Improved Probabilistic Packet Marking for Multiple-source Attacks.

Micah Adler*
Department of Computer Science
University of Massachusetts
Amherst, MA 01003-4610.
Email: micah@cs.umass.edu

March 18, 2003

Abstract

We give a probabilistic packet marking protocol for the multiple source version of the IP Traceback problem. Our protocol is successful with high probability, regardless of the initial distribution over packets used by the attacker, provided that the k nodes of the attacker are chosen uniformly at random from the set of all possible nodes, and that the intermediate nodes have a minimal amount of information concerning their location along the path of attack.

1 Introduction

In this paper, we provide further results on probabilistic packet marking for the IP Traceback problem. In particular, we provide a significant improvement for the case of distributed denial of service attacks, where the packets are being sent to the victim from multiple locations simultaneously. This is a very important consideration, since this is a common, and also quite destructive, form for these attacks.

In [1], it was shown that for the case of k paths of attack, $\log(2k-1)$ header bits were required, and if the attacker is limited in an appropriate manner, then $\log(2k+1)$ header bits are sufficient. In this paper, we address the question of how the attacker should be limited. In [1], it is assumed that the attacker sets the marking bits to 0 for every packet. This is quite restrictive, and leads to a protocol with no guarantees as to the power of the victim to determine where the attack comes from if the attacker simply sets some of the initial header bits to 1. In this paper, we demonstrate a protocol that provides the victim with the source of the attack even in the case where the attacker sets the initial bits arbitrarily.

Instead, we make two alternative assumptions that are much more realistic in terms of the practical applications of probabilistic packet marking. First, we assume that the k paths used by the attacker are chosen randomly (instead of allowing the attacker to use an arbitrary set of paths). Note that this is a reasonable assumption in the Internet, since an attacker cannot chose arbitrary nodes

*This work supported in part by the National Science Foundation under NSF Faculty Early Career Development Award CCR-0133664 and NSF Research Infrastructure Award EIA-0080119.

to corrupt; rather, it is only able to target nodes that are compromised. The lower bound of $\log(2k-1)$ header bits from [1] still applies to this case. Second, we assume that the intermediate nodes have a small amount of information concerning their location along the path of attack (the exact assumption is described below). This is also a reasonable assumption in the Internet, and the lower bound still applies to this case as well.

The new protocol we introduce is a modified form of the protocol for the case of multiple paths of attack from [1]. The main new contribution of this work is substantially improved analysis over that provided in [1].

2 Model

We assume the following: there is a set of k parallel linear arrays (henceforth referred to as *paths*) consisting of n nodes between the attacker and the victim. Each node of a path holds a single bit. The victim must determine k strings, each consisting of n bits, corresponding to the bits held by each path. The victim does not need to determine which string corresponds to which path. We assume that each of the bits is chosen by an independent and fair coin toss.

For each packet, the attacker chooses which of the k paths that packet uses to travel to the victim. When the packet passes the ith node of that path, that node has access to the incoming bits of the packet, its single bit, as well as an unlimited supply of randomness (but the random bits cannot be remembered). It also has access to a small amount of location information described below, but no other information. In particular, the intermediate nodes have no state information. Based on the information a node has, it chooses the bits for the packet it sends to its successor on the path. The victim sees the bits sent by the last node along the path, but does not receive the information of which path the bits travelled along. After receiving sufficiently many packets, the victim must (w.h.p.) determine the strings that were on paths used for a fraction of at least $\frac{\alpha}{k}$ of the packets, for a parameter $\alpha \leq 1$.

We also assume that the nodes along each path have a small amount of information as to their location along that path. Note that this is a reasonable assumption in the Internet, since a router has access to the destination of a given packet, and nodes are likely to have some knowledge of whether that destination is close by or not. In particular, we assume that each node i has a predicate C such that if i has distance of at most $\log^2 k$ hops from the victim of the attack, then $C(i) = TRUE$, and if i is the node adjacent to the attacker, then $C(i) = FALSE$. For the remainder of the nodes along the path, the value of $C(i)$ can be either $TRUE$ or $FALSE$. For example, if $\log^2 k \leq n/2$, it is sufficient for a node to know if it is in the first or second half of the routing path. For ease of presentation, we make two assumptions that are not difficult to remove: (1) we assume here that $C(i)$ is the same for all paths, and we assume that all i for which $C(i) = TRUE$ are closer to the victim than any i for which $C(i) = FALSE$. We denote by $C_{\max}$ the number of i for which $C(i) = TRUE$.

3 The protocol

Let $d = 2^b - 1$. We define two different processes for mapping a probability distribution over packets to another probability distribution over packets. For each of these, let $p_{i,j}$ be the probability that

the packet i gets mapped to packet j. Consider first the mapping **zero**:

- For $0 < i \leq d$, $p_{i,i} = 2^{-i}$, and $p_{i,0} = 1 - 2^{-i}$.
- For $i \neq j$, and $j \neq 0$, $p_{i,j} = 0$.
- $p_{0,0} = 1$.

The second mapping is called **one**:

- For $1 \leq i \leq j \leq d$, $p_{i,j} = 2^{2i-3j}\binom{j}{i} + 2^{-3j}$.
- For $1 \leq j < i \leq d$, or $i = 0 < j \leq d$, $p_{i,j} = 2^{-3j}$.
- For $j = 0 \leq i \leq d$, $p_{i,j} = 1 - \sum_{j=1}^{d} p_{i,j}$.

The protocol from [1] consisted of a node with the bit 0 simply applying mapping **zero**. A node with the bit 1 applies mapping **one**. In the new protocol, a node i with the bit 0 and $C(i) = TRUE$ applies the mapping **zero** twice, followed by the mapping **one** once, followed by three more applications of the mapping **zero**. A node i with the bit 1 and $C(i) = TRUE$ applies the same process, except that the last mapping **zero** is replaced with a **one**. A node i with the bit 0 and $C(i) = FALSE$ applies the mapping **zero** $ck + 1$ times, for a suitable constant c to be described below. A node i with the bit 1 and $C(i) = FALSE$ applies the mapping **zero** ck times, followed by the mapping **one**.

It will also be convenient for us to think of the victim applying the transformation **one** on each packet it receives.

Theorem 1 *After the victim has received sufficiently many packets, with probability at least* $1 - \Delta$*, the victim has enough information to determine every string that is on a path used for at least a fraction of* $\frac{\alpha}{k}$ *of the packets the attacker sends.*

Proof: Denote the input of k n-bit strings available to the attacker as $P_1 \ldots P_k$. Assume first that the attacker sets the initial bits of every packet to 0 (as was assumed throughout in the protocol of [1]). Later, we shall see how to relax this assumption. Let $p_i(P_j)$ be the probability that a packet sent on a path with string P_j arrives at the victim with its bits set to the value i.

Let B_j^r be the rth bit (starting from the attacker) of the string P_j. Let

$$X_{P_j} = \frac{1}{8} + \sum_{r=1}^{C_{\max}} (\frac{1}{2})^{6(r-1)+4}(B_j^r + \frac{1}{8}) + \sum_{r=C_{\max}+1}^{n} (\frac{1}{2})^{(ck+1)(r-C_{\max}-1)+6C_{\max}+4} B_j^r.$$

We shall refer to X_{P_j} as the value of the string P_j. Note that if the victim is informed of the value of a string or even a sufficiently good estimate of this value, then this gives it sufficient information to determine all the bits of that string. With the assumption that the initial bits are set to 0, Claim 9 from [1] demonstrates that for $0 < i \leq d$:

$$p_i(P_j) = \left(X_{P_j}\right)^i, \tag{1}$$

Let λ_i be the fraction of the received packets that are sent by the attacker with string P_i. The probability that a randomly chosen packet from the set of received packets has its bits set to i is $q_i = \sum_{j=1}^{k} \lambda_j p_i(P_j)$. The set of received packets provides the victim with an estimate on the values of the q_i. Although the stochastic variance inherent to the communication process means that it is unlikely for the victim to know the q_is exactly, we first assume that the victim is given the exact values of the q_is, and demonstrate that this uniquely determines the entire set of strings used by the attacker. This allows us to build some intuition for why the victim is able to decode the set of strings. We shall then remove both this assumption, as well as the assumption that the attacker set the initial bits to 0.

We show that if we assume that the q_is do not determine the strings uniquely, this leads to a contradiction. Let $V(P_j)$ be the $2k$-dimensional vector where component i of $V(P_j)$ is $p_i(P_j)$. We shall refer to $V(P_j)$ as the *string vector* for P_j. Assume that there is some set of strings $P_{k+1} \ldots P_{2k}$ and probabilities $\lambda_{k+1} \ldots \lambda_{2k}$ such that $\sum_{j=1}^{k} \lambda_j V(P_j) = \sum_{j=k+1}^{2k} \lambda_j V(P_j)$. For the set of strings to not be uniquely determined, it must be the case that there is some string P_j with $\lambda_j > 0$ such that if $j \leq k$ then $P_j \notin \{P_{k+1}, \ldots, P_{2k}\}$, and if $j > k$ then $P_j \notin \{P_1, \ldots, P_k\}$. Assume here that such a string is P_{2k}; the case where $j \leq k$ is similar. In this case, we see that

$$\lambda_{2k} V(P_{2k}) = \sum_{j=1}^{k} \lambda_j V(P_j) - \sum_{j=k+1}^{2k-1} \lambda_j V(P_j). \tag{2}$$

There may be strings that appear in both $P_1, \ldots, P_k$ and $P_{k+1}, \ldots, P_{2k}$. However, by replacing any such string with another unused string, we see that (2) implies that there is some set of $2k$ distinct strings $P'_1 \ldots P'_{2k}$ and real numbers $\lambda'_1 \ldots \lambda'_{2k}$, with $\lambda'_{2k} > 0$, such that

$$\lambda'_{2k} V(P'_{2k}) = \sum_{j=1}^{2k-1} \lambda'_j V(P'_j). \tag{3}$$

Now, consider the $2k \times 2k$ matrix M where entry $M_{i,j} = p_i(P'_j)$. From (3), we see that M does not have full rank. However, from (1), we see that $M_{i,j} = \left(\frac{X_{P'_j}}{4}\right)^i$. The matrix M', where entry $M'_{i,j} = \left(\frac{X_{P'_j}}{4}\right)^{i-1}$, is a Vandermonde matrix. Since the strings $P'_1 \ldots P'_{2k}$ are distinct, if $i \neq j$ then $X_{P'_i} \neq X_{P'_j}$, and thus M' has full rank. Since the victim applies the mapping **one** on each received packet, we see that for all strings P, $X_P \neq 0$. This implies that the matrix M must have full rank as well, which is a contradiction. Therefore, the exact values of the q_i exactly determines all strings P_j, $1 \leq j \leq k$, such that $\lambda_k > 0$.

We next examine the effect of removing our two assumptions. In particular, 1) instead of the victim knowing the values of the q_i exactly, it only has the information provided it by the packets it has received: a series of samples from the probability distribution. Also, 2) the attacker, instead of being restricted to setting the initial bits to 0 on each packet, is allowed to employ any strategy it wants for the initial bits.

We can think of the values q_i as a point in $2k$-dimensional space, where the coordinate for dimension i is q_i. The effect of removing both of the two assumptions above is that instead of knowing the

exact point defined by the q_is, we instead know a point that we shall show is (whp) sufficiently close to determine any string that is used to send a large enough fraction of the packets. Let Q be the point defined by the q_is. Let $D_0 = \frac{6}{2^{6C_{\max}+(ck+1)(n-C_{\max})}}$. The estimate of the point Q that is used is as follows: the victim collects $N = \frac{6k}{D_0^2} \ln \frac{2k}{\Delta}$ packets. For $1 \leq i \leq 2k$, let Y_i be the number of times that packet i is seen in the N packets. We set $\bar{q}_i = Y_i/N$.

The victim only returns sets of strings that are likely to lead to seeing the $\bar{q}_i$s that it computes. Furthermore, it restricts its attention to those sets of strings that are not too close together, since it is unlikely that randomly chosen strings will be too close together. In particular, consider the following definition:

Definition 1 *We say that a set of k strings $P_1, \ldots, P_k$ is* well dispersed *if $\forall j, 1 \leq j \leq k, \Pi_{i \neq j} |X_{P_j} - X_{P_i}| \geq 2^{-14k}$.*

The victim returns any string P_j such that P_j is contained in a convex combination of at most k string vectors, with the coefficient associated with P_j being at least $\frac{\alpha}{k}$, such that (a) the Euclidean distance of the resulting convex combination from the corresponding point defined by the $\bar{q}_i$s is at most D_0, and (b) the set of k strings is well dispersed.

We first point out that it is likely that the attacker has a set of strings that is well dispersed:

Claim 1 *Say we choose a set R of k strings independently and uniformly at random. The probability that R is not well dispersed is at most $e^{-\epsilon(\min(k, C_{\max} - 2\log k)}$, for some constant ϵ.*

Proof: Note that the value for a randomly chosen string consists of the second bit being chosen randomly, the subsequent 5 bits being fixed, and then every 6th bit being chosen randomly and the other 5 bits being fixed, until $C_{\max}$ bits have been chosen randomly, and then one in every $ck+1$ bits being chosen randomly.

The probability that any fixed pair of strings i and j have a string value that agrees on the first $6C_{\max}+1$ bits is at most $2^{-C_{\max}}$. Thus, the probability that any pair of strings agrees on the first $6C_{\max}+1$ bits is at most $2^{-C_{\max}+2\log k}$. Thus, we henceforth assume that any pair of string values disagrees somewhere on the first $6C_{\max}+1$ bits.

We next examine a single string P_j, and bound the probability that the pairwise products with respect to this string are too small. We see that the distribution on $|X_{P_j} - X_{P_i}|$ stochastically dominates the distribution on $(\frac{1}{2} - \frac{1}{64})^{1+6h}$, where h is the number of heads seen before the first tail in a sequence of flips of a fair coin. Thus, $\Pi_{i \neq j} |X_{P_j} - X_{P_i}|$ stochastically dominates $(\frac{31}{64})^{k+6\hat{h}_k}$, where $\hat{h}_k$ is the number of heads seen before a total of k tails have been seen in a sequence of flips of a fair coin. Standard Chernoff bound techniques suffice to show that $\Pr[\hat{h}_k \geq 2k] \leq e^{\epsilon' k}$, for some positive constant ϵ'.

Thus, by taking a union bound over all possible strings j, $\Pr[\exists j \text{ s.t. } \Pi_{i \neq j} |X_{P_j} - X_{P_i}| \geq (\frac{31}{64})^{13k}] \leq e^{\epsilon' k - \log k}$. The claim now follows from the fact that $(\frac{31}{64})^{13k} \geq (\frac{1}{2})^{14k}$ ■

We demonstrate that with probability at least $1 - \Delta$, the victim returns every string P such that a fraction of at least $\frac{\alpha}{k}$ of the packets travel on P, and no strings that are not used by the attacker at all. To do so, we prove two lemmas: We first demonstrate that (whp) the point determined by the

victim is not more than D_0 distance from Q. We then demonstrate that every convex combination of string vectors that has a coefficient associated with string P_j of at least $\frac{\alpha}{k}$, where P_j is not used by the attacker, has a Euclidean distance from Q of more than $2D_0$. Let $D_q = \sqrt{\sum_{i=1}^{2k}(q_i - \bar{q}_i)^2}$.

Lemma 1 $\Pr[D_q > D_0] \leq \Delta$.

Proof: Note that for each i, $|\bar{q}_i - \mathrm{E}[\bar{q}_i]|$ is the distance caused by stochastic variation, and $|q_i - \mathrm{E}[\bar{q}_i]|$ is the distance caused by the attacker not setting the initial bits to 0. Standard Chernoff bound techniques demonstrate that with N packets, $\Pr[\sqrt{\sum_{i=1}^{2k}(\mathrm{E}[\bar{q}_i] - \bar{q}_i)^2} \geq D_0/2] \leq \Delta$. Thus, we only need to demonstrate that the effect of the attacker setting the initial bits arbitrarily cannot cause the distance from the point Q to be more than $D_0/2$.

To examine the effect of arbitrary settings of the initial bits, note that since the mappings performed by the routers are linear, it is sufficient for us to consider each of the cases where the attacker always sets the initial bits to the same value, for all possible values, and to show that for each of these individually, the distance from Q is at most $D_0/2$. This is sufficient, since the strategy used by the attacker must be some convex combination of these strategies.

The lemma follows from the following claim, which demonstrates that the distance from Q is at most $\frac{3}{2^{6C_{\max}+(ck+1)(n-C_{\max})}}$.

Claim 2 *For i a positive integer, let $\mu(i) = \max(0, i-2)$. After a packet has had $\ell \geq 1$ sets of three mappings applied to it, where the first two mappings in each set are the mapping* **zero**, $|q_i - E[\bar{q}_i]| \leq \frac{1}{2^{3\ell+\mu(i)}}$.

Proof: We prove this by induction on ℓ. For the base case, consider $\ell = 1$. When the last mapping in the set of three is **zero**, the claim follows simply from the definition of the mapping **zero**. When the last mapping is **one**, the portion of the mapping from i to j (which is only relevant when $j \geq i$) is $\binom{j}{i}\frac{2^i}{4^j}$. With the combination of the 2 **zero** mappings that are applied before the **one**, we see that the amount of i that goes to j is $\binom{j}{i}\frac{2^i}{2^{4j}}$. For $j = 1$, only $i = 1$ is relevant, and thus we see that in the case that the incoming packet is a 1, after the first router has applied its mapping, $|q_1 - \mathrm{E}[q_1]| \leq \frac{1}{8}$, as desired. For $j > 1$, we see that the amount of i that goes to j is at most $\frac{1}{2^{2j}}$. Summing over all relevant i, we get at most $\frac{j}{2^{2j}}$, which is at most $\frac{1}{2^{j+1}}$, as desired.

For the inductive step, if we assume that the inductive hypothesis holds, then the case where the last mapping is a **zero** is easy. For the case where the last mapping is a **one**, we saw for the base case that the total relevant probability of going from $i = 1$ to $j = 1$ is at most $\frac{1}{8}$, and so the inductive step works for $|q_1 - \mathrm{E}[q_1]|$. For the case of $j > 1$ we also saw in the base case that the total relevant probability of being j after this step is at most $\frac{1}{2^{j+1}}$. Even if this all comes from the largest possible value at the previous router (i.e., $i = 1$), this is still sufficient for the inductive step. ■ ■

Note that Lemma 1 implies that with high probability, the victim returns all strings that it is required to return. To show that with high probability the victim does not return any strings that it should not return, we show that $D_q \leq D_0$ also implies that there can be no string P not used by the attacker such that P is returned by the victim.

Lemma 2 *If the set of strings used by the attacker is well dispersed, then every convex combination of k well dispersed string vectors that contains a string P_j, not used by the attacker, with a coefficient of at least $\frac{\alpha}{k}$, has a Euclidean distance from Q of at least $2D_0$.*

Proof: If such a string exists, then there must be some set of strings $P_1 \ldots P_{2k}$, where $P_1 \ldots P_k$ are the well dispersed strings used by the attacker, $P_{k+1} \ldots P_{2k}$ are the well dispersed strings contained in the incorrect convex combination, and P_{2k} is the string returned incorrectly. Thus, $P_{2k} \notin \{P_1, \ldots, P_k\}$, and there exist probabilities $\lambda_1 \ldots \lambda_{2k}$, with $\lambda_{2k} \geq \frac{\alpha}{k}$, such that

$$\sqrt{\sum_{i=1}^{2k} \left(\sum_{j=k+1}^{2k} \lambda_j p_i(P_j) - \sum_{j=1}^{k} \lambda_j p_i(P_j) \right)^2} \leq 2D_0$$

This in turn implies that there are $2k$ distinct strings $P'_1, \ldots, P'_{2k}$ and real numbers $\lambda'_1 \ldots \lambda'_{2k}$, with $\lambda'_{2k} \geq \frac{\alpha}{k}$, such that

$$\sqrt{\sum_{i=1}^{2k} \left(\lambda'_{2k} p_i(P'_{2k}) - \sum_{j=1}^{2k-1} \lambda'_j p_i(P'_j) \right)^2} \leq 2D_0 \tag{4}$$

Let D_1 be the Euclidean distance in $\Re^{2k}$ from the point $\lambda'_{2k} V(P'_{2k})$ to the subspace spanned by $V(P'_1), \ldots V(P'_{2k-1})$. For (4) to be true, it must be the case that $D_1 \leq 2D_0$. Thus, to demonstrate that no such incorrectly returned string P_{2k} can exist, it is sufficient to show that $D_1 > 2D_0$. Let $\mathcal{V}_{2k}$ be the $2k$-dimensional volume of the parallelepiped defined by the vectors $V(P'_1), \ldots, V(P'_{2k-1}), \lambda_{2k} V(P'_{2k})$ in $\Re^{2k}$. Let $\mathcal{V}_{2k-1}$ be the $(2k-1)$-dimensional volume of the parallelepiped defined by the vectors $V(P'_1), \ldots, V(P'_{2k-1})$ in $\Re^{2k}$. We see that $D_1 = \frac{\mathcal{V}_{2k}}{\mathcal{V}_{2k-1}}$, and thus we consider each of $\mathcal{V}_{2k}$ and $\mathcal{V}_{2k-1}$ separately.

Lemma 3

$$\mathcal{V}_{2k} = \lambda_{2k} \prod_{1 \leq i < j \leq 2k} \left(X_{P'_i} - X_{P'_j} \right) \prod_{i=1}^{2k} X_{P'_i}$$

Proof: Due to the convenient form of the vectors $V(P'_1), \ldots, V(P'_{2k})$, we can easily determine $\mathcal{V}_{2k}$. In particular, a standard result from linear algebra is that $\mathcal{V}_{2k}$ is equal to the absolute value of the determinant of the matrix T, where column j of T, for $1 \leq j \leq 2k-1$, is $V(P'_j)$, and column $2k$ is the vector $\lambda_{2k} V(P'_{2k})$.

To compute $|det(T)|$, consider the matrix T', where column j of T', for $1 \leq j \leq 2k$, is $\frac{V_j}{X_{P'_j}}$. By (1), the matrix T' is Vandermonde, and thus

$$det(T') = \prod_{1 \leq i < j \leq 2k} \left(X_{P'_i} - X_{P'_j} \right).$$

The lemma then follows from the fact that to get T from T', we merely multiply each column i of T' by X_{P_i}, with the exception of column $2k$, which is multiplied by $\lambda_{2k} X_{P_i}$. ■

Lemma 4

$$\mathcal{V}_{2k-1} \leq \prod_{1 \leq i < j \leq 2k-1} \left(X_{P'_i} - X_{P'_j} \right) \prod_{i=1}^{2k-1} [X_{P'_i}(1 + X_{P'_i}^{2k-1})].$$

Proof: Let $V^2(P_j)$ be the vector consisting of the components $1, X_{P_j}, X_{P_j}^2, \ldots, X_{P_j}^{2k-1}$. Let $V^3(P_j)$ be the vector consisting of the components $0, X_{P_j}, X_{P_j}^2, \ldots, X_{P_j}^{2k-1}$. Let $V^4(P_j)$ be the vector consisting of the components $0, 1, X_{P_j}, X_{P_j}^2, \ldots, X_{P_j}^{2k-2}$.

For $e \in \{2, 3, 4\}$, let $\mathcal{V}_{2k-1}^e$ be the $(2k-1)$-dimensional volume of the parallelepiped defined by the vectors $V^e(P'_1), \ldots, V^e(P'_{2k-1})$ in $\Re^{2k}$.

Since $V(P_j)$ is simply $V_2(P_j)$ with every component multiplied by X_{P_j}, $\mathcal{V}_{2k-1} = \mathcal{V}_{2k-1}^2 \cdot \prod_{i=1}^{2k-1} X_{P'_i}$. Similarly, $\mathcal{V}_{2k-1}^3 = \mathcal{V}_{2k-1}^4 \cdot \prod_{i=1}^{2k-1} X_{P'_i}$. Since $\mathcal{V}_{2k-1}^4$ is the $2k-1$ dimensional volume of a set of $2k-1$ vectors in $2k-1$ dimensions, $\mathcal{V}_{2k-1}^4$ is the absolute value of the determinant of the matrix formed by the vectors $V^4(P'_1), \ldots, V^4(P'_{2k-1})$. Since this matrix is Vandermonde, its determinant is

$$\prod_{1 \leq i < j \leq 2k-1} \left(X_{P'_i} - X_{P'_j} \right).$$

Thus, the lemma follows from the following claim:

Claim 3 $\mathcal{V}_{2k-1}^2 \leq \mathcal{V}_{2k-1}^3 \prod_{i=1}^{2k-1} \frac{1 + X_{P'_i}^{2k-1}}{X_{P'_i}}$.

Proof: Consider the process of changing from the vectors $V^2(P'_1), \ldots, V^2(P'_{2k-1})$ to the vectors $V^3(P'_1), \ldots, V^3(P'_{2k-1})$, and consider the pairing of each vector of the type V^2 with the corresponding vector of the type V^3. This process has two effects on the parellelapipid defined by these vectors: it changes the length of the vectors, and it changes the angle between vectors. Note first that for any two pairs of corresponding vectors, the angle between those two vectors for V^3 is at least as large as the angle between those two vectors for V^2. Since all angles are between 0 and 90 degrees, the effect of the change in angles can only increase the volume of the parallelapipid. Thus, we only need to consider the change in length for each vector.

The length of $V^2(P_j)$ is $L_1 = \sqrt{1 + X_{P_j}^2 + X_{P_j}^4 + \ldots + X_{P_j}^{4k-2}}$. The length of $V^3(P_j)$ is $L_2 = \sqrt{X_{P_j}^2 + X_{P_j}^4 + \ldots + X_{P_j}^{4k-2}}$. It is easy to see from this that $\forall j, L_1 \leq \frac{1 + X_{P_j}^{2k-1}}{X_{P_j}} L_2$. ■ ■

Since $D_1 = \frac{\mathcal{V}_{2k}}{\mathcal{V}_{2k-1}}$, we see that

$$D_1 \geq \frac{\lambda_{2k} X_{P'_{2k}} \prod_{i=1}^{2k-1} \left(X_{P'_i} - X_{P'_{2k}} \right)}{\prod_{i=1}^{2k-1} (1 + X_{P'_i}^{2k-1})} \geq \frac{\lambda_{2k}}{16} \prod_{i=1}^{2k-1} \left(X_{P'_i} - X_{P'_{2k}} \right),$$

where the second inequality follows from the fact that for $1 \leq i \leq 2k$, $\frac{1}{8} \leq X_{P'_i} \leq \frac{1}{4}$. To complete the proof, we need to demonstrate that for any set of $2k$ string vectors formed from two well dispersed sets of k string vectors, this quantity will not be too large.

Claim 4 *Let* $S_1 = \{X_{P_1}, \ldots, X_{P_k}\}$ *and* $S_2 = \{X_{P_{k+1}}, \ldots, X_{P_{2k}}\}$ *be two sets of well dispersed string vectors such that* $X_{P_{2k}} \notin S_1$.

$$\prod_{X_{P_i} \in S_1 \cup S_2 - X_{P_{2k}}} |X_{P_i} - X_{P_{2k}}| \geq \frac{1}{2^{29k+6C_{\max}+(n-C_{\max}-1)(ck+1)+1}}.$$

Proof: Let X_{P_m} be the element of S_1 that minimizes $|X_{P_m} - X_{P_{2k}}|$. Note that since the last bit of the strings must be different, $|X_{P_m} - X_{P_{2k}}| \geq \frac{1}{2^{6C_{\max}+(n-C_{\max}-1)(ck+1)+1}}$. Furthermore, since X_{P_m} is closer to $X_{P_{2k}}$ than any other element in S_1, it must be the case that $\forall i, |X_{P_i} - X_{P_{2k}}| \geq |X_{P_i} - X_{P_m}|/2$. The claim then follows from the definition of a well dispersed set of string vectors. ■

The Lemma (and hence the Theorem) now follows by observing that if we set $c = 36 + \log \frac{1}{\alpha}$, then for $k \geq 2$ it must be the case that $D_1 \geq 2D_0$. ■

■

4 Conclusion

References

[1] Micah Adler, Tradeoffs in Probabilistic Packet Marking for IP Traceback, In *Proc. of ACM Symposium on Theory of Computing*, May 2002.

[2] Sigal Ar, Richard Lipton, Ronitt Rubinfeld, and Madhu Sudan, Reconstructing Algebraic Functions from Mixed Data. In *Proc. of 33rd Annual Symposium on Foundations of Computer Science*, pp. 503-512, October 1992.

[3] S. M. Bellovin, ICMP Traceback Messages. Internet Draft: draft-bellovin-itrace-00.txt, Mar. 2000.

[4] Hal Burch and Bill Cheswick, Tracing Anonymous Packets to Their Approximate Source. In *Proc. Usenix LISA '00*, 2000.

[5] Sven Dietrich, Neil Long, and David Dittrich, Analyzing Distributed Denial of Service Attack Tools: The Shaft Case. In *Proc. 14th Systems Administration Conference, LISA 2000.*

[6] Drew Dean, Matt Franklin, and Adam Stubblefield, An Algebraic Approach to IP Traceback. In *Proc. 2001 Network and Distributed System Security Symposium.*

[7] Thomas Doeppner, Philip Klein, and Andrew Koyfman. Using router stamping to identify the source of IP packets. In*Proceedings of the 7th ACM Conference on Computer and Communications Security*, pages 184–189, Athens, Greece, November 2000.

[8] P. Ferguson and D. Senie, RFC 2267: Network Ingress Filetring: Defeating Denial of Service Attacks which employ IP Source Address Spoofing. *The Internet Society*, 1998.

[9] S. Floyd and V. Jacobson, Random Early Detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking*, 1(4), August 1997.

[10] S. Lee and C. Shields, Tracing the Source of Network Attack: A Technical, Legal and Societal Problem. In *Proceedings of the 2001 IEEE Workshop on Information Assurance and Security*, June 2001.

[11] Rajeev Motwani and Prabhakar Raghavan, *Randomized Algorithms*. Cambridge University Press, New York, NY, 1995.

[12] K. Park and H. Lee. On the effectiveness of probabilistic packet marking for IP traceback under denial of service attack. In *Proc. IEEE INFOCOM '01*, pp. 338-347, 2001.

[13] K. Park and H. Lee. On the effectiveness of route-based packet filtering for distributed DoS attack prevention in power-law internets. To appear in *Proc. ACM SIGCOMM '01*, August 2001.

[14] C. Perkins, IP Mobility Support. RFC 2002, Oct. 1996.

[15] L. Rizzo. Effective Erasure Codes for Reliable Computer Communication Protocols. *ACM Computer Communication Review*, Vol. 27, n.2, pp. 24-36, April 1997.

[16] Stefan Savage, David Wetherall, Anna Karlin and Tom Anderson, Practical Network Support for IP Traceback. In *Proceedings of ACM SIGCOMM 2000* , pp. 295-306, August 2000.

[17] Alex C. Snoeren, Craig Partridge, Luis A. Sanchez, Christine E. Jones, Fabrice Tchakountio, Stephen T. Kent, and W. Timothy Strayer. Hash-Based IP Traceback. To appear in *Proc. ACM SIGCOMM 2001*, August 2001.

[18] Dawn X. Song and Adrian Perrig, Advanced and authenticated marking schemes for IP traceback, In *Proc. IEEE INFOCOM '01*, 2001.