On the Trade-off between Relationship Anonymity and Communication Overhead in Anonymity Networks

Ognjen Vuković
School of Electrical Engineering
KTH, Royal Institute of Technology,
Stockholm, Sweden
Email: vukovic@ee.kth.se

György Dán
School of Electrical Engineering
KTH, Royal Institute of Technology,
Stockholm, Sweden
Email: gyuri@ee.kth.se

Gunnar Karlsson
School of Electrical Engineering
KTH, Royal Institute of Technology,
Stockholm, Sweden
Email: gk@kth.se

***Abstract*—Motivated by protection and privacy in industrial communication networks, in this paper we consider the trade-off between relationship anonymity and communication overhead. We consider two anonymity networks: Crowds, which has unbounded communication delay and Minstrels, proposed in this paper, which provides bounded communication delay. While Crowds hides the sender's identity only, Minstrels aims at hiding the receiver's identity as well. However, to achieve bounded communication delay it has to expose the sender's identity to a greater extent than Crowds. We derive exact and approximate analytical expressions for the relationship anonymity for these systems. While Minstrels achieves close to optimal anonymity under certain conditions, our results show that, contrary to expectations, increased overhead does not always improve anonymity.**

I. Introduction

Many communication systems, for example modern industrial networks [1], [2], require high availability between a fixed set of nodes on a pairwise basis. The nodes can be the subsidiaries of an enterprise connected by a virtual private network over the public Internet, or they can be sensors, actuators and operation centres in a wide area industrial control system, e.g., in a supervisory control and data acquisition (SCADA) network. Cryptography may provide authentication, confidentiality and data integrity for the communication, but source and destination addresses could still be visible to an outside attacker who is able to observe one or more network links. The outside attacker may identify traffic patterns: who is communicating with whom, when and how often. Using this information the attacker can infer the importance of the messages, and may perform targeted attacks on the communication between any two nodes. These targeted attacks might be hard to detect and can lead to incorrect system operation.

Mix networks [3] are a way to mitigate outside attacks by providing relationship anonymity, i.e., by making it untraceable who communicates with whom [4]. Nodes in a mix network relay and delay messages such that an outside attacker cannot trace the route of the individual messages through the mix. While relaying renders outside attacks more difficult, it introduces the possibility of inside attacks. Due to the often long life-cycles of industrial systems software corruption is a threat, and the complexity of the code-base makes corruption hard to detect. Corrupted nodes that are part of the mix network can perform inside attacks to determine the sender-receiver pair for messages that are relayed through them. Anonymity networks can provide some level of relationship anonymity against inside attackers (e.g., [5], [6]) by hiding the sender or the receiver from the relay nodes. Good sender (or receiver) anonymity in itself does not necessarily lead to good relationship anonymity [8], hence we focus on relationship anonymity in this paper.

The relationship anonymity provided by mix networks and anonymity networks comes at the price of delay and communication overhead. Excessive delays can negatively impact the system performance, while overhead leads to high resource requirements, so that in practice both have to be kept low. Our goal in this paper is to investigate the trade-off between the communication overhead introduced and the level of relationship anonymity provided by anonymity networks.

Intuition says that increased overhead should result in increased anonymity. In this paper we show that this is not necessarily the case. We use two anonymity networks for our study. First, Crowds, proposed in [6], which hides the sender by introducing unbounded message delivery delay (it still exposes the receiver's identity). Crowds was shown to provide optimal sender anonymity for given overhead [7], i.e., path length. Second, Minstrels, described in this paper, which provides both sender and receiver anonymity, i.e., relationship anonymity. Minstrels has bounded message delivery delay. We do not consider long term intersection attacks, such as [8], [9], [10], which exploit cases when the sender's anonymity is not *beyond suspicion*, i.e., the sender is distinguishable from other nodes. These attacks consider that the receiver is outside the anonymity network, and they exploit the distribution of message destinations to decrease the relationship anonymity. In our system the receiver is part of the anonymity network, and message destinations can have an arbitrary distribution; but an attacker does not have a-priori knowledge of the traffic matrix.

The rest of the paper is organized as follows. Section II describes our system model and the anonymity metrics. Section III provides a description of the Minstrels anonymity network. In Section IV we develop analytical models of the relationship anonymity provided by Crowds and Minstrels, and we show numerical results based on the models in Section V. Section VI concludes the paper.

II. System model and metrics

We consider an anonymity network with N nodes. The nodes act as sources, destinations and as relay nodes for each others' messages. The underlying communication network is a complete graph. The *inside attacker* is in control of C nodes, and can observe the messages traversing those nodes and the protocol specific information contained in the messages. Its goal is to identify the source and the destination of the messages that it observes.

We quantify the relationship anonymity by the probability $P_{rel}(s,r)$ that the attacker assigns to a sender-receiver pair (s,r) for a message. In general, the relationship anonymity depends on two factors. First, on the probability of having an attacker on the path. Second, on the probability that the attacker assigns to the sender (that it sent the message) and to the receiver (that it is the destination) when it gets the message. These probabilities are a function of the anonymity protocol, the number of nodes N and the number C of inside attackers

$$P_{rel}(s,r) = \sum_{i=1}^{\infty} P(\hat{S}(s),\hat{R}(r)|H_i,S(s),R(r))P(H_i|S(s),R(r)), \tag{1}$$

where $S(s)$ and $R(r)$ denote the events that the sender is node s and the receiver is node r, respectively; $\hat{S}(s)$ and $\hat{R}(r)$ denote the events that the attacker correctly identifies node s as the sender and node r as the receiver, respectively; $P(H_i|S(s),R(r))$ is the probability that the position of the first attacker on the path is i given that (s,r) is the sender-receiver pair, and $P(\hat{S}(s),\hat{R}(r)|H_i,S(s),R(r))$ is the probability that the attacker identifies (s,r) as the sender-receiver pair given its position on the path.

Finally, we define the overhead of the anonymity network as the average path length (number of relay hops) $E[K]$ of the messages.

III. Minstrels system description

Minstrels, described below, uses nodes as message relays in the same way as Crowds [6] with the difference that the number of nodes visited by a message is bounded.

Consider the system described in Section II. When a node s wants to send a message to a node r it picks a node uniformly at random among the other $N-1$ nodes (excluding s) and forwards the message. The next node forwards the message to one of the other $N-2$ nodes (excluding itself and the sender node s) chosen uniformly at random. Every subsequent forwarder picks one of the non-visited nodes to forward the message. When node r receives the message, it will send the message further in order to improve the receiver anonymity. The path ends when all N nodes have been visited.

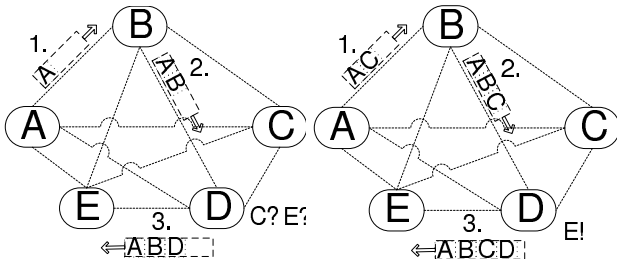

Fig. 1. A simple example of Minstrels with five nodes.

The message, or part of it, is encrypted with the receiver's public key. When a node receives the message, it checks if it is the receiver by trying to decrypt the encrypted part of the message. If the decrypted part of the message represents valid data, the node is the receiver. Note that a node does not know who is the receiver, it can only check whether it is the receiver itself (unlike in Crowds).

To bound the path length, the messages record a list of the visited nodes in the header. The list can be implemented, for example, using a Bloom filter, to keep its size small. When a relaying node receives a message, it will relay the message only to non-visited nodes. To control the maximum path length (i.e., delay) the sender can initialize the list of visited nodes with a number $M \in \{0,...,N-1\}$ of the nodes in the system. These initialized nodes are considered as visited so that the message can not be relayed to them. Hence, a message traverses all nodes except for the initialized nodes in the list. The sender picks the number of initialized nodes at random: it initializes the list with M nodes with probability $P(M)$, where $\sum_{M=0}^{N-1} P(M) = 1$. For $M = 0$ the list is empty, for $M = 1$ the list is initialized only with the sender and for $M > 1$ the list is initialized with the sender and $M-1$ other nodes. The sender must not initialize the list with the receiver. The distribution of $P(M)$ is a system parameter, and we use it to explore the anonymity-overhead trade-off. Fig. 1 shows two simple examples with five nodes, node A as sender and node D as receiver. Fig. 1 (left) shows a case when the list is initialized with the sender node A and the message is forwarded to node C. Node C checks if it is the receiver, puts itself in the list and chooses the next hop uniformly at random among nodes (B,D,E). The next hop, node D, follows the same procedure with only two forwarding options (B,E). Fig. 1 (right) shows another case when the list is initialized with the sender and node C, and the message is forwarded to node B. Node B adds itself to the list and decides to which of the remaining nodes (D,E) to forward the message. Node C is considered as already visited.

IV. Overhead and Anonymity

In the following we derive expressions for the communication overhead and the anonymity provided against inside attackers for Crowds and for Minstrels.

A. Communication Overhead

We start with calculating the communication overhead of Crowds and Minstrels. The mean number of hops for Crowds

is the expected value of a geometric distribution with success probability $1-p_f$, i.e.,

$$E[K] = \frac{p_f}{1-p_f} + 2 \tag{2}$$

where p_f is the probability that a node will relay a message. For Minstrels for a given number M of initialized nodes in the list the path length is equal to $K = N - M$. The mean number of hops depends on the distribution $P(M)$ and can be expressed as

$$E[K] = \sum_{M=0}^{N-1} P(M)(N-M). \tag{3}$$

B. Relationship Anonymity Against Inside Attackers

We consider attackers without any *a priori* knowledge of the system traffic matrix. All nodes are equally likely to be senders or receivers. The attacker can only decrease the relationship anonymity by knowing the protocol and by observing traffic that goes over the nodes it controls. In order to calculate the relationship anonymity in the following we express the probabilities in (1) for Crowds and for Minstrels.

1) Crowds: For Crowds the first attacker is on position i if the message is first relayed $i-1$ times through trusted nodes but the last hop is an attacker. We denote this event by H_i. The probability $P(H_i|S(s),R(r))$ can be expressed as

$$P(H_i|S(s),R(r)) = P(H_i) = p_f^{i-1}\left(\frac{N-C-1}{N-1}\right)^{i-1}\frac{C}{N-1}. \tag{4}$$

Let I denote the event that the first attacker on the path is immediately preceded on the path by the sender. Note that $H_1 \Rightarrow I$ but the opposite is not true since the sender may appear multiple times on the path. For a given attacker on the path, $P(I|H_{1+})$ is the probability that the attacker's predecessor is the sender. $P(\bar{I}|H_{1+})$ is the probability that another node (i.e., not the predecessor) is the sender. The probability that the attacker assigns to the actual sender of the message can be expressed as

$$P(\hat{S}(s)|H_i,S(s),R(r)) = P(I|H_i)P(I|H_{1+}) + P(\bar{I}|H_i)P(\bar{I}|H_{1+}), \tag{5}$$

where $P(I|H_i)$ is the probability that for a given position i of an attacker on the path the sender appears as the predecessor (on position $i-1$). For $i=1$ we have $P(I|H_1)=1$ while for $i>1$ we have $P(I|H_i) = P(I|H_{2+}) = \frac{1}{N-C-1}$. Intuitively, $P(\bar{I}|H_i)$ is the probability that for a given position i of an attacker on the path, some other node, a relay, appears as the predecessor. For $i=1$ we have $P(\bar{I}|H_1)=0$, while for $i>1$ we have $P(\bar{I}|H_i) = \frac{N-C-2}{N-C-1}$.

The expression for $P(I|H_{1+})$ is given in [6] for the case when there are n possible relays (including the sender). Since in our case there are $n = N-1$ possible relays the expression for $P(I|H_{1+})$ becomes $P(I|H_{1+}) = \frac{N-1-p_f(N-C-2)}{N-1}$. $P(\bar{I}|H_{1+})$ can be expressed as $P(\bar{I}|H_{1+}) = \frac{1-P(I|H_{1+})}{N-C-2}$.

The receiver is exposed in Crowds, hence $P(\hat{S}(s),\hat{R}(r)|H_i,S(s),R(r)) = P(\hat{S}(s)|H_i,S(s),R(r))$.

2) Minstrels: For Minstrels we rewrite (1) as

$$P_{rel}(s,r) = P(\hat{S}(s),\hat{R}(r)|H_{1+},S(s),R(r))P(H_{1+}|S(s),R(r)), \tag{6}$$

where $P(H_{1+}|S(s),R(r))$ is the probability of having an attacker on the path for sender-receiver pair (s,r), and $P(\hat{S}(s),\hat{R}(r)|H_{1+},S(s),R(r))$ is the probability that the attacker identifies (s,r) as the sender-receiver pair. We consider coordinated attackers that keep track of the received messages, so that every attacker knows whether a particular message was already received by an attacker. Hence, when the first attacker on the path gets the message, it knows the number m_C of attackers that the list of visited nodes was initialized with by the sender. m_C is a realization of the random variable M_C, whose distribution depends on the value of M.

In Minstrels the probability that the attacker assigns to a sender-receiver pair does not only depend on the node that the message is received from, i.e., the predecessor p, but also on the contents of the list of visited nodes ($\mathcal{L}$) that the message carries. Consequently, the attacker distinguishes between three disjoint sets of nodes: the predecessor node ($\{p\}$), nodes in the list of visited nodes except the predecessor ($\mathcal{L}\setminus\{p\}$), and nodes not in the list of visited nodes ($\overline{\mathcal{L}\cup\{p\}}$). These sets form a partition of the set of all trusted nodes in the system, and nodes belonging to the same set are equally likely to be the sender (and the receiver). As a shorthand for the universe of distinguishable events we use the notation $\Omega_s = \{s=p, s\in\mathcal{L}\setminus\{p\}, s\in\overline{\mathcal{L}\cup\{p\}}\}$, where, for example, $s=p$ is the event that the predecessor is the sender. Similarly, we define $\Omega_r = \{r=p, r\in\mathcal{L}\setminus\{p\}, r\in\overline{\mathcal{L}\cup\{p\}}\}$ for the distinguishable events regarding the receiver.

Given the information on $\mathcal{L}$, m_C, and p available to the attacker, we can use the law of total probability to expand (6) conditional on the list length $||\mathcal{L}|| = l$, $\omega_s \in \Omega_s$, $\omega_r \in \Omega_r$, and $M_C = m_C$,

$$P_{rel}(s,r) = \sum_{m_C}\sum_{l}\sum_{\omega_s}\sum_{\omega_r} P(\hat{S}(s),\hat{R}(r)|\omega_r,\omega_s,m_C,H_{1+},l,S(s),R(r)) \tag{7}$$

$$\cdot P(\omega_r,\omega_s,m_C,H_{1+},l|S(s),R(r)). \tag{8}$$

The summands in (7) are the probabilities that the attacker correctly identifies the sender-receiver pair of the message that contains the information ($||\mathcal{L}|| = l$, $\omega_s \in \Omega_s$, $\omega_r \in \Omega_r$, and $M_C = m_C$), and are independent of $S(s),R(r)$. Eq. (8) is the probability that a message with (s,r) as sender-receiver pair is received by an attacker and carries particular information.

Before we turn to the calculation of the probability $P(\omega_r,\omega_s,l,m_C,H_{1+}|S(s),R(r))$ we introduce the notation $H(l,m_C|M)$ for the joint event $||\mathcal{L}|| = l$, H_{1+}, and $M_C = m_C$ for a given number of initialized nodes M. Clearly, $l \geq M$. The probability of this event can be expressed as

$$P(H(l,m_C|M)) = \begin{cases} \frac{C}{N-1} & l=0, M=0 \\ P(M_C=0|M)\frac{N-C-1}{N-1}\frac{C}{N-l}\prod_{z=1}^{l-1}\frac{N-C-z}{N-z} & l\geq 1, M=0 \\ P(M_C=m_C|M)\frac{C-m_C}{N-l}\prod_{z=M}^{l-1}\frac{N-C+m_C-z}{N-z} & l\geq 1, M>0, \end{cases} \tag{9}$$

TABLE I
$P(\Omega_r, \Omega_s, ||\mathcal{L}|| = 0, M_C = 0, H_{1+}|S(s), R(r))$

Ω_s, Ω_r	
$s = p,\ r \in \overline{\mathcal{L} \cup \{p\}}$	$P(M=0)P(H(0,0\|M=0))$

where $P(M_C|M)$ is the probability that the list of visited nodes is initialized with M_C attacker nodes, given that it is initialized with M nodes by the sender. Due to the rules of prefilling, $M_C \in \{max(0, M-1-(N-2-C)), min(M-1, C)\}$. For $M = 0$ and $M = 1$ there cannot be any initialized attackers, hence $P(M_C = 0|M \in \{0,1\}) = 1$ and $P(M_C > 0|M \in \{0,1\}) = 0$. For $M > 1$ we have

$$P(M_C|M) = \begin{pmatrix} M-1 \\ M_C \end{pmatrix} \frac{\prod_{k=2}^{M-M_C}(N-C-k)\prod_{k=0}^{M_C-1}(C-k)}{\prod_{k=2}^{M}(N-k)}. \quad (10)$$

We now turn to the calculation of the probability $P(\omega_r, \omega_s, l, m_C, H_{1+}|S(s), R(r))$, i.e., the probability that the attacker would receive a particular message sent by s to r. If the sender is the predecessor ($s = p$) the receiver cannot be the predecessor, hence $P(r = p, s = p, l, m_C, H_{1+}|S(s), R(r)) = 0$. For the rest of the cases we show the probabilities in a tabular form to improve readability.

For $||\mathcal{L}|| = 0$ and $||\mathcal{L}|| = 1$ there can be no attackers in the list of visited nodes (when received by the first attacker), because if the sender prefills the list of visited nodes it has to include itself in the list. Hence, for $||\mathcal{L}|| = 0$ and $||\mathcal{L}|| = 1$ we have $M_C > 0$ with probability 0. Furthermore, for $||\mathcal{L}|| = 0$ the sender must be the predecessor ($s = p$) and the receiver cannot be in the list of visited nodes ($r \in \overline{\mathcal{L} \cup \{p\}}$), every other tuple in $\{(\omega_s, \omega_r) : \omega_s \in \Omega_s, \omega_r \in \Omega_r\}$ has probability 0. Table I shows the corresponding probability, i.e., the probability that the sender initializes the message with an empty list, and chooses the attacker as next hop. For $||\mathcal{L}|| = 1$ the sender and the receiver cannot both be in the list of visited nodes. Furthermore, if the sender or the receiver is in the list of visited nodes, it must be the predecessor, hence $s \in \mathcal{L} \setminus \{p\}$ and $r \in \mathcal{L} \setminus \{p\}$ have probability 0. Table II shows the probabilities for the remaining cases for $||\mathcal{L}|| = 1$. As an example, the second row in the table is the probability that the sender initializes the list empty, forwards the message to the receiver, which then forwards the message to the attacker.

For $||\mathcal{L}|| > 1$ there may or may not be attackers in the list of initialized nodes. Table III shows the probabilities for $||\mathcal{L}|| > 1$ when there are no attackers in the list of initialized nodes ($M_C = 0$). When there are attackers in the list of initialized nodes ($M_C > 0$), the sender has to be in the list of visited nodes. Furthermore, if the sender is the predecessor ($s = p$) then the receiver cannot be in the list of visited nodes ($r \in \mathcal{L} \setminus \{p\}$), because this could only happen if the sender had prefilled the list of visited nodes with the receiver, but then the receiver would never receive the message. The corresponding probabilities for $||\mathcal{L}|| > 1$ and $M_C > 0$ are shown in Table IV.

Let us now turn to the calculation of the probabilities that

TABLE II
$P(\Omega_r, \Omega_s, ||\mathcal{L}|| = 1, M_C = 0, H_{1+}|S(s), R(r))$

Ω_s, Ω_r	
$s = p,\ r \in \overline{\mathcal{L} \cup \{p\}}$	$P(M=1)P(H(1,0\|M=1))$
$s \in \overline{\mathcal{L} \cup \{p\}},\ r = p$	$P(M=0)P(H(1,0\|M=0))\frac{1}{N-C-1}$
$s \in \overline{\mathcal{L} \cup \{p\}},\ r \in \overline{\mathcal{L} \cup \{p\}}$	$P(M=0)P(H(1,0\|M=0))\frac{N-C-2}{N-C-1}$

TABLE III
$P(\Omega_r, \Omega_s, ||\mathcal{L}|| > 1, M_C = 0, H_{1+}|S(s), R(r))$

Ω_s, Ω_r	
$s = p,\ r \in \mathcal{L} \setminus \{p\}$	$P(M=0)P(H(l,0\|M=0))\frac{l-1}{(N-C-1)^2}$
$s = p,$ $r \in \overline{\mathcal{L} \cup \{p\}}$	$P(M=0)P(H(l,0\|M=0))\frac{(N-C-l)}{(N-C-1)^2}$ $+P(M=l)P(H(l,0\|M=l))$
$s \in \mathcal{L} \setminus \{p\},$ $r = p$	$P(M=0)P(H(l,0\|M=0))\frac{l-2}{(N-C-1)^2}$ $+\sum_{k=1}^{l-1}P(M=k)P(H(l,0\|M=k))\frac{1}{N-C-k}$
$s \in \mathcal{L} \setminus \{p\},$ $r \in \mathcal{L} \setminus \{p\}$	$P(M=0)P(H(l,0\|M=0))\frac{(l-2)^2}{(N-C-1)^2}$ $+\sum_{k=1}^{l-2}P(M=k)P(H(l,0\|M=k))\frac{l-k-1}{N-C-k}$
$s \in \mathcal{L} \setminus \{p\},$ $r \in \overline{\mathcal{L} \cup \{p\}}$	$P(M=0)P(H(l,0\|M=0))\frac{(N-C-l)(l-2)}{(N-C-1)^2}$ $+\sum_{k=1}^{l-1}P(M=k)P(H(l,0\|M=k))\frac{N-C-l}{N-C-k}$
$s \in \overline{\mathcal{L} \cup \{p\}},\ r = p$	$P(M=0)P(H(l,0\|M=0))\frac{(N-C-l)}{(N-C-1)^2}$
$s \in \overline{\mathcal{L} \cup \{p\}},\ r \in \mathcal{L} \setminus \{p\}$	$P(M=0)P(H(l,0\|M=0))\frac{(l-1)(N-C-l)}{(N-C-1)^2}$
$s \in \overline{\mathcal{L} \cup \{p\}},\ r \in \overline{\mathcal{L} \cup \{p\}}$	$P(M=0)P(H(l,0\|M=0))\frac{(N-C-l)(N-C-l-1)}{(N-C-1)^2}$

TABLE IV
$P(\Omega_r, \Omega_s, ||\mathcal{L}|| > 1, M_C > 0, H_{1+}|S(s), R(r))$

Ω_s, Ω_r	
$s = p,\ r \in \overline{\mathcal{L} \cup \{p\}}$	$P(M=l)P(H(l,m_C\|M=l))$
$s \in \mathcal{L} \setminus \{p\},\ r = p$	$\sum_{k=m_C+1}^{l-1}P(M=k)P(H(l,m_C\|M=k))\frac{1}{N-C+m_C-k}$
$s \in \mathcal{L} \setminus \{p\},$ $r \in \mathcal{L} \setminus \{p\}$	$\sum_{k=m_C+1}^{l-2}P(M=k)P(H(l,m_C\|M=k))\frac{l-k-1}{N-C+m_C-k}$
$s \in \mathcal{L} \setminus \{p\},$ $r \in \overline{\mathcal{L} \cup \{p\}}$	$\sum_{k=m_C+1}^{l-1}P(M=k)P(H(l,m_C\|M=k))\frac{N-C+m_C-l}{N-C+m_C-k}$

the attacker correctly identifies the sender-receiver pair (s, r) used in (7). Given a message received by an attacker that contains information ($||\mathcal{L}|| = l$, $\omega_s \in \Omega_s$, $\omega_r \in \Omega_r$, and $M_C = m_C$) the attacker would identify (s, r) as the sender-receiver pair with probability

$$P(\hat{R}(r), \hat{S}(s)|\omega_r, \omega_s, m_C, H_{1+}, l) = \frac{P(\omega_r, \omega_s, l, m_C, H_{1+}|S(s), R(r)) \cdot P(R(r)|S(s)) \cdot P(S(s))}{\sum_{(a,b)} P(\omega_r, \omega_s, l, m_C, H_{1+}|S(a), R(b)) \cdot P(R(b)|S(a)) \cdot P(S(a))} \quad (11)$$

where the summation in the denominator is over all possible non-attacker sender-receiver pairs (a, b). $P(S(s))$ is the (a priory) probability that node s sends a message, and $P(R(r)|S(s))$ is the probability that node s selects node r as the destination of a message. Since the traffic matrix is homogeneous and attackers are informed about each other, all trusted nodes are equally likely to be the sender, $P(S(s)) = \frac{1}{N-C}$, and any trusted node (except the sender) is equally likely to be chosen as the receiver, i.e., with probability $P(R(r)|S(s)) = \frac{1}{N-C-1}$. The same observation holds for $P(S(a))$ and $P(R(b))$, so that these probabilities cancel out each other in (11).

We already calculated the numerator of (11), so in order to finish our calculations we only have to express

TABLE V
$P(\Omega_r, \Omega_s, ||\mathcal{L}|| = 0, M_C = 0, H_{1+} | S(a), R(b))$

Ω_s, Ω_r, a, b	
$s = p,\ r \in \overline{\mathcal{L} \cup \{p\}},\ a = s,\ \forall b$	$P(M=0)P(H(0,0\|M=0))$

TABLE VI
$P(\Omega_r, \Omega_s, ||\mathcal{L}|| = 1, M_C = 0, H_{1+} | S(a), R(b))$

Ω_s, Ω_r, a, b	
$s = p,\ r \in \overline{\mathcal{L} \cup \{p\}},\ a = s,\ \forall b$	$P(M=1)P(H(1,0\|M=1))$
$s = p,\ r \in \overline{\mathcal{L} \cup \{p\}},\ a \neq s,\ \forall b$	$P(M=0)P(H(1,0\|M=0))\frac{1}{N-C-1}$
$s \in \overline{\mathcal{L} \cup \{p\}},\ r = p,\ a = r,\ \forall b$	$P(M=1)P(H(1,0\|M=1))$
$s \in \overline{\mathcal{L} \cup \{p\}},\ r = p,\ a \neq r,\ \forall b$	$P(M=0)P(H(1,0\|M=0))\frac{1}{N-C-1}$
$s \in \overline{\mathcal{L} \cup \{p\}},\ r \in \overline{\mathcal{L} \cup \{p\}},$ $a \in \{s,r\},\ \forall b$	$P(M=0)P(H(1,0\|M=0))\frac{N-C-2}{N-C-1}$
$s \in \overline{\mathcal{L} \cup \{p\}},\ r \in \overline{\mathcal{L} \cup \{p\}},$ $a \notin \{s,r\},\ \forall b$	$P(M=0)P(H(1,0\|M=0))\frac{N-C-3}{N-C-1}$ $+P(M=1)P(H(1,0\|M=1))$

$P(\omega_r, \omega_s, l, m_C, H_{1+} | S(a), R(b))$ and only for the cases when the numerator of (11) is non-zero, and when $a \neq s$ or $b \neq r$.

The attacker can receive a message with an empty list of visited nodes ($||\mathcal{L}|| = 0, M_C = 0$) only if the sender is the predecessor, hence, $P(\omega_r, \omega_s, ||\mathcal{L}|| = 0, M_C = 0, H_{1+} | S(a), R(b)) > 0$ only for $a = s$. Nevertheless, the receiver of the message can be any trusted node $b \neq s$ (we use $\forall b$ as a shorthand notation). The corresponding probability $P(\Omega_r, \Omega_s, ||\mathcal{L}|| = 0, M_C = 0, H_{1+} | S(a), R(b))$ is given in Table V.

The attacker can receive a message with only one node in the list of visited nodes ($||\mathcal{L}|| = 1$), in which case the node in the list is the predecessor. The list could have been sent by the predecessor ($a = p$) or by a node not in the list ($a \in \overline{\mathcal{L} \cup \{p\}}$), but in either case there cannot be any attacker node prefilled in the list ($M_C = 0$). The receiver could be any other node ($\forall b$). The probability of receiving such a message $P(\Omega_r, \Omega_s, ||\mathcal{L}|| = 1, M_C = 0, H_{1+} | S(a), R(b))$ is given in Table VI.

For brevity, we omit the calculation of the probabilities for $||\mathcal{L}|| > 1$, they can be obtained following a similar reasoning, and can be found in [11].

3) A Bound For Relationship Anonymity: In order to obtain a lower bound of the probability assigned to a sender-receiver pair we use (1) for Crowds and (6) for Minstrels. If there is an attacker on the path, it would assume that any of the $N-C$ trusted nodes is equally likely to be the sender, and any other trusted node is equally likely to be the receiver,

$$P(\hat{S}(s), \hat{R}(r) | H_{1+}) = P(\hat{S}(s), \hat{R}(r) | H_i) = \frac{1}{(N-C)(N-C-1)}. \tag{12}$$

The probability $P(H_{1+})$, from (6), is expressed as

$$P(H_{1+}) = \sum_{M=0}^{N-1} \sum_{i=0}^{N-M} \sum_{M_C=0}^{min(max(0,M-1),C)} P(H_i | M_C, M) P(M_C | M) P(M), \tag{13}$$

where for $M = M_C = 0$ we have $P(H_1) = \frac{C}{N-1}$ and

$$P(H_i | M_C, M) = \frac{(N-C-1)C}{(N-1)(N-i+1)} \prod_{k=1}^{i-2} \frac{N-C-k}{N-k} \tag{14}$$

for $i > 1$, and for $M > 0$ we have

$$P(H_i | M_C, M) = \frac{C - M_C}{N-M-i+1} \prod_{k=1}^{i-1} \frac{N-M-C+M_C-k+1}{N-M-k+1}. \tag{15}$$

We use these bounds in the following as a baseline for comparison for the relationship anonymity provided by Crowds and by Minstrels.

V. Numerical Results

In the following we use the analytical models described above to get insight into the overhead-anonymity trade-off. To explore the trade-off we use $p_f \in (0,1)$ for Crowds, and various uniform and binomial distributions for $P(M)$ for Minstrels.

Fig. 2 shows the probability $P_{rel}(s,r)$ assigned to a sender-receiver pair as a function of the overhead (i.e., the mean path length) for $C = 1$ and $N = 10$. A higher value of $P_{rel}(s,r)$ means that the sender-receiver pair is more exposed, i.e., has less relationship anonymity. One would expect that high overhead provides good relationship anonymity (i.e., low assigned probability), but surprisingly this is not the case. Above a certain point more overhead (more relaying) has a negative effect on anonymity for both anonymity networks. The reason is that as the number of relays increases the probability $P(H_{1+})$ of having an attacker on the path increases faster than the certainty of the attacker about the identity of the sender-receiver pair decreases.

Fig. 3 shows results obtained with $N = 10$ nodes and $C = 3$ attackers. Interestingly, while for Minstrels the relationship anonymity decreases above a certain level of overhead, for Crowds the relationship anonymity improves monotonically. Hence, for $C = 3$ the probability that the attacker can assign to the sender decreases faster than the probability $P(H_{1+})$ of having an attacker on the path increases.

Fig. 4 shows results for $N = 50$ and $C = 1$. The figure has a logarithmic scale on the vertical axis to make the small probabilities easily distinguishable. For this scenario, in which the system size is bigger than in Fig. 2 but the number of attackers is smaller than in Fig. 3, it is now Crowds for which relationship anonymity deteriorates above a certain overhead. For Minstrels the probability $P_{rel}(s,r)$ decreases monotonically with increasing overhead. The reason is that for $N = 50$ the attacker appears later on the path than for $N = 10$ so the sender does not appear as predecessor that often. Hence the attacker assigns the same probability to the sender as to any other node in the list. This does not apply to Crowds. The sender can be revisited and may appear as predecessor at any position on a path and the predecessor is always more likely to be the sender than any other node [6].

Finally, Fig. 5 shows results for $N = 50$ nodes and $C = 5$ attackers. It is only the results shown in this figure that coincide with what one would expect, that is, increased overhead provides better relationship anonymity.

Figs. 2, 3, 4, and 5 also show the lower bounds for the probabilities $P_{rel}(s,r)$ for Crowds and for Minstrels. The lower

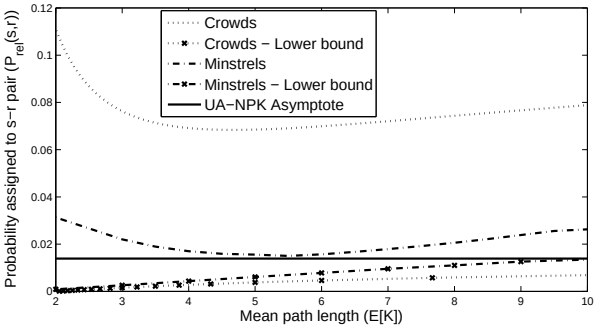

Fig. 2. Relationship anonymity vs. overhead for $N = 10$, $C = 1$

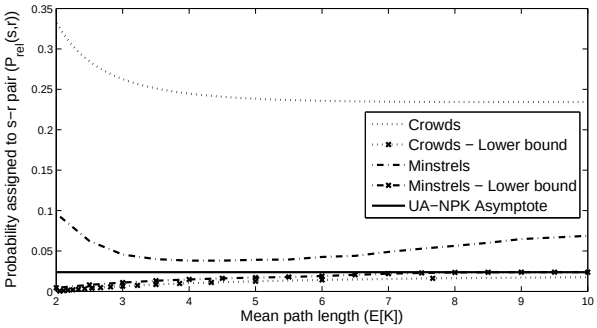

Fig. 3. Relationship anonymity vs. overhead for $N = 10$, $C = 3$

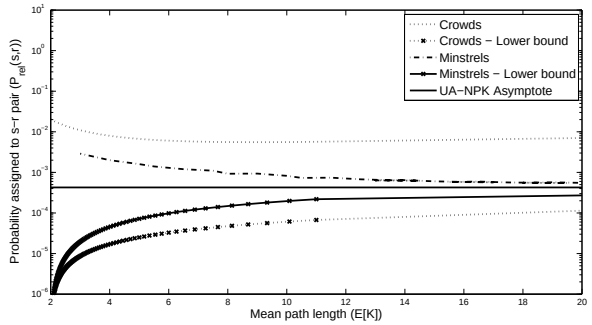

Fig. 4. Relationship anonymity vs. overhead for $N = 50$, $C = 1$

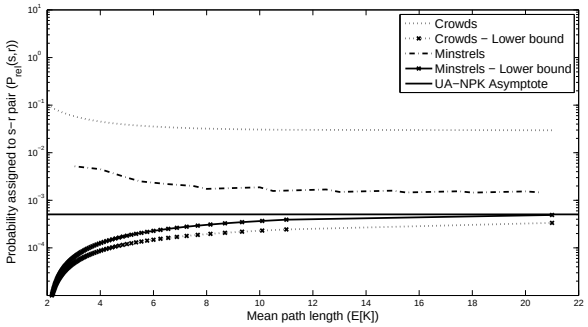

Fig. 5. Relationship anonymity vs. overhead for $N = 50$, $C = 5$

bounds converge to an asymptote, which corresponds to the case when there is always an attacker on the path ($P(H_{1+}) = 1$), and the attacker assigns $P_{rel}(s,r) = \frac{1}{(N-C)(N-C-1)}$ to every possible sender-receiver pair. We refer to this asymptote as the ubiquitous attacker with no protocol knowledge (UA-NPK). The anonymity provided by Minstrels closely approaches the asymptote, but only for moderately high overheads. For low and high overheads the protocol exposes information that the attacker can use to narrow down the possible set of sender-receiver pairs. The relationship anonymity provided by Crowds is significantly worse than the lower bound, which is primarily due to the lack of receiver anonymity.

These results show that the selection of the overhead and hence the average path length in anonymity systems requires a careful consideration of the attacker model. In general, the best possible relationship anonymity might not be provided by the highest allowable overhead.

VI. Conclusions and Future Work

In this paper we made a first attempt to analyze the trade-off between relationship anonymity and communication overhead in anonymity networks. We considered two anonymity networks, Crowds proposed in [6] and Minstrels proposed in this work. Minstrels aims at providing relationship anonymity by hiding both the sender and the receiver, but in order to bound the maximum path length the anonymity provided to the sender is slightly weaker than in Crowds. We expressed the relationship anonymity for these networks, and provided simple lower bounds on the probability assigned to a sender-receiver pair. While intuition says that increased overhead should lead to improved relationship anonymity, our results show this is not the case in general. Instead, anonymity is often easiest to provide at medium levels of overhead, when attackers are still unlikely to be on the path, but the sender-receiver identity is already reasonably well protected. It is subject of our future work to provide a more complete characterization of the overhead-anonymity trade-off for anonymity networks, including networks that provide probabilistic message delivery.

References

[1] D. Dzung, M. Naedele, T. V. Hoff, and M. Crevatin "Security for Industrial Communication Systems." *Proc. IEEE* vol. 82, pp. 6 1152-1177, 2005.

[2] C. W. Ten, C. C. Liu and M. Govindarasu "Vulnerability Assessment of Cybersecurity for SCADA Systems." *IEEE Trans. Power Syst.*, vol. 23, no. 4, 2008.

[3] D. Chaum "Untraceable electronic mail, return addresses and digital pseudonyms" *Commun. of the ACM 24(2)*, pp. 84-88, 1981

[4] A. Pfitzman, M. Köhntopp "Anonymity, unobservability, and pseudonymity - a proposal for terminology" *Anonymity 2000*, pp. 1-9, 2000

[5] P. Syverson, D. Goldschlag, and M. Reed "Anonymous connections and onion routing." *in Proc. IEEE Symp. on Security and Privacy*, pp. 44-54, Oakland, California, May 1997.

[6] M. Reiter and A. Rubin "Crowds: Anonymity for Web Transactions." *ACM Trans. Inform. Syst. Security*, pp. 66-92, 1998.

[7] G. Danezis, C. Díaz, E. Käsper, and C. Troncoso "The wisdom of Crowds: attacks and optimal constructions" in Proc. of ESORICS 2009.

[8] V. Shmatikov and M. H. Wang "Measuring Relationship Anonymity in Mix Networks" in Proc. of WPES 2006.

[9] J. Feigenbaum, A. Johnson, and P. Syverson "Probabilistic Analysis of Onion Routing in a Black-box Model" in Proc. of WPES 2007.

[10] M. Wright, M. Adler, B. N. Levine, and C. Shields "The Predecessor Attack: An Analysis of a Threat to Anonymous Communications Systems" *ACM Trans. Inform. Syst. Security (TISSEC) 4(7)*, pp. 489-522, November 2004

[11] O. Vuković, G. Dán and G. Karlsson. "On the Trade-off Between Relationship Anonymity and Communication Overhead in Anonymity Networks" *Technical Report, KTH, TRITA-EE 2010:035*, July 2010.