Iness Ben Guirat and Claudia Diaz

Mixnet optimization methods

Abstract: We propose a method to optimally select mix network parameters for a given deployment context and adversarial model. Our method considers both worst-case and average-case anonymity and selects configurations that meet worst-case constraints while maximizing average anonymity. We apply our methods to mixnet size optimization to determine the number and width of mixnet layers, and provide results for various deployment and adversarial scenarios. For cases where the deployment context suddenly changes (drop in user traffic) we evaluate countermeasures based on mix-generated dummy traffic and show that inexpensive link dummies can significantly boost protection in some of these cases.

Keywords: mixnet, anonymity, parameter optimization

DOI 10.56553/popets-2022-0081
Received 2021-11-30; revised 2022-03-15; accepted 2022-03-16.

1 Introduction

A mix network, or *mixnet*, is an overlay network of mix nodes that routes messages anonymously from senders to receivers [6, 7, 14, 20, 30, 36, 39, 44]. Messages are encrypted by senders multiple times using, e.g., the Sphinx packet format [15], and then routed through a sequence of mix nodes. Each of the mix nodes decrypts, pads and re-randomizes messages to make its output messages *cryptographically unlinkable* to its input messages. Mix nodes also retain messages for a randomized amount of time to alter their flow and make node inputs and outputs *probabilistically unlinkable* with respect to message order and timing.

Even though the concept of mixnets [6] predates onion routing [25, 27, 28] by more than a decade, and early mixnet deployments [9, 39] were operative before Tor,[1] their uptake has remained far behind for years, mainly due to their higher computational requirements, added latency, and lack of industrial-quality implementations. In recent years however, given a renewed interest in anonymity systems that resist global network adversaries, novel mixnet-based anonymity network designs have been proposed [30, 34, 44] and currently Nym[2] is developing a mixnet-based anonymity network [20], already deployed as a testnet prototype that counts thousands of nodes.[3]

Traffic analysis is a collection of statistical methods to make inferences from available metadata, in particular: a data transmission's source, destination, size and timing [13]. The traffic analysis of mixnets yields probabilistic information that describes the likelihood of input messages corresponding to outputs [11, 19, 31, 44, 46, 47, 50–52]. This likelihood is greatly affected by the mixnet parameters, with some configurations providing significantly better protection from traffic analysis than others. Notably, this includes the network topology parameters, which describe the network size, how nodes are connected, and how multi-hop routes are selected [3, 49]. A suboptimal design can provide very poor privacy – or even no protection at all: while the *anonymity trilemma* [18] is informative of the *theoretical upper bound* on the anonymity that an abstract system can offer given conditions of traffic volume and end-to-end latency, the *practical lower bound* for those very same traffic and latency conditions is actually *zero* anonymity if the network is inadequately parametrized, e.g., if it is grossly oversized.

Given an expected volume of user traffic, constraints on end-to-end latency, and a threat model of concern, we currently lack methods to optimally select mixnet parameters, e.g., to decide how many nodes the mixnet should have and how they should be arranged to maximize protection from traffic analysis. The observation that a limited network size is desirable so that traffic density per mix node is sufficient for the mixing to be effective has been made in prior work [10, 50]. We are however the first to propose a methodology to systematically select mixnet parameter values given deployment and adversarial constraints, which results in configurations that respect worst-case anonymity bounds while maximizing average anonymity.

Iness Ben Guirat: imec-COSIC KU Leuven, E-mail: iness.benguirat@esat.kuleuven.be
Claudia Diaz: imec-COSIC KU Leuven and Nym Technologies SA, E-mail: claudia.diaz@esat.kuleuven.be

1 https://www.torproject.org

2 https://nymtech.net

3 https://github.com/nymtech/

Our methods apply to a class of mixnets broadly defined by continuous-time mixes [31] arranged in a layered network topology [20, 44], considering adversaries that observe all network connections in addition to controlling a subset of mixes. We specify and justify our system model in Section 2, where we also describe the considered adversarial capabilities.

We consider two anonymity metrics: (i) *worst-case anonymity* (expressed as the probability of selecting a fully compromised route), and *average anonymity* (given by the entropy of the probability distribution that relates a target input to the mixnet outputs [24, 48]). In Section 3 we provide analytical methods to compute worst-case anonymity and empirical methods to compute average anonymity. Given these two metrics, we propose a mixnet parametrization methodology in Section 4 that maximizes average anonymity while respecting worst-case anonymity constraints.

We discuss our experimental setup in Section 5. We first show that variability in network propagation delays and multi-core message processing can help prevent message tracing in practice when the per-mix added latency is very small (these effects become negligible as per-mix added latency grows larger). We then show that uniform node selection per layer offers better anonymity than biased capacity-based selection, which allows modest-budget adversaries to arbitrarily increase the fraction of fully compromised routes.

Section 6 presents our optimization results. We first apply our methods to optimizing the number of mixnet layers and show results for various end-to-end latency constraints, considering different adversarial capabilities and worst-case anonymity thresholds. We then apply the method to the mixnet width, again considering various adversarial models. We finally consider scenarios where a network optimized for a certain level of user traffic suffers a large and sudden drop in traffic volume. We study two countermeasures based on mix-generated dummy traffic. We find that link-based dummies are a cheap yet effective strategy to support anonymity levels in scenarios with moderate mixnet compromise. We expand on all these results in the Appendix, where we consider scenarios with higher levels of adversarial compromise. Finally, we review prior work on mixnet optimization in Section 7 and offer our conclusions in Section 8. We present complementary content in the appendices: Appendix A provides a table with the summary of notation; in Appendix B we provide an analysis analogous to the one introduced in Section 3 for networks with imbalanced layers; and finally in Appendix C we provide empirical results for networks with high rates of corruption.

2 System and threat model

2.1 System model

There are two basic types of entities in a mixnet: *end users* who anonymously send and receive messages, and *mix nodes* that act as intermediaries, routing messages between senders and receivers. We model the user population as sending messages with a rate that follows a Poisson process, considering high and low traffic load scenarios. We consider *source-routed decryption* mixnets of *continuous-time mixes*, with *fully connected layered* network topologies, and three strategies for *dummy traffic*. The rest of this section explains and justifies these choices.

2.1.1 Source routing

We consider *decryption mixnets* that are source routed, i.e., where the sender of a message selects the route through the network until it reaches the receiver. Preparing a message for sending requires encrypting it with public key material of the mix nodes selected by the sender as intermediaries in the route. The encryption is done in reverse order: starting with the recipient, adding a layer of encryption for each predecessor in the route, ending with the first mix node that receives the message directly from the sender. Upon receiving a message, mix nodes use their private keys to strip a layer of encryption and discover the next hop in the route. After a randomized delay, the message is forwarded to the next hop, which is either another intermediary mix node or the end receiver. Sphinx [15] is the best known cryptographic packet format for source-routed mixnet messages [20, 44, 45].

An alternative to decryption mixnets is *re-encryption mixnets*, which are typically cascades where batches of encrypted messages are re-randomized and provably shuffled multiple times before being threshold-decrypted [4, 29, 32, 37]. Such mixnets are specially tailored to voting applications as use cases that have limited and predictable traffic volume, very high latency tolerance, and strict public verifiability requirements. The anonymity provided by such re-encryption cascades is essentially proportional to the size of the batch where

a message is mixed, considering the number of voting choices and distribution of votes (e.g., if *all* voters vote for the same candidate, then there is no voting privacy for anyone as everyone's voting choice is revealed by the tally). Given their limited range of application and straightforward anonymity tradeoffs (simply dependent on batch size), we consider re-encryption mixnets as out of scope in this paper, which focuses on the optimization of decryption mixnets for scalable, general-purpose message-based communications [20].

2.1.2 Topology

The topology of a mixnet defines how mix nodes are inter-connected and which routes (sequences of mixes) messages can follow. The earliest mixnet proposals considered **mix cascades**, where a batch of messages goes through a fixed sequence of mixes [6, 42]. Cascades have however two main drawbacks: scalability and fault tolerance. A single server has a performance limit, and thus parallel cascades must be created in order to serve more users. As parallel cascades are disjoint, they do not combine all users in one large anonymity set, failing to take advantage of user growth to offer better anonymity [10, 21]. This makes cascades rather uninteresting for anonymity optimization. Furthermore, the failure of a single node invalidates the whole cascade, making cascades very vulnerable to server failures compared to other topologies [3, 26].

The other traditional anonymity network topology is **free routes** [9, 25, 35], where nodes form a fully connected graph and any random walk in the path (up to a maximum path length) is a valid message route. The evaluation of anonymity in free route networks requires complex and inefficient analysis methods, even for simple threshold-mix based mixnet designs [51]. Moreover, free route networks have been shown to offer worse anonymity than *layered* (or *stratified*) topologies when compared in the same conditions [21, 50].

In **layered** topologies mixes are arranged in a number of layers where each mix, at any given time, is assigned to exactly one layer. The layers are interconnected such that each mix in layer i receives messages from mixes in layer $i-1$ and sends messages to mixes in layer $i+1$, as shown in Figure 1. Mixes in the first layer receive messages from senders, while those in the last layer send messages to end recipients. The path length of message routes is fixed and determined by the number of layers. Valid message routes traverse a mix of each layer in the correct order. Layers can be *fully connected*, meaning that all nodes of a layer are connected to all predecessors in the previous layer and all successors in the next layer, or *restricted*, i.e. subject to constraints where nodes connect to a subset of predecessors and successors rather than all of them. Prior work has found no significant difference between the anonymity provided by fully connected and restricted layers [21].

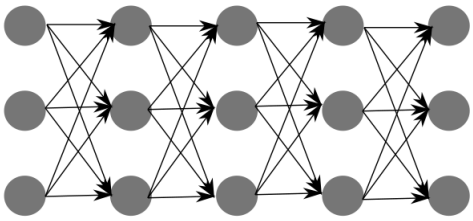


Fig. 1. Layered mixnet with $N = 15$, $L = 5$, $W = 3$.

In this paper we focus on *layered networks with fully connected layers* of the same size. We assume the topology is periodically reshuffled to allow for churn and adjust to changes in network scale. We also assume the assignment is neither predictable nor biasable by an adversary, who cannot influence the placement of malicious nodes in the layers. This can be achieved for example by relying on a public random beacon and a verifiable random function as proposed in [20]. We say a layered network has *balanced layers* if all L layers have the same number of nodes or *width* W, with the total number of nodes being $N = LW$. We choose networks with balanced layers as our baseline for their better load balancing properties. For completeness, we include results for *imbalanced layers*, where some layers may have more nodes than others, in Appendix B.

2.1.3 Mixing

In our model we consider continuous-time mixes with exponential delay [31] as they are known to offer excellent anonymity properties [12] and also allow fine-tuning the added latency per mix node to offer predictable end-to-end latency [20, 44]. This is in contrast to threshold and pool mixes [6, 36] where latency varies with the traffic volume per mix (the more traffic, the less latency), making them impractical for use cases that require latency to be within certain bounds [22]. The per-mix delays are sampled by the message sender and encoded in the Sphinx headers. Upon receiving and decrypting a message, a mix extracts the delay from the header, keeps the message in its internal memory for that amount of time, and then forwards it to its next destination.

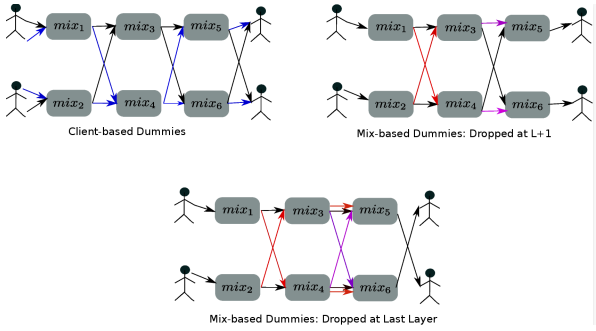


Fig. 2. Types of Dummy Traffic

2.1.4 Dummy traffic

Dummy or *cover* traffic are automatically generated messages introduced for privacy purposes. Dummy messages contain no data payload and are discarded by their final recipient. If the dummy messages follow multi-hop paths, they are considered indistinguishable from actual messages at intermediate hops as well as towards the underlying network, i.e., only the source and destination of a dummy message know that it is a dummy message. Intuitively dummy traffic enables unobservability properties [41], meaning that it is not possible to tell whether a user is idle or actively communicating. In addition, by virtue of increasing the mixnet traffic dummies also contribute to higher anonymity for actual messages.

In this paper we consider three types of cover traffic, illustrated in Figure 2. First, *user-generated dummies* destined to themselves (*loops*, as in Loopix [44]) or to others. We note that for all network purposes this traffic is equivalent to real user traffic. Second, *link-based dummies*, this type of dummy traffic is generated by mixes and it is dropped at the next hop. Third, *partial-route dummies* are also generated by the mixes in all but the last layer, and dropped at the last mixnet layer. This dummy traffic causes a linear increase of traffic load transiting the network after each layer. Other dummy strategies are possible, e.g., dummies can be generated by mixes and dropped by end users [38]. We however consider this impractical as it requires mixes to maintain a list of end user keys and addresses.

2.1.5 User traffic

We consider a user population $\mathcal{U}$ that, as a whole, generates messages following a Poisson process with parameter $\lambda_{\mathcal{U}}$ messages per second, i.e., messages arrive to the mixnet at intervals that follow an exponential distribution of mean $\frac{1}{\lambda_{\mathcal{U}}}$ seconds. We note that only messages generated by honest users (not controlled by the adversary) are relevant to anonymity, and thus $\mathcal{U}$ and $\lambda_{\mathcal{U}}$ exclude malicious users. Furthermore, users may generate end-to-end dummy traffic destined to themselves or to other users. Since this end-to-end traffic follows the same mixnet routes as real traffic and it is fully indistinguishable, we consider that $\lambda_{\mathcal{U}}$ accounts for *all* honest-user-generated traffic, whether real or dummy. Internet traffic varies per day and per hour of the day and large variations in the amount of user traffic arriving to a mixnet are possible. We consider two scenarios: (i) *High volume* user traffic ($\lambda_{\mathcal{U}} = 5000$ messages per second) and (ii) *Low Volume* user traffic ($\lambda_{\mathcal{U}} = 100$ messages per second). We note that $\lambda_{\mathcal{U}}$ is an external deployment constraint (not a parameter chosen by the mixnet designer), and our methodology can be applied to any concrete value of $\lambda_{\mathcal{U}}$.

2.2 Threat model

We consider **global passive network adversaries** that have a *global view* on the network, meaning that they can observe all network links and take into account all messages sent between any two participants (end users or mixes) with their timing information. We assume messages have the same size, and thus only timing information is exploitable to correlate a node's inputs to outputs. If the adversary in addition controls a set of **malicious users**, any messages generated by those users are excluded from $\lambda_{\mathcal{U}}$.

In addition, the adversary controls a subset of mix nodes. Mix nodes controlled by the adversary provide no anonymity to the messages they route, as the adversary knows the mapping between the inputs and outputs of malicious nodes. This is contrast to honest mixes, for which the adversary can only obtain probabilistic information linking their inputs and outputs based on message arrival and departure observations [12, 31]. We consider two adversaries of interest: the *constant fraction adversary* and the *constant budget adversary*.

The **constant fraction adversary** controls a subset of B mix nodes that is a constant fraction b of the total number of nodes N, i.e., $B = bN$, while $A = N - B$ denotes the number of honest nodes. When considering this adversary, the number B of malicious mixes grows proportionally to network size N. The **constant budget adversary** on the other hand controls a constant number B of malicious mix nodes that does not change when the network grows, with adversarial nodes there-

fore becoming a smaller fraction of the total network as it scales up. This adversary is of interest for systems such as the Nym network [20], where competition among mix nodes for finite resources (representing node reputation) may impose a practical cap on the number of new nodes the adversary is able to introduce when the network grows.

3 Anonymity metrics

We evaluate anonymity in mixnets using two metrics: (i) **fraction of fully compromised routes** [1, 5], and (ii) **entropy** [24, 48]. The metrics express different aspects of anonymity, with the first focusing on worst-case scenarios and the second on average-case scenarios. Table 1 in Appendix A provides a summary of the notation used for the various relevant parameters.

3.1 Fraction of fully compromised routes

The fraction $\alpha_{\mathcal{F}}$ of fully compromised routes focuses on *worst-case scenarios*, i.e. messages for which all anonymity is lost as the adversary can determine *with certainty* the <sender, time, receiver> of the message. This happens when the message passes through a fully compromised route, meaning that at every mixnet layer, the node in the message's route is adversarial. Note that the inverse $\frac{1}{\alpha_{\mathcal{F}}}$ expresses the *average number of messages that need to be sent to choose one fully compromised route.*

We analytically compute the expected fraction $\alpha_{\mathcal{F}}$ of fully compromised message routes in a mixnet with L layers of width W and $N = LW$ mix nodes, of which A nodes are honest and B nodes are adversarial ($A + B = N$). We consider networks of equal-capacity mix nodes where the topology is periodically reshuffled, so that the adversary cannot choose where malicious nodes are placed (in which layer). Let $\mathcal{F}$ denote the event of a fully compromised route. We compute $\alpha_{\mathcal{F}}$ as a weighted average of the fraction of compromised routes over all possible valid topologies $\mathcal{T}_v$:

$$\alpha_{\mathcal{F}} = \sum_{\mathcal{T}_v} \Pr(\mathcal{F}|\mathcal{T}_v) \Pr(\mathcal{T}_v) \tag{1}$$

A valid topology $\mathcal{T}_v = (\mathcal{A}, \mathcal{B})$ is defined by the number of honest and malicious nodes present in each layer, $\mathcal{A} = \{a_1, a_2, .., a_L\}$ and $\mathcal{B} = \{b_1, b_2, .., b_L\}$ such that it meets the following constraints:

$$\forall i \quad 0 \leq a_i \leq A, \quad 0 \leq b_i \leq B \tag{2}$$

$$\sum_{i=1}^{L} a_i = A, \qquad \sum_{i=1}^{L} b_i = B \tag{3}$$

$$\forall i \quad a_i + b_i = W \tag{4}$$

$$N = A + B = LW \tag{5}$$

$\Pr(\mathcal{T}_v)$ expresses the likelihood of occurrence of a certain valid topology $\mathcal{T}_v$, and given $\mathcal{T}_v$, $\Pr(\mathcal{F}|\mathcal{T}_v)$ expresses the probability of choosing a fully compromised route in that topology. In layered networks this means choosing a malicious mix node at *every* layer. The choice of nodes in a message's route is made uniformly at random and independently per layer, and thus $\Pr(\mathcal{F}|\mathcal{T}_v)$ corresponds to the product of the fraction of compromised nodes in each layer:

$$\Pr(\mathcal{F}|\mathcal{T}_v) = \prod_{i=1}^{L} \frac{b_i}{a_i + b_i} \tag{6}$$

Note that the adversary does not compromise any full route if he fails to populate one of the mixnet layers, i.e. if $b_i = 0$ for any layer i. At the other end of the spectrum, the optimal topology $\mathcal{T}_{opt}$ for the adversary (with highest fraction of compromised routes) is when adversarial nodes are equally distributed across layers, i.e. when $b_i = \frac{B}{L}$, $i = 1..L$. In this adversarial best case, the fraction of fully compromised routes is:

$$\Pr(\mathcal{F}|\mathcal{T}_{opt}) = \prod_{i=1}^{L} \frac{\frac{B}{L}}{W} = (\frac{B}{N})^L \tag{7}$$

To compute the likelihood $\Pr(\mathcal{T}_v)$ of a valid topology $\mathcal{T}_v = (\mathcal{A}, \mathcal{B})$, we note that in networks with balanced layers $\mathcal{A}$ and $\mathcal{B}$ are not independent. The mapping of honest nodes $\mathcal{A} = \{a_1, a_2, .., a_L\}$ is deterministic with respect to $\mathcal{B}$ as $a_i = W - b_i$, and thus $\Pr(\mathcal{B})$ fully determines the likelihood of a topology $\Pr(\mathcal{T}_v)$ (the inverse is equivalent: fixing $\mathcal{A}$ fully determines $\mathcal{B}$ as $b_i = W - a_i$).

$\Pr(\mathcal{B})$ is modeled by a hypergeometric distribution that initially has a population of size N, with B objects of interest, and W draws without replacement. The number b_1 of malicious nodes selected for the the first layer is given by:

$$\Pr(b_1) = \frac{\binom{B}{b_1}\binom{N-B}{W-b_1}}{\binom{N}{W}} \tag{8}$$

The number b_j of malicious nodes in the subsequent layers $j = 2 \dots L-1$ is given by a hypergeometric distribution with updated parameters to account for the (honest and malicious) nodes already assigned to the previous layers:

$$N_j = N - W(j-1) \tag{9}$$

$$B_j = B - \sum_{k=1}^{j-1} b_k \tag{10}$$

$$\Pr(b_j | b_1, ..b_{j-1}) = \frac{\binom{B_j}{b_j}\binom{N_j - B_j}{W - b_j}}{\binom{N_j}{W}} \tag{11}$$

The last layer is deterministically composed by the leftover nodes:

$$b_L = B - \sum_{k=1}^{L-1} b_k \tag{12}$$

Thus, the probability of a valid topology $\mathcal{T}_v = (\mathcal{A}, \mathcal{B})$ with an assignment of nodes to layers $\mathcal{B} = \{b_1, b_2, .., b_L\}$ and $\mathcal{A} = \{a_1, a_2, .., a_L\}$ with $a_i = W - b_i \ \forall i$ is given by:

$$\Pr(\mathcal{T}_v) = \Pr(\mathcal{B}) = \prod_{j=1}^{L-1} \frac{\binom{B_j}{b_j}\binom{N_j - B_j}{W - b_j}}{\binom{N_j}{W}} \tag{13}$$

Putting everything together, we obtain:

$$\alpha_{\mathcal{F}} = \sum_{\mathcal{B}} \left(\frac{b_i}{W}\right)^L \prod_{j=1}^{L-1} \frac{\binom{B_j}{b_j}\binom{N_j - B_j}{W - b_j}}{\binom{N_j}{W}} \tag{14}$$

Figure 3 shows $\alpha_{\mathcal{F}}$ in networks of a hundred nodes organized in two to five layers, considering 10% to 30% adversarial nodes. We depict with stars the value given by $\Pr(\mathcal{F}|\mathcal{T}_{opt}) = (\frac{B}{N})^L$ and find that it is a close approximation of $\alpha_{\mathcal{F}}$ due to the small variance of the distribution (by the law of big numbers, the variance of $\alpha_{\mathcal{F}}$ becomes smaller as the network size grows). Given that $\frac{B}{N} < 1$, increasing the number L of layers exponentially decreases the fraction of compromised routes, e.g., in a network where the adversary controls 10% of the nodes, 1% of messages are compromised with 2 layers, one in a thousand messages with 3 layers, one in ten thousand with 4 layers, and so on. Combined with the message sending rate of users, $\alpha_{\mathcal{F}}$ determines the de-anonymisation of messages over time. For example, if $\alpha_{\mathcal{F}} = 0.001$ and $\lambda_u = 5$ messages per second for a user u, it will take on average $\frac{1}{\alpha_{\mathcal{F}} \cdot \lambda_u} = 200$ seconds for one of u's messages to be routed via a fully compromised route.

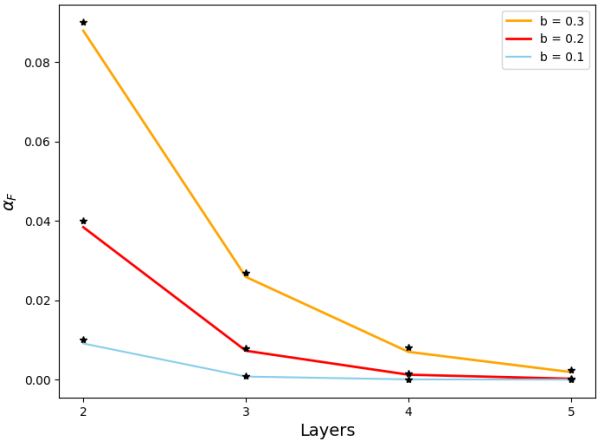


Fig. 3. Fraction $\alpha_{\mathcal{F}}$ for different values of L and B in a network of a hundred nodes.

3.2 Entropy

Instead of a worst-case metric, entropy provides an *average measure* of the number of candidate messages that the adversary confuses with a target message [24, 48]. Entropy metrics capture network scaling as their maximum possible value grows with the number of users. An entropy of, e.g., 10 bits, indicates that a message is as anonymous as if it was perfectly indistinguishable among about a thousand ($2^{10} = 1024$) other messages, while 11 bits correspond to perfect indistinguishability among $2^{11} = 2048$ messages. Note that the scale is logarithmic, and that an increase of one bit of entropy *doubles* the size of the equivalent perfect indistinguishability set, while a drop of one bit *halves* it.

Compared to worst-case metrics that only account for rates of fully compromised routes, entropy metrics account for anonymity in all possible scenarios, weighted by their likelihood of occurrence. For example, from a worst-case perspective it does not matter whether the adversary can guess the sender of an anonymous message with probability 1% or 99.9%; it only matters whether the adversary can *fully determine* the sender (100% certainty), or not. Entropy metrics are not as blunt as this binary determination and instead consider that messages can be *more* or *less* anonymous depending on the uncertainty of the adversary about the real sender. Thus, with entropy metrics a message for which the adversary is *almost* certain of the sender is considered very similarly to a message for which the adversary is *completely* certain of the sender – in contrast to worst case metrics where the *'almost'* case is considered adversarial failure and the *'completely'* case adversarial success. Furthermore, entropy metrics account for

the probabilistic information obtained by network adversaries in addition to corrupt adversarial nodes, while the worst-case metric is only dependent on adversarial nodes and disregards probabilistic inferences made by network adversaries (because nothing short of full route compromise is relevant to the worst case).

Computing entropy metrics requires obtaining the probability distribution that links a target input message to all possible output messages, or conversely one target output to all possible inputs. Given the complexity of mixnets, obtaining the relevant distributions cannot be done in a closed analytical form. In line with prior work [21, 22, 44, 50], we resort to using a discrete-event mixnet simulator [2] that given an experimental setup generates message traces, defines a subset of the traces as adversarial observations, and computes anonymity given those observations.

We consider a user population that generates messages following a Poisson process with rate $\lambda_{\mathcal{U}}$ messages per second sent to the mixnet. Messages are routed through the mixnet until they reach their destination, and in the process they leave traces that are used for anonymity evaluation. The simulation environment allows the adversary to choose a target message m_t and compute the probability $0 \leq \Pr_L[m_i = m_t] \leq 1$ linking that target input to all possible outputs m_i after the last mixnet layer L, as illustrated in Figure 4.

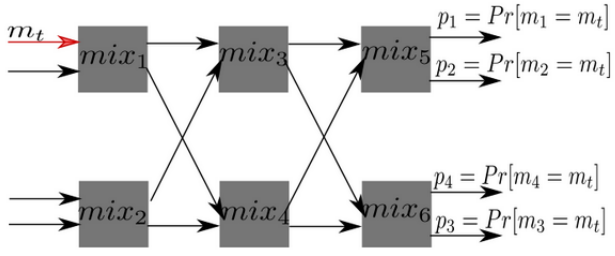


Fig. 4. Probability distribution $\Pr_L[m_i = m_t]$ for a target input m_t and all output messages m_i.

For each target input m_t, we are interested in the probability that each output message m_i may correspond to that target. We do so by associating a probability $\Pr_l[m_i = m_t]$ to each message at layer l. Message probabilities are computed iteratively per layer and updated each time a message enters and leaves a non-adversarial mix, as described in Algorithm 1. Messages that go through an adversarial mix do not alter their associated probability, i.e., $\Pr_l[m_i = m_t] = \Pr_{l-1}[m_i = m_t]$ if the mix at layer l is adversarial.

Algorithm 1: Per-mix entropy update for mix at layer $l = 1 \dots L$

Result: Updated $\Pr_l[m_i = m_t]$.
Initialize:
$\Pr_{Mix}[m_t] = 0$;
$\Pr_0[m_i = m_t] = 1$ if $m_i = m_t$;
$\Pr_0[m_i = m_t] = 0$ if $m_i \neq m_t$;
while *Simulation running* **do**
  if *event(receive(m_i))* **then**
    $\Pr_{Mix}[m_t] += \Pr_{l-1}[m_i = m_t]$;
    $pool += 1$;
  end
  if *event(send(m_i))* **then**
    $\Pr_l[m_i = m_t] = \frac{Pr_{Mix}[m_t]}{pool}$;
    $\Pr_{Mix}[m_t] -= \Pr_l[m_i = m_t]$;
    $pool -= 1$;
    Forward Message (m_i)
  end
end

Before entering the first layer, $\Pr_0[m_i = m_t]$ is one for the target input m_t and zero for the rest of the input messages sent by users. $\Pr_{Mix}[m_t]$ denotes the probability that the target is one of the messages in the current internal memory (pool) of the mix and its initial value (before receiving messages that may be the target) is zero. The *pool* variable simply denotes the number of messages that are currently inside a mix, waiting to be forwarded.

When a message m_i is received by a mix in layer l, its associated probability $\Pr_{l-1}[m_i = m_t]$ is added to $\Pr_{Mix}[m_t]$ to account for the increased probability of m_t being now in that mix. When a message m_i leaves the mix, its updated $\Pr_l[m_i = m_t]$ is a fraction of $\Pr_{Mix}[m_t]$, which is evenly divided by the number of messages currently in the mix's internal pool. This is because in continuous-time mixes with exponential delays all of the messages inside a mix are equally likely to be sent next, and thus the probability that a message in the pool is the target, is uniformly distributed across all messages in the mix at any given time [12]. The outputs of a mix in layer l have an associated updated probability $\Pr_l[m_i = m_t]$ that is the input probability of m_i for the receiving mix in the next layer $l + 1$. If the message is being delivered to its final recipient, the probability $\Pr_L[m_i = m_t]$ is the value needed for the entropy calculation. The anonymity of the target message

is computed considering the probabilities associated to the mixnet outputs m_i sent to final recipients, as:

$$H = -\sum_i Pr_L[m_i = m_t] * \log_2(Pr_L[m_i = m_t]) \quad (15)$$

We generate message traces using the open source MiXiM discrete event simulator [2] (run on an Intel(R) Core(TM) i9-9920X with 3.50GHz CPU and 132 GB RAM) and compute entropy for hundreds of targets m_t. We treat each target as an anonymity sample and then show average values or full distributions (as boxplots). The number of targets depends on the scenario: for pure network adversaries ($b = 0$) we choose 200 targets, while for adversaries that corrupt a fraction of the mixnet ($b > 0$) we increase to 1000 targets. This is because scenarios with corrupted nodes have outliers for messages going through corrupt nodes that need to be properly sampled.

4 Methodology

Our proposed method for optimizing mixnet design parameters consists of three main steps. First, we set the variables that define the considered adversary (selection of b corrupted fraction or B corrupted nodes) and the deployment scenario (average end-to-end latency D_{e2e} and traffic volume $\lambda_{\mathcal{U}}$). These variables represent external constraints that the system design needs to optimize for. Second, we set a threshold β ($0 < \beta < 1$) that defines the maximum tolerable fraction of compromised routes. Note that $\frac{1}{\beta}$ represents the average number of messages that need to be sent to have one fully compromised route: if we set the worst case threshold at 'one in a thousand' messages then $\beta = 0.001$, while lowering the tolerance to 'one in a million' messages corresponds to $\beta = 10^{-6}$. Given the range of values for a mixnet design parameter we want to optimize, we compute $\alpha_{\mathcal{F}}$ for each of the values. We then discard parameter values that result in $\alpha_{\mathcal{F}} > \beta$, while keeping those that result in $\alpha_{\mathcal{F}} \leq \beta$ as candidates for the next step. The third and final step computes entropy-based anonymity, in order to find the value that maximizes *average anonymity* in addition to satisfying *worst-case anonymity* constraints.

5 Experimental setup

5.1 Baseline parameters

As part of the first step of the methodology previously outlined, we select baseline values for the adversary and deployment models as follows:

- As baseline, we consider scenarios where adversaries control 10% of the nodes, i.e., where $b = 0.1$. In specific experiments, we also consider scenarios where adversaries do not control any nodes ($b = 0$), scenarios with larger fraction of corrupted nodes ($b = 0.2$ and $b = 0.3$), and scenarios with a constant number of corrupted nodes ($B = 9$, $B = 15$ and $B = 30$).
- In terms of end-to-end latency, we consider as baseline that $D_{e2e} = 1$ second. In specific experiments however we also consider end-to-end latencies of 2, 5, 0.5, and 0.25 seconds.
- In terms of traffic load, the baseline scenario considers $\lambda_{\mathcal{U}} = 5000$ messages per second. When evaluating dummy traffic we also study scenarios where the traffic suddenly drops to just $\lambda_{\mathcal{U}} = 100$ messages per second.

In terms of parameter choices, we argue that $D_{e2e} = 1$ second is a tolerable end-to-end average latency for, e.g., broadcasting transactions to be included in a blockchain or for email applications. In terms of volume, Mastercard processes 5000 transactions per second,[4] which gives a sense of the volume that could be expected in a broadly used payment application if transactions would be routed via a mixnet. Besides these baseline parameters we test other values for comparison (e.g., lower traffic volumes of just 100 m/s, and latencies between 0.25s and 5s). We note that our main contribution is a method that can be used for any latency and volume constraints of interest in concrete deployments, rather than specific results for a specific configuration.

5.2 Per-mix exponential delay

In multi-hop overlay routing, the end-to-end latency is the aggregation of the latencies incurred at the intermediate hops in the route, each corresponding to a layer in the mixnet. In turn, the latency at each hop is composed of three elements: the network propagation time

4 https://applevisaservices.com/blog/faq-how-many-visa-transactions-per-second.html

τ, the packet processing time δ, and the time that the packet dwells in the mix for anonymity purposes, which is sampled from an exponential distribution with mean μ seconds. Given a mixnet with L layers, the message passes by L mix nodes and $L+1$ links, and needs to be processed by L mixes in addition to the final recipient. The average end-to-end latency can be expressed as:

$$D_{e2e} = \mu L + (\tau + \delta)(L + 1) \tag{16}$$

In Section 6.1 we adjust the per-mix latency μ when comparing mixnets with different number of layers L, to fairly compare configurations that provide the same average end-to-end latency D_{e2e}. For this we consider average network propagation and packet processing times $\tau + \delta = 50$ms, and set μ as:

$$\mu = \frac{D_{e2e} - (\tau + \delta)(L + 1)}{L} \tag{17}$$

In the next two sections we study the impact of variable propagation and packet processing times on anonymity calculations.

5.3 Network propagation delay

In practice, the time τ taken by messages to travel through the internet in each hop may be highly variable. Nodes in an overlay network may be located all around the world, and network propagation times are proportional to geographical distance (ultimately bounded by the speed of light and in practice by a fraction of that speed). For example, distances of 500 Km can be covered in just 10ms while intercontinental distances may take over 100ms [33]. Thus, the propagation latency of a route is dependent on the relative geolocations of the nodes in the route. Furthermore, varying transmission medium characteristics, asymmetric and dynamic routing, congestion, and other effects introduce further variance in network propagation latency. Building a model of network propagation latency into a simulator that accurately predicts specific real-world deployment scenarios is a challenging task. Thus, we study the anonymity impact of propagation latency variability by comparing three scenarios with the same average τ: (i) constant propagation latency of $\tau = 50$ms for all links; (ii) variable latency per link sampled from a uniform distribution $\mathcal{U}[10, 90]$ ms; and (iii) a different propagation delay per mix that is randomly assigned but kept constant for all the received messages throughout the simulation. These three simplified network propagation models provide a sense of the impact of inter-mix propagation variability on anonymity.

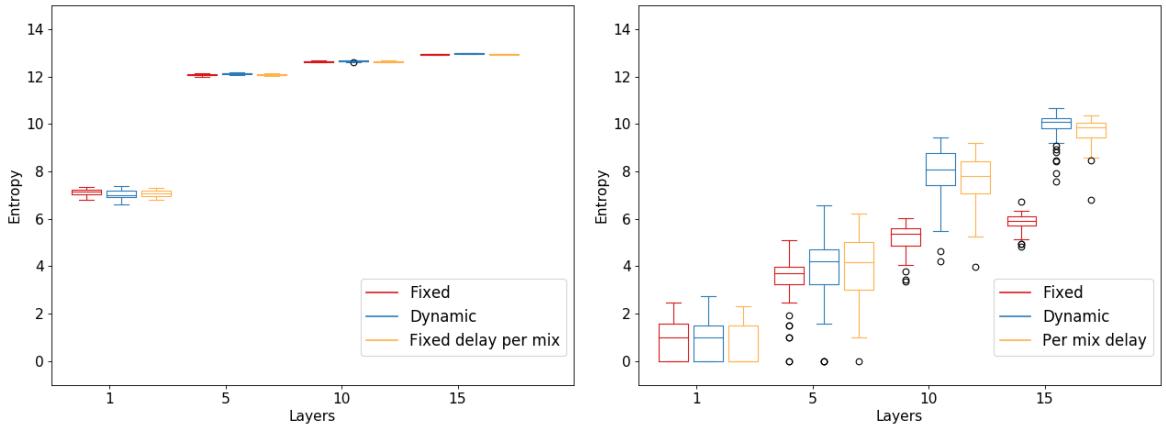
Our results are shown in Figure 5 for mixnets with different per-mix average latency μ and number of layers L (and thus various D_{e2e} latencies). When μ is larger than the propagation latency τ (Fig. 5a) the average anonymity measured in simulations is the same regardless of whether network propagation times are considered fixed or variable. On the other hand, if μ is orders of magnitude smaller than τ (Fig. 5b), the variation of τ has an anonymity impact that makes message tracing harder for an adversary, and this impact is exacerbated with the number of layers in the mixnet. The main effect leading to this anonymity increase when considering variable propagation times whether changing per message or just per mix, is that more output messages are possible matches for a target input, since the window of likely output matches starts earlier (the message could have been lucky to travel via links with low propagation delay) and ends later (the message could have been unlucky and travel via links with a lot of delay).

Based on these results, we conclude that considering constant propagation delays is a conservative assumption that seems to provide a lower bound on anonymity. Considering variable τ risks overestimating anonymity if the modelled variance is larger than the actual variance present in a concrete real-world mixnet deployment.

5.4 Non-uniform mix capacities

So far we have assumed 'uniform routing', i.e., that routing choices per layer are uniform in the number of nodes W in the layer, spreading the traffic load equally over all mix nodes in the network. In this section we consider networks with 'biased (capacity-based) routing', i.e., that allow for different node capacities and that select nodes for a route proportionally to the share of capacity that each node contributes to its mixnet layer. Capacity-based routing has two advantages: first, it is more inclusive, as even participants with limited resources can contribute to the network; and second, it better utilizes available resources, as some mix nodes are able to process more packets than others, and their extra capacity is wasted with uniform routing.

We compare anonymity for both types of routing (uniform and biased) in a small mixnet of $N = 30$ nodes organized as a WxL = 10x3 network, considering the baseline parameters provided in Section 5.1: $\lambda_{\mathcal{U}} = 5000$ messages per second, $D_{e2e} = 1$ second, and $b = 0.1$ fraction of corrupted nodes, i.e., $B = 3$ adversarial nodes.



(a) Fixed vs variable propagation delay when $\mu = 100$ms

(b) Fixed vs variable propagation delay when $\mu = 0.1$ms

Fig. 5. Anonymity with fixed vs variable propagation delays τ

Regardless of the type of routing, from a worst-case anonymity perspective the adversary compromises zero routes whenever no adversarial nodes are present in a layer. In this example $\alpha_{\mathcal{F}} = 0$ whenever the network topology is different from the optimal adversarial topology $\mathcal{T}_{opt} = (\mathcal{A}, \mathcal{B})$ that corresponds to $\mathcal{A} = \{9, 9, 9\}$ and $\mathcal{B} = \{1, 1, 1\}$. We thus focus our comparison on topologies $\mathcal{T}_{opt}$.

Next, we observe that adversaries can be expected to introduce high-capacity nodes in order to maximize route captures ($\alpha_{\mathcal{F}}$). Introducing many nodes in a staking-based system such as the Nym network can be very costly, as the adversary may need to spend millions of dollars to acquire enough stake to control a high percentage of nodes, or otherwise build enough reputation to persuade other stakeholders to delegate millions to support adversarial nodes [20]. In contrast, the additional cost of computing and bandwidth resources that provide significantly larger-than-average node capacity is in the range of thousands of dollars, orders of magnitude less and within the budget of a broader set of adversaries.

We consider that the adversary introduces nodes with 6x more capacity than the average honest node. Thus, in each layer of $W = 10$ nodes the adversarial node has 40% of the layer's capacity and is thus chosen for 40% of the routes. This is in contrast to the uniform routing scenario where each node, including the adversary's, routes 10% of the messages. Using Eq. (7), we can see that in this example biased routing allows the adversary to fully compromise $\alpha_F = 6.4\%$ of routes, in contrast to $\alpha_F = 0.1\%$ of routes in the case of uniform routing; i.e., a 6-fold increase in adversarial bandwidth and computing resources yields a 64-fold increase in the rate of worst-case compromise.

As final step we examine the effect of uniform vs biased routing on average anonymity, and show the results in Figure 6. The red bloxplots show the entropy distribution when considering a network adversary that does not control any mixnet nodes. In this case both uniform and biased routing provide the same level of average anonymity. The blue boxplots show results when the mixnet contains three adversarial nodes, which route 10% of messages per layer in the uniform case and 40% in the biased case. In this case we can see that compared to uniform routing, biased routing enables the adversary to not only compromise many more routes (worst-case anonymity) but also diminish average anonymity for the remaining messages. Based on these results we conclude that uniform routing is the best choice from an anonymity perspective and consider uniform routing policies in our remaining experiments. We note that volunteer-based networks like Tor [28] benefit from flexibility as that allows everyone to contribute even if they have limited capacity – and thus enforcing uniform routing in such networks comes with the cost of excluding prospective node operators with capacity limitations. In commercial networks like Nym [20] however, nodes are rewarded for operating the network, and it is thus possible to set a minimum capacity requirements and penalize with lower rewards the nodes that fail to perform.

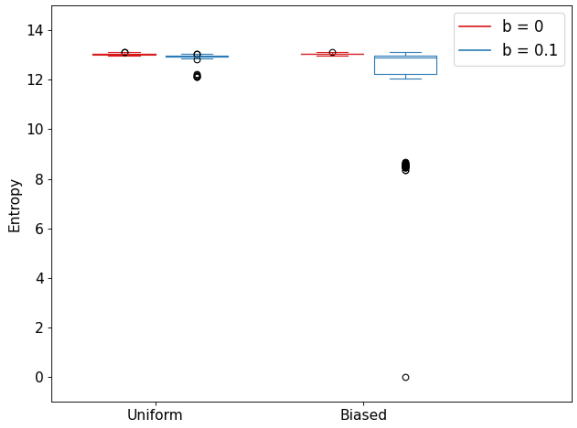


Fig. 6. Anonymity with uniform vs biased routing considering no adversarial nodes (red) and 10% of adversarial nodes (blue)

6 Optimization results

6.1 Optimizing the number of layers L

We first apply the methodology outlined in Section 4 to address the question: *given a deployment scenario and adversary model, what is the optimal number L of mixnet layers?* We consider a mixnet that routes $\lambda_{\mathcal{U}} = 5000$ messages per second and that has a width of $W = 10$ mixes per layer, meaning that each mix routes on average 500 messages per second.[5] We consider that the average end-to-end latency D_{e2e} is fixed per optimization experiment and evaluate anonymity for a range of possible D_{e2e} values, from 0.25 to 5 seconds. We first consider a global network adversary that can observe all links but does not control any mix nodes, i.e., $B = b = 0$. Next, we consider an adversary that, in addition to globally observing the network, also controls $b = 10\%$ of nodes in the mixnet.

6.1.1 Global network adversary

In the case of adversaries that do not control any nodes in the mixnet ($B = 0$), the fraction of fully compromised routes $\alpha_{\mathcal{F}}$ is zero for any number of mixnet layers $L \geq 1$. Worst-case anonymity constraints are therefore satisfied for all possible values of β.

Next we turn to examining average anonymity. Figure 7a shows the mean entropy as a function of the number of layers L for different values of D_{e2e}. As we can see in the results, when $B = 0$ the optimal number of layers is $L = 2$ for all values of the end-to-end latency D_{e2e}. As expected, anonymity values are higher for higher D_{e2e} [18]. Note that for $L = 1$, messages are partitioned in W subsets with disjoint anonymity sets (similarly to how they would be in parallel cascades), and thus the anonymity of $L = 1$ is naturally inferior to $L = 2$, which aggregates all messages in one large anonymity set. This effect would be further exacerbated with higher W, as W increases the partitioning.

Thus, assuming that all mix nodes are honest, a second mixnet layer achieves the best possible mixing for any end-to-end latency. Adding more layers implies wasting more time in propagation between layers, and leaving less latency budget for delaying messages inside the nodes (and thus *mixing* them in bigger pools). Entropy drops to zero and messages are fully distinguishable when all the latency is wasted on propagation and mixes simply forward messages without adding any random latency to reorder them. Considering a propagation latency per link of 50ms, this happens at $L = 4$ for $D_{e2e} = 0.25$s, $L = 9$ for $D_{e2e} = 0.5$s, and $L = 19$ for $D_{e2e} = 1$s. When entropy drops to zero this means that the adversary can identify which output message corresponds to a target input and the system provides no anonymity – though recall that, as shown in Section 5.3, variations of propagation time may make message tracing more uncertain in practical scenarios.

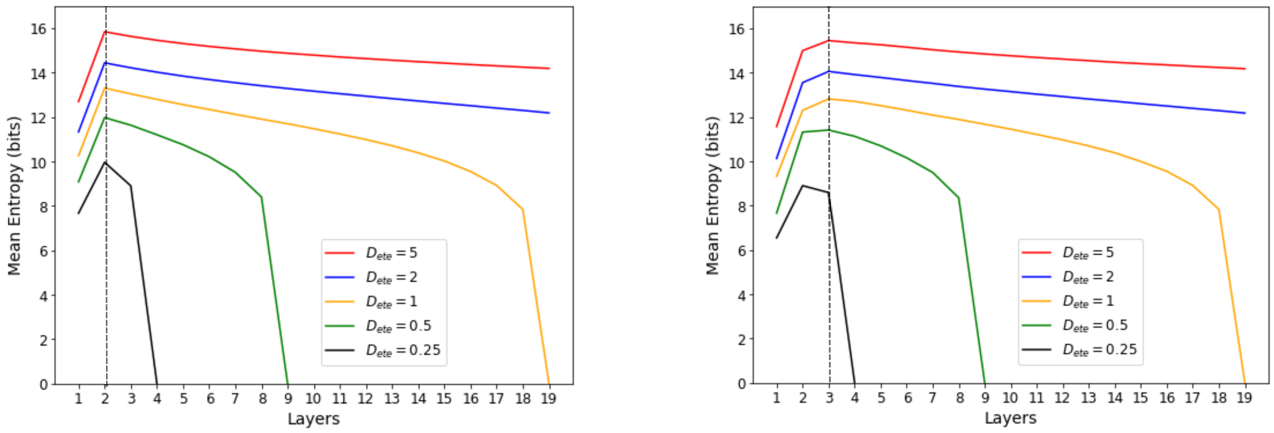
6.1.2 Fraction of corrupted nodes

We now examine scenarios where, *in addition to* observing all links, the adversary corrupts a fraction b of the nodes. We evaluate $b = 0.1$ in this section and present additional results for $b = 0.2$ and $b = 0.3$ in Appendix C.1.

First, from a worst-case perspective the fraction of fully compromised messages in a mixnet with L layers can be approximated as $\alpha_{\mathcal{F}} = b^L$ (Eq. (7)). Considering $b = 0.1$, a worst-case threshold $\beta = 0.001$ (one in a thousand messages is compromised by adversaries controlling 10% of the mixnet) implies that the minimum number of mixnet layers is $L = 3$. Setting a requirement of $\beta = 10^{-6}$ (just one in a million messages is compromised) raises the minimum number of layers to $L = 6$.

We then evaluate average anonymity for various end-to-end latencies D_{e2e} and layers L. Figure 7b shows

5 The currently available Nym implementation is benchmarked at 3125 Sphinx packet decryptions per second per processing core. An average node load of 500 messages per second enables mix nodes to tolerate traffic peaks of up to 6x the average load.



(a) Global network adversary, $b = 0$, $B = 0$

(b) Corruption level $b = 0.1$, $B = bLW$

Fig. 7. Mean Entropy wrt number of layers L for various D_{e2e}, considering $\lambda_{\mathcal{U}} = 5000$ and mixnet width $W = 10$.

the results, where we can observe that from an average anonymity perspective, the optimal number of layers L is dependent on the end-to-end latency D_{e2e}. For the more relaxed latency constraints $D_{e2e} \geq 1$s (yellow, blue and red lines), the optimal L is now $L = 3$; while scenarios with stricter latency constraints $D_{e2e} < 1$s (green and black lines) have their maximum at $L = 2$.

For $D_{e2e} \geq 1$s, the new optimum at $L = 3$ instead of $L = 2$ (as in the scenario without corrupted nodes) is consistent with the worst-case effect of node corruption, which brings anonymity to zero for a fraction of samples, and is mitigated by increasing L. Adding layers not only exponentially reduces the number of cases where anonymity is zero due to full route compromise, but also the number of outlier cases where anonymity is very low due to messages passing through a single honest mix. Beyond a certain point however, adding layers is more detrimental than beneficial, since the fraction of fully compromised routes is already too negligible for any further reduction to make a difference in the average, while the smaller mixing time (due to adding layers) takes a toll on anonymity.

When the latency D_{e2e} is more constrained, the optimal L still maxes out at a lower $L = 2$. This is explainable because for $D_{e2e} = 0.25$s, $L = 3$ implies that 80% of the available end-to-end latency budget is wasted on propagation across four links, leaving less than 17ms for mixing at each of the three nodes in the route. The fact that little mixing takes place per node facilitates message tracking for network adversaries and makes this configuration offer worse average anonymity than $L = 2$, where only 60% of latency budget is spent on propagation leaving 50ms for mixing at each of the two nodes.

For our optimization we consider $\beta = 0.001$ for adversaries that control a fraction $b = 0.1$ of the mixnet, and thus we discard mixnet configurations with $L < 3$. We select $L = 3$ and consider this number of mixnet layers in the remaining experiments. We note that $L = 3$ is commonly used in deployed anonymity networks [20, 25] as well as default experimental setting in prior literature [21, 44]. We are however the first to show that $L = 3$ is the choice that optimizes anonymity for layered mixnets in conditions of moderate rate of compromise ($b = 10\%$ and $\beta = 0.001$) for end-to-end latency tolerances of up to five seconds.

Lowering the worst-case threshold to $\beta = 10^{-6}$ while considering $b = 0.1$ sets the minimum number of layers at $L = 6$. In networks with end-to-end latency D_{e2e} of half a second or more this sets the optimal number of layers at $L = 6$ (since this offers better average anonymity than networks with $L > 6$). For networks with $D_{e2e} = 0.25$s there is no solution that can meet both the latency ($D_{e2e} = 0.25$s) and anonymity ($\beta = 10^{-6}$) requirements. We also note that in practical terms, adding layers to a mixnet comes with significant costs, as it requires additional resources per message (servers, computation and bandwidth) as well as incurring in increased rates of message loss (since it is enough for one node in the route to fail for the message to be lost).

6.2 Optimizing the network width W

Once we have fixed the number of layers L, we proceed to the next question: *how does the width W of the mixnet impact anonymity?* We consider a mean end-to-

end latency $D_{e2e} = 1$ second, mixnets with $L = 3$ layers, and a worst-case compromise threshold $\beta = 0.001$. As before, we consider $\lambda_{\mathcal{U}} = 5000$ messages per second and a minimum width $W_{min} = 10$ nodes, each routing 500 messages per second on average. We consider three threat models: network adversaries that do not control any nodes ($b = B = 0$), adversaries that control a fraction $b = 0.1$ of nodes regardless of network size, and adversaries that control a fixed number of nodes $B = \{9, 15, 30\}$ regardless of network size.

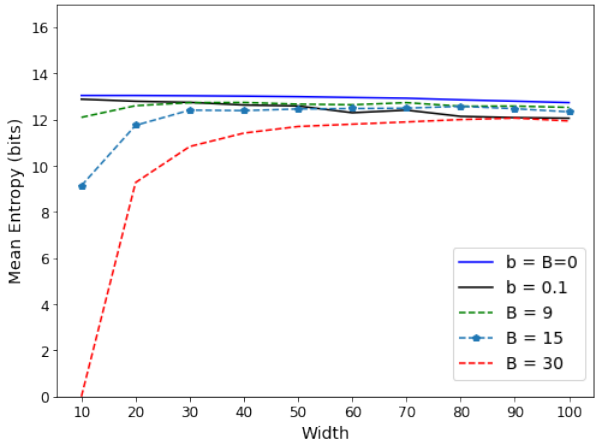


Fig. 8. Mean entropy as a function of the mixnet width W for various b and B ($L = 3$, $\lambda_{\mathcal{U}} = 5000$, $D_{e2e} = 1$s)

6.2.1 Constant fraction of corrupted nodes

We first consider adversaries that control a constant fraction b of the total nodes. This means that as the network grows in width W, the number of adversarial nodes B increases proportionally to W, as $B = bLW$.

In terms of worst-case anonymity, note that the fraction $\alpha_{\mathcal{F}}$ of fully compromised routes remains constant as W grows, because the probability of selecting fully corrupted routes is given by $\alpha_{\mathcal{F}} = b^L$ and thus remains constant. For $b = 0.1$ and $L = 3$ this corresponds to $\alpha_{\mathcal{F}} = 0.001$, which matches the worst-case threshold β. For $b = 0$, all $L \geq 1$ meet any possible value for the β threshold.

Figure 8 shows average anonymity for various scenarios, with the blue and black solid lines representing the cases where $b = 0$ and $b = 0.1$, respectively. As we can see in the figure, anonymity slowly but steadily decreases as the network width W increases. For $b = 0.1$, anonymity decreases by one bit when the width is $W = 100$ compared to the minimum $W_{min} = 10$, meaning that anonymity sets are halved due to the 10-fold width increase. This (modest) decrease happens because higher width means *thinner traffic* per mix, and thus lower level of mixing at each node. This result indicates that when considering a constant fraction adversary, the optimal network width is the minimum W that is sufficient to route the required traffic volume.

6.2.2 Constant number of corrupted nodes

Next we consider adversaries that can corrupt a fixed number B of nodes. In this case, the fraction $b = \frac{B}{LW}$ of adversarial nodes diminishes when the network grows in width W. Therefore, from a worst-case perspective, increasing W can be a strategy to meet constraints on worst-case rates. For example, considering $\beta = 0.001$ and $B = 15$, a mixnet of $L = 3$ with the minimum width $W_{min} = 10$ fails to meet worst-case constraints, as $\alpha_{\mathcal{F}} = (\frac{15}{30})^3 = 0.125$ is orders of magnitude larger than the threshold β. The width W that satisfies β constraints is given by $W = \frac{B}{L\sqrt[L]{\beta}}$. Applying this analysis to $B = 9$, $B = 15$ and $B = 30$ malicious nodes, results in minimum widths of $W = 10$, $W = 50$ and $W = 100$, respectively.

We then evaluate average anonymity in the same scenarios as a function of mixnet width. The results are shown with dashed lines in Figure 8 for 3 corrupted nodes per layer ($B = 9$), 5 corrupted nodes per layer ($B = 15$) and 10 corrupted nodes per layer ($B = 30$). Note that $B = 30$ at width $W = 10$ is a corner case where the *entire* network is controlled by the adversary, and thus all messages are fully traceable and average anonymity is zero. Compared to the previous adversary defined by $b = 0.1$, anonymity levels are lower for small W because relative corruption levels are higher. Overall, in scenarios with constant levels of node corruption we see that increasing W is initially beneficial for anonymity, as the diminishing share b of corrupted nodes dominates an anonymity improvement. After some point however, further increasing W begins to lower anonymity, as the dominant factor becomes the overall thinning of traffic (exploitable by a network adversary) rather than the fraction of compromised routes (which has already reached negligible levels). Based on these results we choose $W = 50$ for our remaining experiments.

6.3 Mix-based dummy strategies to compensate for low traffic volume

We finally turn our attention to the question of what happens when a network that has been optimized for an average traffic volume suddenly sees input traffic drop by more than an order of magnitude. A steep drop in traffic rate $\lambda_{\mathcal{U}}$, particularly in networks with a large width W, can significantly reduce average anonymity levels due to *thin traffic* effects (note that $\alpha_{\mathcal{F}}$ is independent of overall traffic volume, and thus worst-case anonymity is unaffected by traffic load fluctuations). Mixnet parameter adjustments that can counter the drop in user traffic include increasing the end-to-end latency D_{e2e} and reducing the network width W. Increasing latency however may not be possible without seriously undermining the usability of the system. As for network resizing, typically information on active mixnet nodes is updated every hour or few hours [20, 25], and thus structural changes to the network width may not be possible to effect immediately, or fast enough to follow fluctuations in the traffic volume coming from users.

Designs such as Loopix [44] and Vuvuzela [30] rely on client-based dummy traffic to ensure that traffic volumes are sustained and provide an adequate level of anonymity. Client-based dummies are a very effective solution to users going idle while staying online. If end users go offline however, all their traffic ceases, and it becomes unreasonable to expect that they will continue to generate dummy traffic.

Upon detection of low traffic volumes, mix nodes may intervene by generating an increased volume of dummy traffic to support anonymity levels. We note that various prior works propose mix-based dummies with more or less sophisticated strategies for generation and routing [16, 30, 44]. Here we consider two very simple strategies introduced in Section 2.1.4: *link-based dummies* and *partial-route dummies*.

Link-based dummies are generated by a mix and discarded by the successor. Assuming that mixes in adjacent layers communicate via a link-encrypted connection (e.g., TLS), link-based dummies need not be actual Sphinx packets that require expensive public key operations, but simply random data blocks the size of a Sphinx message that can be detected and discarded by the receiving mix node with just symmetric key operations. Link-based dummies are thus very cheap to implement for mixes, which makes them a low-cost countermeasure to use on-demand in case of decreased user traffic. On the downside, link-based dummies only protect towards network adversaries. If any of the two nodes sharing a link is compromised, the adversary can trivially filter out all the link-based dummy messages, rendering the protection ineffective for that link.

Partial-route dummies are generated by mixes in all but the last layer, routed through the mixnet and discarded by mixes in the last layer. In a network of $L = 3$ layers, compared to link-based dummies, partial-route dummies increase protection against adversarial nodes in the middle layer, who can no longer distinguish user messages from dummy messages generated and discarded by honest mixes. Note that this sort of indistinguishability towards middle-layer nodes requires that dummies are encoded as Sphinx packets, which significantly increases the cost of the dummy strategy compared to link-based dummies, as the processing of a Sphinx message requires senders, receivers and intermediaries to perform expensive public key operations.

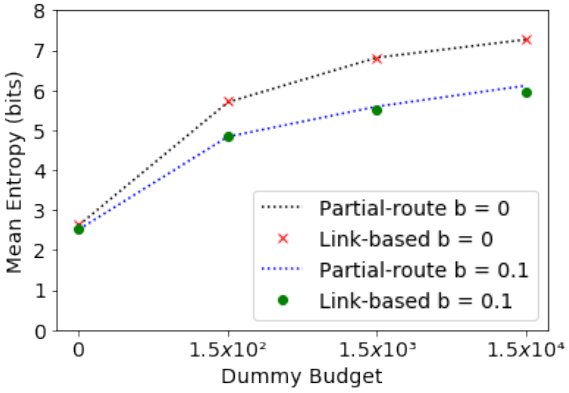


Fig. 9. Average anonymity in low-traffic conditions ($\lambda_{\mathcal{U}} = 100$ m/s) with link-based and partial-route dummy strategies towards network adversaries corrupting $b = 0$ and $b = 0.1$ of a mixnet with $L = 3$, $W = 50$, and $D_{e2e} = 1$s.

Contrary to link-based dummies, partial-route dummies are not evenly distributed across layers but instead increase linearly with the layers, because the dummies generated by each layer of mixes are added to the dummies from earlier layers being forwarded, until they are all discarded by the last layer. In order to enable a fair comparison between both dummy strategies we compare scenarios that have the same overall dummy rate. Considering $L = 3$, a network where each mix generates $\lambda_{\mathcal{M}} = 1$ partial-route dummy message per second is equivalent in terms of dummy traffic volume to a network where each mix generates $\lambda_{\mathcal{M}} = 1.5$ link-dummies per second. In the former case, one third of the dummies is sent from the first to the second layer and two thirds are sent from the second to the third layer, where they

are discarded. In the latter case, half the dummies are sent from the first to the second layer and the other half from the second to the third. No dummies are generated by the last layer in either case. We compare scenarios considering the same network-wide level of dummy traffic.

Our evaluation of both dummy strategies is shown in Figure 9 for various dummy rates, considering both network adversaries ($b = 0$) and adversaries that corrupt 10% of the mixnet ($b = 0.1$). Appendix C.3 includes additional results for adversaries with higher corruption rates ($b = 0.2$ and $b = 0.3$). The x axis in the figure corresponds to the overall dummy traffic in the network. Thus, $x = 1.5\text{x}10^2$ corresponds to $\lambda_{\mathcal{M}} = 1$ *partial-route dummy* generated by each mix per second. Given that $W = 50$, this implies that there are 50 dummies per second between the first and second layers, and 100 dummies per second between the second and third layers, for a total of 150. This is compared to a rate of $\lambda_{\mathcal{M}} = 1.5$ *link-based dummies* per second, with 75 dummies per second in each of the layers adding to the same 150 total. At $x = 1.5\text{x}10^4$ mixes generate $\lambda_{\mathcal{M}} = 100$ *partial-route dummies* or $\lambda_{\mathcal{M}} = 150$ *link-based dummies* per second. Considering $\lambda_{\mathcal{U}} = 100$ messages per second, dummy traffic makes up 43% of inter-mix traffic for $x = 1.5\text{x}10^2$, 88% for $x = 1.5\text{x}10^3$, and 98.6% for $x = 1.5\text{x}10^4$.

As we can see in the figure, in the absence of dummy traffic ($\lambda_{\mathcal{M}} = 0$) anonymity in this network configuration ($N = 3\text{x}50$) is very low due to the low levels of user traffic. Even where there is no corruption ($b = 0$), average anonymity is below 3 bits of entropy, meaning that effective anonymity set sizes are just a handful of messages. Anonymity levels significantly improve once mixes generate dummy traffic, with diminishing returns as the dummy rate increases and anonymity levels approach their upper bound. The best improvement is in the case of $b = 0$, where anonymity goes up by 5 bits, meaning that the anonymity set size multiplies 32-fold thanks to the dummy traffic. Compared to a mixnet that has the same three layers and minimal width ($W = 1$), we find that the average anonymity is the same for $W = 1$ (with no dummies) and for $W = 50$ with $\lambda_{\mathcal{M}} = 1.5\text{x}10^4$, meaning that a high rate of mix-generated dummy traffic succeeds in fully making up for the loss of anonymity caused by traffic thinning.

Link-based and partial-route dummies provide the same protection when there is no adversarial corruption ($b = 0$). This is to be expected since partial-route dummies offer extra protection towards intermediate corrupted nodes, but the same protection as link dummies against external network adversaries. The fact that dummies are distributed across layers $33\% - 66\%$ for partial-route dummies and $50\% - 50\%$ for link-based dummies seems to make no difference to the effectiveness of the dummy strategy. Once a fraction of nodes is compromised ($b = 0.1$), the gains obtained from dummy traffic are mitigated. A the lower levels of $\lambda_{\mathcal{M}}$ dummy traffic still significantly improves anonymity compared to not generating any dummies at all, and both link-based and partial-route strategies provide similar protection. As the dummy rate $\lambda_{\mathcal{M}}$ increases, partial-route dummies provide slightly better anonymity than link-based dummies. This effect becomes more pronounced with higher corruption rates, as shown in Appendix C.3.

Given the huge difference in cost of the two considered dummy strategies and their comparable impact, we conclude that link-based dummies are a simple and low-cost, yet effective option for mixes to support anonymity levels when there are sudden dips in user traffic.

7 Related work

Since Chaum's seminal work on untraceable email in 1981 [6], there has been a great amount of research related to mixnets, both in design [7, 8, 23, 30, 31, 34–36, 42, 44] as in evaluation and optimization [10, 12, 22, 38, 40, 43, 46, 50, 51]. We highlight in this section the most relevant prior work in terms of mixnet parameter optimization.

Rebollo-Monedero et al. [46] provide a method for optimizing the threshold and pool parameters of individual batch-based mixes. Their optimization problem is similar to ours: given a traffic volume and latency constraint, what are the optimal parameters that maximize entropy-based anonymity? Their anonymity system model is however vastly simpler than ours: where we consider full mixnets that may be partially compromised, they restrict themselves to a single (trusted) mix. In terms of optimization methods, their simpler model allows for multiobjective optimization of entropy-based anonymity, while we have to resort to empirical analysis to compare configurations and find optimal parameters that maximize average (entropy-based) anonymity while meeting worst-case anonymity constraints. In another result on mixing algorithm optimization but this time concerning individual continuous-time mixes [31], Danezis [12] showed that for a given mean latency exponentially-distributed delays provide optimal anonymity, thanks to the memory-less

properties of the exponential distribution. Prior results on mixnet topology optimization [21] are taken into consideration in our choice of focusing on layered networks. To the best of our knowledge, our work is the first to tackle mixnet size optimization.

Proposed systems that are reliant on dummy traffic, such as Loopix [44] and Vuvuzela [30], leave the tuning of parameters for the dummy traffic strategies as out of scope. In terms of dummy traffic optimization, Oya et al. [38] consider long-term disclosure attacks [17], which exploit persistent communication patterns to infer communication profiles over time, and propose dummy traffic strategies for networks of pool mixes. Their method, based on solving a least squares problem, optimizes the amount of dummies needed to achieve a desired level of protection against these long-term disclosure attacks. Our model does not make assumptions about repeated user behaviour, focusing instead on the anonymity offered by the mixnet to individual messages. Given specified models for user recipient selection, the methods of Oya et al. may be applied to the mixnet configuration resulting from our methods to further mitigate long-term attacks.

Finally, a first version of the MiXiM simulator that we use in our evaluations was first presented in [2]. The contribution in [2] is the evaluation of different mixing strategies and network connectivity topologies, which shows that Poisson mixing and stratified topologies provide better anonymity than pool mixing and topologies such as XRD [34]. We build on those results by considering the strategy and topology identified as providing the best anonymity properties, and proceeding to parameter optimization within the resulting design space.

8 Conclusion

Given deployment constraints on end-to-end latency, traffic load, and adversarial compromise, we propose a method to systematically optimize mixnet parameters, taking into account worst-case anonymity thresholds while maximizing average anonymity. To our knowledge, this problem had not been addressed as prior work on mixnet design [30, 34, 44] typically leaves network parametrization as out of scope. A real world mixnet deployment can use our methodology to select system parameter values to maximize anonymity for the bulk of messages in the assumed conditions. We note that the results omit corner cases, e.g., due to bootstrapping effects, as they ignore transitory initialization phases to focus on the steady state. A systematic study of anonymity considering fine-grained effects of, e.g., possible mixes geolocations and resulting distribution of propagation delays, is beyond out the scope of this work, which makes some simplifications in order to enable decisions on mixnet parameters such as the network width and number of layers.

Our method includes (i) an analytical framework to compute the rate of fully corrupted paths for a level of adversarial compromise, which defines *worst-case anonymity*; (ii) an empirical method for selecting the network width and number of layers to maximize *average anonymity*, and (iii) an evaluation of the effectiveness of mix-based dummy traffic strategies to support anonymity levels in low-traffic scenarios. Furthermore, we study the effect of parallel message processing by multi-core mixes and the effect of variable inter-mix propagation latency, concluding that real world effects make message tracing by adversaries more challenging, with simulations providing a lower anonymity bound. We compare uniform routing policies to biased capacity-based routing, and show that biased routing allows adversaries to arbitrarily increase their rate of worst-case compromise besides diminishing average anonymity.

Our results show that the optimal number of mixnet layers L depends on the combination of adversarial compromise and end-to-end latency. Tighter latency constraints lower L, while higher adversarial compromise increases L. We note that L remains small (maximum six layers) in all considered scenarios, due to the harmful effect of thinning traffic per mix when layers are added as end-to-end latency remains constrained. In terms of network width, narrower networks are better towards adversaries that compromise a fraction of the network, while slightly wider networks become optimal when adversaries are limited in the number of nodes they can compromise. Finally, we address the challenge of low traffic in a large (oversized) network. We evaluate two simple mix-based dummy traffic strategies and find that, considering global network adversaries that compromise 10% of the mixnet or less, inexpensive link-based dummies significantly improve anonymity up to the same level as if the network size had been optimized for a low traffic volume. If adversarial control of the mixnet is expected to be between 20% and 30%, the more computationally expensive partial-route dummies offer more robust protection.

Acknowledgments

This research is partially supported by the Research Council of KU Leuven under the grant C24/18/049, by CyberSecurity Research Flanders with reference number VR20192203 and by the United States Air Force and DARPA under Contract No. FA8750-19-C-0502. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of any of the funders.

References

[1] Bauer, K., McCoy, D., Grunwald, D., Kohno, T., and Sicker, D. Low-resource routing attacks against tor. In *Proceedings of the 2007 ACM Workshop on Privacy in Electronic Society* (New York, NY, USA, 2007), WPES '07, Association for Computing Machinery, p. 11–20.

[2] Ben Guirat, I., Gosain, D., and Diaz, C. Mixim: Mixnet design decisions and empirical evaluation. In *Proceedings of the 20th Workshop on Workshop on Privacy in the Electronic Society* (2021), pp. 33–37.

[3] Bohme, R., Danezis, G., Diaz, C., Kopsell, S., and Pfitzmann, A. Mix Cascades vs. Peer-to-Peer: Is One Concept Superior? In *Proceedings of Privacy Enhancing Technologies, PET 2004* (2004), vol. 3424 of *LNCS*, Springer-Verlag, pp. 243–255.

[4] Boneh, D., and Golle, P. Almost entirely correct mixing with application to voting. In *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS 2002)* (November 2002), pp. 68–77.

[5] Borisov, N., Danezis, G., Mittal, P., and Tabriz, P. Denial of service or denial of security? In *Proceedings of the 14th ACM conference on Computer and Communications Security Security (CCS 2017)* (2007), ACM, pp. 92–102.

[6] Chaum, D. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM 24*, 2 (1981), 84–88.

[7] Chaum, D., Javani, F., Kate, A., Krasnova, A., Ruiter, J., Sherman, A. T., and Das, D. cmix: Anonymization by high-performance scalable mixing. Tech. rep., Technical report, 2016.

[8] Chen, C., Asoni, D. E., Perrig, A., Barrera, D., Danezis, G., and Troncoso, C. TARANET: traffic-analysis resistant anonymity at the network layer. *CoRR abs/1802.08415* (2018).

[9] Cottrell, L. Mixmaster and remailer attacks, 1995.

[10] Danezis, G. Mix-networks with restricted routes. In *Proceedings of Privacy Enhancing Technologies workshop (PET 2003)* (March 2003), Springer-Verlag, LNCS 2760, pp. 1–17.

[11] Danezis, G. *On the anonymity of anonymity systems*. PhD dissertation, University of Cambridge, Computer Laboratory Queens' College, January 2004.

[12] Danezis, G. The traffic analysis of continuous-time mixes. In *Privacy Enhancing Technologies* (Berlin, Heidelberg, 2005), D. Martin and A. Serjantov, Eds., Springer Berlin Heidelberg, pp. 35–50.

[13] Danezis, G., and Clayton, R. Introducing traffic analysis. In *Digital Privacy: Theory, Technologies, and Practices (1st ed.)* (2007), Auerbach Publications, p. 22.

[14] Danezis, G., Dingledine, R., and Mathewson, N. Mixminion: Design of a Type III Anonymous Remailer Protocol. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy* (May 2003), IEEE, pp. 2–15.

[15] Danezis, G., and Goldberg, I. Sphinx: A compact and provably secure mix format. In *2009 30th IEEE Symposium on Security and Privacy* (2009), pp. 269–282.

[16] Danezis, G., and Sassaman, L. Heartbeat traffic to counter (n-1) attacks. In *Proceedings of the Workshop on Privacy in the Electronic Society (WPES 2003)* (October 2003).

[17] Danezis, G., and Serjantov, A. Statistical disclosure or intersection attacks on anonymity systems. In *International Workshop on Information Hiding* (2004), Springer, pp. 293–308.

[18] Das, D., Meiser, S., Mohammadi, E., and Kate, A. Anonymity trilemma: Strong anonymity, low bandwidth overhead, low latency - choose two. In *2018 IEEE Symposium on Security and Privacy (SP)* (2018), pp. 108–126.

[19] Diaz, C. *Anonymity and Privacy in Electronic Services*. PhD dissertation, KU leuven, December 2005.

[20] Diaz, C., Halpin, H., and Kiayias, A. The Nym Network. https://nymtech.net/nym-whitepaper.pdf, February 2021.

[21] Diaz, C., Murdoch, S. J., and Troncoso, C. Impact of network topology on anonymity and overhead in low-latency anonymity networks. In *Proceedings of the 10th International Conference on Privacy Enhancing Technologies* (Berlin, Heidelberg, 2010), PETS'10, Springer-Verlag, pp. 184–201.

[22] Diaz, C., Sassaman, L., and Dewitte, E. Comparison between two practical mix designs. In *Proceedings of ESORICS 2004* (September 2004), LNCS.

[23] Diaz, C., and Serjantov, A. Generalising mixes. In *Proceedings of Privacy Enhancing Technologies workshop (PET 2003)* (March 2003), Springer-Verlag, LNCS 2760, pp. 18–31.

[24] Diaz, C., Seys, S., Claessens, J., and Preneel, B. Towards measuring anonymity. In *Proceedings of the 2nd International Conference on Privacy Enhancing Technologies* (Berlin, Heidelberg, 2002), PET'02, Springer-Verlag, pp. 54–68.

[25] Dingledine, R., Mathewson, N., and Syverson, P. Tor: The second-generation onion router. In *Proceedings of the 13th Conference on USENIX Security Symposium - Volume 13* (USA, 2004), SSYM'04, USENIX Association, p. 21.

[26] Dingledine, R., Shmatikov, V., and Syverson, P. Synchronous batching: From cascades to free routes. In *Proceedings of Privacy Enhancing Technologies workshop (PET 2004)* (May 2004), vol. 3424 of *LNCS*, pp. 186–206.

[27] Goldschlag, D., Reed, M., and Syverson, P. Onion routing. *Communications of the ACM 42*, 2 (1999), 39–41.

[28] Goldschlag, D. M., Reed, M. G., and Syverson,

P. F. Hiding routing information. In *Proceedings of the First International Workshop on Information Hiding* (Berlin, Heidelberg, 1996), Springer-Verlag, p. 137–150.

[29] Gomulkiewicz, M., Klonowski, M., and Kutylowski, M. Rapid mixing and security of chaum's visual electronic voting. In *Proceedings of ESORICS 2003* (October 2003).

[30] Hooff, J. V. D., Lazar, D., Zaharia, M., and Zeldovich, N. Vuvuzela: Scalable private messaging resistant to traffic analysis. In *Proceedings of the 25th Symposium on Operating Systems Principles* (2015), pp. 137–152.

[31] Kesdogan, D., Egner, J., and Büschkes, R. Stop-and- Go-MIXes providing probabilistic anonymity in an open system. In *Information Hiding* (Berlin, Heidelberg, 1998), D. Aucsmith, Ed., Springer Berlin Heidelberg, pp. 83–98.

[32] Kilian, J., and Sako, K. Receipt-free MIX-type voting scheme - a practical solution to the implementation of a voting booth. In *Proceedings of EUROCRYPT 1995* (May 1995), Springer-Verlag.

[33] Kohls, K., and Diaz, C. Verloc: Verifiable localization in decentralized systems. *arXiv preprint arXiv:2105.11928* (2021).

[34] Kwon, A., Lu, D., and Devadas, S. {XRD}: Scalable messaging system with cryptographic privacy. In *17th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 20)* (2020), pp. 759–776.

[35] Mathewson, N., and Dingledine, R. Mixminion: Strong anonymity for financial cryptography. In *Proceedings of Financial Cryptography (FC '04)* (February 2004), Springer-Verlag, LNCS 3110, pp. 227–232.

[36] Möller, U., Cottrell, L., Palfrader, P., and Sassaman, L. Mixmaster Protocol — Version 2. IETF Internet Draft, July 2003.

[37] Neff, C. A. A verifiable secret shuffle and its application to e-voting. In *Proceedings of the 8th ACM conference on Computer and Communications Security* (2001).

[38] Oya, S., Troncoso, C., and Pérez-González, F. Do dummies pay off? limits of dummy traffic protection in anonymous communications. In *International Symposium on Privacy Enhancing Technologies Symposium* (2014), Springer, pp. 204–223.

[39] Parekh, S. Prospects for remailers. *First Monday 1* (August 1996).

[40] Pérez-González, F., Troncoso, C., and Oya, S. A least squares approach to the static traffic analysis of high-latency anonymous communication systems. *IEEE Trans. Inf. Forensics Secur. 9*, 9 (2014), 1341–1355.

[41] Pfitzmann, A., and Hansen, M. Anonymity, unobservability, and pseudonymity—a proposal for terminology. In *Designing privacy enhancing technologies* (2001), Springer, pp. 1–9.

[42] Pfitzmann, A., Pfitzmann, B., and Waidner, M. ISDN-mixes: Untraceable communication with very small bandwidth overhead. In *Proceedings of the GI/ITG Conference on Communication in Distributed Systems* (February 1991), pp. 451–463.

[43] Piotrowska, A. M. An empirical study of privacy, scalability, and latency of nym mixnet, 2 2021.

[44] Piotrowska, A. M., Hayes, J., Elahi, T., Meiser, S., and Danezis, G. The loopix anonymity system. In *26th {USENIX} Security Symposium ({USENIX} Security 17)* (2017), pp. 1199–1216.

[45] Poon, J., and Dryja, T. The bitcoin lightning network: Scalable off-chain instant payments, 2016.

[46] Rebollo-Monedero, D., Parra-Arnau, J., Forné, J., and Diaz, C. Optimizing the design parameters of threshold pool mixes for anonymity and delay. *Computer networks 67* (2014), 180–200.

[47] Serjantov, A. *Better Anonymous Communications*. PhD dissertation, University of Cambridge, Computer Laboratory, October 2004.

[48] Serjantov, A., and Danezis, G. Towards an information theoretic metric for anonymity. In *Proceedings of the 2nd International Conference on Privacy Enhancing Technologies* (Berlin, Heidelberg, 2002), PET'02, Springer-Verlag, pp. 41–53.

[49] Shirazi, F., Simeonovski, M., Asghar, M. R., Backes, M., and Diaz, C. A survey on routing in anonymous communication protocols. *ACM Computing Surveys (CSUR) 51*, 3 (2018), 1–39.

[50] Shmatikov, V., and Wang, M.-H. Timing analysis in low-latency mix networks: Attacks and defenses. In *Proceedings of ESORICS 2006* (September 2006).

[51] Troncoso, C., and Danezis, G. The bayesian traffic analysis of mix networks. In *Proceedings of the 16th ACM conference on Computer and communications security Security (CCS 2009)* (2009), pp. 369–379.

[52] Zhu, Y., Fu, X., Graham, B., Bettati, R., and Zhao, W. On flow correlation attacks and countermeasures in mix networks. In *Proceedings of Privacy Enhancing Technologies workshop (PET 2004)* (May 2004), vol. 3424 of *LNCS*, pp. 207–225.

A Notation

Table 1 summarizes our notation.

B Imbalanced layers

In this section we develop an analysis analogous to the one in Section 3.1 for the case of layered mixnets with imbalanced layers, i.e., mixnets where layers have variable width. This is the case if the algorithm assigns nodes to layers independently for each node, e.g., deriving the layer from the node's public key and a public random beacon [20]. We represent layer widths with a vector $(w_1, w_2, ..., w_L)$, where w_i is the width of layer i. Since the choice of layer is made independently per node, the assignment is modeled by a multinomial distribution with N trials, L categories and uniform probability $\frac{1}{L}$ over the categories. The probability of having

Notation	Description
N	Total number of nodes
L	Number of layers
W	Width of the network
W_{min}	Minimum width of the network
b	Fraction of adversarial nodes ($\frac{B}{N}$)
D_{e2e}	Average end-to-end latency
μ	Average per-mix delay
τ	Link propagation time
δ	Per-mix processing time
$\lambda_{\mathcal{U}}$	Rate of user-generated traffic
$\lambda_{\mathcal{M}}$	Rate of mix-generated dummy traffic
A	Number of honest nodes
a_i	Number of honest nodes in layer i
B	Number of corrupted nodes
b_j	Number of corrupted nodes in layer j
$\mathcal{F}$	Event of fully compromised route
$\alpha_{\mathcal{F}}$	Fraction of fully compromised routes
β	Maximum tolerated $\alpha_{\mathcal{F}}$
$\mathcal{T}_v$	Valid topology
$\mathcal{T}_e$	Topology with at least one empty layer
$\mathcal{T}_{opt}$	Optimal adversarial topology
m_i	$i-$th message
m_t	Target message
$\mathbf{Pr}_L[m_i = m_t]$	Probability that output m_i is the target
$\mathbf{Pr}_{Mix}[m_t]$	Probability that m_t is in the mix

Table 1. Notation parameters.

a layer size distribution $(w_1, w_2, ..., w_L)$ is subject to the constraint that $\sum_{i=1}^{L} w_i = N$ and given by:

$$Pr(w_1, w_2, ..., w_L) = \frac{N!}{\prod_{i=1}^{L} w_i!}(\frac{1}{L})^N \tag{18}$$

We consider that only mixnet topologies with at least one node per layer are considered valid, i.e., we require that $\forall i\ w_i > 0$, to ensure that messages always go through L mixes. If a topology selection results in a mixnet where $w_i = 0$ for any layer i, the selection is discarded and the assignment is re-sampled with updated randomness.

We consider that the N nodes are split into A honest nodes and B malicious nodes, $N = A+B$, with distribution over the layers given by vectors $\mathcal{A} = \{a_1, a_2, .., a_L\}$ and $\mathcal{B} = \{b_1, b_2, .., b_L\}$, where $w_i = a_i + b_i$. The fraction of compromised routes $\alpha_{\mathcal{F}}$ is computed with Eq. (1), subject to the constraints expressed in Eq. (2), Eq. (3), and Eq. (5). The difference in node assignment however does invalidate the constraint in Eq. (4), which is instead substituted by:

$$\forall i, \quad a_i + b_i > 0 \tag{19}$$

Given a valid mixnet topology $\mathcal{T}_v = (\mathcal{A}, \mathcal{B})$, the fraction $\Pr(\mathcal{F}|\mathcal{T}_v)$ is computed with Eq. (6). We now derive the distribution $\Pr(\mathcal{T}_v)$ of valid topologies $\mathcal{T}_v = (\mathcal{A}, \mathcal{B})$, defined by the number of honest and malicious nodes in each layer, $\mathcal{A} = \{a_1, a_2, .., a_L\}$ and $\mathcal{B} = \{b_1, b_2, .., b_L\}$, subject to the already mentioned constraints.

In imbalanced networks each node's layer assignment is done independently, and thus the probability $\Pr(\mathcal{A}, \mathcal{B})$ of a topology $(\mathcal{A}, \mathcal{B})$ can be computed as the probability of two independent assignments $\mathcal{A}$ and $\mathcal{B}$, i.e., $\Pr(\mathcal{A}, \mathcal{B}) = \Pr(\mathcal{A})\Pr(\mathcal{B})$. $\Pr(\mathcal{A})$ and $\Pr(\mathcal{B})$ are each described by a multinomial:

$$\Pr(\mathcal{A} = \{a_1, a_2, .., a_L\}) = \frac{A!}{\prod_{i=1}^{L} a_i!}(\frac{1}{L})^{\sum_{i=1}^{L} a_i} \tag{20}$$

$$\Pr(\mathcal{A} = \{a_1, a_2, .., a_L\}) = \frac{A!}{L^A \prod_{i=1}^{L} a_i!} \tag{21}$$

$$\Pr(\mathcal{B} = \{b_1, b_2, \ldots, b_L\}) = \frac{B!}{L^B \cdot \prod_{i=1}^{L} b_i!} \tag{22}$$

We recall that topologies with empty layers are discarded. We define $\mathcal{T}_e$ as the set of topologies with at least one empty layer, i.e., topologies that meet the constraints in Eq. (2), Eq. (3), and Eq. (5), but violate the constraint in Eq. (19) for at least one layer. We define a normalization factor Z that accounts for the aggregate probability of choosing a topology that is discarded due to empty layers:

$$Z = \frac{A!B!}{L^N} \sum_{(\mathcal{A},\mathcal{B})\in\mathcal{T}_e} (\prod_{j=1}^{L} a_j! \prod_{k=1}^{L} b_k!)^{-1} \tag{23}$$

The probability of selecting a valid topology $\mathcal{T}_v = (\mathcal{A}, \mathcal{B})$ that meets all constraints is re-normalized considering $(1 - Z)$, to account for discarded topologies $\mathcal{T}_e$:

$$\Pr(\mathcal{A}, \mathcal{B}) = \frac{A!B!}{(1-Z)L^N} \sum_{(\mathcal{A},\mathcal{B})\in\mathcal{T}_v} (\prod_{j=1}^{L} a_j! \prod_{k=1}^{L} b_k!)^{-1} \tag{24}$$

Putting everything together we obtain:

$$\alpha_{\mathcal{F}} = \frac{A!B!}{(1-Z)L^N} \sum_{(\mathcal{A},\mathcal{B})\in\mathcal{T}_v} \prod_{i=1}^{L} \frac{b_i}{a_i + b_i} (\prod_{j=1}^{L} a_j! \prod_{k=1}^{L} b_k!)^{-1} \tag{25}$$

The results for $\alpha_{\mathcal{F}}$ in this case are nearly identical to those obtained for balanced networks and shown in Figure 3, meaning that the expected fraction of fully corrupt paths is the same regardless of whether layers are balanced or imbalanced. Furthermore, in networks of a hundred nodes the variance is so low that $\alpha_{\mathcal{F}}$ can be safely approximated by $(\frac{B}{N})^L$.

As side observation, note that the optimal topology for the adversary in *imbalanced* networks is a corner case where all mixnet layers but one have a single node, which happens to be adversarial, with a lone layer containing all the rest of nodes. The fraction of compromise in this case would be $\frac{B-(L-1)}{N-(L-1)}$. Such scenarios are rare but possible in toy-sized networks but their likelihood quickly becomes negligible for any realistic network sizes. Large networks have overwhelming probability of being close to balanced for the same reason that casting a fair die many times yields roughly the same counts for each side, with relative variance only affecting small sample sizes.

We finally note that imbalanced layers do not present any advantage over balanced layers, and as disadvantage they typically incur in a small loss of overall mixnet throughput, which is bounded by the layer with the least capacity. Furthermore, particularly in small networks, imbalanced layers present worst-cases that provide more advantage to the adversary than the worst-case of balanced networks. Therefore, we argue that network topologies with balanced layers should be preferred when designing a mixnet, as they minimize capacity waste caused by imbalances in the sizes of different layers and avoid scenarios that could give outsize advantage to the adversary.

C Optimization results with high rates of adversarial corruption

In this section we expand on the results presented in Section 6 with scenarios where the adversary corrupts a higher percentage of nodes $b = 0.2$ and $b = 0.3$.

C.1 Number of layers L

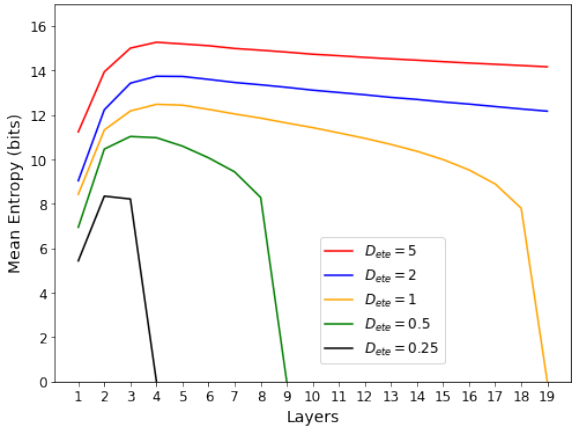
First we examine results for selecting the optimal number of layers L with a high corruption rate b. In terms of worst-case anonymity, the same threshold $\beta = 0.001$ imposes a higher L than in the cases with lower b shown in Section 6.1: while $L \geq 1$ was enough for $b = 0$ and $L \geq 3$ for $b = 0.1$, $b = 0.2$ raises the minimum required layers to $L \geq 5$ and $b = 0.3$ to $L \geq 6$. Increasing the worst-case tolerance to $\beta = 0.01$ (on average one message out of 100 has a fully compromised route) allows for configurations where $L \geq 3$ for $b = 0.2$ and $L \geq 4$ for $b = 0.3$.

We then examine average anonymity in these scenarios, showing the results for $b = 0.2$ and $b = 0.3$ in Figure 10. As we can see in the figure, higher corruption rates b increase the optimal L for a given end-to-end latency D_{e2e}. For example, for $D_{e2e} = 5$s and $D_{e2e} = 2$s, the optimal L increases from $L = 4$ for $b = 0.2$ to $L = 5$ for $b = 0.3$; while, as shown in Figure 7, $b = 0$ had the optimum at $L = 2$ and $b = 0.1$ at $L = 3$. Increasing the number of mixnet layers beyond the optimal L makes the average anonymity, up to the point where it drops to zero because all available latency budget is spent on propagation delays. For the more constraining latency $D_{e2e} = 0.25$s, increasing the corruption rate to $b = 0.3$ makes $L = 3$ become the optimum instead of $L = 2$, which is the optimum for lower rates of corruption.

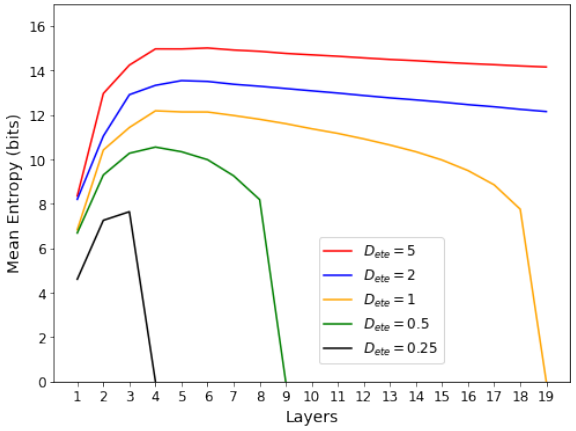
Combining worst-case anonymity constraints and average anonymity optima for the different scenarios, we conclude that for $\beta = 0.001$ the minimum layers required by the worst-case dominate, determining that the number of layers should be $L = 5$ for $b = 0.2$ and $L = 6$ for $b = 0.3$. This choice is the same for all $D_{e2e} \geq 0.5$s, while no solution exists for $D_{e2e} = 0.25$s that can satisfy both anonymity and latency constraints. Considering a more relaxed worst-case anonymity constraint $\beta = 0.01$ for $b = 0.2$ would lead to selecting the L that maximizes average anonymity, which is $L = 3$ for $D_{e2e} \leq 0.5$s and $L = 4$ for larger D_{e2e}. In the case of $b = 0.3$, the choices would be $L = 4$ for $D_{e2e} = 0.5$s and $D_{e2e} = 1$s, and $L = 5$ for larger D_{e2e}; while again no solution exists for $D_{e2e} = 0.25$s that satisfies both worst-case anonymity and latency constraints.

C.2 Mixnet width W

We now examine the effects of mixnet width W with increasing percentages b of adversarial node corruption. As mentioned in Section 6.2, a fixed b and L determine the worst-case anonymity rate as $\alpha_{\mathcal{F}} = b^L$, regardless of the mixnet width W. We thus examine average anonymity in scenarios with higher values of b, and show the results in Figure 11. As expected, an increased b lowers the average anonymity for any width W, and the decline of average anonymity caused by traffic thinning with larger W is slightly faster for higher b. Overall, the decline in average anonymity is noticeable but moderate: between one and two bits of decline when the mixnet width W is increased by an order of magnitude from $W = 10$ to $W = 100$.



(a) Average entropy for $b = 0.2$



(b) Average entropy for $b = 0.3$

Fig. 10. Mean Entropy wrt number of layers L for various D_{e2e}, with $W = 10$ and $\lambda_{\mathcal{U}} = 5000$.

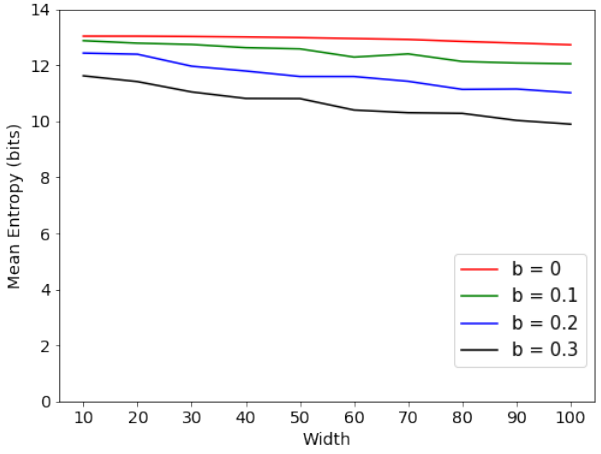


Fig. 11. Mean entropy as a function of the mixnet width W for various levels of corruption b ($L = 3$, $\lambda_{\mathcal{U}} = 5000$, $D_{e2e} = 1$s)

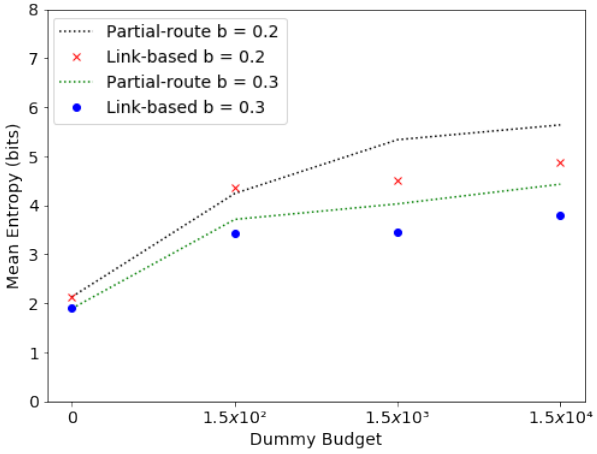


Fig. 12. Average anonymity in low-traffic conditions $\lambda_{\mathcal{U}} = 100$ m/s with link-based and partial-route dummy strategies towards adversaries corrupting $b = 0.2$ and $b = 0.3$ of a mixnet with $L = 3$, $W = 50$, and $D_{e2e} = 1$s.

C.3 Effectiveness of dummy strategies

Finally, we examine the effectiveness of partial-route and link-based dummy strategies when there is very low traffic from users ($\lambda_{\mathcal{U}} = 100$m/s) in a mixnet dimensioned for higher traffic loads (width $W = 50$), in the presence of adversaries that corrupt $b = 0.2$ and $b = 0.3$ of the mixnet. Our results, shown in Figure 12, illustrate that high levels of mixnet corruption diminish the effect of dummies (compared to the results for lower b shown in Figure 9) and in particular of link-based dummies — many of which are now identified and discarded by the large number of adversarial nodes. When $b = 0.3$, even high levels of link-based dummies result in anonymity below 4 bits; while for $b = 0.2$ the anonymity set triples (1.4 bit increase), reaching an average entropy of 5 bits.

Partial-link dummies fare moderately better in this challenging adversarial scenario with low traffic and high level of compromise. When $b = 0.2$, partial-route dummies can increase anonymity up to 6 bits, a 16-fold increase in anonymity set size compared to the 2 bits obtained when not using dummies. In the case of $b = 0.3$ however, even high levels of partial-route dummies result in a mean entropy of 4.5 bits, corresponding to a perfect indistinguishability set of about twenty other users, which may be too small to provide meaningful protection.

We conclude from these results that the more expensive partial-route dummies should be preferred in deployment scenarios where high levels of node corruption are expected. On the other hand, in scenarios mainly

concerned with global network adversaries, where the degree of mixnet corruption is expected to remain below 10%, link-based dummies should be preferred as they provide practically the same level of protection as partial-route dummies for a fraction of the cost.