misleading information—and give a comprehensive analysis of the effectiveness of PPM under single-source and distributed DoS attacks, complemented by numerical evaluations.

3 Probabilistic Packet Marking and Traceback

3.1 Network Model

The network is given as a directed graph $G = (V, E)$ where V is the set of nodes and E is the set of edges. V can be further partitioned into end systems (leaf nodes) and routers (internal nodes). The edges denote physical links (directed media) between elements in V. Let $S \subset V$ denote the set of *attackers* and let $t \in V \setminus S$ denote the *victim*. We will first consider the case when $|S| = 1$ (single-source attack) and treat the distributed DoS attack case separately. We assume that routes are fixed,[1] and

$$\mathcal{A} = (s, v_1, v_2, \ldots, v_{d-1}, t)$$

comprised of $d-1$ routers (or hops) $v_1, \ldots, v_{d-1}$, and of path length d is called an *attack path*. A path $\mathcal{B}$, $\mathcal{B} \neq \mathcal{A}$, with destination node t and source node u ($u \neq s$) is called a *forgeable path*.

3.2 Probabilistic Marking

3.2.1 Definition

Let N denote the number of packets sent from s to t. We will leave the time duration or interval unspecified (typically $N \gg 1$ and DoS attacks occur over a concentrated time period). A packet x is assumed to have a *marking field* where the identity of a link $(v, v') \in E$ just traversed can be inscribed. A packet travels on the attack path $\mathcal{A}$ sequentially. At a hop $v_i \in \{v_1, \ldots, v_d\}$, packet x is marked with the edge value (v_{i-1}, v_i), $i = 1, \ldots, d$, with probability p ($0 \leq p \leq 1$) where $v_0 = s$ and $v_d = t$. This process is called *probabilistic marking*. If a packet x was already marked by a previous router, the new mark will replace or overwrite the old one. Let x_j, $j = 0, 1, \ldots, d$ denote the value inscribed in the marking field at node v_i. Let $X_1, X_2, \ldots, X_d$ be a set of i.i.d. binary random variables where $\Pr\{X_i = 1\} = p$, $\Pr\{X_i = 0\} = 1 - p$, and $X_i = 1$ indicates that marking

[1] On the IP Internet, about 95% of TCP sessions do not experience route changes during their connection lifetime. Generalization of PPM under dynamic routing (the routing process must be specified) is a problem for future work.

was performed at node v_i. Thus x_j is a random variable depending on $X_j, X_{j-1}, \ldots, X_1$, and we will be interested in the behavior of x_d.

3.2.2 Path Sampling

Let $\alpha_i(p)$ denote the probability that the arriving packet is marked at node v_i but nowhere else. Thus

$$\alpha_i(p) = \Pr\{x_d = (v_{i-1}, v_i)\} = p(1-p)^{d-i}.$$

The probability that a packet sent from the attacker reaches the victim without being marked at any of the routers is $\alpha_0(p) = (1-p)^d$. As with IP source address spoofing, the attacker may choose to inscribe a value x_0 which serves the purpose of hiding the attacker's identity. When N packets are transmitted, the expected number of packets reaching target t marked with the edge value (v_{i-1}, v_i) is $n_i(p) = N\alpha_i(p)$. Note that

$$\alpha_1(p) \leq \alpha_2(p) \leq \cdots \leq \alpha_d(p),$$

and to receive a marked packet from v_1 containing the first link value (s, v_1) requires $N \geq 1/\alpha_1(p)$. Since N (the attack volume) is a variable under the attacker's control, from a purely sampling point-of-view, edge (s, v_1) is the "weakest link" requiring the most samples (i.e., packet transmissions) to recover. The expected number of samples needed to receive marked packets from *all* routers requires a logarithmic correction term, and is bounded above by $(\text{const} \cdot \ln d)/\alpha_1(p)$. This follows from the disjointness of $\alpha_i(p)$ and an application of the coupon collector's problem using the relaxed probability $\alpha_1(p)$, which yields the well-known solution $d \ln d + O(d)$. This has also been noted in [18].

3.2.3 Marking Field Spoofing

When N packets are sent in the course of a DoS attack, the attacker can expect $n_0(p) = N(1-p)^d$ containing the attackers inscribed value x_0 to reach the target untouched. By "corrupting" the marking field—in addition to spoofing the IP source address—the attacker may negatively impact the path reconstruction capability of the victim based on the N packets received. The larger the fraction of corrupted marking field packets, the more damage the attacker can exact. What values to inscribe to achieve maximum effect is treated in the next section. With respect to the weakest point v_1, we are interested in the p values for which

$$\begin{aligned} n_0(p) \geq n_1(p) \quad &\Leftrightarrow \quad \alpha_0(p) \geq \alpha_1(p) \\ &\Leftrightarrow \quad (1-p)^d \geq p(1-p)^{d-1} \end{aligned} \tag{3.1}$$

which has the solution $p \leq 1/2$. That is, if $p \leq 1/2$ then as many spoofed packets will arrive as packets marked with the link value (s, v_1). In general, we may consider the case

$$\alpha_0(p) \geq \sum_{i=1}^{d} \alpha_i(p) \quad \Leftrightarrow \quad (1-p)^d \geq 1-(1-p)^d \tag{3.2}$$

where the corrupted packets are in the absolute majority which holds for $p \leq 1 - 2^{-1/d}$. Thus, for $d = 10$, the inequality holds if $p \leq 0.067$. Figure 3.1 (left) shows the ratio $\alpha_0(p)/\alpha_1(p)$ as a function of p, and Figure 3.1 (right) shows $\alpha_0(p)/\sum_i \alpha_i(p)$ as a function of p for $d = 10, 25$.

Whereas N and x_0 are under the attacker's control, the marking probability is a system parameter and, thus, the purview of the victim. The optimal selection of N and x_0 by the attacker, and correspondingly optimal selection of p by the victim to achieve their individual, conflicting objectives lies at the heart of the probabilistic PPM approach to source identification.

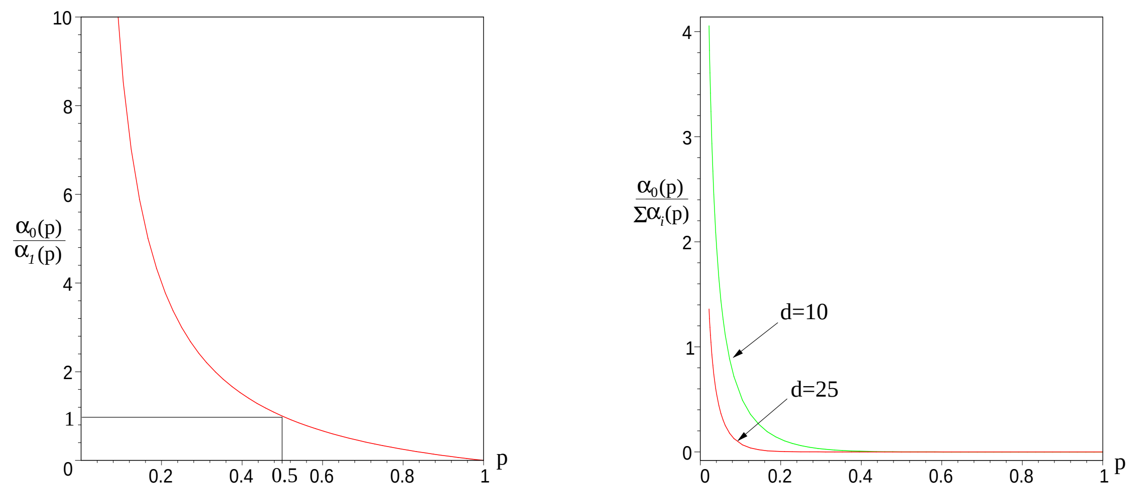

Figure 3.1: Left: $\alpha_0(p)/\alpha_1(p)$ as a function of p. Right: $\alpha_0(p)/\sum_i \alpha_i(p)$ as a function of p for $d = 10, 25$.

3.3 Traceback Problem

Consider an attacker with attack path $\mathcal{A} = (s, v_1, \ldots, v_{d-1}, t)$ and forgeable paths $\mathcal{B}_i = (u_i, v_1, \ldots, v_{d-1}, t)$, $i = 1, \ldots, m$, at distance d, joined at v_1 forming the (caterpillar) subgraph

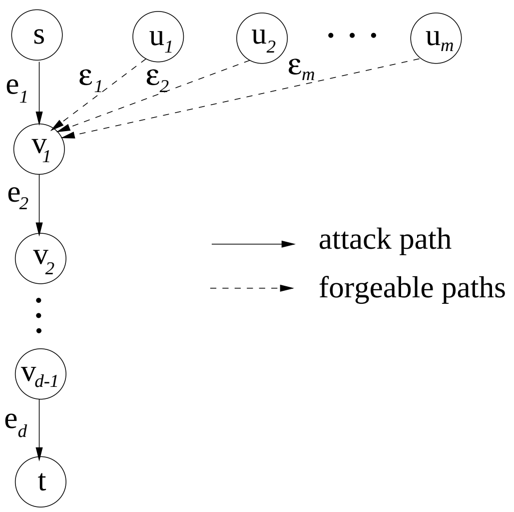

Figure 3.2: The attack path $(s, v_1, v_2, \ldots, v_{d-1}, t)$ and a set of m forgeable paths $(u_i, v_1, \ldots, v_{d-1}, t)$, $i = 1, \ldots, m$, joined at v_1.

shown in Figure 3.2. This particular attack pattern is of interest (i) because it targets the "weakest" point of probable path recovery by the victim according to (3.1), (ii) attacker s can generate packets that, unless marked at v_1, will be indistinguishable from real packets originating at u_i and arriving at t, (iii) other attack configurations can be analyzed using the tools developed for the caterpillar subgraph, and (iv) the concepts underlying optimal decision making by both attacker and victim are easily brought out. The traceback problem in a caterpillar graph is a special case of the traceback problem in general topologies, which is discussed in Appendix A. One of the three decision variables—the attacker's marking field spoof variable x_0—can be fixed by the following information-theoretic argument. Let $n_i^s(p)$ be the number of spoofed packets arriving at t with the marking field containing (u_i, v_1). Assume $n_0(p) = \sum_{i=1}^m n_i^s(p)$. That is, all packets transmitted by the attacker are inscribed with spoofing values from the link set $\{(u_i, v_1) : i = 1, 2, \ldots, m\}$. If it holds that

$$n_1(p) = n_1^s(p) = n_2^s(p) = \cdots = n_m^s(p), \tag{3.3}$$

then by (ii) all $(m+1)$ paths are *equally likely*—i.e., the attack could have been undertaken from any of the nodes $s, u_1, u_2, \ldots, u_m$ yielding the same outcome in terms of collected marking values at t. Of course, by the probabilistic nature of the marking process, exact equality cannot be expected to hold. Instead, if the marginal densities can be equated

$$\alpha_1(p) = \alpha_1^s(p) = \alpha_2^s(p) = \cdots = \alpha_m^s(p), \tag{3.4}$$

entropy is maximal, and by symmetry, each of the nodes $\{s, u_1, u_2, \ldots, u_m\}$ is an equally likely candidate. We will call m—a function of p and spoofing variable x_0—the *uncertainty factor* with respect to marking probability p. In the context of traceback, the uncertainty factor m is the objective function for measuring the effectiveness of traceback. The larger m is, the more the processing cost incurred by the victim to trace back the attack source. Thus, the objective of the attacker is to maximize m, whereas the objective of the victim is to minimize m. A minimax optimization problem for the attacker and victim can be formulated as follows:

$$\begin{gathered} \min_{p} \max_{x_0} \; m(p, x_0) \\ \text{subject to (3.4)} \end{gathered} \tag{3.5}$$

where the maximum is over all distributions of x_0 viewed as a random variable. The minimax formulation biases toward the victim. Interchanging the optimization order yields the corresponding maximin formulation. The formulation in (3.5) does not incorporate the attack volume N and thus unduly favors the victim. A sampling constraint is added by requiring

$$N\alpha_1(p) = Np(1-p)^{d-1} \geq 1. \tag{3.6}$$

Thus the refined minimax optimization reflecting the victim's sampling constraint is given by

$$\begin{gathered} \min_{p} \max_{x_0, N} \; m(p, x_0) \\ \text{subject to (3.4) and (3.6).} \end{gathered} \tag{3.7}$$

Note that N is incorporated as part of the attacker's decision variable due to constraint (3.6). $N\alpha_1(p)$ as a function of p has a unimodal (or bell) shape with peak at $p = 1/d$. Thus decreasing N can shrink the size of the feasibility region defined by (3.6).

4 Analysis of Single-Source DoS Attack

This section analyzes PPM under single-source DoS attacks. We first derive performance bounds for the minimax optimization problem, and then give numerical evaluations using Internet related parameters that complement the analytical results.

4.1 Minimax Optimization

A necessary condition for (3.4) to hold is that when transmitting a packet, the attacker inscribes spoofed link values with uniform probability, i.e.,

$$\Pr\{x_0 = (u_i, v_1)\} = \frac{1}{m}, \qquad i = 1, 2, \ldots, m. \tag{4.1}$$

Condition (4.1) can be further derandomized—i.e., replaced by a deterministic procedure that emulates uniform generation—if information contained in the sequential arrival of marked/spoofed packets is not considered. In conjunction with (4.1), a necessary and sufficient condition for (3.4) is

$$\begin{aligned} m\,\alpha_1(p) = \alpha_0(p) \quad &\Leftrightarrow \quad m\,p(1-p)^{d-1} = (1-p)^d \\ &\Leftrightarrow \quad m = \frac{1}{p} - 1 \end{aligned} \tag{4.2}$$

That is, given p (determined by the victim), the attacker can achieve an uncertainty factor of $m = (1/p) - 1$. Thus $m = (1/p) - 1$ is the *maximal* uncertainty factor satisfying (3.4) for a given p. Without the sampling constraint (3.6), the victim can affect

$$\min_p \left(\frac{1}{p} - 1 \right) = 0$$

since $0 \le p \le 1$. If $p = 1$ is *disallowed*—necessary for probabilistic path discovery when the hop count d is at least 2—then

$$m = (1/p) - 1 \searrow 0 \quad \text{as} \quad p \nearrow 1.$$

That is, the uncertainty factor achievable by the attacker becomes null since, m being an integer representing the in-degree of router v_1, only $\lfloor m \rfloor$ matters. With the sampling constraint (3.6) constraining the victim from choosing p arbitrarily close to 1, we need to compute the min-max over the feasible region

$$L = \{(p, N) : Np(1-p)^{d-1} \ge 1\}$$

defined by (3.6) where L is parameterized by the attack distance d. It can be checked that for all $d \ge 2$, L is convex in p. Thus the feasible region L defined by both the attacker and victim's moves is a union of convex sets L_N (the set L keeping the second coordinate fixed at N) for $N \ge N_0$ where $N_0 = N_0(d)$ is the least number—a function of d—such that $(p, N_0) \in L$ for some p.

Theorem 1 *For all $d \ge 2$, L_N is convex. Furthermore, $L_{N'} \supseteq L_N$ if $N' \ge N$.*

Proof. Since $Np(1-p)^{d-1}$ is a quasi-concave function in p, convexity of L_N follows immediately. The second property follows from $N'p(1-p)^{d-1} \ge Np(1-p)^{d-1}$ if $N' \ge N$. ■

Theorem 1 shows that the minimax problem can be viewed as a sequence of convex minimization problems of the objective function $(1/p) - 1$ over L_N for $N = N_0, N_0 + 1, \ldots$. Thus there is a unique solution. The next result gives a performance bound on the attacker's ability to hide his identity under PPM.

Theorem 2 *Let m^* be the solution to the constrained minimax problem given by* (3.7). *Then* $m^* \leq d - 1$.

Proof. Since constraint (3.4) has been incorporated into the objective function, what remains is to find the minimax value $\min_p \max_N m(p)$ over the feasible region L. For all $N \geq 1$, $d \geq 2$, it can be checked that $n_1(p) = Np(1-p)^{d-1}$ attains its maximum at $p = 1/d$. Substituting $p = 1/d$ into $m(p) = (1/p) - 1$ yields $d - 1$. Noting that $m(p)$ is monotone decreasing in p completes the proof. ■

Theorem 2 shows that the maximum achievable uncertainty factor—i.e., equally likely forged paths—cannot exceed $d - 1$, the distance between the attacker and victim. Thus the farther the attack site from the target, the more uncertainty can be injected. On the Internet [28], most path lengths are bounded by 25, and thus this puts an upper bound on the effectiveness of single-source DoS attacks when subject to probabilistic packet marking. An immediate consequence of Theorem 2 is the following corollary which shows that $d - 1$ can be tight.

Corollary 1 *If $N = d^d/(d-1)^{d-1}$ then $m^* = d - 1$.*

Proof. Substitute $p = 1/d$ into $Np(1-p)^{d-1} = 1$. Noting the unimodal shape of $Np(1-p)^{d-1}$ finishes the proof. ■

Thus the attacker, by judiciously choosing the attack volume, can maximally hide his identity given by $d-1$. Since $d^d/(d-1)^{d-1} \propto d$, this occurs at a drastic cost in reduced attack volume which may fail to affect significant "denial of service" at the target, thus taking the bite out of the attack.

4.2 Approximation of Uncertainty Factor

To find a feasible region of p for $Np(1-p)^{d-1} \geq 1$, we need to solve the equation $Np(1-p)^{d-1} = 1$. This equation is transformed to the polynomial $x^n - x^{n-1} + c$ by substitution of p, N, d with $1 - x$, $1/c$, n, respectively. It is not possible, however, to factor the polynomial with $c = 1/N$ to find its roots. Also, there are no known formulae for the roots of polynomials with degree $n \geq 5$ [22]. Therefore, we derive approximate solutions for the minimax optimization problem in addition to the qualitative results derived in the previous section.

Without loss of generality, we divide $Np(1-p)^{d-1} = 1$ by N, and represent p as $1-x$ ($0 \leq x \leq 1$). Thus, the equation becomes

$$(1-x)x^{d-1} = \frac{1}{N}.$$

Assuming $N \gg 1$ which is justified by N denoting the attack volume of DoS, the right-hand-side becomes $\frac{1}{N} \approx 0$. Thus, the solution is close to 0 or 1. First, consider the case where a root is close to 1. The exponential term will be close to 1, yielding the approximate solution $x = 1 - 1/N$. For this value of x, the exponential term on the left-hand-side becomes $(1 - 1/N)^{d-1}$. This term approaches 1 as $N \to \infty$. For example, its value is 0.99976 when $N = 10^5$ and $d = 25$, which is small compared to unity. Thus, we arrive at an approximation to the root.

Next, consider the case when the root is close to 0. The term $(1 - x)$ will be close to 1, and may be neglected. The equation $x^{d-1} = 1/N$ gives an approximate solution $x = \frac{1}{N}^{\frac{1}{d-1}}$. The value of $x = \frac{1}{N}^{\frac{1}{d-1}}$ is close to 0 for large N so that $(1 - x) \approx 1$.

Thus, since x is approximately $1 - (1/N)$ or $\frac{1}{N}^{\frac{1}{d-1}}$, the corresponding p is $1/N$ or $1 - \frac{1}{N}^{\frac{1}{d-1}}$. Therefore, p is approximately in the following region for satisfying the sampling constraint $Np(1 - p)^{d-1} \geq 1$:

$$\frac{1}{N} \leq p \leq 1 - \left(\frac{1}{N}\right)^{\frac{1}{d-1}}.$$

Hence, the maximum uncertainty value m of the min-max optimization problem is given by

$$m \approx \frac{N^{-\frac{1}{d-1}}}{1 - N^{-\frac{1}{d-1}}}.$$

When $N = 10^5$ and $d = 25$, the uncertainty factor m is approximated by 1.6247. When $N = 10^7$ and $d = 25$, m is further reduced to 1.0446. From the approximate analysis of the maximally attainable uncertainty factor, we conclude that choosing a maximum allowable p by a victim results in the limited ability of an attacker to hide his identity (e.g., $m = 1 \sim 2$ when $N = 10^5 \sim 10^7$).

4.3 Numerical Evaluation

In this section, we give numerical solutions to (3.7) that complement the bounding results and the approximate solutions given in the previous sections. We used *Maple V* [13] for computing the numerical solutions. We first evaluate the impact of marking probability (p), followed by attack distance (d) and attack volume (N).

4.3.1 Marking Probability

Probabilistic marking with respect to its encoding using the IP header's fragmentation field can be efficiently implemented using code distribution over multiple packets. We refer the reader to [18] for a discussion.

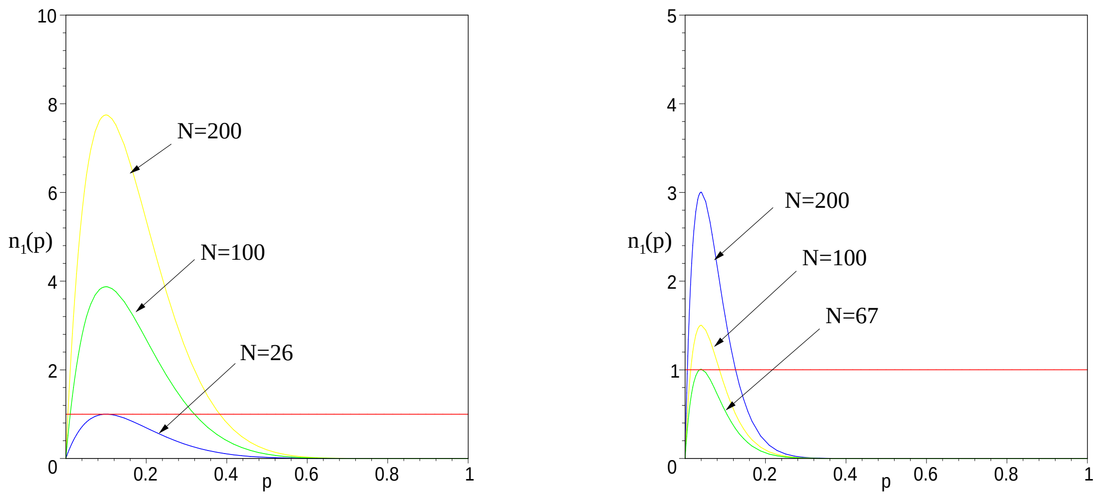

Figure 4.1: Left: $n_1(p)$ as a function of p for $N = 26, 100$, and 200 when $d = 10$. Right: Corresponding plot when $d = 25$.

First we measure the range of p which satisfies $n_1(p) \geq 1$ for different values of N and d. Figure 4.1 (left) shows $n_1(p) = p(1-p)^{d-1}N$ as a function of p for $N = 26, 100$, and 200 when $d = 10$. The allowable range of p (i.e., the set L_N) is the region where values of $n_1(p)$ become larger than 1. This can be discerned by the intersection of $n_1(p)$ with the constant line 1. For this graph, the upper bound of p is minimized at $1/d = 0.1$ with $N = 10^{10}/9^9 \approx 26$. As N decreases, the upper bound of p decreases until N reaches to $d^d/(d-1)^{d-1}$. Figure 4.1 (right) shows the corresponding graphs when $d = 25$.

Figure 4.2 (left) shows the feasible range of p as a function of d when $N = 10^3, 10^5, 10^7$, and their approximations. The plots show that our approximation is close to the solution. In particular, as N increases, the approximation becomes tighter, especially, for d large. The upper graphs represent the upper bounds of p which correspond to the minimax solution of (3.7), and the bottom graphs are of the feasible region L_N which are near zero. We observe that as d increases, the upper bound of p decreases. Since the Internet has a bounded diameter, the upper bound of p stays at "high" values yielding uncertainty factors $m = (1/p) - 1$ that are commensurately "low." Figure 4.2 (right) shows the minimax solution as a function of traffic volume N (i.e., its logarithm $\log N$) for N in the range $100 \sim 10^7$ when $d = 5,\ 10,\ 15,\ 20$, and 25. We observe that to reduce the minimax value of p and thus increase the uncertainty factor m, the attack volume needs to be decreased exponentially which is a high penalty to pay in a DoS attack.

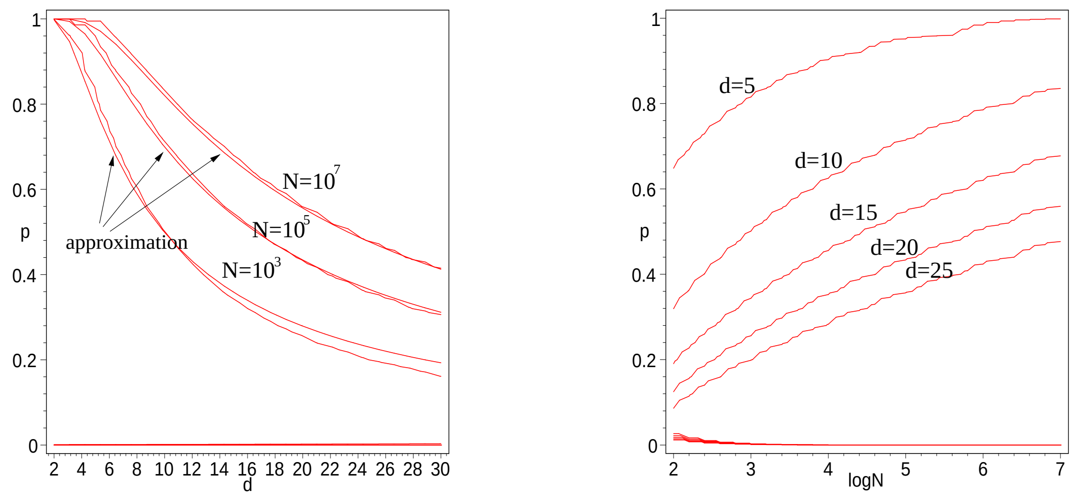

Figure 4.2: Left: Upper bound of marking probability p and its approximation as a function of d for $N = 10^3, 10^5, 10^7$. Right: Upper bound of p as a function of N (i.e., its logarithm) for $d = 5, 10, 15, 20$, and 25.

4.3.2 Attack Distance

Let us consider the range of forgeable paths when $d = 25$, since few paths on the Internet exceed that distance [28]. In the case of $N = 10^5$, the marking probability p must be in the range $0.1 \times 10^{-4} < p < 0.3536$ to satisfy the sampling constraint. For this range, the number of forgeable paths m is shown in Figure 4.3 (left). While the uncertainty factor m lies in the range $1 < m < 10^5$, a victim can reduce m to 2 by choosing the maximal feasible p, i.e., $p = 0.35$. When we increase N to 10^7, p is in the range of $0.1 \times 10^{-6} < p < 0.4729$, and its corresponding value of m is shown in Figure 4.3 (left). From the above instances, we observe that even though PPM cannot pinpoint the attack host's location, the number of possible candidates is a manageable constant which can help facilitate on-line traceback and increase the deterrent factor.

Let us consider the effect of the attacker's location to the traceback. As shown in Figure 4.2, as d increases, the upper bound of p decreases, which increases the uncertainty factor m. Given N, as distance d decreases, the expected number of spoofed packets, N_s, will increase for any given value of p. We note, however, that the ability of an attacker to hide the attack location is not in proportion to the number of spoofed packets received by the victim. Conversely, as d increases, while the number of spoofed packets received by a victim decreases, the uncertainty factor m increases. Therefore, when the source of an attack is far from the victim, the attacker becomes more potent at impeding traceback. Since the distance between an attacker and victim is bounded

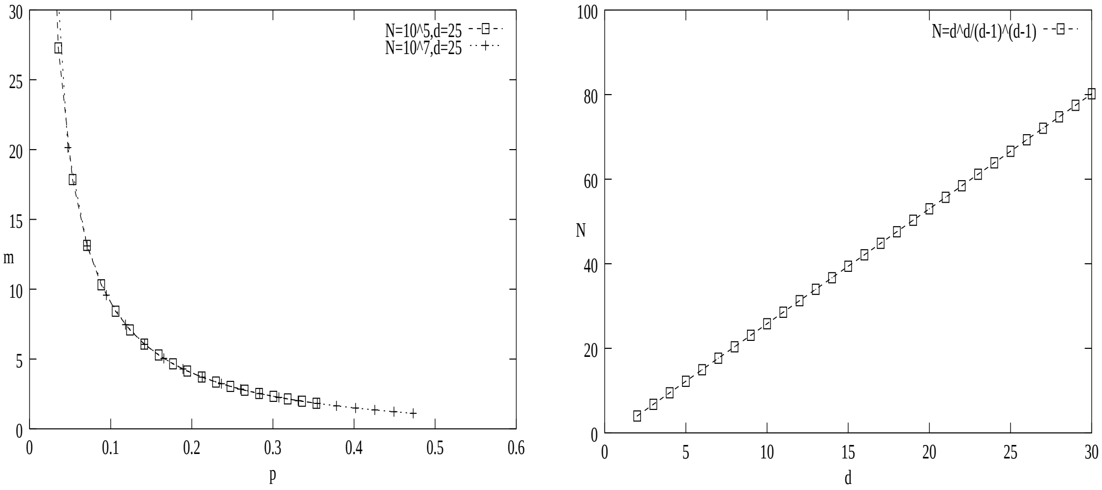

Figure 4.3: Left: The lower bound of forgeable paths m as a function of p for $d = 25$ and $N = 10^5, 10^7$. Right: The least attack volume N needed to satisfy the sampling constraint (3.6).

on the Internet, an attacker has limited ability to hide his location when subject to probabilistic packet marking.

4.3.3 Attack Volume

For the purpose of path reconstruction on the victim side, N needs to be at least $d^d/(d-1)^{d-1}$ to satisfy the sampling constraint. Figure 4.3 (right) shows the sampling lower bound on N when $d = 2 \sim 30$. As N increases, the victim can reduce the number of forgeable paths to less than $d-1$. Therefore, if an attacker transmits a small number of packets near the sampling lower bound, the victim will additionally suffer under a sampling problem. This points toward the fact that amplified confusion can be achieved by mounting distributed DoS attacks where each attack host contributes a small fraction of the total attack volume.

Figure 4.4 shows the minimax solution of uncertainty factor m as a joint function of d and $\log N$. The value of m is plotted for $10^1 \leq N \leq 10^7$ and $2 \leq d \leq 30$. As noted earlier, an increase in d leads to an increase in m. Whereas the impact of d is gradual—in fact, linear (i.e., upper bounded by $d-1$), the impact of N is more pronounced. With a small attack volume, e.g., $N = 10 \sim 100$, an attacker can keep the victim at an uncertainty level approaching 20. As N increases to $N = 10^3 \sim 10^5$, however, m can achieve values only in the range $1 \sim 4$ even at $d = 25$. This means that a victim can effectively localize the physical source of an attack to $2 \sim 5$ candidates. This makes it intrinsically difficult for a DoS attacker to wreck havoc using single-source attacks when PPM is employed by the network to facilitate traceback. Of course, it is unrealistic to assume

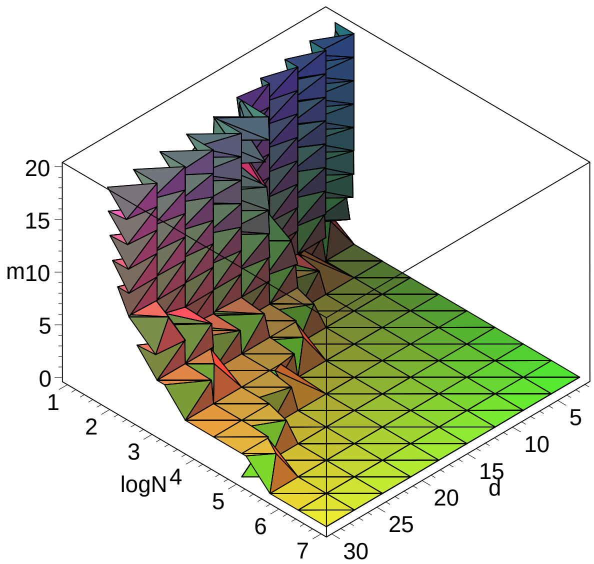

Figure 4.4: The minimax solution of forgeable paths m as a function of N (shown for $\log N$) and d.

that p can be programmed by different users to suit their individual needs. The small constant upper bound on m admits the policy of setting p—once and for all—for a sufficiently large distance d and conservative attack volume N which renders single-source traceback practically feasible.

5 Distributed DoS Attack

5.1 Key Issues

Given the theoretically and practically bounded impact of single-source DoS attack under probabilistic packet marking, distributed DoS attacks present a potentially important dimension to the source identification problem. In Section 4 we showed that the uncertainty factor in single-source attack can be amplified up to 20 if the path length is sufficiently large, however, this occurs at the cost of drastic—i.e., exponential—reduction in traffic volume (cf. Figure 4.4) which may render the attack ineffective with respect to achieving "denial of service." Attack volume may be recovered by mounting concurrent, small volume attacks from a number of sites, but its efficiency needs to be evaluated with respect to the cost of mounting distributed attacks which grows with the number of hosts engaged in the attack. In particular, following the uncertainty optimization framework—minimization for the victim and maximization for the attacker—of Section 4, given a desired attack volume N, an amplification factor of M can be trivially achieved by mounting

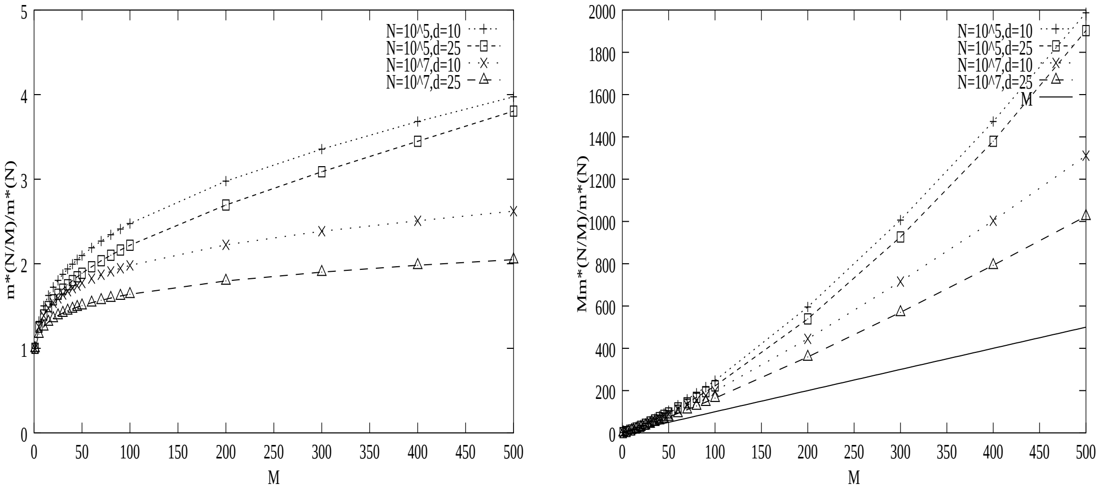

Figure 5.1: Left: The coefficient of expansion $\frac{m^*(N/M)}{m^*(N)}$ to the uncertainty amplification measured in the range $M = 1 \sim 500$. Right: The uncertainty factor $\frac{Mm^*(N/M)}{m^*(N)}$ measured in the range $M = 1 \sim 500$.

N/M-volume attacks from M separate attack sites. That is, even in the absence of forging or spoofing of the marking field, the victim will need to process M total attack paths when fielding a defense, on-line or off-line. With spoofing of the marking field, amplification may be greater. In fact, it is given by the expression $M(m^*(N/M)/m^*(N))$ where $m^*(\cdot)$ is a function depicting the optimum (i.e., minimax) uncertainty factor for the traffic volume given in the argument. Without spoofing, $m^*(N/M) = m^*(N)$, and we arrive at the amplification factor M. With spoofing, $m^*(N/M) \geq m^*(N)$.

5.2 Distributed DoS Attack Model

5.2.1 Classification

The traceback problem in distributed DoS attacks can be classified into two categories in accordance with the objectives underlying the attack and its susceptibility to traceback. In *any-source traceback*, the attacker is assumed to be vulnerable to further traceback once a compromised attack host is identified (e.g., due to state information left on the host). Thus the attacker seeks to fortify the weakest link—i.e., maximize the uncertainty factor of each individual attack host—whereas the victim tries to find a weak attack host. In *all-source traceback*, we assume the attacker is able to mount stateless intrusions when gathering attack hosts, and thus his objective is to maximize total uncertainty (vs. individual uncertainty in the any-source traceback case) since quick traceback of individual attack hosts does not present a danger with respect to revealing traceback information. The attacker's objective is to maximize the number of forged paths that the victim has to process, and the victim's goal is to isolate or shut down traffic flow emanating from comprised hosts.

5.2.2 Traceback Analysis

An environment for distributed DoS attack is described as follows. Given M distinct sources, each source s_i sends N_i packets to victim t at d_i distance for $1 \leq i \leq M$. An attack path $\mathcal{A}_i$ is represented by $\mathcal{A}_i = (s_i,\ v_{i,1},\ v_{i,2}, .., \ v_{i,d_i-1},\ t)$. Without loss of generality, assume $d_i \leq d_j$ for $i < j$. The expected number of spoofed packets received by the victim from attack host s_i is

$$N_{s_i} = \alpha_{i,0}(p)N_i = (1-p)^{d_i}N_i$$

for $1 \leq i \leq M$. The expected number of packets marked by $v_{i,1}$ is

$$n_{i,1}(p) = \alpha_{i,1}(p)N_i = p(1-p)^{d_i-1}N_i.$$

An attack host may use N_{s_i} to increase its uncertainty factor m^i, or it may use its forged packets to help amplify the uncertainty factor m^j of some other attack host $j \neq i$. That is, the attack hosts, in a distributed DoS attack, may engage in cooperative actions to achieve a common objective.

In the case of any-source traceback, the objective of the attacker is to maximize

$$\min_{1 \leq i \leq M} m^i \tag{5.1}$$

which is tantamount to fortifying the weakest attack host with respect to its uncertainty factor. Thus, (5.1) yields

$$\min_{1 \leq i \leq M} \left\{ \frac{\alpha_{i,0}(p)}{\alpha_{i,1}(p)} \right\} = \min_{1 \leq i \leq M} \left\{ \frac{(1-p)^{d_i}}{p(1-p)^{d_i-1}} \right\} = \frac{1}{p} - 1.$$

Thus, the any-source traceback case reduces to the single-source traceback problem as affected by the definition.

In the case of all-source traceback, the objective of the attacker lies in maximizing

$$m = \sum_{i=1}^{M} m^i.$$

To affect an increase in m, an attack host may send spoofed packets whose aim is to amplify another attack host's uncertainty factor rather than its own. The objective function can be further simplified

$$\sum_{i=1}^{M} m^i = \sum_{i=1}^{M} \frac{\alpha_{i,0}(p)}{\alpha_{i,1}(p)} = \sum_{i=1}^{M} \frac{(1-p)^{d_i}}{p(1-p)^{d_i-1}} = M\left(\frac{1}{p} - 1\right)$$

due to its lack of dependence on N_i and d_i. Thus the derivation shows that from the attacker's point-of-view, one way of maximizing $m = \sum_{i=1}^{M} m^i$ is to perform M separate maximizations on each attack host. As with the any-source traceback case, all-source traceback reduces to the single-source traceback problem and does not necessitate cooperation among the attack hosts to achieve maximum uncertainty amplification. When performing the constrained minimax optimization (3.7) on each attack host as given by the single-source formulation in Section 3, d_i and N_i only enter in the M constraints corresponding to (3.6).

5.3 Numerical Evaluation of Traceback

To measure the (in)effectiveness of traceback in a distributed DoS attack setting, we perform comparative evaluation with single-source attack where the total traffic volume is held constant. As discussed in Section 5.1, our aim lies in evaluating the degree to which distributed DoS attack under probabilistic packet marking can achieve uncertainty amplification above and beyond the distribution factor M achievable in the trivial case.

Let N be the total attack traffic volume—the same for single-source attack as well as distributed DoS attack—and let $N_i = N/M$, $d_i = d$, $1 \leq i \leq M$, which facilitates comparability. Let $m^*(N_i)$ be the uncertainty factor achievable by N_i. Then the ratio $m^*(N/M)/m^*(N)$ represents the expansion rate to uncertainty factor with respect to the distribution factor M. Figure 5.1 (left) shows the coefficient of expansion to uncertainty amplification in an M-distributed attack where $M = 1 \sim 500$. As M increases, the coefficient of expansion increases, and achieves higher gains with small N and small d. This implies that traceback to a single source becomes more difficult as the attack volume is scattered into smaller units.

Figure 5.1 (right) shows the amplification factor $\frac{Mm^*(N/M)}{m^*(N)}$ as a function of M. The larger M, the higher the amplification. Thus, given an attack volume, as attack sources are distributed, the uncertainty injected into traceback can be amplified beyond the effect afforded by distribution.

6 Conclusion

Recently probabilistic packet marking has been proposed for tracing the source—i.e., origin—of an DoS attack. While PPM has the advantages of efficiency and implementability over deterministic packet marking and router based logging/messaging, it has the potential drawback that an attacker may impede traceback by sending packets with spoofed marking field values as well as spoofed source IP addresses. This paper analyzed the effectiveness of PPM in a minimax adversarial context where the attacker is allowed to spoof the marking field to achieve maximum confusion at the victim. Our analysis shows that, while it is always possible for an attacker to impede exact traceback by the victim, the attacker's ability to affect uncertainty is limited in internetworks with bounded diameters similar to the Internet, when a suitable marking probability is chosen. Thus, for single-source attacks PPM is effective at localizing the attack origin. In a distributed DoS attack, however, as the number of attack sources mounted increases, traceback is rendered more difficult due to an uncertainty amplification effect above and beyond the distribution factor M.

References

[1] Gaurav Banga, Peter Druschel, and Jeffrey C. 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] Steven M. 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] 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.

[5] David Clark. The design philosophy of the DARPA internet protocols. In *Proc. SIGCOMM '88*, 1988.

[6] John Elliott. Distributed denial of service attack and the zombie ant effect. *IT Professional*, pages 55–57, March|April 2000.

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

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

[9] Jari Hautio and Tom Weckstrom. Denial of service attacks, March 1999. http://www.hut.fi/u/tweckstr/hakkeri/DoS_paper.html.

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

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

[12] Mixter. Protecting against the unknown – a guide to improving network security to protect the Internet against future forms of security hazards, January 2000. http://members.tripod.com/mixtersecurity/protecting.html.

[13] M. Monagan, K. Geddes, K. Heal, G. Labahn, and S. Vorkoetter. *Maple V Programming Guide for Release 5.* Springer-Verlag, 1997.

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

[15] 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.

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

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

[18] Stefan Savage, David Wetherall, Anna Karlin, and Tom Anderson. Practical network support for IP traceback. Technical Report UW-CSE-00-02-01, University of Washington, Seattle, Washington, 2000. http://www.cs.washington.edu/homes/savage/traceback.html.

[19] Christoph L. Schuba, Ivan V. Krsul, Markus G. Kuhn, Eugene H. Spafford, Aurobindo Sundaram, and Diego 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.

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

[21] Oliver Spatscheck and Larry 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.

[22] Josef Stoer and R. Bulirsch. *Introduction to numerical anlaysis.* New York : Springer-Verlag, 1993.

[23] Robert Stone. Centertrack: An IP overlay network for tracking DoS floods, October 1999. Presentation at the NANOG17 Meeting, http://www.nanog.org/mtg-9910/robert.html.

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

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

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

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

[28] Wolfgang Theilmann and Kurt Rothermel. Dynamic distance maps of the Internet. In *Proc. of the 2000 IEEE INFOCOM*, March 2000.

A Traceback Problem in General Topology

We show that the analysis carried out in the caterpillar graph can be extended to general graphs G. Assume that any path in G can be the target of marking field spoofing, i.e., forgeable. Given the set of attack packets received by a victim and an arbitrary set of disjoint paths with the victim as the common end point, the goal of marking field spoofing is to render all paths equally probable as potential originating sources of the attack. We call this the *symmetry property* of forgeable paths. Consider two paths $\mathcal{A}_1 = (s_1, v_{1,1}, .., v_{1,d_1-1}, t)$ and $\mathcal{A}_2 = (s_2, v_{2,1}, .., v_{2,d_2-1}, t)$, as shown in Figure A.1. Let one path be the attack path, and the other the forged path.

When N packets arrive at the victim t, the expected number of packets marked by routers on the paths $\mathcal{A}_1$ and $\mathcal{A}_2$ are $N_m^1 = (1-(1-p)^{d_1})N$ and $N_m^2 = (1-(1-p)^{d_2})N$, respectively. If an attacker at s_1 wants to forge $\mathcal{A}_2$, the number of spoofed packets $N - N_m^1$ must be larger than the marked packets (imaginary) passing through $\mathcal{A}_2$. Thus,

$$N - N_m^1 \geq N_m^2.$$

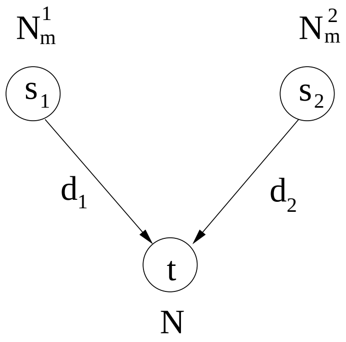

Figure A.1: Symmetry of an attack path and a forgeable path.

Conversely, if an attacker is located at s_2 and wants to forge $\mathcal{A}_1$, the attacker must be able to forge N_m^1 packets with $N - N_m^2$ unmarked spoofing packets. Thus,

$$N - N_m^2 \geq N_m^1.$$

Therefore, the symmetry property is satisfied when

$$N_m^1 + N_m^2 \leq N.$$

Equivalently, perfect symmetry is achieved if, and only if,

$$(1-p)^{d_1} + (1-p)^{d_2} \geq 1.$$

The previous development can be generalized to m paths for defining the symmetry property of forgeable paths. Given m paths $\mathcal{A}_i = (s_i, v_{i,1}, .., v_{i,d_{i-1}}, t)$, $1 \leq i \leq m$, they are *forgeable* satisfying the symmetry property if

$$\sum_{i=1}^{m} \left(1 - (1-p)^{d_i}\right) \leq 1.$$

The objective of an attacker can be formulated as follows. Given $G = (V, E)$, find $S \in V$ which maximizes $|S|$, subject to

$$\sum_{i=1}^{|S|} \left(1 - (1-p)^{d_i}\right) = 1.$$

Of course, the attacker must lie in S. In case $d_i = 1$ for all i, $\max\{|S|\} = \frac{1}{p}$. Thus, the maximum uncertainty factor becomes $m = \frac{1}{p} - 1$. It implies that an attacker can hide his identity with $\frac{1}{p} - 1$ nodes near t, at maximum, and this is the same result as with the caterpillar graph. Although the traceback problem in the caterpillar graph is a special case of the general topology, its conclusions reflect that of the general case in the *best* case for the attacker, which is required for minimax analysis.