Towards Asymptotic Optimality in Probabilistic Packet Marking*

Micah Adler†
Dept. of Computer Science
University of Massachusetts
Amherst, MA 01003-4610
micah@cs.umass.edu

Jeff Edmonds‡
Dept. of Computer Science
York University
Toronto, ONT, Can M3J 1P3
jeff@cs.yorku.ca

Jiří Matoušek
Dept Applied Mathematics and
Inst. of Theoretical Comp. Sci.
Charles University
Malostranské nám. 25
118 00 Praha 1, Czech Republic
matousek@kam.mff.cuni.cz

ABSTRACT

There has been considerable recent interest in probabilistic packet marking schemes for sending information from nodes (routers) along one or more paths traveled by a stream of packets to the end-host receiving that stream. A central consideration for such schemes is the tradeoff between the number B of possible states of the marking bits in a packet, the number of bits n of information being sent by the nodes, and the expected number of packets T required to reconstruct this information. For the case where the packets all travel along the same path, we prove a lower bound of $T \geq \Omega(B2^{2n/(B-1)})$, roughly the square of an earlier lower bound of Adler.

For an upper bound, we consider a model where each of m nodes along a single path must send one of s possible messages (thus $n = m \log_2 s$ total bits are sent). We prove that $T \leq O(m \cdot 2^{2m(\log_2 s)/(B-1)})$ suffices (the implicit constant depends on B and s); this almost matches the lower bound, and is roughly the square root of an earlier upper bound of Adler. The new bound holds for all B and s in two slightly relaxed models, while under the strictest requirements we prove it only for some special values of B and s. This is related to a challenging geometric problem: the existence of an s-reptile $(B-1)$-dimensional simplex, i.e. a simplex S that can be tiled by s congruent simplices similar to S.

We also consider the case where the packets travel along multiple paths to the same destination. In this case, we present a new protocol and analysis technique that together allow us to significantly generalize over previous work the scenarios where the protocol is effective.

*Part of this research was done at the workshop "Random GRAALS '02" in Bertinoro.

†This work supported in part by the National Science Foundation under NSF Faculty Early Career Development Award CCR-0133664, NSF Award ITR-0325726, and NSF Research Infrastructure Award EIA-0080119.

‡This work supported in part by NSERC grants.

STOC'05, May 22-24, 2005, Baltimore, Maryland, USA.

Categories and Subject Descriptors

C.2.2 [**Network Protocols**]: Computer-Communication Networks—*Computer Systems Organization*; E.4 [**Coding and Information Theory**]: Data; F.1.1 [**Models of Computation**]: Computation By Abstract Devices—*Theory of Computation*; F.2.3 [**Tradeoffs between Complexity Measures**]: Analysis Of Algorithms and Problem Complexity—*Theory of Computation*; G.3 [**Probability and Statistics**]: Mathematics of Computing

General Terms

Algorithms, Security, Theory

Keywords

Denial of Service

1. INTRODUCTION

Probabilistic packet marking (PPM) is a recently discovered, powerful technique for extracting information from nodes (typically routers) along a path travelled by a sequence of packets. In PPM, a small number of header bits in each packet is reserved for this transfer of information, and each node along the path can update these header bits. The goal is to inform the node of the network that receives the sequence of packets of some information stored in a distributed fashion across the nodes along the path. For example, a simple case is where each node along the path has a single bit of information to send to the receiving node. The challenge (and power) of this technique is that each of the intermediate nodes is required to perform this transmission in a memoryless fashion: it can only update the header bits to a value based on its own piece of information, the value of the header bits on the incoming packet, and some number of random bits.

The memoryless requirement is crucial in a network such as the Internet, where routers typically handle a large number of simultaneous flows of packets, and thus it is not conceivable to have per-flow state. This requirement also means that the nodes must mark the packets probabilistically. PPM was first suggested in [4], with the first extensively analyzed protocols being introduced in [17]. This early work used PPM to solve the IP Traceback problem: determine the source of a stream of packets that hides its origin by "spoofing" the source node field in the packet header.

Solutions to the IP Traceback problem are crucial to combating Denial-of-Service attacks.

Such a scenario leads to two additional difficulties for designing a PPM scheme. First, a malicious *adversary* sets the initial value of all header bits in the packet, including those allocated to PPM, and thus this adversary will attempt to use this ability to hide the information being transmitted. We henceforth refer to the node receiving the stream of packets, which is trying to determine this information, as the *victim*. Second, many Denial-of-Service attacks are performed in a distributed fashion, with multiple sources sending packets to the victim. In such a scenario, a PPM scheme must be able to handle packets arriving at the victim along multiple paths, where each path has different information to send to the victim, but the victim is not able to determine the path travelled by a given packet.

The application of PPM to the IP Traceback problem has generated considerable interest in this technique [6, 16, 19, 12, 8, 11, 9]. Furthermore, PPM has the potential to be useful in a number of other scenarios as well: congestion control [2], robust routing [6], dynamic network configuration [6], and identification of bottleneck routers [9].

In addition to the potential practical impact of PPM schemes, it turns out that designing optimal PPM techniques is an interesting theoretical problem. In particular, [1] demonstrates that there are inherent tradeoffs between the parameters s, B, m, k, and T, which are defined as follows. We assume that each node is given one element from a set of size $s \geq 2$, and it must inform the victim of which element it has. We also assume that the header bits can take on B different values. If all possible settings of b binary bits are available, then $B = 2^b$. In [20], an application of packet marking to congestion control is described where $B = 3$, since two bits are available, but one of the possible settings of these bits is reserved for another use. We assume that there are m nodes along a path. Finally, k represents the number of paths being used by the adversary, and T is the number of packets that the victim must receive to reliably reconstruct the messages from the nodes.

In [1], the following results quantifying the tradeoffs between these parameters are presented. For the case where $k = 1$ (i.e., all packets travel along the same path), $s = 2$, and $B = 2^b$ for an integer b, $T \leq 2^{(4+o(1))n/B}$ packets are sufficient with high probability (here $n = m$). An information-theoretic lower bound shows that $T \geq 2^{(1-o(1))n/B}$ is necessary, for any values of s and m such that $n = m \log s$. For the case of multiple paths, $B \geq 2k - 1$ must hold, regardless of how large T is. Furthermore, if the adversary sets the initial marking bits to 0 in every packet, then there is a protocol with $B \leq 2k + 1$.

In this paper, we provide several significant improvements to the results of [1]. For the case of a single path, we prove a new lower bound, demonstrating that $T \geq \Omega(B 2^{2n/(B-1)})$ packets are necessary, for any values of s and m such that $n = m \log s$. This value is roughly the square of the lower bound shown in [1], and, for the case $B = 2$, matches an upper bound provided in [1]. For brevity, we call protocols (upper bounds) *quasioptimal* if $T \leq 2^{(2+o(1))n/(B-1)}$ is sufficient with high probability. For quasioptimality, we consider k, s and B fixed, and thus asymptotic notation refers to $m \to \infty$. Note that quasioptimality refers to the asymptotics of the exponent and thus leaves plenty of room for improvement, especially for small values of m.

In our efforts to achieve quasioptimality, we provide a reduction from the PPM problem for a given s and B to the problem of finding a $(B-1)$-dimensional simplex that is an *s-reptile*; that is, it can be partitioned into s equal sized pieces all congruent and similar to the whole. This allows us to use known results on s-reptiles to provide quasioptimal protocols for various values of s and B. In particular, we show that if $B = 3$ and s is of the form 2, i^2, $3i^2$, or i^2+j^2 for integers i and j or if $B \geq 4$ is arbitrary and $s = i^{B-1}$, then there is a quasioptimal protocol. The upper bound provided by these protocols is roughly the square root of the bound shown in [1].

The reduction to finding s-reptiles also allows us to provide a new protocol for the scenario where the marking bits are restricted in their initial distribution. This would be the case in cooperative scenarios, such as applying PPM to congestion control, where the source of the packets sets the initial bits in a predictable fashion. We demonstrate that in this scenario, $T \leq O(s^2 B^2 2^{2m(\log_2 s)/(B-1)})$ is sufficient for any values of B and s. This differs from the new lower bound (which also applies to this cooperative scenario) by a factor of only $s^2 B$, and thus the dependence on m is asymptotically optimal for this scenario. A similar technique also achieves quasioptimality for any values of B and s if the adversary sets the value of the initial bits, but the victim is allowed to lose the information held by a few nodes farthest from it.

Finally, we turn to the case of multiple paths. We remove the assumption made in the protocols of [1] that the adversary sets the initial marking bits to 0 in every packet. While such restrictions on the initial distribution are natural for some scenarios of the single path case, the multiple path case is mostly motivated by Denial-of-Service attacks, and thus the most relevant case is where a malicious adversary is setting the initial value of the header bits. We here introduce a protocol where $B \leq 2k + 1$ is still sufficient, but this protocol makes two alternative assumptions that are much more realistic in terms of the application of PPM to IP Traceback. First, we assume that the element of the set of size s at each of the nodes are chosen randomly (instead of allowing the adversary to choose worst case elements). Note that this is a reasonable assumption for IP Traceback in the Internet, since an adversary cannot chose arbitrary nodes to corrupt; rather, it is only able to target nodes that are compromised. 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. The lower bound of $B \geq 2k - 1$ from [1] still applies to this case.

Due to space restrictions, most of our proofs are omitted, or are only sketched. The exception to this is the proof of the result on multiple paths. A significant portion of the contribution of that result is the proof technique used, and thus this proof is provided in the appendix.

2. MODELS FOR THE PPM PROBLEM

We first describe the model we use for the protocols. For the case of a single path of attack, we assume that packets are traveling across a sequence of intermediate nodes. We shall refer to the node on the path from the victim to the adversary at distance i from the victim as N_i (where the victim is N_0, and the adversary is N_{m+1}). Each node has one of exactly s values to send to the victim. We shall refer to this information as $W = w_1 w_2 \ldots w_m$, where $w_i \in [s]$ indicates the value known by N_i.

Information is sent to the victim via the header bits in the packets traveling from the adversary to the victim. We ignore the contents of each packet other than that allocated to PPM, and thus we simply assume that each packet can take on exactly one of B values. Each node can update the contents of each packet that it forwards towards the victim. However, this update can only be based on the contents of the incoming packet, the value the node holds locally, and probabilistic choices made by the node. The victim does have storage. Typically, we assume that the contents of the packet received by the node N_m is set by a malicious adversary N_{m+1}. However, as was described above, we sometimes assume that the initial distribution of packets is restricted.

For the case of multiple paths, we assume the same model, except that now there are up to k different paths, each of which contains a different set of m nodes. Thus, there are as many as km values that the victim must determine. Each packet travels along one of the k paths, and the contents of that packet can be updated by the nodes along that path. The adversary chooses which path each packet travels on; the victim only sees the contents of the final packet it receives - it does not know which path that packet traveled on. After receiving sufficiently many packets, the victim attempts to 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 provide more details on this model in Section 6.

For the lower bound, we assume a stronger model (i.e., a model where the problem is at least as easy to solve as in the model for the protocols). For the lower bound model, we assume a system consisting of only two parties, called the *Victim* and the *Network*, where we here capitalize Victim to distinguish it from the victim of the upper bound model. The Network has an n-bit string to send to the victim. No communication occurs from the victim to the Network. The Network is allowed to send B-valued packets to the victim, but it is stateless: for each packet it sends, it has no memory of the previous packets that it has sent. This lower bound model actually captures the difficulty of sending information from a memoryless node using packets consisting of a bounded number of bits.

Note that any protocol for the upper bound model implies a protocol for this model as well, and thus lower bounds for this model imply lower bounds for the protocol model. In fact, it might seem like this model is quite a bit more powerful: there is no adversary setting the initial bits of the packets and all of the information to be transmitted is stored at a single node. Despite these seeming advantages, we can prove lower bounds in this model that are close to matching the upper bounds shown in the model for the protocols.

3. PROTOCOLS FOR A SINGLE PATH OF ATTACK

THEOREM 1. *If (B, s) has one of the forms $(2, i)$, $(3, i^2)$, $(3, 3i^2)$, $(3, i^2 + j^2)$, (i, j^{i-1}) for positive integers i, j, then there is a protocol that recovers all values along the path with high probability as long as the victim receives at least $T \geq Cm \cdot 2^{2m(\log_2 s)/(B-1)}$ packets, with a suitable constant C depending on B and s. In particular, the protocol is quasioptimal.*

Encoding in the probability distribution of packets. The basic idea, introduced in [1], is to encode the given sequence $W \in [s]^n$ into the probability distribution of the packets received by the victim. This probability distribution can be specified by a vector $X = (x_1, x_2, \ldots, x_B)$, where x_u is the probability that a packet reaching the victim has value $u \in [B]$. Geometrically, such an X can be regarded as a point of the $(B-1)$-dimensional standard simplex $\Delta^{B-1} = \{(x_1, x_2, \ldots, x_B) \in \mathbf{R}^B : x_1, x_2, \ldots, x_B \geq 0, x_1 + x_2 + \cdots + x_B = 1\}$ in $\mathbf{R}^B$ (for example, for $B = 3$, Δ^2 is an equilateral triangle placed in $\mathbf{R}^3$).

If the victim receives sufficiently many packets, then he can "read off" the incoming distribution X with a prescribed precision. This allows him to distinguish between different sets of possible distributions, and is expressed quantitatively in the following lemma.

LEMMA 2. *Let $X \in \Delta^{B-1}$ be a distribution of the packets. Suppose that the victim receives T packets generated according to X, let Z_u be the number of received packets with value u, $u \in [B]$, and let $Y = (\frac{Z_1}{T}, \frac{Z_2}{T}, \ldots, \frac{Z_B}{T})$. Let $\beta > 0$ and $a > 0$ be real parameters (that can be chosen at will) and suppose that $T \geq a/\beta^2$. Then the probability that $\|X - Y\| \geq \beta$ is at most $1/a$.*

PROOF. We calculate that the expectation $\mathrm{E}\left[\|X - Y\|^2\right] = 1/T$; then the claim follows from Markov's inequality. By linearity of expectation we have $\mathrm{E}\left[\|X - Y\|^2\right] = \sum_{u=1}^{B} \mathrm{E}\left[(x_u - Z_u/T)^2\right] = T^{-2} \sum_{u=1}^{B} \mathrm{E}\left[(Tx_u - Z_u)^2\right]$. The term in the sum is the variance of Z_u. Now Z_u is the sum of T independent random variables, each attaining value 1 with probability x_u and value 0 with probability $1 - x_u$, and its variance is thus bounded above by Tx_u. Hence $\mathrm{E}\left[\|X - Y\|^2\right] \leq T^{-1} \sum_{u \in B} x_u = 1/T$. $\square$

Protocols as affine maps. Now the main question is, how can the nodes encode their messages into the probability distribution of packets received by the victim? Each node, being memoryless, has no way of "knowing" the distribution of the incoming packets. It has to process one packet at a time in a uniform way (but, of course, depending on the message $w \in [s]$ it wants to send). Having received a packet with value u, it must choose in a probabilistic way some value v to put in the packet that it sends. Let $p_{w,uv}$ be the (fixed) probability that when it receives $u \in [B]$, it sends $v \in [B]$. The behavior of the node is fully specified by the choice of $p_{w,uv}$ for every $w \in [s]$ and $u, v \in [B]$. Any such choices that obey the natural restrictions $p_{w,uv} \in [0, 1]$ and $\sum_{v \in B} p_{w,uv} = 1$ can be implemented.

Let us assume that the $p_{w,uv}$ are fixed, and that all nodes N_i, $i = 1, 2, \ldots, m$, follow the same protocol given by these values. If the packets reaching N_i have some probability distribution $X_i \in \Delta^{B-1}$, then the packets leaving it have the distribution X_{i-1} given by

$$X_{i-1,v} = \sum_{u \in [B]} p_{w,uv} \cdot X_{i,u}, \tag{1}$$

where $w \in [s]$ is the message N_i wants to send. In other words, we have $X_{i-1} = f(X_i)$, where $f_w \colon \Delta^{B-1} \to \Delta^{B-1}$ is the affine map given by (1). We note that *any* affine map $f \colon \Delta^{B-1} \to \Delta^{B-1}$ can appear as f_w. Indeed, given such an f, the corresponding $p_{w,uv}$ is the vth coordinate of $f(\mathbf{e}_u)$, where $\mathbf{e}_u = (0, 0, \ldots, 0, 1, 0, \ldots, 0)$ (with the 1 at position u) is the uth vertex of Δ^{B-1}.

We prefer this way of regarding the protocol executed by the nodes as affine maps. Thus, by a **protocol** for the nodes we mean an s-tuple $F = (f_1, f_2, \dots, f_s)$ of affine maps $\Delta^{B-1} \to \Delta^{B-1}$, where f_w is the map "executed" by a node with message $w \in [s]$.

If the adversary (node N_{m+1}) sends packets to N_m with some distribution X_m, the path is described by the string $W = w_1 w_2 \cdots w_m \in [s]^m$, and each node N_i, $i = 1, 2, \dots, m$, follows the protocol F, then the distribution X_0 "seen" by the victim is $f_W(X_m)$, where f_W is the composed map $f_{w_1} \circ f_{w_2} \circ \cdots \circ f_{w_m}$. In this section we assume X_m arbitrary, i.e., the adversary is free to choose any initial distribution.

DEFINITION 3. *A protocol F as above has* **m-step resolution** *at least β if for every two distinct strings $W, W' \in [s]^m$, the (Euclidean) distance of the sets $f_W(\Delta^{B-1})$ and $f_{W'}(\Delta^{B-1})$ is at least β.*

By Lemma 2, if F has m-step resolution at least some $\beta > 0$, then the victim can reconstruct the string sent by the m nodes (with probability close to 1) after receiving $T = O(1/\beta^2)$ packets, no matter what initial distribution X_m the adversary chooses. Thus, in order to prove Theorem 1, it suffices to exhibit protocols F with m-step resolution $\Omega(m^{-1}s^{-m/(B-1)})$ for the values of B and s listed in the theorem.

An example. We begin with an example where $B = 3$ and $s = 2$. Let $a_1 = \mathbf{e}_1$, $a_2 = \mathbf{e}_2$, $a_3 = \mathbf{e}_3$ be the vertices of the equilateral triangle Δ^2. First we introduce two auxiliary affine maps $g_1, g_2 \colon \Delta^2 \to \Delta^2$. The map g_1 is given by $g_1(a_1) = a_1$, $g_1(a_2) = a_3$, and $g_1(a_3) = \bar{a} = \frac{1}{2}(a_1 + a_2)$; see Fig. 1(a).The image $g_1(\Delta^2)$ is the gray triangle Δ_1. Similarly, g_2 is given by $g_2(a_1) = a_3$, $g_2(a_2) = a_2$, $g_2(a_3) = \bar{a}$, and it maps Δ^2 to Δ_2.

For an affine map $f \colon \Delta^{B-1} \to \Delta^{B-1}$, a point $c \in \Delta^{B-1}$, and a real number $\varepsilon \in (0, 1)$, we define the *$(1-\varepsilon)$-shrinking* of f (with center c) as the affine map $\tilde{f} \colon \Delta^{B-1} \to \Delta^{B-1}$ given by $\tilde{f}(x) = f((1-\varepsilon)(x - c) + c)$. Intuitively, we first shrink Δ^{B-1} by the factor $1 - \varepsilon$ from the center c and then we apply f.

Continuing with our example, we let c be the center of gravity of Δ^2, we choose a small $\varepsilon > 0$, and we define f_u as the $(1-\varepsilon)$-shrinking of g_u, $u = 1, 2$. In Fig. 1(b) we have $\varepsilon = 0.1$, f_1 maps Δ^2 to the shaded left triangle and f_2 maps it to the right shaded triangle. For $m = 4$, the sets $f_W(\Delta^2)$ for the 16 different strings $W \in [s]^m = [2]^4$ are the 16 small shaded triangles in Fig. 1(c).

We claim that if we set $\varepsilon = \frac{1}{m}$ in the above construction, then the m-step resolution of $F = (f_1, f_2)$ is $O(m^{-1}2^{-m/2})$; thus, it is quasioptimal (for $s = 2$ and $B = 3$). The definition below captures properties of g_1 and g_2 used in the proof of this fact, and we state it for arbitrary B and s.

α-tilings. First we introduce, for a map $f \colon \Delta^{B-1} \to \Delta^{B-1}$, the quantity $\mathrm{contr}(f)$, which is defined to be the smallest factor by which f contracts distances; that is, $\mathrm{contr}(f) = \inf_{x,y \in \Delta^{B-1}, x \neq y} \frac{\|f(x)-f(y)\|}{\|x-y\|}$.

DEFINITION 4. *Let $g_1, g_2, \dots, g_s \colon \Delta^{B-1} \to \Delta^{B-1}$ be affine maps such that the sets $g_u(\Delta^{B-1})$ for $u \in [s]$ have disjoint interiors. Let $\alpha \in (0, 1)$. We call $(g_1, g_2, \dots, g_s)$ an* **α-tiling** *(of Δ^{B-1}) if for all $m \geq 1$ and for each $W \in [s]^m$ we have $\mathrm{contr}(g_W) \geq \delta\alpha^{-m}$ for a positive constant δ (independent of m but possibly depending on B, s, and the g_u). An $s^{-1/(B-1)}$-tiling is called an* **asymptotically optimal tiling**.

LEMMA 5. *Let $(g_1, \dots, g_s)$ be an α-tiling of Δ^{B-1} and let c be an interior point of Δ^{B-1}. For a given m and $u \in [s]$, let f_u be the $(1-\frac{1}{m})$-shrinking of g_u with center c. Then the protocol $(f_1, \dots, f_s)$ has m-step resolution at least $\Omega(m^{-1}\alpha^m)$ (implicit constants depending on the g_u and on c).*

The proof is not difficult and we omit it. For our example above, it remains to verify that (g_1, g_2) is an asymptotically optimal tiling. The condition that is not obvious is $\mathrm{contr}(g_W) \geq \Omega(2^{-m/2})$ for all $W \in [2]^m$. It can be checked by direct geometric arguments, but we offer a more conceptual proof that generalizes easily.

Let us consider the affine map h that maps the right isosceles triangle S in Fig. 1(d)to the equilateral triangle Δ^2, with $h(a_j') = a_j$. The map $r_1 = h^{-1} \circ g_1 \circ h \colon S \to S$ maps S to its left half S_1, and $r_2 = h^{-1} \circ g_2 \circ h$ maps S to S_2. The important point is that both S_1 and S_2 are similar to S with ratio $2^{-1/2}$, and r_1 and r_2 are isometries followed by scaling by the factor $2^{-1/2}$. Hence $\mathrm{contr}(r_W) = 2^{-m/2}$ for any $W \in [2]^m$. For g_W we then easily find that $g_W = h \circ r_W \circ h^{-1}$. Therefore $\mathrm{contr}(g_W) \geq \mathrm{contr}(h) \cdot \mathrm{contr}(r_W) \cdot \mathrm{contr}(h^{-1}) = \Omega(2^{-n/2})$, and we have thus verified that (g_1, g_2) is an asymptotically optimal tiling.

Asymptotically optimal tilings and simplex s-reptiles. The technique for showing that our example yields an asymptotically optimal tiling can be generalized in an obvious manner and it connects the problem to a classical area of combinatorial geometry.

The following notion has been studied in various contexts (see, e.g., [3, 18, 10, 15]): A closed set $S \subset \mathbf{R}^d$ with nonempty interior is called an **s-reptile** if there are sets $S_1, S_2, \dots, S_s$ with disjoint interiors and with $S = S_1 \cup S_2 \cup \cdots \cup S_s$ that are all congruent and similar to S. For each S_u, let $r_u \colon S \to S_i$ be an affine map of S onto S_u that witnesses the similarity of S_u to S; that is, it is an isometry followed by scaling by the factor $s^{-1/d}$. We call $r_1, r_2, \dots, r_s$ a *reptiling map system* of S.

The above example was based on the fact that the right isosceles triangle S is a 2-reptile. The following result establishes a close connection between asymptotically optimal tilings and simplex reptiles:

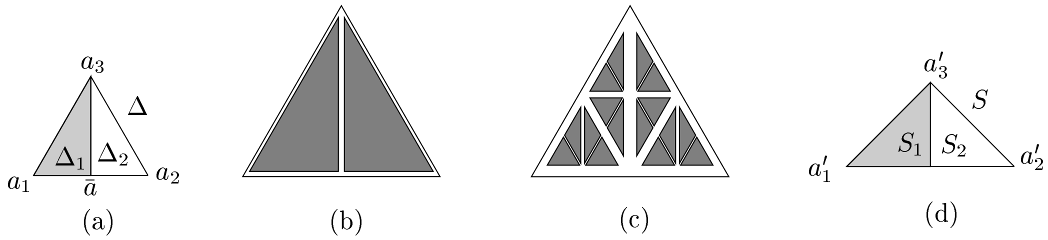
THEOREM 6. *(i) Let S be a $(B-1)$-dimensional simplex that is an s-reptile, and let $h \colon S \to \Delta^{B-1}$ be an affine bijection. Let us put $g_u = h \circ r_u \circ h^{-1}$, $u \in [s]$. Then $(g_1, g_2, \dots, g_s)$ is an asymptotically optimal tiling of Δ^{B-1}.*

(ii) Every asymptotically optimal tiling of Δ^{B-1} can be obtained from some s-reptile simplex as in (i). Moreover, given affine maps $g_1, g_2, \dots, g_s \colon \Delta^{B-1} \to \Delta^{B-1}$ such that the sets $g_u(\Delta^{B-1})$ have disjoint interiors, it can be checked in polynomial time whether $(g_1, \dots, g_s)$ is an asymptotically optimal tiling.

PROOF. Part (i) is proved by repeating the considerations in the above example almost verbatim. It remains to deal with part (ii).

Let us assume that $(g_1, \dots, g_s)$ is an asymptotically optimal tiling of Δ^{B-1}. Easy volume considerations show that the simplices $g_u(\Delta^{B-1})$ have equal volumes and tile Δ^{B-1} without overlap.

Let us write $d = B - 1$ and let us assume that Δ^d is isometrically embedded in $\mathbf{R}^d$. Let $\ell_u \colon \mathbf{R}^d \to \mathbf{R}^d$ be the "linear part" of g_u, given by $\ell_u(x) = g_u(x) - g_u(0)$. Then $\mathrm{contr}(g_W) = \mathrm{contr}(\ell_W) = \inf\{\|\ell_W(x)\| : x \in \mathbf{R}^d, \|x\| = 1\}$.



Next, let L_u be ℓ_u scaled by $s^{1/d}$, i.e., $L_u(x) = s^{1/d} \cdot \ell_u(x)$. Then L_u is volume-preserving.

For $W \in [s]^m$ we set $C_W = L_W(B^d) = \{L_W(x) : x \in \mathbf{R}^d, \|x\| \leq 1\}$, and we let $C = \bigcap\{C_W : W \in [s]^m, m = 1, 2, \ldots\}$.Each C_W is convex, and thus C is convex as well. Since each L_u is a nonsingular linear map, C_u is bounded, and hence C is bounded. Since $(g_1, \ldots, g_s)$ is an $s^{-1/d}$-tiling, there is some $\delta > 0$ such that $\mathrm{contr}(L_W) \geq \delta$ for all W. Then C contains the ball of radius δ centered at 0.

We have $L_u(C_W) = C_{uW}$, and so $L_u(C) \subseteq C$. Since L_u preserves volume and since the volume of C is finite and positive, we have $L_u(C) = C$ for all $u \in [s]$.

Let E be the ellipsoid of the smallest volume containing C (the Löwner-John ellipsoid). As is well known, the smallest-volume ellipsoid containing a given convex body is unique (see [5] for references). Hence we have $L_u(E) = E$ for all $u \in [s]$ (for otherwise, E and $L_u(E)$ would be two different smallest-volume ellipsoids containing C).

Let $h\colon \mathbf{R}^d \to \mathbf{R}^d$ be a linear map that maps the unit ball B^d onto the ellipsoid E. Then each of the linear maps $R_u = h^{-1} \circ L_u \circ h$ maps B^d to B^d, and hence it is an isometry. Thus each of the affine maps $r_u = h^{-1} \circ g_u \circ h$ is an isometry followed by scaling by $s^{-1/d}$. The simplex $S = h^{-1}(\Delta^d)$ is tiled by the images $r_u(S)$ without overlap, and it follows that S is an s-reptile.

Next, we consider the algorithmic question: Do given g_u constitute an asymptotically optimal tiling? By what we have already proved it is enough to check whether there exists a nonsingular linear map $h\colon \mathbf{R}^d \to \mathbf{R}^d$ such that each $r_u = h^{-1} \circ g_u \circ h$ is an isometry followed by scaling by $s^{-1/d}$.

The affine map g_u has the form $g_u(x) = A_u x + b_u$, where A_u is a nonsingular $d \times d$ matrix and b_u is a translation vector. In order to describe the input to the algorithm easily, we suppose that the A_u and b_u are rational (using standard machinery, the algorithm can be extended to work with algebraic numbers, say).

In the matrix language, we ask whether there is a nonsingular $d \times d$ matrix T (the matrix of h) such that all the matrices $Q_u = s^{1/d} \cdot T^{-1} A_u T$, $u \in [s]$, are orthogonal, i.e. such that $Q_u^T = Q_u^{-1}$. This condition can be rewritten to $s^{2/d} \cdot P A_u^T = A_u^{-1} P$ for all u, where P is the matrix TT^T. As is well known, a square matrix P can be written in the form TT^T iff it is positive semidefinite. So we have a semidefinite programming problem with an unknown matrix P, which is solvable in polynomial time (see, e.g., [13]). We do not know whether there is any more direct algorithm. □

We are thus led to the question, for what s and d does there exist a d-dimensional simplex that is an s-reptile? Obviously, the answer is positive for $d = 1$ and all s. For $d = 2$, all s-reptile triangles have been characterized [18], and in particular, they exist iff s is of the form i^2, $3i^2$, or $i^2 + j^2$, for integers i and j. The only known examples for $d \geq 3$ seem to be the Hill simplices (or Hadwiger-Hill simplices; see, e.g., [7]), which provide s-reptile d-simplices for all s of the form $s = i^d$. These examples and the above discussion conclude the proof of Theorem 1.

Further research. (i) No s-reptile simplex is known for $d \geq 3$ and $s < 2^d$. Motivated by the present paper, it was shown in [14] that no 2-reptile simplices exist for $d \geq 3$. The general existence problem for d-dimensional s-reptile simplices appears challenging.

(ii) We have shown that an asymptotically optimal tiling has to come from an s-reptile simplex. It would be interesting to decide whether this is also the only possible way to obtain a *quasioptimal tiling* of Δ^{B-1}, i.e. one with the condition on $\mathrm{contr}(g_W)$ weakened to $\mathrm{contr}(g_w) \geq (1 - o(1))^m s^{-m/(B-1)}$.

(iii) How efficiently can we find (or approximate) the largest α such that given affine maps $g_1, \ldots, g_s\colon \Delta^{B-1} \to \Delta^{B-1}$, such that the images $g_u(\Delta^{B-1})$ have disjoint interiors, constitute an α-tiling? If we rephrase this in a matrix language, we arrive at the at the following (more general) problem, about which we haven't found anything in the literature: Given $d \times d$ matrices $M_1, M_2, \ldots, M_s$, can we decide (at least in some approximate sense) whether $\sup\{\|M_W x\| : x \in \mathbf{R}^d, \|x\| \leq 1, W \in [s]^m, m = 1, 2, \ldots\} < \infty$?

(iv) What is the best possible m-step resolution of a protocol, for given s and B? Our current bounds are between $O(m^{-1/(B-1)} s^{-m/(B-1)})$ (always) and $\Omega(m^{-1} s^{-m/(B-1)})$ (if an asymptotically optimal tiling exists). Suppose that for some B and s and for infinitely many m there is a protocol with m-step resolution $\Omega(m^{-1} s^{-m/(B-1)})$. Does this imply the existence of an asymptotically optimal tiling?

4. RESTRICTED PROTOCOLS FOR ANY B AND s

In the previous section we have constructed quasioptimal protocols for certain combinations of values of B and s. For other cases, such as $s = 2$, $B \geq 4$ we suspect that no quasioptimal protocols exist, although we cannot prove it.

In such cases, several approaches are possible. First, we can try to construct suboptimal but still good protocols, under the same requirements. The results of [1] yields a protocol with $T \leq s^{4n/B}$. One can try to look for α-tilings of Δ^{B-1} with α as large as possible. In this context, the problems raised at the end of the last section become even more relevant, since we would like to be able to estimate the best α for given candidate tilings. Another possibility is to relax the requirements on the protocol. In the following theorem we offer two versions (attaining quasioptimality).

THEOREM 7. *(i) For every s and B there exist a region $D \subset \Delta^{B-1}$ (convex and with nonempty interior) and a protocol such that such that if the distribution X_m of packets* *generated by the adversary is guaranteed to lie in D, then the victim can reconstruct all m messages sent by the nodes (with constant probability) using* $T \leq O(s^2 B^2 2^{2m(\log_2 s)/(B-1)})$ *packets.*

(ii) For every s, B, and every function φ *on the natural numbers with* $\lim_{n\to\infty} \varphi(n) = \infty$*, there exists a (quasioptimal) protocol such that no matter what distribution* X_m *is generated by the adversary, the victim can reconstruct (with high probability) the messages of the nodes* $N_{m-\varphi(m)}$ *through* N_1 *using* $T \leq 2^{(2+o(1))m(\log_2 s)/(B-1)}$ *packets.*

Part (i) is proved using a suitable s-reptile, in a way similar to Theorem 6(i). This time the s-reptile is not a simplex, but rather, a suitable d-dimensional rectangular box R, where write $d = B - 1$. First we define an auxiliary simplex $S = \{X \in \mathbf{R}^d : x_1, \ldots, x_d \geq 0, \sum_{i=1}^d x_i \leq 1\}$. Then we set $\rho = s^{-1/d}$ and $\lambda = 1 - \rho$, and we define R as the rectangular box $\prod_{i=1}^d [0, \lambda\rho^{i-1}]$. The reptiling map r_u is given by $r_u(x_1, x_2, \ldots, x_d) = (\rho x_d + \frac{\lambda}{s}(u - 1), \rho x_1, \rho x_2, \ldots, \rho x_{d-1})$. That is, the box is sliced into s congruent boxes by parallel slices perpendicular to the longest side. We let $h\colon S \to \Delta^{B-1}$ be an affine bijection, and we define a protocol $(f_1, f_2, \ldots, f_s)$ by $f_u = h \circ r_u \circ h^{-1}$. It is easily checked that $r_u(S) \subseteq S$, and hence the maps f_u indeed map Δ^{B-1} into Δ^{B-1} and constitute a protocol.

To define the region D where the initial distributions X_m are permitted to lie, we define $R' = \prod_{i=1}^d [\frac{\lambda}{4}\rho^{i-1} \frac{3\lambda}{4}\rho^{i-1}]$ as the $\frac{1}{2}$-shrinking of R from its center and we set $D = h(R')$. It remains to verify that the protocol restricted to D has m-resolution $\Omega((sB)^{-1} s^{-m/(B-1)})$. This is a simple calculation which we omit.

For part (ii), we start with the protocol $(f_1, \ldots, f_s)$ as in (i). Then we choose a parameter $\eta > 0$, depending on s, B and φ and tending to 0 as $m \to \infty$ but very slowly, and we replace the reptiling maps r_u at the beginning of the construction by their $(1-\eta)$-shrinking with center at the origin (i.e. at a vertex of S; this is different from our previous shrinking operations, where the center was always an interior point of the considered region). Let $(\tilde{f}_1, \ldots, \tilde{f}_u)$ be the protocol obtained by the construction from the shrunk r_u's. The m-step resolution decreases somewhat by the $(1-\eta)$-shrinking of the r_u but asymptotically this won't matter. The point is that no matter what initial distribution X_m is generated by the adversary, its images after $\varphi(m)$ steps, i.e. $\tilde{f}_V(X_m)$ for all $V \in [s]^{\varphi(m)}$, are guaranteed to lie in the region $h(R)$, for which the protocol already "works." This is verified by a direct, although not entirely short, calculation.

This finishes a sketch of the proof of Theorem 7.

5. LOWER BOUND FOR A SINGLE PATH OF ATTACK

THEOREM 8. *For any protocol* $\mathcal{P}$*, let T be the expected number of packets received by the Victim and* $w(\mathcal{P})$ *be the probability that the Victim does not return the input string given to the Network when that input is chosen uniformly at random from the set of all* 2^n *possible n-bit strings. If* $w(\mathcal{P}) \leq 1/2$*, then* $T \geq \Omega\left(B \cdot 2^{2n/(B-1)}\right)$.

PROOF. (Sketch.) Let *permutation oblivious* protocols be the restricted class of protocols where the Victim waits until it has received exactly T packets, where T depends only on n and B. It then ignores the order that the packets arrive and considers only the *receipt profile* X, which is the B-tuple from $\Delta_{\langle B,T\rangle} = \left\{X = (x_1, \ldots, x_B) : \sum_{u=1}^B x_u = T\right\}$, where x_u is the number of packets of type u received by the Victim. In a permutation oblivious protocol, the Victim's strategy is specified by the function $V(X, W)$ which is the probability that when the Victim receives receipt profile X, it guesses that the Network's n-bit string is W.

Adler in [1] proves that for any general protocol $\mathcal{P}$, there is a permutation oblivious protocol $\mathcal{P}'$ for the Victim, where $w(\mathcal{P}') \leq w(\mathcal{P}) + 1/4$, and $\mathcal{P}'$ uses at most 4 times as many packets as $\mathcal{P}$. Intuitively, this is because the Network has no memory and thus no sense of time, and so the order in which the packets arrive is not useful information for the Victim. Thus, to prove the lower bound for general protocols that make a mistake with probability at most 1/2, it suffices to prove a lower bound on permutation oblivious protocols that make a mistake with probability at most 1/4.

The lower bound in [1] follows from the fact that if the n-bit string W is communicated by the Network "sending" a receipt profile X, then the number of packets T must be large enough that the number of different receipt profiles is at least the number of possible values held by the Network. We improve on this idea via a technique to account for the fact that the Network does not have full control over the receipt profile that it sends. In particular, all that a protocol is able to do is specify the probability $N(W, u)$ that the Network sends a packet of type u when its n-bit string is W. This same probability is used independently when sending each of the T packets. This in turn induces the probability $N(W, X)$ that receipt profile X is "sent" by the Network when its n-bit string is W. We demonstrate that no matter what the Network does, the probability that it sends a particular profile X is exponentially small.

LEMMA 9. *Given n-bit string W and any receipt profile* $X = (x_1, \ldots, x_B)$*, the probability* $N(W, X)$ *that the Network "sends" X when having W is at most* $N(X) = \frac{\sqrt{T}}{\mathrm{e}} \cdot \Pi_{u=1}^B \frac{\mathrm{e}}{\sqrt{x_u}}$.

We do not prove this lemma in this extended abstract, but for intuition consider an easier situation in which each packet is obtained by an independent Bernoulli trial. If the Network can color each packet either red or blue independently with any fixed probability p of its choice and it wants to color exactly x of them red, then the best it can do is to set $p = \frac{x}{T}$. As such, the expected number of red packets is $x = pd$. Furthermore, with constant probability the actual number of packets is fairly uniformly distributed within a range of $\sqrt{x} = \sqrt{pT}$ of this expected number. As a result of this, the probability of getting exactly x reds is approximately $\frac{\mathrm{e}}{\sqrt{x}}$. Note that this probability does not depend on the number of packets T. The reason that the probability of the Network sending exactly X is at most $\frac{\sqrt{T}}{\mathrm{e}} \cdot \Pi_{u=1}^B \frac{\mathrm{e}}{\sqrt{x_u}}$ is that it must get the exact number x_u of each type of packet.

We use this Lemma to prove Theorem 8. We consider the quantity $\rho(\mathcal{P}) = (1 - w(\mathcal{P})) \cdot 2^n$. Since $1 - w(\mathcal{P})$ is the probability that using $\mathcal{P}$, the Victim returns the input string given to the Network when that input is chosen uniformly at random from the set of all 2^n possible n-bit strings, we have $\frac{1}{4} \cdot 2^n \leq \rho(\mathcal{P})$. Furthermore, $\rho(\mathcal{P}) = \sum_{W\in\{0,1\}^n} \Pr[\text{Victim returns } W \mid \text{Network has } W] =$

$\sum_{W\in\{0,1\}^n}\sum_{X\in\Delta_{\langle B,T\rangle}}$ Pr[Network sends profile X and Victim returns W | Network has W] $= \sum_{W\in\{0,1\}^n}\sum_{X\in\Delta_{\langle B,T\rangle}} N(W,X)\cdot V(X,W)$. Computing this sum becomes difficult because of the interplay between what the Network and the Victim do. However, we decouple them by not forcing the Network to run a fixed protocol when it has the message W, but instead for each distribution of packets X allow it to use the protocol that will maximize its probability of sending X. This can increase the probability of success. However, we know from Lemma 9 that no matter what the Network does, $N(W,X) \leq N(X)$. This conveniently decouples our sum, giving us that $\rho(\mathcal{P}) \leq \sum_{X\in\Delta_{\langle B,T\rangle}} N(X)\cdot\sum_{W\in\{0,1\}^n} V(X,W) \leq \sum_{X\in\Delta_{\langle B,T\rangle}} N(X)\cdot 1$. By plugging in the value for $N(X)$, we get $\rho(\mathcal{P}) \leq \sum_{X\in\Delta_{\langle B,T\rangle}} \frac{\sqrt{T}}{\mathbf{e}}\cdot\Pi_{u=1}^{B}\frac{\mathbf{e}}{\sqrt{x_u}}$.

What remains is to bound this sum. In particular, we can demonstrate that it implies $\rho(\mathcal{P}) \leq \left(\frac{(3.4)^2\mathbf{e}^3 T}{B-1}\right)^{(B-1)/2}$. Again, we do not have space for the full proof, but for intuition, the requirement that $\sum_{u=1}^{B} x_u = T$ leaves $B-1$ independent values of x_u, each roughly T/B. Hence, there are approximately $(T/B)^{B-1}$ terms in the sum $\sum_{X\in\Delta_{\langle B,T\rangle}} \Pi_{u=1}^{B}\frac{1}{\sqrt{x_u}}$. Each term is roughly $(\frac{1}{\sqrt{T/B}})^B$. This gives a total of about $(T/B)^{B-1}\cdot(\frac{1}{\sqrt{T/B}})^B = \sqrt{T/B}^{B-2}$. Finally, using $\frac{1}{4}\cdot 2^n \leq \rho(\mathcal{P})$ and solving for T gives $T \geq \Omega\left(B\cdot 2^{2n/(B-1)}\right)$. □

6. PROTOCOL FOR MULTIPLE PATHS

In this section, we provide our new protocol for the case of multiple paths of attack. Recall that for this case of the problem, there are up to k different paths, each of which contains a different set of m nodes. The adversary chooses which path each packet travels on; the victim only sees the contents of the final packet it receives - it does not know which path that packet traveled on. The goal is to design a protocol where the victim can 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 refer to the string of information along the jth path as W_j. Also, we refer to the node at distance i from the victim on the jth path as N_i^j.

The minimum value of B required by this protocol is $2k+1$, which is the same as that achieved by a protocol provided in [1]. The main improvement of the new protocol is the assumptions made on the underlying network. In particular, the protocol from [1] assumes that the adversary always sets the initial bits to 0. Thus, it would be quite easy for the adversary to disable that protocol. We here provide a protocol that works for *any* scheme by the adversary to set the initial bits. Instead, we show that the combination of two other and more realistic assumptions is sufficient. We point out that the lower bound of $B \geq 2k-1$, shown in [1], still applies with the two assumptions made here.

First, we assume that the value to be sent by each node in the system is chosen independently and uniformly at random. This is justified in the IP Traceback scenario, since the attacks these techniques protect against occur from compromised nodes in the Internet. Our assumption here corresponds to the assumption that the descriptions of the paths to compromised nodes are distributed randomly, as opposed to being a worst case distribution.

Our second assumption is that the nodes along each path have a small amount of information as to their location along that path. Note that this is also a reasonable assumption in the Internet, since a node 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 N_i^j has a predicate C such that if N_i^j has distance of at most $2\log k+1$ hops from the victim of the attack, then $C(N_i^j)$ = TRUE, and if i is the node adjacent to the adversary, then $C(N_i^j)$ = FALSE (and hence $m \geq 2\log k+2$). For the remainder of the nodes along the path, the value of $C(N_i^j)$ can be either TRUE or FALSE. For example, if $2\log k < m/2$, 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 $C(N_i^j)$ is constant for all j, and we assume that all N_i^j for which $C(N_i^j)$ = TRUE are closer to the victim than any N_i^j for which $C(i)$ = FALSE. We denote by $C_{\max}$ the number of i for which $C(N_i^j)$ = TRUE.

We also point out that the protocol we describe here is based on that introduced in [1], with a number of changes. The real innovation of the result presented here is not as much these changes as a greatly improved analysis technique. This new analysis technique allows us to prove these results. It also has considerable potential to address the question of tight tradeoffs for the number of packets required in the multiple path case, an interesting open problem.

6.1 The protocol

We here describe the protocol for the case where $s=2$ (i.e., each node has a single bit), but our technique can easily be adapted to any s that is a power of 2. We here assume that the protocol is designed for a specific upper bound on k: the protocol is not required to know how many paths the adversary is using. Instead, it works (with high probability), as long as the the adversary does not use more than k paths. For simplicity, we also here assume that $B = 2k+1$. For larger values of B, the remaining states of the marking bits are treated as being equivalent to state 0, and thus are not used. Let $d = 2k = B-1$. We define two different mappings from a probability distribution over packets to a probability distribution over packets. For each of these, let $p_{u,v}$ be the probability that the packet u gets mapped to packet v. Consider first the mapping **zero**:

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

The second mapping is called **one**:

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

The protocol from [1] consists of a node with the bit 0 simply applying mapping **zero**, and a node with the bit 1 applying mapping **one**. In the new protocol, a node N_i^j with the bit 0 and $C(N_i^j)$ = TRUE applies the mapping **zero** twice, followed by the mapping **one** once, followed by three more applications of the mapping **zero**. A node N_i^j with the bit 1 and $C(N_i^j)$ = TRUE applies the same process, except that the last mapping **zero** is replaced with a **one**. A node N_i^j with the bit 0 and $C(N_i^j)$ = FALSE applies the mapping

zero $ck+1$ times, for a suitable constant c to be described below. A node N_i^j with the bit 1 and $C(N_i^j)$ = FALSE applies the mapping **zero** ck times, followed by the mapping **one**. This completes the description of the encoding portion of the protocol.

THEOREM 10. *For any* α*, and any* $\delta > e^{-\frac{2}{3}k} + 2^{2\log k - C_{\max}}$*, there is a value* $T(\alpha,\delta)$*, such that after the victim has received at least* $T(\alpha,\delta)$ *packets, with probability at least* $1-\delta$*, he 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 adversary sends.*

We point out that the lower bound on δ is due to a requirement that the set of k strings "look" random (in a manner we make formal below). For any set of strings that meet this requirement, the probability of success can be made arbitrarily close to 1.

PROOF. Assume first that the adversary sets the initial value of every packet to 0 (as was assumed throughout in the protocol of [1]). Later, we shall see how to relax this assumption. Let $p_u(W_j)$ be the probability that a packet, with initial value 0, sent on a path with string W_j, arrives at the victim set to the value u. Let w_i^j be the ith bit (starting from the victim) of the string W_j. Let

$$X_{W_j} = \sum_{i=1}^{C_{\max}} (\frac{1}{2})^{6(i-1)+1}(w_i^j + \frac{1}{8}) + \sum_{i=C_{\max}+1}^{m} (\frac{1}{2})^{(ck+1)(i-C_{\max}-1)+6C_{\max}+1} w_i^j.$$

We shall refer to X_{W_j} as the value of the string W_j. Note that X_{W_j} is the real number with a binary representation where the bit representing 2^{-t} is a 1 if and only if the tth mapping (counting from last to first) applied to the probability distribution is the mapping **one**. Thus, 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 $0 < u \le d$:

$$p_u(W_j) = \left(\frac{X_{W_j}}{4}\right)^u, \tag{2}$$

Thus, if the victim could determine a sufficiently good estimate on $p_u(W_j)$, for any u, $0 < u \le d$, it would have enough information to determine the string W_j. However, the adversary is able to "hide" the $p_u(W_j)$s by choosing what fraction of the packets are sent on each of the different paths. Let λ_j be the fraction of the received packets that are sent by the adversary with string W_j. The probability that a randomly chosen packet from the set of packets received by the victim has its bits set to u is $q_u = \sum_{j=1}^k \lambda_j p_u(W_j)$. The set of received packets provides the victim with an estimate on the values of the q_u. Although the stochastic variance inherent to the communication process means that it is unlikely for the victim to know the q_us exactly, we first assume that the victim is given the exact values of the q_us, and demonstrate that this uniquely determines the entire set of strings used by the adversary. This allows us to build some intuition for why the victim is able to decode the set of strings in the actual scenario. We shall then remove both this assumption, as well as the assumption that the adversary set the initial bits to 0.

We show that if we assume that the q_us do not determine the strings uniquely, this leads to a contradiction. Let $V(W_j)$ be the $2k$-dimensional vector where component u of $V(W_j)$, for $1 \le u \le 2k$, is $p_u(W_j)$. We shall refer to $V(W_j)$ as the *string vector* for W_j. Assume that there is some set of strings $W_{k+1} \dots W_{2k}$ and probabilities $\lambda_{k+1} \dots \lambda_{2k}$ such that $\sum_{j=1}^k \lambda_j V(W_j) = \sum_{j=k+1}^{2k} \lambda_j V(W_j)$. For the set of strings to not be uniquely determined, it must be the case that there is some string W_j with $\lambda_j > 0$ such that if $j \le k$ then $W_j \notin \{W_{k+1},\dots,W_{2k}\}$, and if $j > k$ then $W_j \notin \{W_1,\dots,W_k\}$. Assume here that such a string is W_{2k}; the case where $j \le k$ is similar. In this case, we see that

$$\lambda_{2k} V(W_{2k}) = \sum_{j=1}^{k} \lambda_j V(W_j) - \sum_{j=k+1}^{2k-1} \lambda_j V(W_j). \tag{3}$$

There may be strings that appear in both $W_1,\dots,W_k$ and $W_{k+1},\dots,W_{2k}$. However, by replacing any such string with another unused string, we see that (3) implies that there is some set of $2k$ distinct strings $W'_1 \dots W'_{2k}$ and real numbers $\lambda'_1 \dots \lambda'_{2k}$, with $\lambda'_{2k} > 0$, such that

$$\lambda'_{2k} V(W'_{2k}) = \sum_{j=1}^{2k-1} \lambda'_j V(W'_j). \tag{4}$$

Now, consider the $2k \times 2k$ matrix M where entry $M_{u,j} = p_u(W'_j)$. From (4), we see that M does not have full rank. However, from (2), we see that $M_{u,j} = \left(\frac{X_{W'_j}}{4}\right)^u$. The $2k \times 2k$ matrix M', where entry $M'_{u,j} = \left(\frac{X_{W'_j}}{4}\right)^{u-1}$, is a Vandermonde matrix. Since the strings $W'_1 \dots W'_{2k}$ are distinct, if $j \ne j'$ then $X_{W'_j} \ne X_{W'_{j'}}$, and thus M' has full rank. Since, for all strings W_j, $X_{W_j} \ne 0$, the matrix M must have full rank as well, which is a contradiction. Therefore, the exact values of the q_u exactly determines all strings W_j, $1 \le j \le k$, such that $\lambda_j > 0$.

We next examine the effect of removing our two assumptions. In particular, 1) instead of the victim knowing the values of the q_u exactly, it only has the information provided by the packets it has received: a series of samples from the probability distribution. Also, 2) the adversary, 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_u as a point in B-dimensional space, where the coordinate for dimension u is q_u. The effect of removing both of the two assumptions above is that instead of knowing the exact point defined by the q_us, 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_us. Let $D_0 = \frac{6}{2^{6C_{\max}+(ck+1)(m-C_{\max})}}$. The estimate of the point Q that is used is as follows: the victim collects $T = \frac{6k}{D_0^2}\ln\frac{2k}{\delta}$ packets. For $1 \le u \le B$, let Y_u be the number of times that packet u is seen in the T packets. We set $\bar{q}_u = Y_u/T$.

The victim only returns sets of strings that are likely to lead to seeing the $\bar{q}_u$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.

DEFINITION 11. *We say that a set of k strings $W_1, \ldots, W_k$ is* well dispersed *if* $\forall j, 1 \le j \le k, \Pi_{i \ne j} |X_{W_i} - X_{W_j}| \ge 2^{-32k}$.

The victim returns any string W_j such that W_j is contained in a convex combination of at most k string vectors, with the coefficient associated with W_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 $\bar{q}_u$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 adversary has a set of strings that is well dispersed. The proof of the following is omitted due to lack of space.

CLAIM 12. *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 $e^{-\frac{2}{3}k} + 2^{2\log k - C_{\max}}$.*

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 adversary 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 W_j of at least $\frac{\alpha}{k}$, where W_j is not used by the adversary, has a Euclidean distance from Q of more than $2D_0$. Let $D_q = \sqrt{\sum_{i=1}^{2k}(q_u - \bar{q}_u)^2}$.

LEMMA 13. $\Pr[D_q > D_0] \le \delta$.

PROOF. Note that for each u, $|\bar{q}_u - \mathrm{E}[\bar{q}_u]|$ is the distance caused by stochastic variation, and $|q_u - \mathrm{E}[\bar{q}_u]|$ is the distance caused by the adversary not setting the initial bits to 0. Standard Chernoff bound techniques demonstrate that with N packets, $\Pr[\sqrt{\sum_{i=1}^{2k}(\mathrm{E}[\bar{q}_u] - \bar{q}_u)^2} \ge D_0/2] \le \delta$. Thus, we only need to demonstrate that the effect of the adversary 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 nodes are linear, it is sufficient for us to consider each of the cases where the adversary 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 adversary must be some convex combination of these strategies.

CLAIM 14. *For u a positive integer, let $\mu(u) = \max(0, u - 2)$. After a packet has had $\ell \ge 1$ sets of three mappings applied to it, where the first two mappings in each set are the mapping* **zero**, $|q_u - E[\bar{q}_u]| \le \frac{1}{2^{3\ell + \mu(u)}}$.

(Proof omitted.)This implies the Lemma, since the distance from Q is at most $\frac{3}{2^{6C_{\max} + (ck+1)(m - C_{\max})}}$. □

Note that Lemma 13 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 $D_q \le D_0$ also implies that there can be no string P not used by the adversary such that P is returned by the victim.

LEMMA 15. *If the set of strings used by the adversary is well dispersed, then every convex combination of k well dispersed string vectors that contains a string W_j, not used by the adversary, with a coefficient of more than $\frac{\alpha}{k}$, has a Euclidean distance from Q of at least $2D_0$.*

PROOF. If a string W_j as described by the Lemma exists, then there must be some set of strings $W_1 \ldots W_{2k}$, where $W_1 \ldots W_k$ are the well dispersed strings used by the adversary, $W_{k+1} \ldots W_{2k}$ are the well dispersed strings contained in the incorrect convex combination, and W_{2k} is the string returned incorrectly. Thus, $W_{2k} \notin \{W_1, \ldots, W_k\}$, and there exist probabilities $\lambda_1 \ldots \lambda_{2k}$, with $\lambda_{2k} \ge \frac{\alpha}{k}$, such that

$$\sqrt{\sum_{u=1}^{B}\left(\sum_{j=k+1}^{2k} \lambda_j p_u(W_j) - \sum_{j=1}^{k} \lambda_j p_u(W_j)\right)^2} \le 2D_0$$

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

$$\sqrt{\sum_{u=1}^{B}\left(\lambda'_{2k} p_u(P'_{2k}) - \sum_{j=1}^{2k-1} \lambda'_j p_u(P'_j)\right)^2} \le 2D_0 \qquad (5)$$

Let D_1 be the Euclidean distance in $\Re^{2k}$ from the point $\lambda'_{2k} V(W'_{2k})$ to the subspace spanned by $V(W'_1), \ldots V(W'_{2k-1})$. For (5) to be true, it must be the case that $D_1 \le 2D_0$. Thus, to demonstrate that no such incorrectly returned string W_{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 $V(W'_1), \ldots, V(W'_{2k-1}), \lambda_{2k} V(W'_{2k})$ in $\Re^{2k}$. Let $\mathcal{V}_{2k-1}$ be the $(2k-1)$-dimensional volume of the parallelepiped defined by the vectors $V(W'_1), \ldots, V(W'_{2k-1})$ 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. In what follows, for any string W_i, let $Y_{W_i} = \frac{X_{W_i}}{4}$.

LEMMA 16. $\mathcal{V}_{2k} = \lambda_{2k} \prod_{1 \le i < j \le 2k} \left| Y_{W'_i} - Y_{W'_j} \right| \prod_{i=1}^{2k} Y_{W'_i}$.

PROOF. Due to the convenient form of the vectors $V(W'_1), \ldots, V(W'_{2k})$, 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 $1 \le j \le 2k-1$, is $V(W'_j)$, and column $2k$ is the vector $\lambda_{2k} V(W'_{2k})$.

To compute $|det(T)|$, consider the matrix T', where column j of T', for $1 \le j \le 2k$, is $\frac{V_j}{Y_{W'_j}}$. By (2), the matrix T' is Vandermonde, and thus $det(T') = \prod_{1 \le i < j \le 2k} \left| Y_{W'_i} - Y_{W'_j} \right|$. The lemma then follows from the fact that to get T from T', we merely multiply each column i of T' by Y_{W_i}, with the exception of column $2k$, which is multiplied by $\lambda_{2k} Y_{W_i}$. □

LEMMA 17.

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

PROOF. Let $V^2(W_j)$ be the vector consisting of the components $1, Y_{W_j}, Y_{W_j}^2, \ldots, Y_{W_j}^{2k-1}$. Let $V^3(W_j)$ be the vector consisting of the components $0, Y_{W_j}, Y_{W_j}^2, \ldots, Y_{W_j}^{2k-1}$. Let

$V^4(W_j)$ be the vector consisting of the components $0, 1, Y_{W_j}, Y_{W_j}^2, \ldots, Y_{W_j}^{2k-2}$.

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 $V^e(W_1'), \ldots, V^e(W_{2k-1}')$ in $\Re^{2k}$.

Since $V(W_j)$ is simply $V_2(W_j)$ with every component multiplied by Y_{W_j}, $\mathcal{V}_{2k-1} = \mathcal{V}_{2k-1}^2 \cdot \prod_{i=1}^{2k-1} Y_{W_i'}$. Similarly, $\mathcal{V}_{2k-1}^3 = \mathcal{V}_{2k-1}^4 \cdot \prod_{i=1}^{2k-1} Y_{W_i'}$. 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 $V^4(W_1'), \ldots, V^4(W_{2k-1}')$. Since this matrix is Vandermonde, its determinant is $\prod_{1 \le i < j \le 2k-1} \left| Y_{W_i'} - Y_{W_j'} \right|$.

Claim 18. $\mathcal{V}_{2k-1}^2 \le \mathcal{V}_{2k-1}^3 \prod_{i=1}^{2k-1} \frac{1+Y_{W_i'}^{2k-1}}{Y_{W_i'}}$.

The Lemma follows (proof omitted). □

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

$$\frac{\lambda_{2k} Y_{W_{2k}'} \prod_{i=1}^{2k-1} \left(Y_{W_i'} - Y_{W_{2k}'}\right)}{\prod_{i=1}^{2k-1}(1+Y_{W_i'}^{2k-1})} \ge \frac{\lambda_{2k}}{128} \prod_{i=1}^{2k-1} \left(Y_{W_i'} - Y_{W_{2k}'}\right),$$

where the second inequality follows from the fact that for $1 \le i \le 2k$, $\frac{1}{64} \le Y_{W_i'} \le \frac{1}{4}$. 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. The proof of the following is omitted due to lack of space.

Claim 19. *Let* $S_1 = \{Y_{W_1}, \ldots, Y_{W_k}\}$ *and* $S_2 = \{Y_{W_{k+1}}, \ldots, Y_{W_{2k}}\}$ *be two sets of well dispersed string vectors such that* $Y_{W_{2k}} \notin S_1$.

$$\prod_{Y_{W_i} \in S_1 \cup S_2 - Y_{W_{2k}}} |Y_{W_i} - Y_{W_{2k}}| \ge \frac{1}{2^{65k+6C_{\max}+(m-C_{\max}-1)(ck+1)+2}}.$$

Lemma 15 (and hence the Theorem) now follows by observing that if we set $c = 72 + \log \frac{1}{\alpha}$, then for $k \ge 2$ it must be the case that $D_1 > 2D_0$.

Finally, we mention the computational efficiency of the decoding procedure. A simple solution is to try all possible $\binom{2^m}{k}$ sets of k paths, and for each set, (a) check if it is well dispersed, and (b) check to see if the corresponding string vectors have a convex combination that is sufficiently close to the observed sample from the received packets. Part (b) can be done via linear programming (strictly speaking, this requires using an L_1 norm, instead of the L_2 norm we have used in the proofs, but adapting our proofs to L_1 is not difficult). While this procedure is not very fast, the number of packets that are required is $2^{\Omega(mk)}$, and thus the decoding procedure is polynomial in the number of packets received. Determining if there exists a faster decoding algorithm is an interesting open problem.

Acknowledgments

We thank Uri Zwick for fruitful discussions at the beginning of this research. We also thank William Hesse for being the first to point out a protocol of the type depicted in Figure 1. The third author (JM) would like to thank the following people for kindly answering his queries concerning rep-tiles, matrix powers, and other things related to this paper: Christoph Bandt, Maurice Cochand, David Eppstein, Eike Hertel, Marie Hušková, Krystyna Kuperberg, Wlodzimierz Kuperberg, Petr Plecháč, and Jiří Rohn. We are particularly indebted to Yuri Lyubich for a suggestion that eventually led to the use of the convex body C in the proof of Theorem 6, and to Pavel Valtr for the right remark about smallest-volume ellipsoids at the right moment.

7. REFERENCES

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

[2] M. Adler, J. Cai, J. K. Shapiro, and D. Towsley, Estimation of Congestion Price Using Probabilistic Packet Marking. In *Proceedings of Infocom* 2003.

[3] C. Bandt, Self-similar sets. V. Integer matrices and fractal tilings of $\mathbf{R}^n$. *Proc. Amer. Math. Soc.*, 112(2), 1991.

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

[5] L. Danzer, B. Grünbaum, and V. Klee, Helly's theorem and its relatives. In *Convexity*, volume 7 of *Proc. Symp. Pure Math.*, pages 101–180. American Mathematical Society, Providence, 1963.

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

[7] H. E. Debrunner, Tiling Euclidean d-space with congruent simplexes. In *Discrete geometry and convexity (New York, 1982)*, volume 440 of *Ann. New York Acad. Sci.*, pages 230–261. New York Acad. Sci., New York, 1985.

[8] T. Doeppner, P. Klein, and A. 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.

[9] Q. Dong, M. Adler, and K. Hirata, Efficient Schemes for Probabilistic Packet Marking. University of Massachusetts, Amherst Technical Report 2004-67.

[10] G. Gelbrich, Crystallographic reptiles. *Geom. Dedicata*, 51(3):235–256, 1994.

[11] M. Goodrich, Efficient Packet Marking for Large-Scale IP Traceback. *Proc. of 9th ACM Conf. on Computer and Communications Security (CCS)*, 2002, 117-126.

[12] 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.

[13] L. Lovász, Semidefinite programs and combinatorial optimization. In B. Reed and C. Linhares-Sales, editors, *Recent Advances in Algorithms and Combinatorics*, pages 137–194. Springer, New York, 2003.

[14] J. Matoušek, Nonexistence of 2-reptile simplices. *Discrete Comput. Geom.*, 2004. Submitted.

[15] S.-M. Ngai, V. F. Sirvent, J. J. P. Veerman, and Y. Wang, On 2-reptiles in the plane. *Geom. Dedicata*, 82(1-3):325–344, 2000.

[16] 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.

[17] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, Practical Network Support for IP Traceback. In *Proceedings of ACM SIGCOMM 2000* , pp. 295–306, August 2000.

[18] S. L. Snover, C. Waiveris, and J. K. Williams, Rep-tiling for triangles. *Discrete Math.*, 91(2):193–200, 1991.

[19] D.X. Song and A. Perrig, Advanced and authenticated marking schemes for IP traceback. In *Proc. IEEE INFOCOM '01*, 2001.

[20] R. Thommes and M.J. Coates, Deterministic Packet Marking for Congestion Price Estimation. In *Proc. IEEE INFOCOM '04*.