An Analysis of the Statistical Disclosure Attack and Receiver-Bound Cover

Nayantara Mallesh and Matthew Wright

Abstract

Anonymous communications provides an important privacy service by keeping passive eavesdroppers from linking communicating parties. However, an attacker can use long-term statistical analysis of traffic sent to and from such a system to link senders with their receivers. Cover traffic is an effective, but somewhat limited, counter strategy against this attack. Earlier work in this area proposes that privacy-sensitive users generate and send cover traffic to the system. However, users are not online all the time and cannot be expected to send consistent levels of cover traffic; use of inconsistent cover traffic drastically reduces its impact. We propose that the anonymity system generate cover traffic that mimics the sending patterns of users in the system. This *receiver-bound cover* (RBC) helps to make up for users that aren't there, confusing the attacker. To study the statistical disclosure attack and different cover traffic methods, we introduce an analytical method to bound the time for an attacker to identify a contact of Alice with high probability. We use these bounds to show that cover traffic sent by Alice greatly increases the time for attacker success, especially as the amount of traffic from other users increases. Further, we show that RBC greatly enhances the defense, forcing the attacker to take additional time proportional to the amount of cover used. We also examine the effectiveness of the attack and cover traffic when the attacker can only observe part of the traffic in the system. We validate our analysis through simulations that extend to realistic social networks. When RBC is used in combination with user-generated cover traffic, the attack takes a very long time to succeed.

Keywords:

1. Introduction

Anonymity systems are fundamentally challenging to build on top of the existing Internet architecture. The simplest and most secure approaches require all participants to send messages at the same rates, e.g. one message per given time interval. Users without a message to send must send fake messages, known as *cover traffic* or *dummies*, to ensure anonymity for themselves as well as for others. This provides no allowance for the realities of node failure and network partitions. Furthermore, users are not online all the time and at the cost of their own benefit are unable to provide consistent cover traffic.

Existing implementations based on the mixes paradigm introduced by Chaum (Chaum, 1981) remove this unrealistic requirement for constant participation, but at a cost to their security. The changing group of users can be observed, along with outgoing messages, leading to powerful *intersection attacks*. In these attacks, differences in the membership of the set of users are matched with the differences in the message-sending behavior, leading to links between users and their receivers. Effectively, the attacker can observe information leaks over time.

The *statistical disclosure attack* is a particularly effective form of intersection, in which the attacker isolates his attack against a single user, whom we will call Alice. The primary statistic used in this attack is the number of messages that each recipient gets during the observation period. By taking the difference between the amount of messages observed when Alice is active and the amount observed when she is not active, the attacker can estimate Alice's contribution to the recipient. This attack has been studied previously and is well understood (Danezis, 2003; Mathewson and Dingledine, 2004).

In this work, we explore defenses against intersection attacks such as statistical disclosure. In particular, we study the relative effectiveness of different defenses, and we present the first in-depth study of the idea of sending cover traffic to recipients that are outside of the system, which we call *receiver-bound cover* (RBC). To date, the possible defenses against intersection attacks have been limited to two basic techniques: the user sending more cover traffic into the anonymity system and increasing random delays for messages in the system.

We conduct an analysis to bound from above the time an attacker needs to identify Alice's contact. For our analysis, we begin with a basic scenario in which Alice has a single contact and uses no cover traffic. We extend our analysis to study the effect of both cover traffic generate by Alice and RBC on the time required for attacker success. We further extend our analysis to investigate the impact of a partial attacker on the time for attacker success. Although our bounds are loose, they provide a way to compare between different amounts and types of cover traffic in various scenarios.

To further study statistical disclosure and the effect of cover traffic, we conducted a set of simulation experiments. The results of these simulations validate the bounds given by the analyses described above. From both analysis and simulation, we find that RBC substantially increases the amount of time required for the attacker to succeed in linking Alice with her contacts. The attack takes a very long time as the amount of cover traffic from Alice and the amount of background traffic increases. Additionally, we demonstrate how the sending activity and cover traffic of other users surprisingly fails to provide any help to Alice.

FiXme Fatal: T:Discuss dummy messages used to enhance non-traceability in ISDN mixes

In the next section, we describe our model and the statistical disclosure attack in more detail. We then motivate the two types of cover traffic that we are studying and analyze their effects in Section 3. In Section 4, we present a detailed analysis of the time needed for attacker success in the presence and absence of cover traffic as a counter measure. Section 5 presents our simulation model and results. Discussion and analysis of the feasibility and costs of the cover traffic methods are presented in Section 7. We discuss related work in Section 8 and then conclude.

2. Statistical Disclosure

The *Statistical Disclosure Attack (SDA)*, first described by Danezis (Danezis, 2003) is a long-term intersection attack against mix-based anonymity systems. SDA is an extension of the *disclosure attack* introduced by Kesdogan (Kesdogan et al., 2002). In this section, we first explain the network model used and then describe statistical disclosure attacks. We discuss why cover traffic delays statistical disclosure and how it can be used to counter this attack.

2.1. Model

Let us assume that there are N senders that wish to communicate with a set of R recipients using a mix network. The relationship between senders and receivers is many-to-many. In each *round* of communication, a set of senders send messages via the mix network to a set of receivers. The mix network may consist of a single mix or a network of connected mixes. For simplicity, we abstract away the details of the number of mixes in the mix network and refer to a single mix or a cascade of mixes as a *mix*.

2.2. Mix Types

For our study we consider two types of mixes. The first type of mix is a simple threshold mix, which collects a batch of B messages in each round and forwards them in a random order to their destinations. The second type of mix, called a binomial mix (Díaz and Serjantov, 2003), applies a weighted coin-toss to each incoming message to decide if the message leaves the mix in the current round or is delayed until a later round. The binomial mix uses a delay probability of P_{delay} to bias the decision.

2.3. Communication Links and Sending Behavior

We consider two models of the relationships between senders and receivers. The first model is a simple uniform model,**FiXme Fatal: Tara – please describe the uniform model, since it's used in the analysis and some sims.** We model the relationships between senders and receivers as a scale-free network, in which the distribution of node degrees follows a power law relationship (Barabási and Albert, 1999). This means that most senders communicate with a few well-known recipients in addition to many lesser-known recipients. The well-known recipients hence receive more messages during their communication lifetime. Senders other than Alice are called *background senders*. Background senders send more frequently to their well-known recipients than their lesser-known recipients. Alice uses a *uniform method* of sending and sends uniformly to all her recipients.

Alice and other senders also send cover traffic. We describe the types of cover traffic and our model in Section 3.

2.4. Attacker Model

The attacker is a global passive adversary who can observe all links from senders to the mix and all links from the mix to recipients. The target of the attacker is a sender Alice and the attacker's aim is to expose the set of recipients with whom Alice communicates. The attacker observes multiple rounds, including rounds with and without Alice's participation, and tries to identify Alice's recipients. The attacker can observe only the incoming and outgoing links from the mix and cannot observe activity inside the mix network. This assumption is for the simplicity of the model, as there are many configurations for a mix network, but also because the statistical disclosure attack is effective without observations of activity inside the network.

2.4.1. Partial Eavesdropper

For an attacker to be able to see all communication going into and out of the anonymity system, he must be very powerful. To have a more realistic evaluation, we also consider a *partial eavesdropper*, an eavesdropper who can observe only some of the links from the senders to the mix and from the mix to the recipients. One way a partial attacker can be implemented is as described in (Murdoch and Zieliński, 2007). By gaining control of a number of internet exchanges (IXes), the attacker is able to see a part of the traffic going into and out of a number of autonomous systems (ASes). Anonymity systems such as Mixminion and Tor contain nodes in many different AS zones. The adversary who has control of a number of IXes would be able to observe all of the traffic going in and out of these IXes.**FiXme Fatal: Tara, please explain how this is modeled, not just why it's realistic.**

2.5. Statistical Disclosure

SDA is a probability-based approach to the disclosure attack and is a practical way to expose Alice's set of recipients (Danezis, 2003). The attacker makes observations in a number of rounds in which Alice participates. In each round he records the number of messages Alice sends, the number of messages background senders send, and the number of messages each receiver receives. The attacker stores the number of messages each receiver get in an observation vector $\overline{O}$. To learn the background senders' behavior, the attacker makes observations in multiple rounds in which Alice does not participate. He records the number of messages that background senders send and the number of messages each receiver receives in a background vector $\overline{U}$. The observation vector $\overline{O}$ can be expressed as a sum of Alice's sending behavior D_A and the background vector $\overline{U}$.

$$\overline{O} = n_A D_A + \overline{U} \tag{1}$$

In Equation (1), n_A is the total number of messages sent by Alice during the period of observation. Alice's likely set of recipients can be determined by solving Equation (1) for D_A. The indices with the highest values in D_A correspond to the most likely recipients of Alice.

3. Cover Traffic

Cover traffic consists of dummy messages that are inserted into the network along with real user messages. Dummy messages have long been recognized as a useful tool to increase anonymity provided by mix-based systems. In the context of our model, cover traffic can be classified into three types based on where it is generated. *user cover* is cover traffic generated by Alice herself and *background cover* is cover traffic generated by other senders connecting to the mix. On the other hand, *receiver-bound cover (RB)* is generated by the mix and sent to message recipients.

Mathewson et.al. have shown that user cover helps delay statistical analysis (Mathewson and Dingledine, 2004). When Alice generates cover traffic randomly, e.g. with a geometric distribution, she can significantly delay the SDA. A more effective approach is for Alice to send a fixed threshold number of messages in every round. If the number of real messages is less than the threshold, then Alice inserts dummy messages to compensate for the shortage. Both of these approaches become more effective as the mix exhibits higher delay variability, since the number of possibilities that the attacker must consider increases. Even if the sender is online 100% of the time, however, sender-originated dummy packets alone are not enough to protect against statistical analysis.

3.1. Background Cover Traffic

Background cover is created when many mix users generate dummies along with their real messages. Cover traffic from users other than Alice could be seen as providing cover for Alice's messages. Note that the users have a clear incentive to provide these dummies, as it helps to protect their own privacy. As we show in Section 6, this can be very effective in confusing a naive attacker. However, a slightly more sophisticated attacker can account for background cover and reduce its effectiveness. **FiXme Fatal: How do we actually model the sending of background cover?**

We now describe how the naive attacker proceeds in the presence of background cover traffic. The attacker uses Equation **??** to find $\overrightarrow{v}$, which contains an estimate of Alice's sending behavior. In each round, the attacker observes a number of messages entering and exiting the mix. He estimates the number of (i) Alice's messages exiting the mix, n_{Alice} and (ii) the number of background messages exiting the mix per round, $n_{Background}$. These estimates are calculated based on the mix's delay policy and on the number of messages seen entering the mix from Alice and from the other users. The attacker records the set of recipients who receive messages in each round in $\overrightarrow{r}$, which contains an element for every recipient in the system. $\overrightarrow{r}[i]$ contains the number of messages received by the i^{th} recipient in a particular round. $\overline{O}$ is updated each round as follows:

$$\overline{O}[i] = \frac{\overrightarrow{r}[i] * n_{Alice}}{n_{Alice} + n_{Background}}$$

When background dummies are sent, the attacker sees more messages entering the mix. The dummies get dropped inside the mix and do not exit the mix along with real messages. The attacker, however, expects the messages to exit the mix and wrongly estimates the value of $n_{Background}$. As a result the calculation of $\overline{O}$ is upset, thereby affecting the number of rounds to correctly identify Alice's recipients.

To counter background cover, the attacker can discount away a percentage of incoming messages that he knows are dummies. We assume that the background user's policies for sending dummies are known to the attacker. This can be reasonable in many systems, as only the aggregate behavior is needed. Additionally, the attacker can estimate the long-term behavior of the aggregate dummy-sending behavior by subtracting the number of real output messages from the number of input messages over a period of time in which Alice is not active. We show in Section 6 that background dummies do not help against this informed attacker.

3.2. Receiver-bound Cover Traffic

Cover traffic is most effective when users send it consistently. In reality, however, most senders are not online all the time. It is difficult for users to consistently send cover traffic, as it requires them to be online and connected to the mix network all the time. This problem is potentially alleviated when the mix carries the onus of sending cover traffic.

Receiver-bound (RB) cover consists of dummy messages generated by the mix. The dummies are inserted into outgoing user traffic in every round. The mix chooses the recipients of cover traffic uniformly and randomly from the list of recipients. $\overline{O}[i]$ contains the probability that a message received by the i^{th} recipient has originated at Alice. The attacker updates elements in $\overline{O}$ in every round according to Equation 3.1. When RB dummies are present, elements in $\overline{O}$ are updated for messages that were in fact never sent by any sender. This upsets the attackers' statistical calculations. We note that, unlike with background cover, the attacker cannot simply discount the cover traffic volume. This is because the observations used in the SDA are the volumes of traffic to each recipient; there is no information available to help the attacker discern how much of the volume is cover traffic. Thus, for the attack to be successful, the number of rounds the attacker must observe increases significantly. We discuss the practical issues in implementing this approach in Section 7.

4. An Analysis of SDA

In this section, we analyze the performance of SDA and bound the number of rounds for SDA to be successful. We compare the bounds for different scenarios in order to understand how different design choices impact the performance of SDA. We first discuss the assumptions we use in this analysis and then give an overview of our approach before presenting the bounds. We present numerical results of this analysis in Section 6.

4.1. Assumptions

Background senders send m messages per round to receivers. These messages are distributed uniformly and randomly among the R receivers. The average number of messages that a single receiver gets in any round is $\frac{m}{R}$. Receivers who do not receive from Alice are called non-Alice receivers.

Alice has a single recipient called Bob. In each round, Bob receives one message from Alice and $\frac{m}{R}$ messages on average from other senders. Thus, Bob receives on average $1 + \frac{m}{R}$ messages in rounds that Alice sends and $\frac{m}{R}$ messages in rounds that Alice does not send.

The attacker's goal is to identify Bob. The attacker makes observations for T number of rounds in order to successfully identify Bob.

4.2. Approach

The analysis proceeds by considering the number of messages that each receiver gets in each round of observation. After a sufficient number of rounds of observation T, the attacker will observe that Bob receives more messages than a typical receiver in rounds in which Alice sends. We find an upper bound on T such that the attacker observes a minimum difference between typical receivers and Bob.

We derive equations to calculate bounds on T in different scenarios, beginning with the case when there is no cover traffic. We extend the analysis to bound T from above in the presence of Alice cover and receiver-bound cover. Finally, we derive bounds for the number of rounds of observation needed by a partial adversary who observes only a fraction of the network.

4.3. Bounding the time for attacker success

Let $X = \sum_{i=1}^{T} X_i$ be the number of messages Bob receives after T rounds of observation. $X_i = 1 + \frac{m}{R}$ is the total number of messages sent to Bob from both Alice and the background senders in round i. Let μ_x be the expected value of X after T rounds, and we note that $\mu_x = T\left(1 + \frac{m}{R}\right)$.

Let $Y = \sum_{i=1}^{T} Y_i$ be the number of messages received by each receiver other than Bob, after T rounds of observation. $Y_i = \frac{m}{R}$ is the total number of messages from background senders to a typical receiver in round i. Let the expected value of Y after T rounds of observation is $\mu_y = T\left(\frac{m}{R}\right)$.

We use a Chernoff bound to find the lower bound T_{low}, of the number of rounds for Bob to receive less than a threshold B messages with probability $p_L = 0.5$. By setting $p_L = 0.5$, our analytical results are somewhat comparable with the median rounds to attacker success, which is the metric used in our simulations. It is possible, using the same framework, to bound the number of rounds until the attacker has a high confidence of success by setting p_L to a higher rate.

For the threshold B, we choose the mid-point between μ_x, the expected messages per round for Bob, and μ_y, the expected messages per round for a typical non-Alice receiver. The mid-point is an arbitrary boundary chosen for the value of B because it provides, in the long run, a point between the X and Y distributions. Although there may be better boundary points for the attacker, if the attacker can identify Bob in a given number of rounds using B as defined, he can certainly identify Bob using a better boundary.

$$B = T\left(\frac{m}{R} + \frac{1}{2}\right) \tag{2}$$

Using a Chernoff bound (Motwani and Raghavan, 1996) for $0 < \delta_x \leq 1$, we get

$$Pr[X < (1-\delta_x)\mu_x] < exp\left(\frac{\mu_x(\delta_x)^2}{2}\right) \tag{3}$$

We want the probability that $X < B$ to be $p_L = 0.5$. So,

$$exp\left(\frac{\mu_x(\delta_x)^2}{2}\right) = \frac{1}{2} \tag{4}$$

Using B as the lower bound

$$(1-\delta_x)\mu_x = B$$

From (2), $B = \frac{Tm}{R} + \frac{T}{2}$

$$(1-\delta_x)\mu_x = T\left(\frac{m}{R} + \frac{1}{2}\right)$$

$$(1-\delta_x)T\left(1 + \frac{m}{R}\right) = T\left(\frac{m}{R} + \frac{1}{2}\right)$$

$$\delta_x = \frac{R}{2(m+R)} \tag{5}$$

Substituting (5) in (4),

$$exp\left(\frac{-T_{low}(1+\frac{m}{R})\left(\frac{R}{2(m+R)}\right)^2}{2}\right) = \frac{1}{2}$$

$$T_{low} = 8\ln(2)\left(1 + \frac{m}{R}\right) \tag{6}$$

We now use a Chernoff bound to derive an upper bound T_{up} on the number of rounds of observation required for a 0.5 or lower probability that a receiver other than Bob will get more than B messages. As before, we choose a probability of 0.5 because it maps to the median rounds metric used to measure attacker success in our simulations and allows for a comparison between the analysis and simulation results.

Let p_U be the probability that a receiver other than Bob receives more than B messages. Then, the probability that at least one of the $R-1$ non-Alice receivers gets more than B messages is $p < 1-(1-p_U)^{R-1}$. We want $p < 0.5$. Hence, we choose $p_U = \frac{1}{2R}$ for deriving the upper bound T_{up}.

Applying the Chernoff bound with $\delta_x > 0$,

$$Pr[Y > (1 + \delta_y)\mu_y] < \left(\frac{e_y^{\delta}}{(1 + \delta_y)^{1+\delta_y}} \right)^{\mu_y} \tag{7}$$

We want $Pr[Y > B]$ to be $\frac{1}{2R}$. So,

$$Pr[Y > (1 + \delta_y)\mu_y] < \frac{1}{2R} \tag{8}$$

Using B as the upper bound

$$(1 + \delta_y)\mu_y = B$$

$$(1 + \delta_y)\mu_y = T\left(\frac{m}{R} + \frac{1}{2}\right)$$

$$\delta_y = \frac{R}{2m} \tag{9}$$

Substituting (9) in (8) and taking the log of 2 both sides, we get

$$\frac{mT_{up}}{R} \ln\left(\frac{e^{\delta_y}}{(1 + \delta_y)^{1+\delta_y}} \right) = \ln\left(\frac{1}{2R}\right)$$

$$T_{up} = \frac{R}{m} \ln\left(\frac{1}{2R}\right) \frac{1}{\delta_y - (1 + \delta_y)\ln(1 + \delta_y)} \tag{10}$$

4.4. Bounding the time for attacker success in the presence of Alice cover

We now study the SDA in the case that Alice uses cover traffic to increase her privacy. Let p_d be the probability that a message from Alice is a dummy message. The probability that a message from Alice is a real message is $r = 1 - p_d$. Thus, the expected number of messages that Bob receives after T rounds is $\mu_x = T\left(\frac{m}{R} + r\right)$. We find the lower bound for μ_x in the same way we did before:

$$Pr[X < (1 - \delta_x)\mu_x] < \frac{1}{2}$$

$$exp\left(\frac{-\mu_x \delta_x^2}{2}\right) = \frac{1}{2} \tag{11}$$

We set $(1-\delta_x)\mu_x = B$,

$$(1-\delta_x)\left(\frac{m}{R}+r\right) = \left(\frac{m}{R}+\frac{r}{2}\right)$$

$$\delta_x = \frac{Rr}{2(m+Rr)}$$

Substituting for δ_x in (11) we get

$$exp\left(\frac{-T_{low}\left(\frac{m}{R}+r\right)\left(\frac{Rr}{2(m+Rr)}\right)^2}{2}\right) = \frac{1}{2}$$

$$T_{low} = 8\ln(2)\frac{m+Rr}{Rr^2} \tag{12}$$

where $r = (1-p_d)$

To get the upper bound on the number of rounds of observation we proceed as before and obtain $\delta_y = \frac{Rr}{2m}$ and

$$T_{up} = \frac{R}{m}\ln\left(\frac{1}{2R}\right)\frac{1}{\delta_y-(1+\delta_y)\ln(1+\delta_y)} \tag{13}$$

4.5. Bounding the time for attacker success in the presence of RBC

In the presence of receiver-bound cover traffic, Bob receives $\frac{mV_{rbc}}{R}$ more messages than before. V_{rbc} is the volume of receiver-bound cover. So if the outgoing traffic in a round is $\frac{m}{R}$, the amount of RBC sent in that round is $V_{rbc}\frac{m}{R}$. This means $\mu_x = T\left(\frac{m}{R}(1+V_{rbc})+r\right)$. Following the derivation steps as before, we get $\delta_x = \frac{Rr}{2(m+mV_{rbc}+Rr)}$ and

$$T_{low} = 8\ln(2)\frac{m+Rr+mV_{rbc}}{Rr^2} \tag{14}$$

In the presence of RBC, other recievers also receive $\frac{mV_{rbc}}{R}$ more messages than before. This means $\mu_y = T.\frac{m}{R}(1+V_{rbc})$. Using Chernoff bounds to bound the number of rounds of observation, we get $\delta_y = \frac{Rr}{2m(1+V_{rbc})}$ and

$$T_{up} = \ln\left(\frac{1}{2R}\right)\frac{R}{m(1+V_{rbc})}\frac{1}{\delta_y-(1+\delta_y)\ln(1+\delta_y)} \tag{15}$$

4.6. Bounding the time for success of a partial attacker

In this subsection we use Chernoff bounds to derive an upper limit on the number of rounds for a partial attacker to successfully identify Bob as a contact of Alice. Since the partial adversary can observe only parts of the network, we assume that a message entering or exiting the anonymity network has a probability p_p of being observed by the attacker. p_p varies from 0.0 to 1.0 depending on how much of the network the attacker can observe; 0.0 means that he can not observe any messages to or from the mix network and 1.0 means that he is a global adversary and is able to observe every message going in and coming out of the mix network.

We proceed similarly as in Section 4.3 and include the observation capability of the attacker, p_p, into our derivation steps. The number of messages the adversary sees Bob receiving after T rounds is now $X = \sum_{i=1}^{T} X_i p_p$. The number of messages the adversary observes other receivers receiving is $Y = \sum_{i=1}^{T} Y_i p_p$. The expected values of X and Y after T rounds of observation is $E[X] = \mu_x = T\left(\frac{m}{R}+1\right)p_p$ and $E[Y] = \mu_y = T\left(\frac{m}{R}\right)p_p$. We set B to be half the distance between μ_x and μ_y, which means $B = Tp_p\left(\frac{m}{R}+\frac{1}{2}\right)$. Using Chernoff bounds and proceeding as before we get $\delta_x = \frac{R}{2(m+R)}$, which is the same as (5) and

$$T_{low} = \frac{8\ln(2)}{p_p}\left(1+\frac{m}{R}\right) \tag{16}$$

$\delta_y = \frac{R}{2m}$, which is the same as (9) and

$$T_{up} = \frac{R}{mp_p}\ln\left(\frac{1}{2R}\right)\frac{1}{\delta_y - (1+\delta_y)\ln(1+\delta_y)} \tag{17}$$

From the above bounds we note that for a partial adversary, the number of rounds required to achieve the same certainty as a full attacker increases by a factor of p_p.

4.7. Combined Equations

FiXme Fatal: **What is this subsection about?** Combining equations (6), (12), (14), and (16) we get

$$T_{low} = \frac{8\ln(2)}{p_p}\frac{m + Rr + mV_{rbc}}{Rr^2} \tag{18}$$

and combining equations (10), (13), (15), and (17) we get

$$T_{up} = \frac{R}{mp_pV_{rbc}}\ln\left(\frac{1}{2R}\right)\frac{1}{\delta_y - (1+\delta_y)\ln(1+\delta_y)} \tag{19}$$

5. Simulation

Using the basic sender-mix-receiver model described in Section 2, we simulate the process of sending messages, cover traffic, and the corresponding SDA. We first discuss the three main elements of the simulation design, which are the attacker algorithm, the generation of real traffic, and our metrics for attacker success. We then describe how we generate cover traffic.

5.1. Simulator Design

We built our simulations around the core simulator used by Mathewson and Dingledine, and we refer the reader to that paper for further detail (Mathewson and Dingledine, 2004).

5.1.1. Attacker Algorithm

A full attacker is able to see all messages from senders into the anonymity system and all messages exiting the system to receivers. A partial attacker can see part of the network and can only see some of the messages from senders to the mix system (inbound) and from the mix system to receivers (outbound). To simulate a partial attacker we use a probability $p_p = 0.5$ to decide whether the attacker sees a particular inbound message or outbound message.

The attacker algorithm is based on the statistical analysis approach *Attacking pool mixes and mix networks* described in (Mathewson and Dingledine, 2004). Beyond this, we assume that the attacker makes reasonable adjustments to the algorithm in response to changes in the system, such as adjustments to background cover described in Section 3.

5.1.2. Attacker adjustment to background cover

The attacker can estimate the average total background cover from the set of background sends. In the presence of background cover, the attacker discounts the expected background cover per round from the total number of messages sent by background senders in every round.

5.1.3. Attacker adjustment to receiver-bound cover

FiXme Fatal: check review comment: Note difference from BGC as he can't discount all messages and discounting proportionally does not help When discounting RBC the attacker cannot simply discount the number of estimated dummy messages from the total number of messages as he does in the case of background cover. In the case of RBC, the attacker must discount RBC on a per receiver basis in order to preserve the distribution of the number of messages received by each receiver in a round. By proportionally discounting dummy messages from all receivers, the attacker discounts on average the total numberof estimated dummy messages in each round.

In our simulations, we discount receiver-bound cover by applying a discount to each message coming out of the mix. If the mix's RBC volume is V_{rbc}, then the discount is $\frac{1}{1+\frac{V_{rbc}}{100}}$ per message. This means in the presence of RBC, the attacker counts a fraction of the volume of actual traffic each receiver gets in any given round.

FiXme Fatal: end check

5.1.4. Real Message Generation

Major elements in the simulated generation of real messages include:

- Background Traffic: To ensure comparability with previous empirical work, the number of messages sent by the background follows a normal distribution with mean 125 and standard deviation of 12.5. Additionally, we consider a more active set of users, with means of 1700 and 9000 messages per round. The senders follow a scale-free model in sending to recipients. We first created a scale-free network and then created a weighted recipient distribution for background senders. The weighted distribution allows background senders to send more messages to popular recipients. A uniform recipient distribution is created for Alice, which allows Alice to send uniformly to all of her recipients.

- Alice's Traffic: Alice has a recipient set of 32 recipients. In each round she sends messages to recipients chosen with uniform probability from this set. Alice generates real messages according to a geometric distribution with a distribution parameter of 0.6, which means that she sends about 1.5 real messages per round.

- Mix Behavior: We use two different mix types for our simulations. For the threshold mix simulations, the batch size is set at 125 messages/round. For simulations to compare the analysis and simulations the batch size is varied from 100 to 2000 messages per round.

  In the case of a binomial mix, the mix applies a probability P_{delay} to each message entering the mix in order to decide if the message will exit the mix in the current round or will be delayed until a later round (Díaz and Serjantov, 2003). For our simulations we varied P_{delay} from 0.1 to 0.9. For simulations where P_{delay} does not vary, we set $P_{delay} = 0.1$.

5.1.5. Measuring Attacker Success

For most of our experiments, we measure the number of rounds that the attacker takes to correctly identify ten of Alice's recipients. This is a deviation from prior work, which chose to determine when the attacker correctly identified all 32 of her recipients. The latter is, in our opinion, an unnecessarily high bar for the attacker to meet. In particular, we discovered that finding the final recipient was a particularly challenging task that took many additional rounds of communication in most experiments. Worse, the variance for obtaining this final recipient is quite high, as it may depend on just a few messages that are sent with low probability.

We propose the lower threshold of ten recipients, although arbitrary, as a point at which the attacker has identified a substantial fraction of Alice's recipients. At this point, the attacker can correctly identify not only the popular members of Alice's recipient set, but also several of the less popular members as well. The attacker may not have the full profile that he seeks, but some of Alice's privacy has been lost, as the attacker has some picture of Alice's communication patterns. Since the attack could take many rounds, a partial picture may be all that the attacker could attain in a reasonable time frame.

It should be noted that we stop all runs after one million rounds. This could equate to almost one hundred and fifteen years, at one hour per round, or nearly two years at one minute per round. If the attacker cannot identify

10 of Alice's recipients in this time, the attack is taking very long. Even if the attacker is that patient, and Alice is that consistent, we focus our attention on stopping the attacker from defeating the system in a faster time frame. When we have strong methods for doing that, longer term attacks can be considered.

5.2. Cover Traffic Scenarios

The simulations in (Mathewson and Dingledine, 2004) focus mainly on the effects of user cover traffic. In this study, we describe the effects of RB cover and background cover. We use three scenarios to evaluate the effect of cover traffic on statistical analysis.

5.2.1. Alice and Background Cover Traffic

We first study how dummy messages sent by users other than Alice affects statistical analysis. We set $N = 2^{16}$ as the number of senders. Each of the $N - 1$ other senders apart from Alice, called background senders, generate zero or more dummy messages in every round. Senders choose the number of dummies according to a geometric distribution with a parameter varying from 0.1 to 0.9. This means each sender sends between 0.11 to 9 dummy messages per round on average.

Alice also generates a number of dummy messages in each round in which she participates. Like other senders, Alice follows a geometric distribution to select the number of dummies to send per round. Alice's dummy parameter, P_{dummy}, is varied from 0.1 to 0.9. In simulations where Alice's cover traffic does not vary, we set P_{dummy} to 0.6, which is about 1.5 messages/round. The geometric distribution parameters for Alice dummies and background dummies are independent of each other. Cover traffic generated by senders is sent to the mix like real traffic. The mix can recognize real messages from dummies and drops all dummies that it receives. Hence, dummies sent from the users are dropped inside the mix network and are not propagated to any receivers.

5.2.2. Receiver-Bound Cover Traffic

We also evaluate how RB cover traffic originating at the mix impacts SDA. At the end of each round, the mix selects a subset of messages in its pool and sends them to their respective recipients. In addition to the real messages, the mix adds a number of dummy messages to the outbound stream. We run simulations with the number of receiver-bound dummy messages per round,

$V_{rbc} = 100\%$, 200%, and 300% of real traffic. The recipient of each dummy message is chosen uniformly at random from the set of recipients. Although the mixes may not know the full set, a reasonable approximation can be constructed by using previously observed recipients and a selection of receiver addresses from the general population. Dummy messages travel from the mix to the receiver and are observed as part of the outgoing traffic by the passive attacker. However, since the attacker cannot distinguish dummy messages from real messages, dummies are included in the attackers analysis. Dummy messages reach the destination nodes and are dropped by the receiver.

5.2.3. Alice and Receiver-bound Cover

In this scenario, Alice sends cover traffic to the mix along with her real messages. These messages are dropped inside the mix. The mix in turn generates dummy messages independent of Alice's dummy messages. The mix dummies are sent out with real outbound user messages.

6. Results

In this section we present the results of our simulations. Please note the use of logarithmic scales in some of our graphs. The Y-axis in all graphs is the number of rounds of observation the attacker needs to expose a subset of Alice's recipients.

6.1. Comparison of Analysis and Simulation Results

We now compare the bounds given by the analysis in Section 4 with simulations of SDA using both a threshold mix and a binomial mix. For the simulations used for comparison with the analysis, we assume that Alice has one contact called Bob. Alice may or may not participate in a given round. She sends one message per round to Bob in rounds in which she participates. The attacker observes only rounds in which Alice participates. We assume that the attacker has already observed rounds in which Alice does not participate and has an understanding of background traffic behavior. Thus, the total number of rounds of observation needed to expose Bob are rounds in which Alice participates.

The attacker is a partial adversary and can observe any message with probability $p_O = 0.5$. The mix is a threshold mix, so all messages are flushed out of the mix at the end of every round. The batch size of the mix is 125

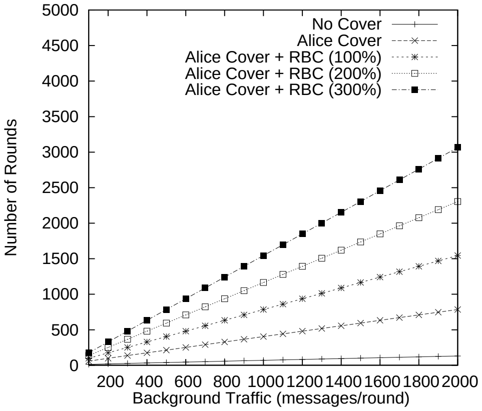

Figure 1: **Analysis:** Number of rounds for an attacker to be 50% sure that Bob received more messages than other user for increasing background traffic

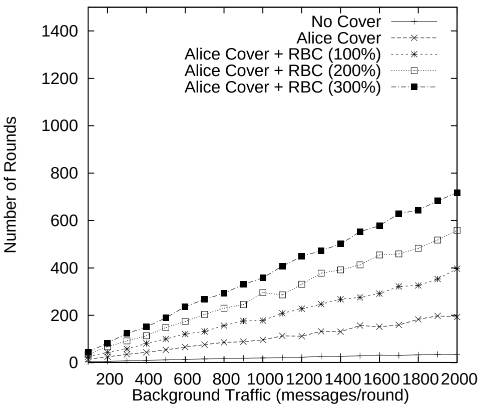

Figure 2: **Simulation:** Median rounds to identify Alice's contact Bob with increasing traffic from background senders.Threshold mix with batch size = 500 messages/round.

messages/round. In non-Alice rounds, the background traffic is 125 messages/round and in Alice rounds it is 124 messages/round.

In the simulations we measure the median rounds for attacker success. This means 50% of the time Bob is the highest receiver in Alice rounds. In the analysis, we set $p_B = \frac{1}{2}$ which means that 50% of the time, Bob has over B messages. This makes the analysis and simulation results somewhat comparable.

Figure 1 shows the result of analysis and Figure 2 shows the result of simulations. When we compare the figures we see that the simulation results closely follow the trend predicted by the analysis but the analysis results are upto 8 times higher than the simulation results.

6.2. Degree of Disclosure

It is easier for an attacker to obtain a subset of Alice's recipients than to find all of Alice's recipients. We ran simulations to evaluate how different cover traffic approaches affect the attackers ability to expose a number of Alice's recipients. The graph in Figure 7 shows that as the attacker tries to expose more number of recipients, the amount of observation rounds significantly increases. In comparison, Figure 8 shows that with more active background senders, the effectiveness of cover traffic is more pronounced. When RB cover is used, the number of rounds sharply increase when more

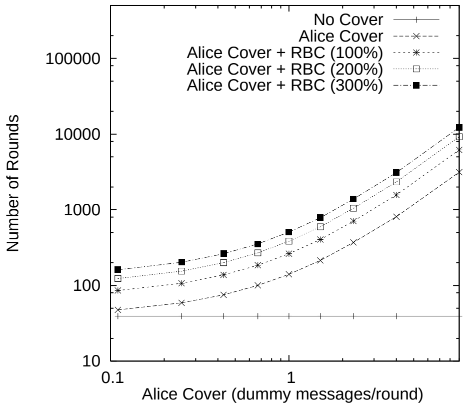

Figure 3: **Analysis:** Number of rounds for an attacker to be 50% sure that Bob received more messages than any other user with increasing Alice cover traffic

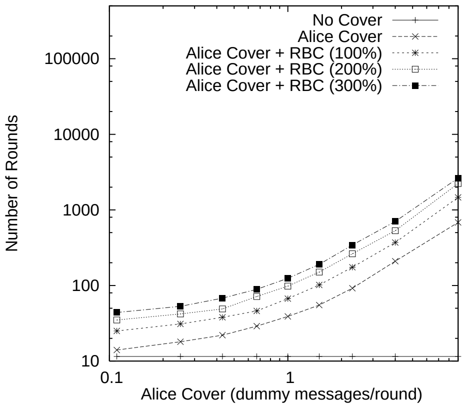

Figure 4: **Simulation:** Median rounds to identify Alice's contact Bob with increasing cover traffic from Alice. Threshold mix with batch size = 500 messages/round.

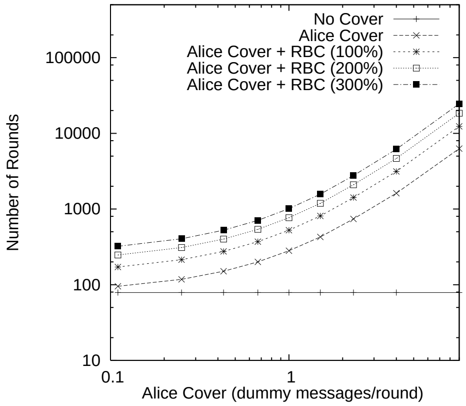

Figure 5: **Analysis:** Number of rounds for a partial attacker to be 50% sure that Bob received more messages than any other user with increasing Alice cover traffic

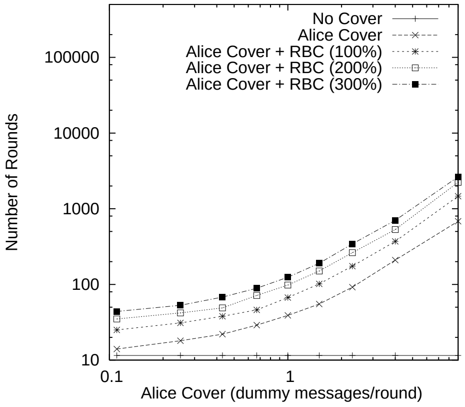

Figure 6: **Simulation:** Median rounds for a partial attacker to identify Bob with increasing cover traffic from Alice. Threshold mix with batch size = 500 messages/round.

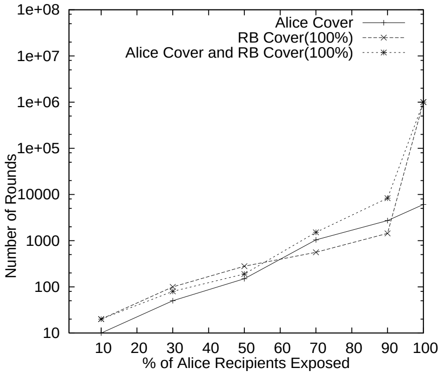

Figure 7: Median rounds to identify a subset of Alice's recipients. Background **(BG)** volume = 125 messages/round. Mix delay probability P_{delay}=0.5

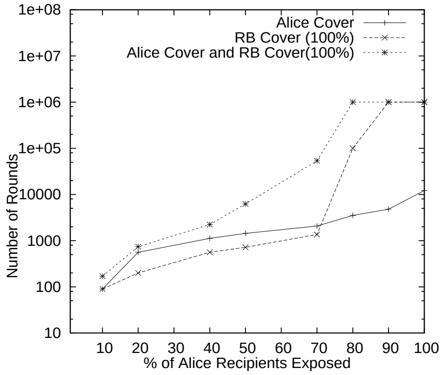

Figure 8: Median rounds to identify a subset of Alice's recipients. Background **(BG)** volume = 1700 messages/round. Mix delay probability P_{delay}=0.5

than 70% of her recipients are exposed. When only Alice sends dummies, the rise in number of rounds is more modest when compared to when RB cover is also used. In our remaining experiments, we fix the number of recipients to be exposed at 30% which we simplify to 10 recipients.

6.3. Effect of Background Senders

The graph in Figure 9 illustrates the effect of background dummy messages on the number of rounds needed to correctly identify 10 of Alice's recipients. Alice generates dummies according to a geometric distribution. Alice's dummy distribution parameter varies from 0.1 to 0.9 as seen along the x-axis. The effect of background traffic volume (BG) is clearly visible in this graph. When $BG = 125$, the effect of background and Alice dummy messages is very low. In the case when $BG = 1700$, cover traffic has a greater impact. As Alice's dummy volume increases, the number of rounds needed to identify Alice's recipients increases. Further, we see that when the background senders also send cover traffic, it becomes increasingly difficult for the attacker to successfully identify Alice's recipients. When the background senders generate cover traffic at 10% of real traffic and Alice increases her dummy distribution parameter to 0.9, it takes more than one million rounds to correctly identify ten of Alice's recipients.

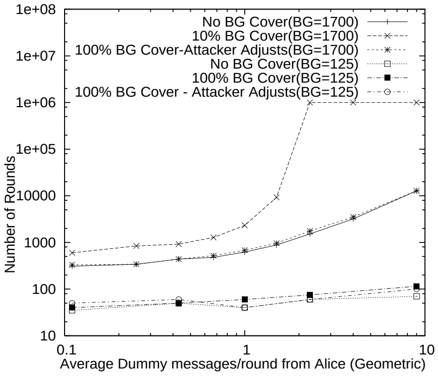

Figure 9: Effect of Background Cover and Attacker Adjustment. Median rounds to guess 10 recipients.

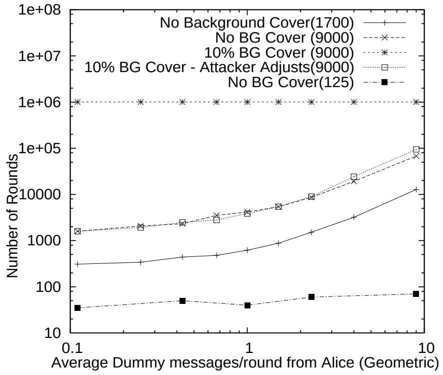

Figure 10: Effect of increase in Background Volume. Median rounds to guess 10 recipients.

6.3.1. Attacker Adjustment

The attacker can counter the effect of background cover by estimating the number of dummies that the background sends per round. The attacker can observe the number of senders sending per round and has knowledge of their dummy policy. Once the estimate is obtained, the attacker simply has to subtract the number of estimated dummies from the number of observed background messages and continue as if there were no dummies. Figure 9 shows how attacker adjustment can completely negate the effect of background cover, even if background senders use 50% or 100% dummies.

The estimation of total background dummies per round is simple if all senders use the same dummy volume parameter. If senders use arbitrary dummy volume parameters, selected independently or even randomly varied over time, it becomes more difficult for the attacker to estimate the background dummy volume. The attacker could attempt to subtract the average system output from the average system input, as this provides an average of the sum of the background dummies plus Alice's dummies. This suggests another benefit of RB cover traffic, as the attacker would have greater difficulty in measuring the background cover traffic if the number of real messages is hidden in the system output as well. To gain this benefit, a dynamic amount of background cover traffic is required, rather than the fixed percentage of real traffic that we have studied in this paper.

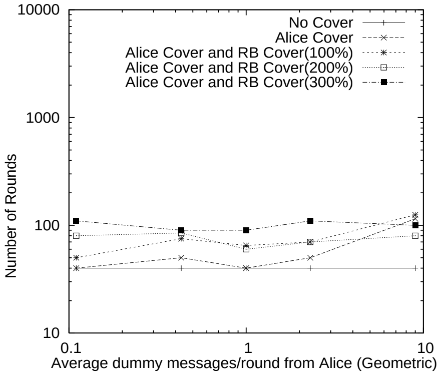

Figure 11: Effect of RB cover traffic. Median rounds to guess 10 recipients. Background **(BG)** volume = 125 messages/round.

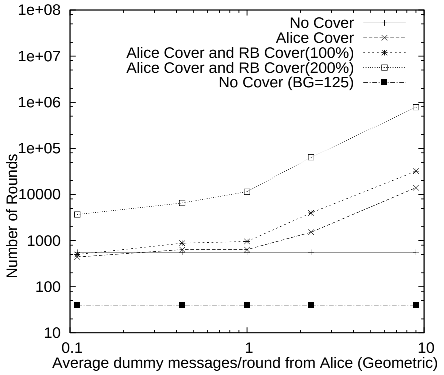

Figure 12: Effect of RB cover traffic. Median rounds to guess 10 recipients. Background **(BG)** volume = 1700 messages/round.

6.3.2. Larger Number of Participants

Figure 10 shows that as the number of participants in the mix increases, the anonymity of individual participants correspondingly increases. In this simulation we increased the volume of background traffic from a normal distribution with mean 1700 to a normal distribution with mean 9000 messages per round. As observed in the graph, the time for the attacker to expose the same number of recipients more than doubles when participants send messages more frequently.

6.4. Effect of Receiver-bound Cover

Figures 11 and 12 show the effect of RB cover traffic. The mix generates RB dummies equal to the number of real messages per round. We also studied whether the presence or absence of cover traffic from Alice would affect the number of rounds needed to identify Alice's recipients. As Figure 12 shows, cover traffic from Alice alone does not have a significant impact on number of rounds. When Alice sends dummies in the presence of RB cover the effects are more pronounced. Compared with Figure 11, we see the extent to which increasing the number of background messages helps improve the effectiveness of RB cover. When $BG = 125$, RB cover up to 300% does not significantly degrade the attack.

Figures 15 and 16 shows how the increase in delay distribution at the mix makes the attack harder. As before, there is greater benefit in increasing

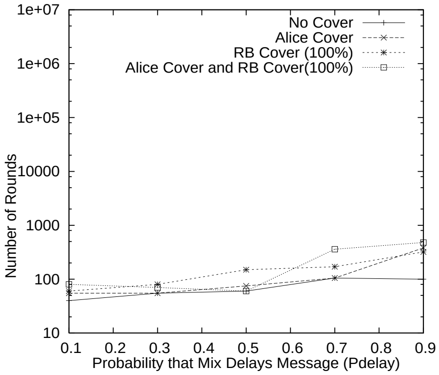

Figure 13: Effect of increased delay distribution in the mix. Median rounds to guess 10 recipients. Background (**BG**) volume = 125 messages/round.

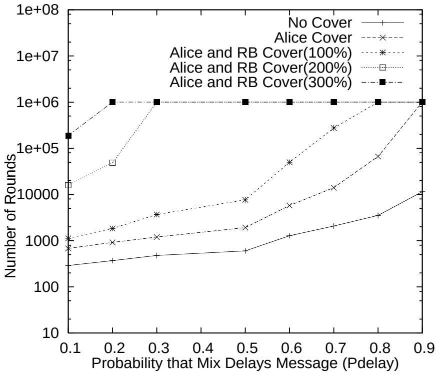

Figure 14: Effect of increased delay distribution in the mix. Median rounds to guess 10 recipients. Background (**BG**) volume = 1700 messages/round.

P_{delay} is when the background senders are more active. When the mix exhibits a delay probability higher than 0.5, the number of rounds increases more rapidly. When RB cover is increased to 200% and P_{delay} is more than 0.3, the attack takes more than one million rounds.

6.5. Partial Observation

The attack becomes slower when the adversary is a partial observer. Partial observation is a more real situation than a full observer for reasons discussed earlier. Figure **??** shows the results of a partial adversary who can observer 50% of all traffic going into and out of the anonymity system.

7. Discussion

In Section 6, we show how RB cover traffic can be used to successfully delay statistical analysis. We now touch upon the implementation aspects that RB cover should exhibit in real-world networks. There are three main considerations:

- Cover traffic must resemble real traffic in order for it to effectively anonymize user traffic.

- Receivers must tolerate the presense of dummy messages.

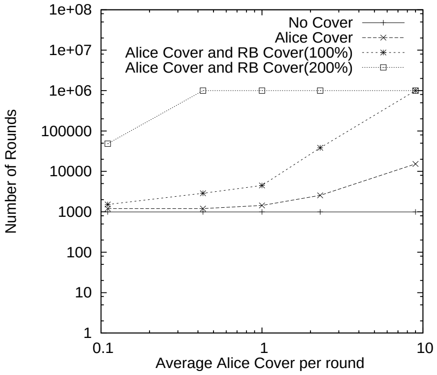

Figure 15: **Partial Observer:** Median rounds to guess 10 contacts with increasing Alice cover traffic. Background traffic volume = 1700 messages/round.

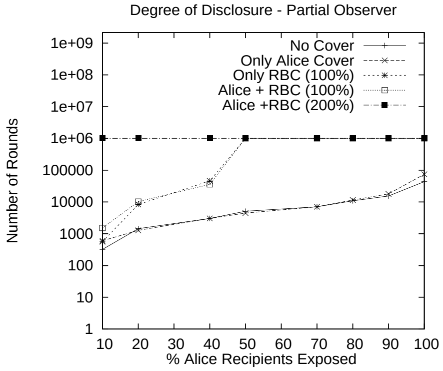

Figure 16: **Partial Observer:** Median rounds to guess increasing number of Alice contacts. Background traffic volume = 1700 messages/round.

- The costs of the cover traffic should not be too high for the mixes or the receivers.

We study these both in the context of high-latency and low-latency mixes, as intersection attacks apply to both types of system. The two forms of cover traffic that we can use are encrypted and unencrypted, each with different advantages and applications.

7.1. Encrypted Dummies

Making cover traffic that looks like real traffic is challenging. Content, timing, and receiver selection must all appear to be the same as users' messages. Realistic content is relatively easy to generate if it is encrypted. For high-latency message delivery, such as anonymous email, we can craft packets that appear to be encrypted using PGP (Zimmermann, 1995) or S-Mime (Dusse et al., 1998) but with random payload bytes (in Radix-64). The receiver could attempt to decrypt the random payload and discard the email when it doesn't decrypt properly. There is some cost to the receiver in this case, although email clients could automate this process and remove most of the cost that the receiver actually notices.

One problem with only sending dummies designed to appear encrypted is that, if some of the real messages are not encrypted, the attacker can discount the presence of those encrypted messages. The attacker takes an estimate d' of the number of RB dummies (say, d), based on knowledge of the mixes' distribution of sending those dummies. If the total number of messages is n, and the number of unencrypted real messages is u, which are both measurable, then the chance that any packet with a random payload is a real message is estimated as $p'_{real} = (n - u - d')/(n - u)$. p'_{real} becomes a discounting factor on the additions to vector $\overrightarrow{o}$ in each round. The impact of this depends on the ratio of encrypted real messages to total real messages. If the ratio is high, we may be able to increase the number of dummies to compensate. If the ratio is low, i.e. there are few real encrypted messages, the attacker can discount much of the cover traffic.

7.2. Unencrypted Dummies

As real traffic may also be unencrypted, we propose the use of unencrypted dummies for some applications. There are many applications where users often do not use encryption, including email. In such a case, the mix has to generate cover traffic that carefully replicates real traffic. Messages with randomly-generated payloads would be useless since they can be easily differentiated from real traffic.

For email, messages must be constructed that look like real messages. Messages could be replayed, but the attacker could detect this. The techniques of email spammers could be employed fruitfully here, as copying real text passages, randomization, and receiver customization could all be used to avoid detection by automated systems. Further, the word choice can be designed to match non-spam emails perfectly, as the emails do not need to sell anything. This negates many of the standard Bayesian filtering methods for detecting spam (Graham, 2002; Meyer and Whateley, 2004; Androutsopoulos et al., 2000). While attackers could use humans to determine which messages are real, and which are dummies, this would be expensive and might require knowledge about the receiver.

A useful tool to help generate realistic dummies is the behavior of real users. In email, this could mean keeping a record of messages sent to each receiver, and then using this record to help generate new messages with appropriate key words.

7.3. Making Receiver-bound Dummies Acceptable

Another critical issue in the use of RB cover traffic is their acceptance by the set of receivers. We have implicitly added some costs to receivers for the privacy of the senders, which may be classified as spam and cause the system to get unwanted negative attention. There are a number of issues and possible solutions which we touch on briefly here.

One way to cast to the problem is to note that RB cover traffic increases the anonymity of the senders connecting to the receivers. It is in the interest of anonymity for these users, so a receiver should allow anonymity networks to send cover traffic to it. Receivers who don't wish to help provide anonymous communications can block messages from the system. Some recipients block connections coming from anonymity systems like Tor (R. Dingledine, 2004) exit nodes. We could publish a 'White List' of servers that allow connections from the anonymity systems, so users can connect to those services via systems like Mixminion (Danezis et al., 2003).

Another way to see the issue is in the light of spam. Today we see that a large percentage of network traffic consists of spam messages (Weinstein, 2003). Receivers have developed a number of effective ways to drop or ignore spam messages. RB cover traffic would be a tiny addition to the millions of unwanted messages that flood the network. Further, these unwanted messages help enhance sender and receiver anonymity. Reciever-bound cover would be a small price to pay for the greater benefit of anonymity that it provides to network users. In some cases, especially in Web-browsing, the extra traffic could generally go unnoticed.

Anonymity systems have become popular over the past few years and the number of users participating these systems is continuing to grow. Currently, however, these users remain a small part of the global Internet community. The volume of traffic exiting anonymity systems is low as compared to non-anonymous traffic in the network. RB cover traffic generated to anonymize this fraction of Internet traffic would hardly burden the massive network resources that are in place.

8. Related Work

We are not the first to propose sending cover traffic to receivers. Berthold et. al have users send pre-generated dummy messages to the recipient when the sender is offline (Berthold and Langos, 2002). Mathewson and Dingledine suggest, and then dismiss, this approach in a footnote of their work on statistical disclosure (Mathewson and Dingledine, 2004). They citep problems with the user sending to all receivers, which we avoid by having the mix generate the cover traffic. Shmatikov and Wang propose cover traffic sent to receivers to prevent active and passive timing analysis attacks in low-latency mix networks (Shmatikov and Wang, 2006). In their approach, senders generate the dummies in advance and send them to the mix, which later sends them when cover traffic is needed. The authors point out that dummy packets sent on the link between the mix and recipient can be easily recognized and dropped by the recipient. Mix-generated cover traffic is also useful in protecting reverse paths from malicious clients that use the Overlier-Syverson attack. The results from Section 6 of our work indicate that this approach can also help prevent intersection attacks.

FiXme Fatal: Discuss M and D in more detail here

Mathewson and Dingledine extend SDA to pool mixes (Mathewson and Dingledine, 2004). Their work relaxes some of the assumptions made in the original work (Danezis, 2003).

System for anonymous peer-to-peer services, such as GNUnet (Bennett et al., 2002), Freenet (Clarke et al., 2001), and APFS (Scarlatta et al., 2001), include receivers in the system by their nature. Sending cover traffic to receivers would be very reasonable in such systems. P5 is an anonymity system that provides sender, receiver, and sender-receiver anonymity(Sherwood et al., 2002). P5 creates a hierarchy of broadcast channels with each level providing a different level of tradeoff between anonymity and communication performance. In P5, noise (dummy) messages are added to prevent statistical correlation of sources and sinks of a communication stream. Real messages and noise messages move from the source to the sink hop by hop across different nodes. Intermediate nodes cannot distinguish real packets from dummy packets and treat all transiting packets similarly. Furthermore, intermediate nodes are also sources and insert dummy packets into outgoing streams. Dummies are dropped at the final destination. By using these channels, each sender effectively creates a form of receiver-bound cover traffic, as each message is sent to a group of receivers. While this multicast approach would be one way to do receiver-bound cover traffic in mix-based anonymity systems, it would only work in non-encrypted communications.

9. Conclusions and Future Work

Anonymous communications remain challenging in the face of determined and powerful attackers. No matter how secure the process of mixing becomes, inconsistent usage patterns can give the attacker enough information to link users with their communication partners over time. Prior work had developed the notion of statistical disclosure as a powerful form of this attack.

In this work, we explored defenses against this attack in greater depth. We found that the cover traffic of other users is surprisingly ineffective in protecting Alice, our user of interest; techniques to hide the amount of real traffic could help. Alice's own cover traffic has a limited effect on its own, or in combination with greater delays in the mix system. We proposed receiver-bound cover traffic and showed that it can have a substantial benefit to the user. We then discussed in detail the implications of using such an approach; we believe that it is feasible, and that the improvement in privacy could well be worth the costs.

Much work remains before receiver-bound cover traffic could be put into place. First, we need to have a deeper study of the use of unencrypted receiver-bound dummies. It is unclear whether it is a pure arms race between defense and attack, or whether one side has a clear advantage. We suggest that the attacker would find that deep content analysis does not scale well, while creating realistic automated messages is a well-understood problem from spam email generation.

References

Androutsopoulos, I., Koutsias, J., Chandrinos, K., Paliouras, G., Spyropoulos, C., May 2000. An evaluation of naive bayesian anti-spam filtering. In: Proc. Workshop on Machine Learning in the New Information Age.

Barabási, A.-L., Albert, R., 1999. Emergence of scaling in random networks. Science 286, 509–512.
URL `http://www.citebase.org/abstract?id=oai:arXiv.org:cond-mat/9910332`

Bennett, K., Grothoff, C., Horozov, T., Patrascu, I., Stef, T., Mar. 2002. Gnunet – a truly anonymous networking infrastructure. In: Proc. Privacy Enhancing Technologies Workshop (PET).

Berthold, O., Langos, H., Apr. 2002. Dummy traffic against long-term intersection attacks. In: Proc. Privacy Enhancing Technologies Workshop (PET).

Chaum, D., Feb. 1981. Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. Communications of the ACM 24 (2), 84–88.

Clarke, I., Sandberg, O., Wiley, B., Hong, T. W., 2001. Freenet: A distributed anonymous information storage and retrieval system. Lecture Notes in Computer Science 2009, 46–66.

Danezis, G., May 2003. Statistical disclosure attacks: Traffic confirmation in open environments. In: Proc. Security and Privacy in the Age of Uncertainty (SEC).

Danezis, G., Dingledine, R., Mathewson, N., May 2003. Mixminion: Design of a type III anonymous remailer protocol. In: Proc. 2003 IEEE Symposium on Security and Privacy.

Díaz, C., Serjantov, A., March 2003. Generalising mixes. In: Proc. Privacy Enhancing Technologies workshop (PET).

Dusse, S., Hoffman, P., Ramsdell, B., Lundblade, L., Repka, L., 1998. S/mime version 2 message specification.

Graham, P., Aug. 2002. A plan for spam.
Available at http://www.paulgraham.com/spam.html.

Kesdogan, D., Agarwal, D., Penz, S., Oct. 2002. Limits of anonymity in open environments. In: Proc. Information Hiding, 5th International Workshop (IH).

Mathewson, N., Dingledine, R., May 2004. Practical traffic analysis: Extending and resisting statistical disclosure. In: Proc. Privacy Enhancing Technologies workshop (PET).

Meyer, T., Whateley, B., Jul. 2004. Spambayes: Effective open-source, bayesian based, email classification. In: Proc. Conference on Email and Anti-Spam (CEAS).

Motwani, R., Raghavan, P., 1996. Randomized algorithms. Vol. 28. ACM, New York, NY, USA.

Murdoch, S. J., Zieliński, P., June 2007. Sampled traffic analysis by internet-exchange-level adversaries. In: Borisov, N., Golle, P. (Eds.), Proceedings of the Seventh Workshop on Privacy Enhancing Technologies (PET 2007). Springer, Ottawa, Canada.

R. Dingledine, N. Mathewson, P. S., Aug. 2004. Tor: The next-generation onion router. In: Proc. 13th USENIX Security Symposium.

Scarlatta, V., Levine, B., Shields, C., Nov. 2001. Responder anonymity and anonymous peer-to-peer file sharing. In: Proc. IEEE Intl. Conference on Network Protocols (ICNP).

Sherwood, R., Bhattacharjee, B., Srinivasan, A., May 2002. P5: A protocol for scalable anonymous communication. In: Proc. 2002 IEEE Sym. on Security and Privacy.

Shmatikov, V., Wang, M.-H., 2006. Timing analysis in low-latency mix networks: attacks and defenses. In: Proceedings of ESORICS 2006. pp. 18–33.

Weinstein, L., 2003. Spam wars. Communications of the ACM 46 (8), 136.

Zimmermann, P. R., 1995. The official PGP user's guide. MIT Press, Cambridge, MA, USA.