# MOCHA: Mixnet Optimization Considering Honest Client Anonymity

Mahdi Rahimi
mahdi.rahimi@esat.kuleuven.be
COSIC, KU Leuven
Leuven, Belgium

## ABSTRACT

Mix networks (mixnets) safeguard client anonymity by forwarding traffic through multiple intermediary nodes (mixnodes), which reorder and delay messages to obscure communication patterns against a global passive adversary capable of monitoring all network transmissions. The anonymity provided by mixnets is usually assessed with a discrete-event simulator, gauging a target message's indistinguishability among output messages. While useful for comparative analysis, this approach only approximates the mixnet's anonymity potential. Hence, this paper sheds light on the necessity of considering the client (originator of messages) itself to gauge anonymity accurately. We further provide an algorithm (simulator) to simulate client anonymity for Loopix mixnets. We conduct experiments to optimize general Loopix mixnet parameters, considering both message and client anonymity. Our findings indicate that message anonymity often provides an upper bound and can yield misleading results for mixnet optimization, underscoring the importance of client anonymity. Additionally, we explore scenarios where client anonymity is significantly compromised due to an insufficient number of clients. To address these cases, we propose a multimixing strategy that enhances client anonymity by effectively merging varied traffic types with different mixing characteristics.

## CCS CONCEPTS

• **Security and privacy → Network security**.

## KEYWORDS

Anonymity, Anonymous systems, Mix networks

## 1 INTRODUCTION

Anonymous communications have been established for over four decades with the goal of concealing who communicates with whom in network exchanges [27]. Despite the diverse strategies aimed at achieving anonymity, mix networks (or mixnets) [4, 6, 14, 27] have proven to be significantly effective. This is particularly evident in scenarios where a global passive adversary possesses extensive surveillance capabilities over all network entities and their communications [5]. The anonymity afforded by mixnets is specifically engineered through multi-hop routing via intermediary nodes within the mixnet (referred to as mixnodes). This mechanism involves mixing and reshuffling client traffic at each intermediary mixnode, effectively obfuscating the correlation between incoming and outgoing traffic patterns. As a result, provided that there is at least one honest intermediary mixnode performing traffic shuffling, the ability of a global adversary to deanonymize clients is significantly impeded [5].

The reordering of input traffic within mixnodes can be achieved through various methodologies. For example, in certain designs, this reordering depends on the volume of received traffic reaching a predetermined threshold, known as a threshold mix [4]. Alternatively, messages entering the mixnodes may be independently reordered, often following an exponential distribution. This implies that each message is dispatched from the mixnode after a period governed by an exponential distribution, a technique referred to as stop-and-go mixing [12]. Threshold-based schemes frequently lead to unpredictable time delays for messages, rendering them unsuitable for time-sensitive applications [6]. Conversely, methods utilizing an exponential distribution can better regulate latency, owing to the predictable average delay inflicted on messages. Moreover, the exponential distribution's memoryless characteristic augments the untraceability of messages within the stop-and-go mix. This attribute renders it more effective compared to other reordering techniques [12], thus enhancing its practical applicability [6].

Similar to different strategies for designing mixing types, one can construct a mixnet using varied approaches. For example, one approach allows clients to randomly pick mixnodes from a pool to establish message routes with $L$ intermediary mixnodes, enabling mixnodes to be utilized at any position in the intermediary hops. This method is referred to as free routing [10]. Alternatively, mix networks can be organized into different sets of $L$ cascades, where each mixnode is predetermined for specific hops, and clients simply choose one of the cascades for forwarding their messages [3]. Lastly, the stratified mixnet structure [6, 14] organizes the mixnet into $L$ groups, each constituting a layer within the mixnet. Here, clients select their first hop from the first layer, their second hop from the second layer, and so forth. The stratified mixnet architecture has been demonstrated to provide superior anonymity for clients compared to other configurations [7]. Consequently, our focus in this paper is on these types of topologies.

Combining the principles of stop-and-go mix design with the stratified mixnet architecture, the Loopix mixnet is introduced [14], a layered network comprising mixnodes that shuffle incoming traffic based on exponential distribution delays. This innovative mixnet design has been demonstrated to be more compatible with practical applications, offering anonymity while imposing a somewhat predictable average latency on traffic. MOCHA also bases its analysis on this popular structure of the mixnet.

### 1.1 Related Work

Mixnets, regardless of their construction or type, are designed to provide anonymity. How this anonymity should be considered in mixnets has been an interesting research question for decades. To answer this, we begin with the seminal work of [9], which

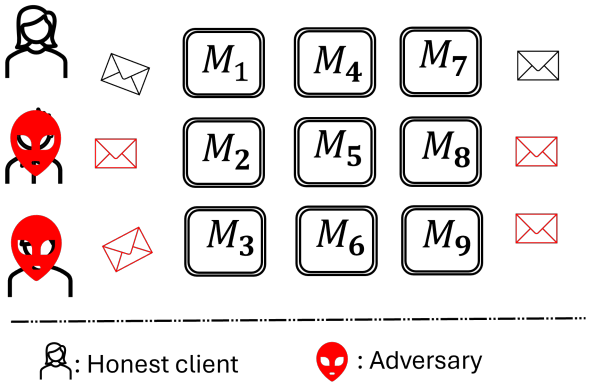



**Figure 1: Adversary compromises clients to deanonymize an honest client in a stratified mixnet architecture with $L = 3$.**

focused on assessing the anonymity of earlier mixnets (such as cascade or free routes mixnets). They introduced the concept of the anonymity set (the number of clients sending messages to a mixnet) and showed its specific importance for an adversary who wants to understand the message originator in the output of the mixnode or mixnet. They further used information-theoretic analysis, mainly employing Shannon entropy [26], to precisely gauge sender (client) anonymity in such early mixnets. Expanding on this, [25] also employed information-theoretic analysis for threshold (or more accurately, pool-based) mixes to assess their anonymity, considering both senders and receivers for cascade or free routes topologies of the mixnet, primarily using entropy concepts.

Contrasting these earlier works, [8] introduced a new notion of anonymity termed message anonymity. They assumed an adversary interested in targeting specific messages at the input/output of the mixnet and aimed to find their associated message in the output/input of the mixnet. This definition provided a novel way to compare different mixnet designs for anonymity purposes. Inspired by this, Loopix [14], while introducing a new mixnet design, proposed the expected difference in likelihood method. This method assesses relative sender anonymity by distinguishing between two different senders who each send roughly the same number of messages. This approach essentially represents a weaker notion of client anonymity. On the contrary, recent works such as [2] and [13] primarily focused on message anonymity and developed methods and simulators to assess message anonymity in Loopix-like mixnets. Their proposed metrics have also been extensively used in subsequent studies [16–24] to evaluate or improve the usability and anonymity guarantees of mixnets.

## 1.2 Problem Statement

Considering the works proposed after the introduction of Loopix-like mixnets [14], we realized that either the proposed metrics represent a weak notion of client (sender) anonymity or indistinguishability, or they are predominantly based on message anonymity. Analyzing the anonymity of individual messages to assess the overall anonymity provided by mixnet designs can be beneficial for comparative analysis. However, this approach can often lead to a false evaluation of mixnets. For instance, the example provided in Fig. 1, where three clients send messages to the mixnet, suggests that analyzing message anonymity gives a high degree of anonymity since each user generates many messages. However, if we consider that the adversary is primarily interested in identifying the original message senders, compromising some of the clients can easily reveal the generator.

Moreover, using message anonymity for optimizing mixnets, as in [11], to analyze parameters such as the number of layers, the number of mixnodes, and the mixing operation might be unrealistic and detrimental to practical cases. Therefore, considering only message anonymity to capture the actual anonymity provided by the mixnet is not a feasible and practical choice. We need a robust way to gauge anonymity to ensure usability in practice.

## 1.3 Contributions

Realizing the necessity of having a concrete simulator based on client anonymity, we take examples from earlier works [9, 25] that were pioneers in introducing client anonymity, primarily focusing on free routes topology and cascade mixnodes. Building on these works, our **first** contribution is to extend their analysis based on the entropy of clients for Loopix-based mixnets. More precisely, we start with a scenario involving a group of clients using the mixnet. Initially, we formalize a method for simulating client anonymity using a single mixnode. This approach is then expanded to encompass a stratified mixnet through the introduction of an algorithm designed for implementation via discrete-event simulations. The proposed algorithm provides the probability distribution of each message being generated by any of the clients using the mixnet. We further represent the entropy of such a distribution as $\mathsf{H}(C)$ and name it client anonymity. Measured in bits, client anonymity implies that, for instance, if $\mathsf{H}(C) = 5$, a message might originate from one of $2^5 = 32$ different clients.

We **second** simulated a Loopix-based mixnet using a discrete-event simulator developed with *SimPy* [15] in *Python*. Our simulator includes both message and client anonymity metrics for the Loopix mixnet and stop-and-go mix. Using our simulator, we assess how message anonymity and client anonymity distinguish the optimum parameters for a mixnet, aiming to determine which anonymity metric is more useful and realistic in practice. To do so, we set up 128 clients using the mixnet, collectively generating 10,000 messages per second sent to the mixnet with 3 layers, each having 32 mixnodes. Each mixnode delays messages with an exponential distribution with an average of 50 ms. We targeted some of the messages and clients in this scenario to measure both client and message anonymity.

Our experiments revealed that the inherently larger number of messages compared to the number of clients results in an increased message-based anonymity metric, approximately 5 bits higher than that for clients measured in the same setting, suggesting that message anonymity might overestimate the mixnet's anonymity. Additionally, we noted that client anonymity is primarily reliant on a single mixnode's shuffling, provided all clients select the initial hop of the mixnet uniformly at random. Specifically, if all clients send sufficient messages to a mixnode, the anonymity for clients for messages inside the mixnode will be maximized. Adding more

mixnodes will not significantly enhance anonymity but will help with scalability and trust distribution purposes. In other words, enhancing anonymity does not necessarily depend on multiple layers in the mixnet. Meanwhile, our experiments measuring message anonymity suggest that anonymity increases by adding more mixing layers. This indicates that such considerations lead to high-cost configurations of the mixnet with minimal achievement in contrast to what is suggested by [11].

We additionally analyzed a scenario with a stronger adversary, capable of compromising the anonymity offered by mixnodes in the mixnet, known as the mixnode adversary. Findings suggest that as the adversary's budget within the mixnet increases, both message and client anonymity levels decrease, with client anonymity suffering less compared to message anonymity. This highlights client anonymity's capability to offer a more realistic analysis, illustrating that a single honest mixnode along the message route can suffice for substantial anonymity. Such insights are crucial for practical design considerations and underscore the potential of deploying mixnets in applications without necessitating numerous intermediary hops to augment message anonymity, as it may not accurately represent the true anonymity afforded to clients.

Considering that mixnet anonymity primarily depends on client anonymity, providing high anonymity in scenarios with fewer clients can be challenging. To address this, we propose mixnodes with multimixing properties (inspired by recent work [1]) offering different levels of functionality in a Loopix-based mixnet. Specifically, we suggest that the Loopix mixnet should handle different types of traffic originating from various clients. For example, clients wanting to use mixnodes as a VPN can set the mixing delay to a minimum; clients using mixnodes for instant messaging can set the mixing delays to a moderate average; and clients utilizing the mixnet for latency-tolerant purposes can have higher delays for mixing. Despite these varied configurations, all clients can still use the same mixnet. Our analysis shows that such a setting can provide up to 150% higher anonymity for clients.

### 1.4 Outline

The remainder of this paper is organized as follows: In Section 2 we describe the client-based anonymity approach. In Section 3 we examine client and message anonymity through extensive experiments. In Section 4 we evaluate client anonymity in the presence of a mixnode adversary. In Section 5 we propose the multimixing mixnet. Finally, we conclude this work in Section 6.

## 2 SYSTEM MODEL AND OVERVIEW

In this section, we first overview the concept of message-level anonymity as a method for measuring the anonymity afforded to targeted messages in the presence of a global passive adversary, utilizing the terminology described in [5]. We then address the inaccuracies inherent in using message anonymity to evaluate the level of anonymity mixnets provide for their clients. To overcome these inaccuracies, we extend the concept of client anonymity pioneered by [9, 25] to the Loopix mixnet. This concept is further elaborated upon through the introduction of Algorithm 1.

### 2.1 Anonymity of Target Messages

Following the introduction of message anonymity as discussed in [11], we explore a scenario depicted in Fig. 2a, which illustrates clients sending messages through a stop-and-go mix. In this setup, a global passive adversary aims to identify the outgoing counterpart of a specific target message, $m_t$, as it enters the mix. Upon $m_t$'s entry into the mixnode, every message exiting the mixnode thereafter is regarded as a potential target by the adversary. Initially, the probability of an outgoing message being $m_t$ is $\frac{1}{N}$, where $N$ represents the total number of messages in the pool at the time of $m_t$'s entry. As the mixnode processes and releases messages, the likelihood of $m_t$ remaining inside adjusts. For instance, the probability for the second outgoing message to be $m_t$ is $\frac{1}{N'}(1 - \frac{1}{N})$, with $N'$ denoting the current message count inside the mixnode.[1] Consequently, a probability is assigned to each outgoing message regarding its potential identity as $m_t$. The Shannon entropy is applied to this distribution of probabilities, denoted as $\mathsf{H}(m)$, to quantify the level of uncertainty—or entropy—associated with identifying $m_t$ among the outgoing messages.

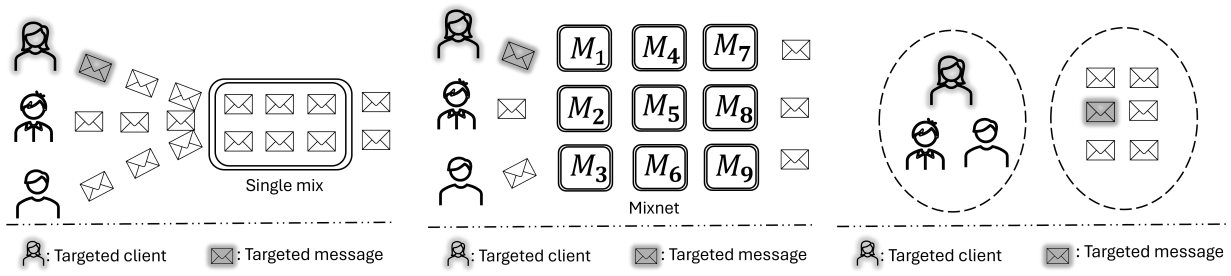
Conversely, Fig. 2b illustrates a scenario where clients leverage a mix network to anonymize their communications. In this case, a global passive adversary endeavors to correlate a specific message entering the mixnet with its potential outputs. In this scenario, [11] advocates for expanding the analysis beyond a single mixnode, proposing the calculation of the probability of a target message traversing through subsequent mixnodes within additional layers. Consequently, every message exiting the mixnet is assigned a probability reflecting its likelihood of being the target message. The measure of message anonymity is then determined by the entropy across the probability distribution of the target message among all exit messages. Using a discrete-event simulator to dynamically assess this entropy across various target messages provides a method to evaluate the anonymity of a target message afforded by the mixnet design [11].

### 2.2 Anonymity of Target Clients

Although the entropy of messages ($\mathsf{H}(m)$) serves as a useful metric for gauging the anonymity provided by a mixnet, it tends to overestimate this anonymity by assuming each client sends only one message. In practice, however, clients often send multiple messages to its destination, indicating that the foundational assumption of single-message transmissions by all clients is not practically viable. Additionally, an adversary's primary goal is likely to map clients using the mixnet to their corresponding destinations, implying that correlating input messages directly to output messages does not fully achieve this objective. For instance, as depicted in Fig. 2a, if three clients connect to a mixnode for message transmission, a global passive adversary would encounter difficulties in correlating each outgoing message to one of the three clients. Consequently, the number of messages sent to the mixnode does not inherently increase client anonymity.

Moreover, as illustrated in Fig. 2c, the concept of an anonymity set—integral to the anonymity afforded to clients—markedly diverges when solely considering message entropy. The anonymity set in the context of message anonymity encompasses all outgoing

[1] Considering that $m_t$ was not among previously exited messages.




**(a) Anonymity in a single mix scenario.** **(b) Anonymity in a mixnet scenario.** **(c) Anonymity sets.**

**Figure 2: This figure illustrates the anonymity guarantees provided to clients in two scenarios: first, when relying on a single mixnode for anonymity (as in traditional mixnets), and second, when using Loopix-like mixnets. Additionally, it highlights the distinction between the set of messages and the set of clients involved in the mixnet, showcasing that the anonymity set of messages is typically much larger than that of clients.**

messages subsequent to the entry of the target message into the mixnet. However, a more precise representation of the anonymity set should include the entire set of clients sending messages to the mixnet around the time of the target message's entry. This pivotal distinction highlights that relying solely on message anonymity could suggest a level of anonymity that is overly optimistic, potentially leading to erroneous conclusions. This observation underscores the necessity of considering client anonymity within mixnets.

To furnish a more precise and tangible evaluation of anonymity within a mixnet, we extend the concept of client anonymity based on entropy from prior works to Loopix mixnet. This approach entails assessing the probability that each outgoing message from the mixnet originates from any of the potential clients who have sent messages to the mixnet. To pursue this, we begin by formulating the anonymity of clients sending messages to a single stop-and-go mix. Consider a scenario akin to that depicted in Fig. 2a, where a global passive adversary targets a specific client. The adversary's goal is to determine which outgoing messages can be attributed to this client, thereby fully deanonymizing the client's connection to their destination.

Consider a scenario where $M$ clients are transmitting messages through a single stop-and-go mix. In this context, a global passive adversary meticulously monitors the inflow of clients' messages within the mixnode. We define $N_{C_i}$ as the count of messages from a particular client $C_i$ inside the mixnode, where $i$ ranges from 0 to $M$. Notably, $i = 0$ corresponds to messages that might serve as cover traffic, generated by the mixnode itself. Adversary aim is to determine the probability $\mathbb{P}(m_o \in C_i)$ that a specific message $m_o$ exiting the mixnode is associated with $i$th client.

Additionally, in the context of the Loopix-based mixnet, which employs stop-and-go mixes, traffic is reordered by imposing delays on messages following an exponential distribution with parameter $\lambda$. The crux of the analysis is to ascertain the likelihood that outgoing messages can be attributed to their originating clients. To quantify this, we consider the probability $\mathbb{P}_{\text{out}}$ of a message from client $C_i$ leaving the mixnode. Given $N_{C_i}$ messages from client $C_i$, each departs from the mixnode based on an independent exponential distribution ($\text{Exp}(\lambda)$). Consequently, we evaluate the probability $\mathbb{P}_{\text{out}}(\min(X_1, X_2, \ldots, X_{N_{C_i}}) < t)$ that at least one message from client $C_i$ exits within a time frame $t$. This calculation, detailed in Eq. (1), corresponds to the cumulative distribution function for an exponential distribution scaled by $N_{C_i}\lambda$, suggesting that on average, a message from client $C_i$ exits the mixnode every $\frac{1}{N_{C_i}\lambda}$ seconds.

To compute $\mathbb{P}(m_o \in C_i)$, we delineate two distinct events. The first, denoted as event $T_1$, encapsulates the probability that any message from client $C_i$ exits the mixnode within a time interval of length $t$. The second event, $T_2$, pertains to the departure of any message, excluding those from client $C_i$, from the mixnode within the same interval. Based on Eq. (1), $T_1$ is characterized by an exponential distribution with parameter $\lambda N_{C_i}$, and similarly, $T_2$ is defined by an exponential distribution with parameter $\sum_{\substack{j=0 \\ j\neq i}}^{M} \lambda N_{C_j}$.

By considering these parameters, the probability of a message from client $C_i$ exiting the mixnode within the interval $t$ can be computed as the probability that $T_1 < t$ while $T_2 > t$. Leveraging the properties of the exponential distribution, $\mathbb{P}(m_o \in C_i)$ can be articulated as depicted in Eq. (2), where the total number of messages, $N$, is the sum $\sum_{j=0}^{M} N_{C_j}$. This framework indicates that the likelihood of any message from client $C_i$ exiting the mixnode is directly proportional to the quantity of messages from $C_i$ within the mixnode, and inversely proportional to the aggregate number of messages within the mixnode.

$$
\begin{aligned}
&\mathbb{P}_{\text{out}}\left(\min(X_1, X_2, \ldots, X_{N_{C_i}}) < t\right) \\
&= 1 - \mathbb{P}_{out}\left(\min(X_1, X_2, \ldots, X_{N_{C_i}}) > t\right), \\
&= 1 - \prod_{j=1}^{N_{C_i}} \mathbb{P}_{out}\left(X_j > t\right), \\
&= 1 - \prod_{j=1}^{N_{C_i}} e^{-\lambda t} = 1 - e^{-\lambda N_{C_i} t}.
\end{aligned} \tag{1}
$$

For a single mixnode, Eq. (2) provides a robust means for accurately predicting the probability of each outgoing message's origin from a specific client. However, in the context of a Loopix mixnet, which comprises multiple layers of mixnodes, the dynamics of

message origin probabilities differ from those in a single mixnode scenario. In a Loopix architecture, the total number of mixnodes is $W \times L$, with each layer hosting $W$ mixnodes. A mixnode within layer $i$ is indexed from $(i-1)W+1$ to $iW$, for $i$ ranging from 1 to $L$ (illustrated in Fig. 2b).

$$
\begin{aligned}
&\mathbb{P}(m_o(t) \in C_i) = \mathbb{P}(T_1 < t) \wedge \mathbb{P}(T_2 > t),\\
&= \mathbb{P}_{\text{out}}(m_i \in C_i | T_1 \le t) \wedge \bigvee_{\substack{j=0\\ j\neq i}}^{M} \mathbb{P}_{\text{out}}(m_i \in C_j | T_2 > t),\\
&= \int_0^\infty \lambda N_{C_i} e^{-N_{C_i}\lambda x_1} \int_{x_1}^\infty \sum_{\substack{j=0\\ j\neq i}}^{M} \left(\lambda N_{C_j}\right),\\
&\qquad\qquad e^{-\sum_{\substack{j=0\\ j\neq i}}^{M}\left(\lambda N_{C_j}\right)x_2} \, \mathrm{d}x_1 \, \mathrm{d}x_2.\\
&= \int_0^\infty \lambda N_{C_i} e^{-N_{C_i}\lambda x_1} e^{-\sum_{\substack{j=0\\ j\neq i}}^{M}\left(\lambda N_{C_j}\right)x_1} \mathrm{d}x_1.\\
&= \frac{N_{C_i}}{N} \int_0^\infty N\lambda e^{-N\lambda x_1} \, \mathrm{d}x_1 = \frac{N_{C_i}}{N}.
\end{aligned} \tag{2}
$$

Within this structure, the count of messages from client $i$ in mixnode $j$ beyond the initial layer is represented by a random variable $N^j_{C_i}$. At any given time $t_0$, this variable equates to the number of messages received at mixnode $j$ from client $i$, subtracting those that have exited mixnode $j$ by $t_0$. For instance, if mixnode $j$ in the second layer is connected to a mixnode in the first layer, with incoming and outgoing messages denoted as $m_{\text{in}}$ and $m_{\text{out}}$, respectively, the expected number of messages in mixnode $j$ can be formulated as:

$$
\begin{aligned}
&= \sum_{\text{in}} \mathbb{P}(m_{\text{in}} \in C_i | t < t_0) - \sum_{\text{out}} \mathbb{P}(m_{\text{out}} \in C_i | t < t_0),\\
&= \sum_{\mathcal{S}_{in} | t<t_0} \frac{N_{C_i}(t)}{N(t)} - \sum_{\mathcal{S}^j_{out} | t<t_0} \frac{N^j_{C_i}(t)}{N^j(t)}.
\end{aligned} \tag{3}
$$

Note that in Eq. (3), $\mathcal{S}_{\text{in}}$ denotes the set of all incoming messages to the second layer of the mixnet; in other words, it represents the complete set of messages sent out from the first layer of the mixnet up to time $t$. Similarly, $\mathcal{S}^j_{\text{out}}$ refers to the set of outgoing messages from mixnode $j$ in the second layer up to time $t$.

Assuming the total number of messages entering the second layer at time $t$ is $N(t)$, the count of incoming messages from client $C_i$ to mixnode $j$ can be approximated by $\frac{1}{W} \times \frac{N_{C_i}(t)}{N(t)}$ over the set $\mathcal{S}_{\text{in}}$, where $W$ denotes the number of nodes in the second layer, and $N_{C_i}(t)$ is the total number of messages originating from client $C_i$ up to time $t$. The probability of a message from $C_i$ exiting mixnode $j$ is then given by $\frac{N^j_{C_i}(t)}{N^j(t)}$, where $N^j_{C_i}(t)$ denotes the number of messages from $C_i$ exiting node $j$ by time $t$, and $N^j(t)$ denotes the total number of messages exiting node $j$ up to time $t$. Summing these probabilities over all messages allows the estimation of the average number of messages from client $C_i$ processed by mixnode $j$ at time $t$.

Furthermore, to meticulously analyze the probability distribution of messages from a specific client exiting the mixnet, one must utilize a discrete-event simulator. This simulation collects data on each mixnode's status, facilitating the estimation of the probability that outgoing messages originate from any client interacting with the mixnet. Additionally, the entropy of each probability distribution of exiting messages constitutes a measure for assessing the mixnet's anonymity, denoted as $\mathsf{H}(C)$. The subsequent section details an algorithm developed to compute this anonymity metric using a discrete-event simulator.

## 2.3 Client Anonymity Algorithm

To accurately assess the anonymity provided by a mixnet, establishing a framework that incorporates a discrete-event simulator for evaluation is crucial. We present Algorithm 1, outlining the methodology for dynamically updating the status of mixnodes. This approach enables the calculation of the probability distribution for each outgoing message, indicating the chance of it originating from any given client. By analyzing this data, we can determine the entropy of these probability distributions, thus gauging the average level of client anonymity afforded by the mixnet.

In a scenario where a global passive adversary monitors clients sending packets to the mixnet, initially, each message $k$ entering the mixnet is assigned a probability distribution $\mathbb{P}(\text{msg}_k)$, which is set to 0 for all clients, except for the client $i$ who sent the message, where it is set to 1. As messages reach the first layer's mixnodes, the adversary updates the record of how many messages each client has in these mixnodes. When a message exits a mixnode from the first layer, its probability distribution is updated according to Eq. (2). The updated distribution, $\mathbb{P}(\text{msg}_k \in C_i)$, becomes proportional to the client's message count at the time of exit, expressed as $\frac{N_{C_i}(t)}{N(t)}$, with $t$ representing the exit time. Simultaneously, the adversary adjusts the message counts for each client within the mixnodes, updating $N^j_{C_i}$ accordingly: $N^j_{C_i} = N^j_{C_i} - \mathbb{P}(\text{msg}_k \in C_i)$.

This procedure is consistently applied to all messages as they navigate through each mixnode within the mixnet. When a message exits the mixnet, its probability distribution—reflecting the potential origins from any client—serves as a measure of clients anonymity. To quantitatively assess client anonymity, we calculate the entropy of this probability distribution ($\mathsf{H}(C)$). Specifically, we run a discrete-event simulator over a prolonged period to gather a broad set of entropy values. The average of these entropy measurements offers an estimate of the overall anonymity provided to clients by the mixnet.

## 2.4 Computation Complexity

Our goal for assessing client anonymity in mixnets is to evaluate, through simulations, the origins of each message entering the mixnet. Specifically, for every message, we track its probability of being generated by any client and update this probability as the message traverses each mixnode along its path.

We note, however, that the computational complexity of running such an anonymity evaluation is essentially at the same level as the message anonymity evaluation introduced in [2, 13], where the prior probability distribution of each outgoing message is updated

**Algorithm 1:** Client's Anonymity in Mixnet

**Input:** Targeted clients sending messages to the mixnet.
**Output:** Client's Anonymity $\mathsf{H}(C)$.
**for** *j in range($L \times W$)* **do**
  **for** *each message k entering mix j* **do**
    **for** $C_i$ *in* $M$ **do**
      Update $N^j_{C_i}$: $N^j_{C_i} = N^j_{C_i} + \mathbb{P}(\text{msg}_k \in C_i)$;
      Update $\mathbb{P}(\text{msg}_k \in C_i) = \frac{N^j_{C_i}}{N^j}$ ;
      **after leaving the mix**;
      Update $N^j_{C_i} = N^j_{C_i} - \mathbb{P}(\text{msg}_k \in C_i)$;
    **end**
  **end**
**end**
**for** *j in range($W$)* **do**
  **for** *each message k exiting mix j in last layer* **do**
    Calculate $\mathsf{H}^k(C) = \text{Entropy}(\mathbb{P}(\text{msg}_k))$;
  **end**
**end**
Compute $\mathsf{H}(C) = \text{mean}(\mathsf{H}^k(C))$.

based on its possible matching to tagged messages entering the mixnet.

Thus, both approaches involve maintaining and updating a probability vector for each message at every mixnode along its path. Nonetheless, the memory requirements of our client anonymity evaluation are significantly lower compared to [2, 13], as the number of clients is typically much smaller than the number of messages.

## 3 EVALUATION

In this section, we assess the client anonymity metric extended for Loopix-like mixnets in this paper. For this purpose, we simulate a mix network using the discrete-event simulator *SimPy* [15] in *Python*. We analyze the anonymity offered by a single mixnode to clients, as well as the anonymity provided by a Loopix mixnet with $L = 3$ layers. The configuration includes $W = 32$ mixnodes per layer and $M = 128$ clients. Messages are generated and enter the mixnode according to a Poisson distribution with an average interval of $\frac{1}{\lambda} = \frac{1}{10}$ ms. Within each mixnode, messages undergo reordering based on an exponential delay with an average of $\mu = 50$ ms, following the parameters used in the real NYM mixnet deployment [6]. We evaluate anonymity by initially exploring the message anonymity concept proposed by [11], considering 200 messages, and subsequently contrasting it with our proposed client anonymity scheme.

To ensure a consistent level of activity at each mixnode and thereby guarantee anonymity, we incorporate a burn-in period during which a substantial volume of messages is transmitted to maintain a minimum message count at each mixnode. The absence of messages implies the absence of anonymity. We emphasize that experiment parameters, unless otherwise noted, are maintained constant across all evaluations. Furthermore, within the Loopix mixnet configuration, we assume that clients uniformly select mixnodes at random at each layer for routing their messages. This ensures that each client has an equal probability of choosing any mixnode from each layer, denoting random routing.

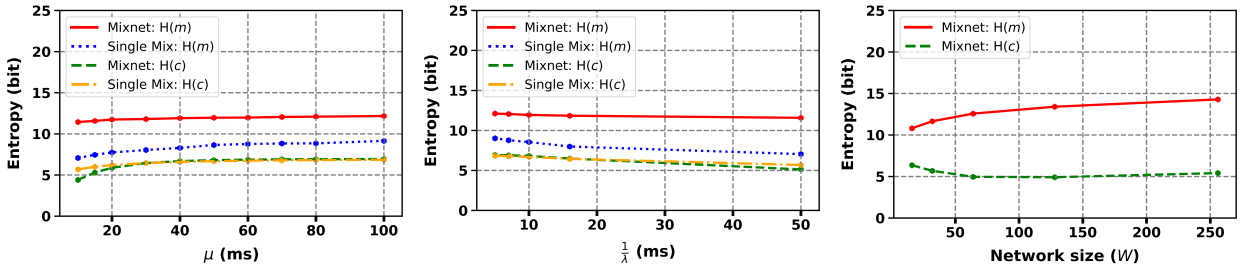
Lastly, note that for statistical significance, we perform evaluations over multiple instantiations of the mixnet (at least 400 runs) to minimize variance and ensure the robustness of results. Furthermore, we apply standard outlier removal techniques to eliminate any faulty measurements. As a result, the reported performance metrics in all our evaluations are tightly concentrated around their true mean values, even if we do not explicitly plot the variance in every figure.

### 3.1 Impact of Mixnet on Anonymity

Our preliminary studies investigate the effects of mixnet parameters on the anonymity provided to messages and clients. Fig. 3a illustrates the anonymity levels for target messages, denoted as $\mathsf{H}(m)$, and target clients, denoted as $\mathsf{H}(C)$, in both individual mix and mixnet configurations, with a focus on the impact of varying the mixing delay $\mu$. It is consistently observed that the anonymity afforded to messages exceeds that provided to clients, regardless of the specific value of $\mu$. More precisely, within a mixnet environment, the anonymity level for messages is found to be quantitatively 5 bits greater than that for clients. Conversely, in a scenario involving a single mix, this difference is reduced to approximately 2 bits. This differential highlights the limitation of relying solely on message anonymity to gauge the overall efficacy of a mixnet in preserving privacy, demonstrating that significant aspects of client anonymity are not adequately captured by message anonymity metrics alone.

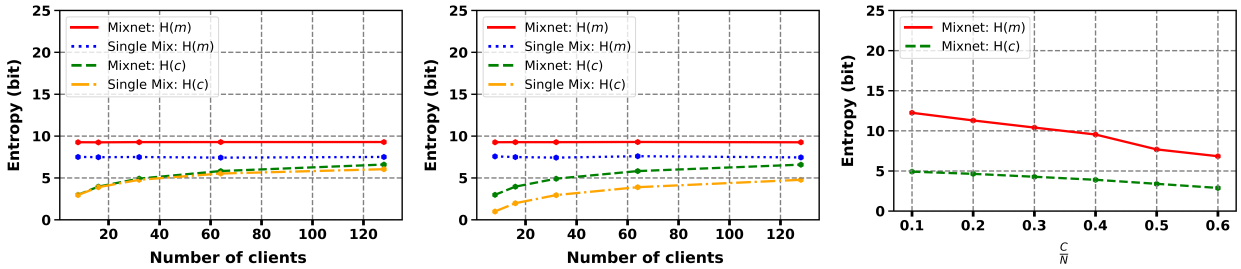
Moreover, increasing the mixing delay $\mu$ invariably enhances anonymity across both messages and clients. A prolonged mixing delay allows for extended mixing periods within mixnodes, increasing the potential for messages to be combined with a larger pool, thereby elevating message anonymity. For clients, an increased message volume enhances the probability of more client messages being present in the mix, thereby involving more clients in the mixing process and enhancing client anonymity.

Additionally, our findings reveal that the message anonymity ($\mathsf{H}(m)$) provided by a single mixnode is consistently outperformed by the enhanced message anonymity observed within a mixnet framework. This enhancement is attributable to messages undergoing additional layers of mixing, where they encounter a broader diversity of messages, thereby elevating their anonymity. In contrast, client anonymity in a single mix scenario demonstrates a marginal difference when compared to its counterpart in a mixnet setting. This minimal disparity arises because client anonymity fundamentally depends on the diversity of clients contributing messages to the mixnet's initial layer. The practice of clients uniformly selecting mixnodes for their message routing effectively complicates the task of a global passive adversary in correlating messages to their senders, even within the confines of a single mixnode scenario. This observation reveals the necessity of additional intermediary hops for amplifying message anonymity but also points to their limited impact on augmenting client anonymity. Such a distinction underscores the limitations inherent in relying solely on message anonymity as a metric for designing mixnets,




(a) Entropy vs. mixing delay. (b) Entropy vs. rate of input traffic. (c) Entropy vs. size of mixnet.

**Figure 3: Comparative analysis of message and client anonymity in mixnets.**




(a) Random selection. (b) Biased selection. (c) Mixnode adversary.

**Figure 4: This figure illustrates the anonymity guarantees for both messages and clients in two scenarios: (i) when anonymity relies on a single honest mixnode, and (ii) when utilizing a Loopix-like mixnet. The evaluation is conducted under two node selection strategies: uniform random selection and biased selection. Additionally, the figure highlights how the anonymity of messages and clients can be impacted by a mixnode adversary capable of corrupting a subset of nodes within the mixnet.**

emphasizing the critical need for considering client anonymity as a more representative measure for evaluating the efficacy of mixnets in preserving the anonymity of its users.

Fig. 3b illustrates the impact of varying the input traffic rate ($\lambda$) on message and client anonymity in both single mix and mixnet scenarios. An increase in the inter-arrival time ($\frac{1}{\lambda}$) leads to a reduction in anonymity, as fewer messages and, consequently, fewer clients contribute to the mix, diminishing both message and client anonymity. This experiment reinforces the notion that message anonymity, being higher than client anonymity, serves as an upper bound but may not precisely reflect the mixnet's effectiveness in preserving client anonymity.

Fig. 3c illustrates the impact on message and client anonymity within a Loopix mixnet scenario as we vary the number of mixnodes per layer ($W$), incrementing from 8 to 256. In this case, maintaining an equal traffic rate across each mixnode is crucial, as imbalances could undermine the anonymity provided. To this end, the input traffic rate is calibrated proportionally to the size of the network. The findings reveal that increasing $W$ notably enhances message anonymity. This is attributed to a larger pool of messages, which ensures that any specific target message is mixed with a higher number of other messages, thus elevating its anonymity. In contrast, the overall effect of changes in $W$ on client anonymity is minimal,[2] showcasing a disparity between the effects on message and client anonymity. This observation suggests that merely adding more mixnodes per layer does not significantly benefit overall anonymity. It further highlights the limitations of using message anonymity as a sole metric for mixnet design, indicating that practical applications may not derive substantial advantage from such adjustments.

## 3.2 Impact of Client Set on Anonymity

Our subsequent experiments investigate the influence of the client count on anonymity within the mixnet, a factor that stands orthogonal to the parameters of the mixnet itself. For example, if only one client utilizes the mixnet (or a single mixnode), the identity of that client becomes transparent, regardless of the mixnet's configuration. To explore how the client count affects anonymity, we conducted experiments depicted in Fig. 4, assessing message and client anonymity in both single mix and mixnet scenarios.

In Fig. 4a, we consider a scenario where clients select mixnodes uniformly at random, allowing for the possibility that each client sends messages to any mixnode within the network. Contrastingly, Fig. 4b depicts a scenario where each mixnode is chosen by a fixed number of clients, such as clients selecting the nearest mixnode to their location.

The results, as shown in both Fig. 4a and Fig. 4b, indicate that message anonymity remains constant irrespective of the client count. This constancy suggests that an increase in clients does not

[2]It is important to note that when the network size ($W$) is exceptionally small, client anonymity marginally increases. This effect arises because a smaller network size enhances the probability of clients selecting the same mix node, thereby facilitating a higher degree of clients' message intermingling among different clients.

inherently alter the traffic rate, thus highlighting the limitations of using message anonymity as a metric for accurately quantifying anonymity. Conversely, client anonymity is observed to increase with the number of clients in both the uniform and biased selection scenarios. The increase in client anonymity is attributed to a broader pool of potential message sources, thereby enhancing the adversary's confusion and, consequently, the level of anonymity.

Additionally, the uniform routing through the mixnet does not markedly differentiate the client anonymity offered by a single mixnode from that of a mixnet. A single mixnode, when chosen by a sufficient number of clients, already ensures an adequate distribution of message probabilities across all clients. Routing messages through additional mixnodes in a mixnet marginally increases anonymity. However, when clients preferentially select the closest mixnode (Fig. 4b), the anonymity provided by a single mixnode is comparatively lower than that in a mixnet scenario. This reduction in anonymity is due to the limited number of clients connected to a single mixnode, which narrows the pool of clients associated with each message and decreases anonymity. In contrast, within a layered mixnet, messages from the first layer might be linked to a few clients, but subsequent layers mix messages from more diverse client locations, suggesting that each message could originate from a larger set of clients. This observation suggests that in scenarios involving biased selection, client anonymity is enhanced by adding more layers to the mixnet. This underscores the significance of a layered approach in mixnets for bolstering client anonymity.

## 4 MIXNODE ADVERSARY

In this section, we delve into a scenario featuring a mixnode adversary, representing a more formidable threat than a global passive adversary. This adversary, with finite resources to control portions of the mixnet [6], endeavors to deanonymize client communications. Unlike the global passive adversary, the mixnode adversary has the capability to directly link the inputs to the outputs of the mixnodes under its control. Working in tandem with the global adversary, it seeks to maximize its influence over the network traffic, potentially leading to a greater risk of client deanonymization.

The mixnode adversary corrupts mixnodes across various layers, determined by its budget, and chooses which mixnodes to corrupt uniformly at random. Suppose the adversary is capable of corrupting $\alpha$ percent of mixnodes within each layer. Then, it would effectively control $\alpha^L$ percent of the entire network's traffic, thereby gaining the ability to unequivocally associate clients with their destinations for that portion of the traffic. This scenario presents a worst-case analysis of the potential damage such an adversary can inflict.

To address and quantify the impact of the mixnode adversary on client anonymity within the mixnet, this paper employs Algorithm 1. This algorithm facilitates the analysis of client anonymity even in the presence of compromised mixnodes. However, it is assumed that once a mixnode is corrupted, the adversary can directly correlate its input messages with their outputs, meaning these compromised nodes no longer contribute to the calculation of client anonymity ($\mathsf{H}(C)$), rendering their impact on enhancing anonymity null.

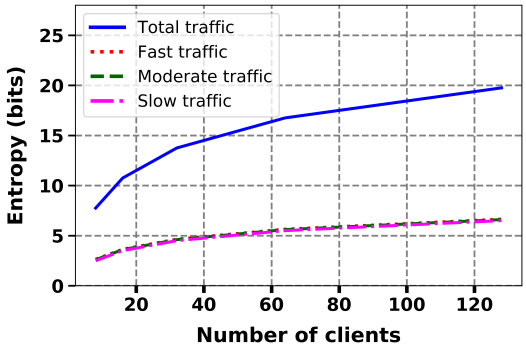



**Figure 5: Improved client anonymity ($\mathsf{H}(C)$) using multimixing methods.**

To analyze client anonymity in conjunction with anonymity that is provided for messages in the context of a mixnode adversary, we initially adopt the general parameter settings mentioned in Section 3. Our investigation also explores varying budgets allocated for the adversary's capability to corrupt mixnodes. We denote $C$ as the count of mixnodes compromised by the adversary, resulting in $\frac{C}{N}$ as the fraction of the mixnet under adversarial control. Utilizing discrete event simulations, we assess both message and client anonymity under these conditions.

Fig. 4c showcases the results from our discrete event simulations concerning both message and client anonymity. It becomes evident that as the fraction of corrupted mixnodes increases within the mixnet, there is a decline in both the entropy of messages ($\mathsf{H}(m)$) and the entropy of clients ($\mathsf{H}(C)$). This trend is primarily due to the diminishing pool of honest mixnodes capable of effectively mixing messages or clients.

A key observation is that message anonymity suffers a more pronounced reduction compared to client anonymity with the increase in corruption. This disparity arises because message anonymity is intricately linked to the interactions a message has with honest mixnodes. Consequently, a decrease in honest mixing opportunities leads to a reduction in message anonymity. On the contrary, while an increase in corrupted mixnode fraction similarly affects client anonymity, the distribution of output messages among all potential clients can still be somewhat maintained by the remaining honest nodes.

Remarkably, client anonymity can be preserved to a considerable extent by even a solitary honest mixnode along the communication path. This insight underscores that client anonymity remains relatively resilient against mixnode adversaries unless the adversary compromises the entire path. Therefore, maintaining at least one honest mixnode in the routing chain can still ensure a substantial degree of anonymity, highlighting the practical importance of employing client anonymity ($\mathsf{H}(C)$) as a metric for practical analysis.

## 5 IMPROVING CLIENT ANONYMITY

As observed so far, clients within mixnets, regardless of the mixnet type, enjoy a degree of anonymity; however, the anonymity at the client level is notably weaker than the anonymity measured at the message level. This discrepancy may discourage the adoption of

mixnets in certain scenarios, particularly when the use case is limited or when an adversary can control a subset of clients—thereby increasing the risk of deanonymization. To address this issue, in this section we provide insights and methods aimed at enhancing client anonymity in mixnets, especially considering the risk that adversaries compromising multiple clients can more easily deanonymize honest client-destination communications.

To address these cases, we highlight recent work proposing beta mixing in mixnets [1]. This work shows that mixnets typically cater to a single type of traffic, meaning the mixing process characteristics applied to the traffic are specific to that type. However, this study demonstrates that adopting a mixnet to carry two different traffic types with distinct mixing characteristics can enhance the anonymity of messages.

We propose a similar strategy but suggest considering multiple traffic types, each with distinct mixing characteristics, instead of just two specific types of traffic as suggested by [1]. We argue that in such a case, each client will be mixed with a larger group of clients, thereby extending the functionality of the mixnet from serving a single group of clients to multiple groups, ultimately providing much stronger anonymity for clients.

To analyze our proposal, we conducted an experiment involving three types of traffic: slow traffic, which should be mixed with a higher mixing delay; moderate traffic, mixed with a moderate delay; and fast traffic, mixed with a minimal delay. Specifically, we considered average mixing delays of 150 ms, 50 ms, and 5 ms, respectively, for slow, moderate, and fast traffic. We then assessed the impact of this multimixing proposal on client anonymity.

The results of our experiments are shown in Fig. 5. As demonstrated, when the fraction of generated messages tied to each traffic type is roughly the same, the anonymity provided by the mixnet is relatively uniform. However, when employing a multimixing scenario, the overall anonymity enjoyed by each client is almost three times higher, showing the effectiveness of the multimixing strategy in improving client anonymity.

## 6 CONCLUSION

In this paper, we extended the concept of client anonymity from prior works to Loopix-like mixnets. Our examination suggested that message-level anonymity metrics may not fully encompass the scope of protection offered by mixnets, potentially leading to an overestimated perception of anonymity. Through various experiments, we showed that client anonymity, despite generally being lower than message anonymity, offers a more accurate and robust measure of anonymity. This is particularly evident in mixnet configurations that incorporate at least one honest intermediary mixnode committed to performing message mixing properly. Additionally, we proposed a multimixing scenario to address situations where the number of clients per traffic type is limited. Our results demonstrated a promising improvement in client anonymity. We hope that future work utilizing these insights will lead to more realistic and practical evaluations of mixnets, enhancing their usability in real-world applications.

## ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their valuable feedback. This work was supported in part by CyberSecurity Research Flanders with reference number VOEWICS02.

## REFERENCES

[1] Iness BEN GUIRAT, Debajyoti Das, and Claudia Diaz. 2023. Blending different latency traffic with beta mixing. *Proceedings on Privacy Enhancing Technologies* (2023).
[2] Iness Ben Guirat, Devashish Gosain, and Claudia Diaz. 2021. Mixim: Mixnet design decisions and empirical evaluation. In *Proceedings of the 20th Workshop on Workshop on Privacy in the Electronic Society*. 33–37.
[3] David Chaum, Debajyoti Das, Farid Javani, Aniket Kate, Anna Krasnova, Joeri De Ruiter, and Alan T Sherman. 2017. cMix: Mixing with minimal real-time asymmetric cryptographic operations. In *Applied Cryptography and Network Security: 15th International Conference, ACNS 2017, Kanazawa, Japan, July 10-12, 2017, Proceedings 15*. Springer, 557–578.
[4] David L Chaum. 1981. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM* 24, 2 (1981), 84–90.
[5] Claudia Diaz. 2005. Anonymity and privacy in electronic services. *Heverlee: Katholieke Universiteit Leuven. Faculteit Ingenieurswetenschappen* (2005).
[6] Claudia Diaz, Harry Halpin, and Aggelos Kiayias. 2021. The Nym Network. (2021).
[7] Claudia Diaz, Steven J Murdoch, and Carmela Troncoso. 2010. Impact of network topology on anonymity and overhead in low-latency anonymity networks. In *Privacy Enhancing Technologies: 10th International Symposium, PETS 2010, Berlin, Germany, July 21-23, 2010. Proceedings 10*. Springer, 184–201.
[8] Claudia Díaz, Len Sassaman, and Evelyne Dewitte. 2004. Comparison Between Two Practical Mix Designs, Pierangela Samarati, Peter Y. A. Ryan, Dieter Gollmann, and Refik Molva (Eds.).
[9] Claudia Diaz, Stefaan Seys, Joris Claessens, and Bart Preneel. 2003. Towards measuring anonymity. In *Privacy Enhancing Technologies: Second International Workshop, PET 2002 San Francisco, CA, USA, April 14–15, 2002 Revised Papers*. Springer, 54–68.
[10] Michael J Freedman and Robert Morris. 2002. Tarzan: A peer-to-peer anonymizing network layer. In *Proceedings of the 9th ACM Conference on Computer and Communications Security*. 193–206.
[11] Iness Ben Guirat and Claudia Diaz. 2022. Mixnet optimization methods. *Proceedings on Privacy Enhancing Technologies* (2022).
[12] Dogan Kesdogan, Jan Egner, and Roland Büschkes. 1998. Stop-and-go-mixes providing probabilistic anonymity in an open system. In *International Workshop on Information Hiding*. Springer, 83–98.
[13] Ania M Piotrowska. 2021. Studying the anonymity trilemma with a discrete-event mix network simulator. In *Proceedings of the 20th Workshop on Workshop on Privacy in the Electronic Society*. 39–44.
[14] Ania M Piotrowska, Jamie Hayes, Tariq Elahi, Sebastian Meiser, and George Danezis. 2017. The loopix anonymity system. In *26th USENIX Security Symposium (USENIX Security 17)*. 1199–1216.
[15] Python. 2013. Event discrete, process based simulation for Python. https://pypi.org/project/simpy/.
[16] Mahdi Rahimi. 2024. CLAM: Client-Aware Routing in Mix Networks. In *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security, IH&MMSec 2024, Baiona, Spain, June 24-26, 2024*. ACM, 199–209. https://doi.org/10.1145/3658664.3659631
[17] Mahdi Rahimi. 2024. LARMix ++ : Latency-Aware Routing in Mix Networks with Free Routes Topology. *Cryptology ePrint Archive* (2024).
[18] Mahdi Rahimi. 2024. LARMix++: Latency-Aware Routing in Mix Networks with Free Routes Topology. In *International Conference on Cryptology and Network Security*. Springer, 187–211.
[19] Mahdi Rahimi. 2024. MALARIA: Management of Low-Latency Routing Impact on Mix Network Anonymity. In *2024 22nd International Symposium on Network Computing and Applications (NCA)*. IEEE, 193–202.
[20] Mahdi Rahimi. 2025. MALARIA: Management of Low-Latency Routing Impact on Mix Network Anonymity (Extended Version). *Cryptology ePrint Archive* (2025).
[21] Mahdi Rahimi. 2025. PARSAN-Mix: Packet-Aware Routing and Shuffling with Additional Noise for Latency Optimization in Mix Networks. In *International Conference on Applied Cryptography and Network Security*.
[22] Mahdi Rahimi. 2025. PARSAN-Mix: Packet-Aware Routing and Shuffling with Additional Noise for Latency Optimization in Mix Networks (Extended Version). *Cryptology ePrint Archive* (2025).
[23] Mahdi Rahimi, Piyush Kumar Sharma, and Claudia Diaz. 2024. LARMix: Latency-Aware Routing in Mix Networks. In *The Network and Distributed System Security Symposium*. Internet Society.

[24] Mahdi Rahimi, Piyush Kumar Sharma, and Claudia Diaz. 2025. LAMP: Lightweight Approaches for Latency Minimization in Mixnets with Practical Deployment Considerations. In *The Network and Distributed System Security Symposium*. Internet Society.

[25] Andrei Serjantov and George Danezis. 2003. Towards an information theoretic metric for anonymity. In *Privacy Enhancing Technologies: Second International Workshop, PET 2002 San Francisco, CA, USA, April 14–15, 2002 Revised Papers 2*. Springer, 41–53.

[26] Claude E Shannon. 1949. Communication theory of secrecy systems. *The Bell system technical journal* 28, 4 (1949), 656–715.

[27] Fatemeh Shirazi, Milivoj Simeonovski, Muhammad Rizwan Asghar, Michael Backes, and Claudia Diaz. 2018. A survey on routing in anonymous communication protocols. *ACM Computing Surveys (CSUR)* 51, 3 (2018), 1–39.