How to apply Database Anonymity Notions to Mix Networks

Marc Roßberger[1], Alperen Aksoy[2], Doğan Kesdoğan[3]

Abstract: Communication networks are an indispensable part of our society. By observing network traffic, one can acquire sensitive information about individuals, businesses, or governments. Thus, the protection of this traffic data is important and a well-researched topic. While encryption can protect the content of these messages, it can not hide the link between communication partners. Mix networks can be used to achieve this anonymity goal. However, these networks can not guarantee anonymity since they only apply technical protocols to improve anonymity. On the contrary, one can guarantee anonymity when releasing information from databases. Here, the principles of k-anonymity, l-diversity, and t-closeness can be applied to determine anonymity. In this work, we transfer these principles towards network anonymity, highlight problems that occur, and give an outlook on how a protocol could look like, with which users can build anonymity groups to guarantee anonymity.

Keywords: anonymity; mix networks; data privacy

1 Introduction

One of the biggest problems in our society is preventing the unauthorised creation of profiles on the Internet. The most basic defense against unauthorized profiling is the careful deployment of cryptographic techniques, which guarantees the privacy of exchanged messages. At the network level, however, a message's address information attributes it to both sender and receiver. Cryptography cannot hide this address information. Consequently, a network operator or intruder can read and collect a user's interactions with which to derive user-specific profiles. Thus the foundation of digital privacy is the network anonymity.

Although network anonymity has a few decades of experience, database anonymity has a solid foundation. In this paper, we ask ourselves at the definition and notions level how these two areas can be brought together. A fruitful discussion and exchange of notions and techniques could enrich both fields.

Contribution We are working on a new approach to measure and improve anonymity in mix networks based on metrics known from the field of database anonymity. For this purpose, we make a novel comparison between the two issues. In particular, we contribute with:

[1] University of Regensburg, Chair for Business Informatics IV, Universitätsstraße 31, 93053 Regensburg, Germany, marc.rossberger@ur.de

[2] Friedrich-Alexander-University Erlangen-Nürnberg, Computer Science Dept. Chair 1, Martensstrasse 3, 91058 Erlangen, Germany, alperen.aksoy@fau.de,
Thanks to the Turkish ministry of education for granting his Ph.D. studies.

[3] University of Regensburg, Chair for Business Informatics IV, Universitätsstraße 31, 93053 Regensburg, Germany, dogan.kesdogan@ur.de

- an overview of anonymity goals in database anonymity and a suggestion on how to apply them to network anonymity,
- an analysis, why current mix networks can not fulfill these objectives, and
- an outlook of a possible protocol to meet these requirements.

Structure This paper is structured as follows. Section 2 gives an overview of related work. In section 3, we present the basics about mix networks and cover traffic. Afterward, section 4 introduces database anonymity and demonstrates how its principles can be applied to network anonymity. We conclude this work in section 5 by giving an outlook to future work.

2 Related Work

Anonymous communication is classified into two categories. These are high-latency and low-latency. Message-based systems are considered to be high-latency [EY09]. One of the well-known high-latency systems is the mix introduced by Chaum [Ch81]. Until 2004, several anonymous communication systems based on the mix were popular such as anon.penet.fi, cyberpunk remailers, and mixminion [DDM03; EY09]. After the release of TOR, which provides low-latency anonymization based on onion routing, mix-based systems went out of fashion because sending a message over mix-based systems can take longer times, even hours or days. In 2017, Pietrowska et al. revealed the Loopix system, which provides lower latency communication based on Mix networks by using cover traffic [Pi17]. In addition, the Loopix system is resistant to Sybil attacks, for which TOR has a vulnerability. Thus, Loopix shows that mix-based systems are able to compete with TOR-based systems.

By using database anonymization techniques, sensitive data can be protected from disclosure while still allowing statistical conclusions. Thus, these techniques hide the link between people and sensitive data. Anonymous communication networks have a similar purpose and hide the link between sender and recipient. To our knowledge, no study in the literature has investigated truly whether database anonymity techniques can be applied to anonymous communication networks in deep. In this work we will apply the principles of database anonymity to communication networks. This analysis also highlights how existing anonymization strategies can be used to try and achieve these goals and shows their flaws.

3 Mix Networks

Mix networks are systems that allow their participants to communicate anonymously by employing a special routing protocol. Most practical anonymous communication networks are inferred from the **Chaum Mix** [Ch81], also known as **Threshold Mix**. Its fundamental functionality is illustrated in figure 1. Here, 4 participants want to send their messages to 4 recipients. Both the sender set S and the recipient set R are part of a global set of users.

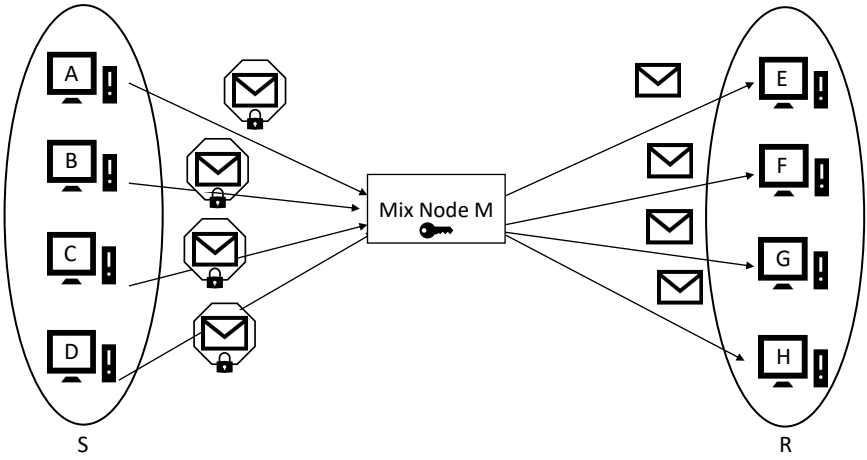

Fig. 1: One round of a Threshold Mix. The mix node decrypts and shuffles the messages. ($n = 4$)

To prevent possible attackers from observing the communication relationships between senders and recipients, they send their messages to a mix node M, encrypted with M's public key. M then decrypts these messages, checks for duplicates to prevent replay attacks, and waits until it has received n messages, where n is its batch size. Once its threshold is reached, M shuffles the messages and relays them to their recipients. This protocol requires the messages to have a common size and appearance to prevent the linkage of messages. This is achieved, e.g., by the sphinx packet format [DG09].

A message can be routed through multiple mix nodes to improve anonymity. As long as at least one mix node on its path is honest, an attacker can not disclose the relationship between sender and receiver. Different types of mixes try to improve anonymity even further, e.g., by introducing indeterminism. One example is a **Pool Mix**, which collects $n + l$ messages but only sends l messages per batch. Another alternative is the **Stop-and-Go Mix**, which keeps messages for a randomly chosen time delay defined by the sender.

3.1 Anonymity Goals

Most mix network anonymity goals can be described using the terminology by Pfitzmann and Köhntopp [PK01]. The **sender-recipient unlinkability** describes that an attacker cannot determine which sender communicated with which recipient. If an attacker observes a message arriving at a receiver, his inability to determine the message's sender is described by **sender anonymity**. The list of all possible senders, which the attacker cannot differentiate, forms the **sender anonymity set**. Analogously, **recipient anonymity** and **recipient anonymity set** are defined. Further goals include **sender** (recipient) **unobservability**, which defines that an attacker cannot determine when a sender (recipient) is sending (receiving) a

message. How cover traffic can be employed to achieve these goals will be shown in the next section.

3.2 Cover traffic

One of the well-known ways to increase the anonymity set size is the usage of cover traffic. It protects messages from disclosure by covering them with dummy messages, making it difficult to track a particular message. Using cover traffic is a useful but costly application to prevent or delay successful disclosure attacks. According to Malleshwright et al. [MW07], cover traffic is beneficial under the conditions that (i) cover messages must be indistinguishable from real messages until they reach their final destination, (ii) dummy messages should be tolerable by receivers because the decryption phase does cost operations, and (iii) the cost of cover traffic should not exceed the limits of the mixes and receivers.

Cover traffic can be used for several purposes. Some of them are sender-recipient unlinkability, sender/recipient unobservability, detection of corrupt nodes, and reduced delays. Cover traffic can be handled in four ways. These are displayed in figure 2.

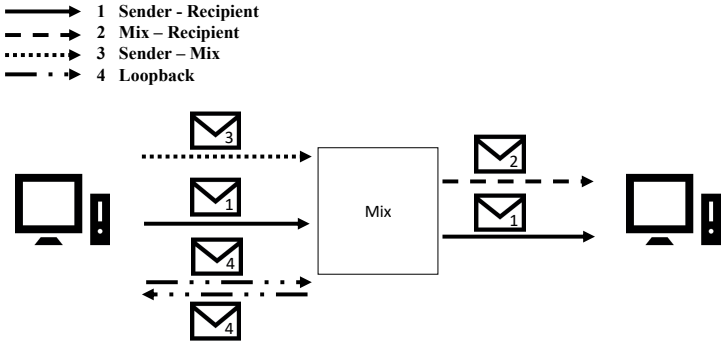

Fig. 2: Types of cover traffic

In the scope of sender-recipient cover traffic, dummy packets are created by the sender and reach the recipient via the mix network. The attacker cannot distinguish these packets from regular packets. Using this method makes a significant contribution to delay the success of disclosure attacks.

Another type of cover traffic is dummy packets created by mix nodes and sent to end-users. Although an attacker can notice a difference in count between packets arriving and leaving the mix, as long as the cover message's frame matches a real message's frame, he cannot distinguish cover packets from real packets. This method is convenient to increase the recipient anonymity set size. However, this method does not increase the sender anonymity set size.

Thirdly, dummy packets are created by a user and discarded by the mix, i.e. it does not increase the recipient anonymity set size. This method is only useful if every sender attends communication in every round. If senders were offline in some periods, an attacker might reveal the recipient set of a specific sender by comparing the recipient list to these rounds. If all users are online constantly, this method also provides sender unobservability for the network because an attacker cannot determine if a specific user is sending a message or not due to cover traffic.

Loop-back cover traffic is a dummy message sent by the user to himself. This type of cover message is used to detect corrupt nodes in the Loopix network anonymity system [Pi17]. If the message does not reach its sender, the user can detect the corruption in the path and use another route for future messages.

4 Comparison to Database Anonymity

We are trying to measure anonymity beyond the classical metrics from section 3.1, and for this purpose, we are looking at a related field: anonymity in databases. When releasing data from a confidential database to the public, e.g., for research purposes, one has to preserve the privacy of individuals listed in the database, including possible sensitive information, such as exact addresses, salaries, or diseases. For this purpose, the data is categorized into different classes, which, by the definition of Domingo-Ferrer et. al. [DSS16], include:

- **Identifiers** can be used to unambiguously identify an individual.
- **Quasi-Identifiers** are attributes, which can not identify a person alone, but a set of quasi-identifiers can. All entries sharing the same value for a quasi-identifier form an **equivalence class**.
- **Confidential attributes** have to be protected, and it should not be possible to determine an individual's confidential attribute (range).

	Identifier	Quasi-Identifiers		Confidential Attribute
	ID	**ZIP Code**	**Gender**	**Salary**
1	17394	93049	m	2794
2	52844	93049	m	3984
3	92961	93049	f	3104

Tab. 1: Example of a database table. Entries 1 and 2 form an equivalence class.

One example for these classes can be seen in table 1. The confidential attributes can be represented by the anonymity goals of section 3.1, and we will analyze these attributes from the perspective of an attacker trying to determine a message's sender. Except for the sender's signature, the only identifier is represented by the act of sending the message. A

global attacker, controlling all mix nodes, can observe this and thus identify the sender, while a local attacker can not follow a message beyond a mix node.

The link between an incoming and outgoing message at a mix node can represent quasi-identifiers in a mix network since the knowledge of one link does not reveal the message's full path, but knowledge of all links on the message's path would reveal the sender-recipient-relationship.

All messages arriving at a mix-node in the same batch form a message's local anonymity set. We will apply anonymity concepts developed for equivalence classes to these local anonymity sets. In contrast, we define the global anonymity set as the set of all sent messages, which could be the original message from an attacker's perspective. If every participant only uses one mix node, the local and global anonymity sets are identical. In addition to quasi-identifiers, the global anonymity set can be reduced even further by recognizing messages as cover traffic or knowledge of some messages' actual senders. Various privacy models from databases will be presented in the following sections.

4.1 k-Anonymity

The notion of k-anonymity indicates if the information for each person contained in the release cannot be distinguished from at least k-1 individuals whose information also appear in the release [Sw02]. This notion can be achieved in two ways. Some characters in quasi-identifier columns can be replaced with an asterisk (*), called suppression, or attributes are replaced with a broader category. This makes it difficult to identify an individual in the table.

When applying k-anonymity to mix networks, a release can be considered as messages that depart from the mix each round. In each round, if the mix flushes at least k messages, the system provides recipient k-anonymity. However, to provide sender k-anonymity, at least k messages must have arrived in the mix. The application of this technique in the Chaum mix, whose batch size is denoted as n, leaves three possibilities:

$$1)\ k < n \quad 2)\ k = n \quad 3)\ k > n$$

The Chaum mix, whose batch size n is equal to or greater than k, guarantees k-anonymity for both sender and recipient anonymity set. So, in the first and second conditions, k-anonymity can be provided by Chaum mixes easily. In the third scenario, this is not possible since it is impossible to send more than n packets per round. To achieve k-anonymity, indeterministic mixes may be used, such as a pool mix or a Stop-And-Go Mix.

When applying the k-anonymity technique to a pool mix, the mix has to wait until enough($\geq k$) messages have been passed into the mix. From this point on, k-anonymity is guaranteed. With a Stop-And-Go mix, the situation is a little more complicated since it

is always possible for the Stop-And-Go mix to go idle. If one can accept this uncertainty, then the same statement as for pool mixes can also be applied to the Stop-And-Go mix. Otherwise, users must constantly send messages to the Stop-And-Go mix to guarantee it is not going empty.

4.2 l-Diversity

However, k-anonymity is not the only model used to evaluate database anonymity since its leaves weaknesses under certain conditions. In particular, two primary attacks can be employed against k-anonymity, even in the context of mix-networks:

- A **homogeneity attack** can occur when all confidential attributes share the same value. In a mix network, this can happen when all messages inside a batch go to the same recipient, thus revealing the relationship between all senders and that recipient.
- In a **background knowledge attack**, an attacker can reduce the (already small) set of possible values for the sensitive attribute with specific background knowledge. In a mix network, this can happen, for example, when the attacker knows that there is no communication between a sender and particular recipients of a batch.

Because of these problems, the concept of l-diversity was defined as follows: ”An equivalence class is said to satisfy l-diversity if there are at least l well-represented values for the sensitive attribute.” In this sentence, ”well-represented” is not very clear, the following sections discuss more specific definitions and their possible implications for anonymity in mix networks.

4.2.1 Distinct l-diversity

This is the most straightforward definition and requires that l different values exist for the sensitive attribute. For a mix network, a message batch would have to include l different recipients, which could be end-users or other mix nodes. There are multiple possible strategies, how this could be achieved. A simple implementation would change a threshold mix to wait until it contains messages with l different recipients before relaying them. However, such a protocol would allow for new attacks. If an attacker observes messages arriving at the mix node and sees that the node is not sending its message batch even though l messages have arrived at it, he knows that at least two messages share the same recipient. He can also conclude that the last message arriving at the node before the batch is sent must have a new recipient. Thus, this protocol would require further adaptions, such as introducing indeterminism.

If the mix node is allowed to take active measures, known strategies from mix networks can be used to achieve l-diversity. One possible protocol would let the mix node generate cover

traffic and add it to its message pool until l distinct recipients exist. An alternative technique employed in Babel Mix [GT96] allows mix nodes to reroute messages. This could be used to spread messages, initially going to the same recipient, across different recipients.

In both previous suggestions, the end-user has to trust the mix node as only it can determine if l-diversity is fulfilled for its current message set. However, an attacker can abuse these options of withholding and rerouting packets to execute active attacks. If end-users do not trust the mix nodes, they can not guarantee l-diversity, but they can improve message diversity by sending cover traffic. This way, users can increase anonymity sets and thus diversity. However, known problems for end-to-end cover traffic apply. These include, e.g., how users should find their set of possible recipients for cover traffic. In the best case, every user is included in this set, but in real systems, knowing every end-user seems unrealistic.

4.2.2 Entropy l-diversity

A different interpretation is Entropy l-diversity, first suggested by Ohrn and Ohno-Machado [ØO99]. In this case, the entropy of a dataset is calculated with equation 1, where $p(s)$ describes the fraction of records in the dataset S with the same sensitive value as s. Entropy l-diversity is fulfilled if $H(S) \geq log(l)$.

$$H(S) = -\sum_{s \in S} p_s(s) * log(p_s(s)) \tag{1}$$

This definition of l-diversity does not alter how l-diversity can be applied to mix networks since only the calculation inside mix nodes has to be changed. However, unlike distinct l-diversity, it does not guarantee the latest message arriving at a mix node to have a new recipient. A new message going to a recipient underrepresented in the node's currently held messages could balance out the ratios of recipients and thus result in the fulfillment of entropy l-diversity, thus giving an attacker less information about the last messages' recipients.

4.2.3 Recursive (c, l)-diversity

In another definition of l-diversity, an attempt is made to limit the most frequent sensitive attribute's occurrence and provide a lower bound for the least common values. Equation 2 is used to determine if (c, l)-diversity is achieved, where $r_1, r_2, ..., r_n$ are the frequencies of the sensitive attribute's occurrence from most to least frequent.

$$r_1 < c * (r_l + r_{l+1} + ... + r_m) \tag{2}$$

A special subcategory is defined as **Positive Disclosure-Recursive Diversity** and reduces the requirements towards diversity by allowing the disclosure or frequent appearance of a sensitive attribute if its value is not very sensitive, e.g., healthy. This concept could be transferred to mix networks since the knowledge of a message going to another mix node does not give a local attacker much information. However, this knowledge would allow an (n-1)-attacker (an attacker who controls all but one mix-node) to track a message completely, revealing the sender-recipient relationship.

4.3 t-Closeness

t-Closeness implies that the distance between the distribution of a sensitive attribute in each equivalence class and the distribution of the attribute in the whole table is no more than a threshold t [LLV07]. This technique is used to overcome some of the well-known attacks that cannot be prevented by l-diversity, such as the skewness attack and similarity attack. Let **P** represent one of the distributions of an equivalence class, and **Q** represents the whole table's distribution. To provide t-closeness, the distance between P and Q must be less than the threshold t.

This technique can also be applied to mix networks. The table can be considered as a union of recipient sets for all rounds (Q), and classes can be considered as a recipient set of each round (P). This distance can be calculated in several ways. The well-known techniques are variational distance and Kullback-Leibler distance. The formula of variational distance is given in Equation 3.

$$Dist[P,Q] = \sum_{i=1}^{m} \frac{1}{2} |p_i - q_i| \tag{3}$$

A global passive attacker, who is observing all routes in the network and wants to learn who a specific sender Alice is connecting with, can record the recipient sets of mixes over rounds in which Alice participates and does not participate. By comparing these recipient sets, he can intercept Alice's possible recipient set. These types of attacks are named intersection attacks and have some variants such as disclosure attack and statistical disclosure attack [Da03; KAP03]. In this scenario, if the distribution of recipient sets throughout all rounds are close to each other, creating an interception about Alice's recipient set takes more time.

In the scenario in figure 3, four users are communicating with four other users via a mix. P_i denotes the distribution of the recipients in the current round. The threshold value of t-closeness is set to 0.4, and the variational distance method is used to measure the distance between distributions. Q denotes the distribution of all rounds. In the first round, P_1 is equal to Q, and the distance is 0. It can be seen that all senders send a message and all recipients receive a message. In the second round, the distance between P_2 and Q is 0.1 and still less than the threshold. In round three, only two users send messages to the two recipients.

This information leakage can be reduced by applying t-closeness to this mix network. The mix can manipulate the third round because the distance between P_3 and Q is bigger than the threshold t. So, the mix can decrease the distance by producing cover packets. As seen in figure 4, these packets can be sent over two rounds, keeping a distance less than the threshold, preventing a decrease of the anonymity set size. In this example, recipient anonymity is considered. If sender anonymity is also considered, cover packets must be created by senders, requiring collaborative work.

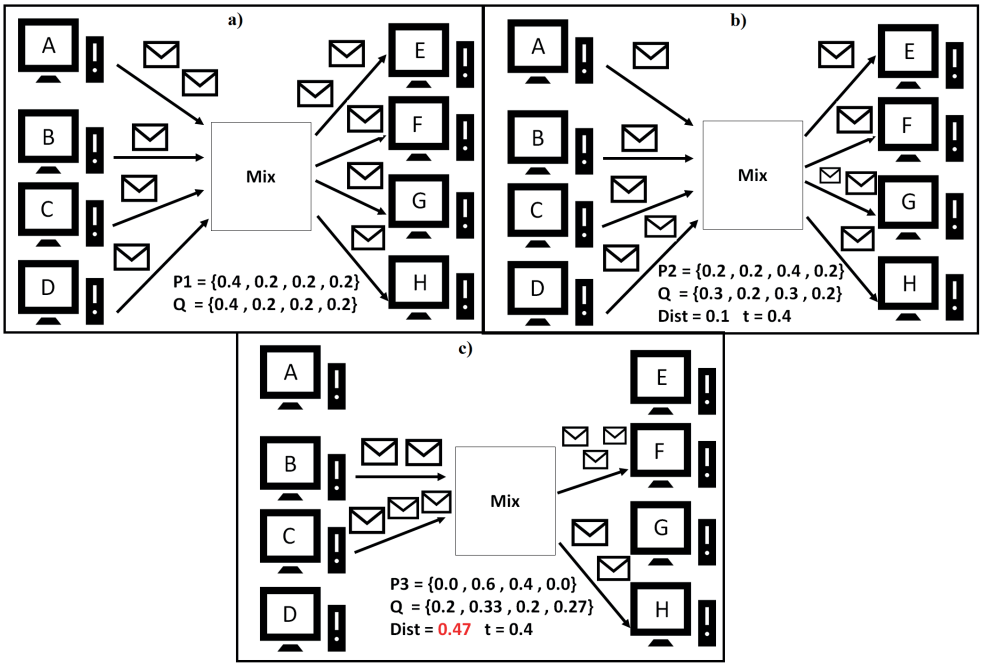

Fig. 3: Sample communication scenario via mix

5 Conclusion

This work has shown that the principles of anonymity used for databases can be applied to mix networks. However, achieving these strict privacy goals is a complex challenge. k-Anonymity, l-diversity, and t-closeness guarantee anonymity with specific parameters, while mix networks only apply techniques to improve anonymity, and we have shown that known attacks against database anonymity can also affect mix networks. The only way to guarantee anonymity under these conditions is by giving the mix nodes more control since only they can determine the diversity/closeness of their current message batch. However, it should not be necessary to trust the mix nodes. Instead, control over messages should be distributed among the end-users. If participants only work independently, they can improve

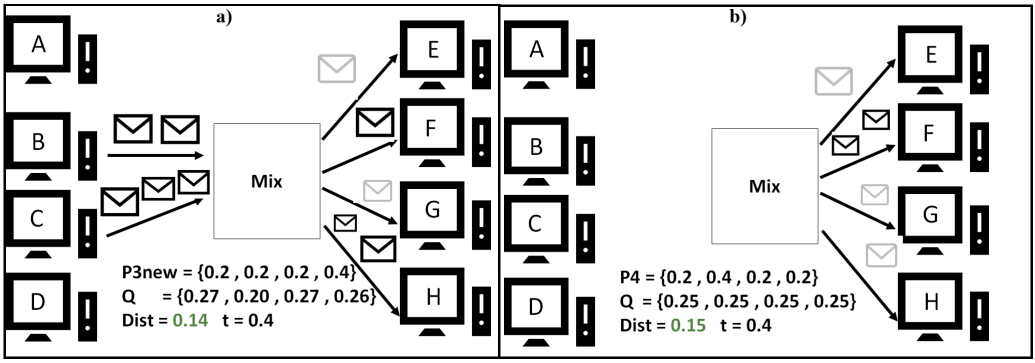

Fig. 4: a) third round after applying t-closeness. b) fourth round by applying t-closeness (cover messages are denoted by light grey color)

anonymity by introducing end-to-end cover traffic, but they can not guarantee it. Thus, multiple users have to work together to achieve privacy. For this purpose, they have to complete three objectives:

1. **Grouping**: Users have to build a group of size n of trusted users. This group needs to be big enough to achieve the anonymity goals, and it should not be possible for an attacker (group) to control most of its participants.
2. **Building**: The group needs to generate a message batch to insert into the network. Therefore, each user has to contribute a message, which can be real or cover traffic. The group has to ensure that this batch fulfills the anonymity requirements of diversity and closeness. If they are not fulfilled, the batch has to be changed, e.g., by randomizing the cover traffic or replacing some real messages with cover traffic.
3. **Embedding**: The group needs to deploy its messages into the mix network. To do this, one user has to transfer the batch into the mix network. The other users must be able to verify that their messages have been sent correctly without modification.

We are working on a protocol that allows mix users to build and send messages in anonymous groups under these conditions and want to present it in a future work.

References

[Ch81] Chaum, D. L.: Untraceable electronic mail, return addresses, and digital pseudonyms. Communications of the ACM 24/2, pp. 84–90, 1981.

[Da03] Danezis, G.: Statistical Disclosure Attacks. In: Security and Privacy in the Age of Uncertainty. Springer US, Boston, MA, pp. 421–426, 2003.

[DDM03] Danezis, G.; Dingledine, R.; Mathewson, N.: Mixminion: design of a type III anonymous remailer protocol. In: 2003 Symposium on Security and Privacy, 2003. Pp. 2–15, 2003.

[DG09] Danezis, G.; Goldberg, I.: Sphinx: A compact and provably secure mix format. In: 2009 30th IEEE Symposium on Security and Privacy. IEEE, pp. 269–282, 2009.

[DSS16] Domingo-Ferrer, J.; Sánchez, D.; Soria-Comas, J.: Database anonymization: privacy models, data utility, and microaggregation-based inter-model connections. Synthesis Lectures on Information Security, Privacy, & Trust 8/1, pp. 1–136, 2016.

[EY09] Edman, M.; Yener, B.: On Anonymity in an Electronic Society: A Survey of Anonymous Communication Systems. ACM Comput. Surv. 42/1, Dec. 2009.

[GT96] Gülcü, C.; Tsudik, G.: Mixing E-mail With Babel. In: Proceedings of the Network and Distributed Security Symposium - NDSS '96. IEEE, pp. 2–16, Feb. 1996.

[KAP03] Kedogan, D.; Agrawal, D.; Penz, S.: Limits of Anonymity in Open Environments. In (Petitcolas, F. A. P., ed.): Information Hiding. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 53–69, 2003.

[LLV07] Li, N.; Li, T.; Venkatasubramanian, S.: t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. In: 2007 IEEE 23rd International Conference on Data Engineering. Pp. 106–115, 2007.

[MW07] Mallesh, N.; Wright, M.: Countering Statistical Disclosure with Receiver-Bound Cover Traffic. In (Biskup, J.; López, J., eds.): Computer Security – ESORICS 2007. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 547–562, 2007.

[ØO99] Øhrn, A.; Ohno-Machado, L.: Using Boolean reasoning to anonymize databases. Artificial Intelligence in Medicine 15/3, pp. 235–254, 1999.

[Pi17] Piotrowska, A. M.; Hayes, J.; Elahi, T.; Meiser, S.; Danezis, G.: The Loopix Anonymity System. In: 26th USENIX Security Symposium (USENIX Security 17). USENIX Association, Vancouver, BC, pp. 1199–1216, Aug. 2017.

[PK01] Pfitzmann, A.; Köhntopp, M.: Anonymity, unobservability, and pseudonymity—a proposal for terminology. In: Designing privacy enhancing technologies. Springer, pp. 1–9, 2001.

[Sw02] Sweeney, L.: K-Anonymity: A Model for Protecting Privacy. Int. J. Uncertain. Fuzziness Knowl.-Based Syst. 10/5, pp. 557–570, Oct. 2002.