Measuring Anonymity in a Non-adaptive, Real-time System

Gergely Tóth, Zoltán Hornák

Budapest University of Technology and Economics
Department of Measurement and Information Systems
H-1111 Budapest, XI. Magyar tudósok krt. 2., Building I. B414.
{tgm,hornak}@mit.bme.hu

Abstract. Anonymous message transmission should be a key feature in network architectures ensuring that delivered messages are impossible—or at least infeasible—to be traced back to their senders. For this purpose the formal model of the non-adaptive, real-time PROB-channel will be introduced. In this model attackers try to circumvent applied protection measures and to link senders to delivered messages. In order to formally measure the level of anonymity provided by the system, the probability will be given, with which observers can determine the senders of delivered messages (source-hiding property) or the recipients of sent messages (destination-hiding property). In order to reduce the certainty of an observer, possible counter-measures will be defined that will ensure specified upper limit for the probability with which an observer can mark someone as the sender or recipient of a message. Finally results of simulations will be shown to demonstrate the strength of the techniques.

1 Introduction

Anonymous message transmission techniques, such as MIX-net [1] or Onion Routing [2] aim to guarantee that no delivered message can be traced back to its sender. Research on such methods is currently under development but their theoretical analysis and description is not complete. Anonymous message transmission may be used for several real-life scenarios: in anonymous electronic election systems, in anonymous on-line shopping, in anonymous medical consulting and education or simply in electronic mailing.

Recent research in the field of anonymity focuses mainly on adaptive techniques [8,17]. Our approach on the contrary analyses a scenario, where the intermediate node providing anonymity is non-adaptive (i.e. message delay is independent of the actual message distribution). This way a truly real-time system can be constructed, where message-delay has a guaranteed maximum. Although there are connection-based systems among the active ones that aim to allow low-latency communication [2,13], they sacrifice aspects of the techniques described in this paper (e.g. mixing, dummy traffic) in order to become fast—on the other hand however they become vulnerable to some attacks as shown in [5].

In this paper we focus largely on the probabilities with which an attacker can compromise the anonymity provided by our system. A similar approach is shown in Kesdogan et al. for the SG-MIX protocol [16]. Our approach is different in that they specify the user to determine the delay of a packet while traversing the channel, whereas in our model the channel is responsible for determining the delay.

In this paper we will consider only one relaying node (the PROB-channel) for providing anonymity. Reason for this is to analyze this simple scenario first as deeply as possible. Afterwards if the provided anonymity was evaluated, cascading our node similarly to de idea of MIX-nodes [1] will enable a more sophisticated construction. However this approach is out of scope for this paper.

We first introduce the formal model of the PROB-channel and explore what conclusions a passive observer can draw by only knowing public parameters and timing of events (sending & delivery time). Based on the model the source and destination hiding properties will be defined, which can act as a numerical measure for anonymity. The aim of these measures is the same as in [6,11]—to quantify the anonymity provided by the system. However instead of the entropy of the probability distribution we use the maximum of the probabilities for our quantitative analysis. Requirements necessary to limit the certainty of the adversary observer and to ensure given level of anonymity will also be introduced. Finally simulation results will be discussed that give a basic understanding about the operation of the channel.

2 Model of the PROB-channel

The PROB-channel is responsible for providing anonymity in a scenario where senders send messages to recipients. First let us define the main characteristics of the channel informally:

- The channel is *real-time*, thus messages will be delivered before a message-invariant maximal delay. Other systems may work on a best effort basis (e.g. connection-based techniques: Onion Routing [2]) or do not consider time limits at all (e.g. MIX-nets [1]).
- The channel is *non-adaptive*, as its operation is not affected by properties and distribution of incoming messages, i.e. delay has static distribution. Other solutions prefer active operation, where the system is adaptive to the traffic at the expense of real-time guarantee (e.g. MIXMaster [15]).
- All input and output of the channel is *observable*, so an observer can detect all incoming and delivered messages.
- The channel is a *black-box*, since it is analyzed as a whole. The internal implementation is not specified and side-channel attacks are not considered. The observer cannot see what happens to the messages inside the channel and how they are encoded and delivered.
- The PROB-channel is required as there should be no direct connection between a sender and the receiver. As only one relaying node is inserted into the network topology, our system is a single *proxy* (just like anonymizer.com [14]). Other solutions, where no single relay can be trusted any more employ distributed systems with many relays forming a graph (e.g. Crowds [7]).

- We furthermore assume that messages passing through the channel are equally sized and properly encrypted, thus an observer can only draw conclusions from the timing of the messages, *content does not provide information*. This condition can easily be satisfied.

Our analysis started with the PROB-channel so that future evaluation of cascaded and active techniques can build on the conclusions drawn from this simple non-adaptive channel. We chose a real-time system, as our aim is to employ anonymity in interactive on-line services (e.g. web-browsing), where delay needs to be reduced below a certain limit. We use a black-box proxy model since we did not go into details about internal structure of the channel and left it as an open question how the transformation between sent and delivered messages will be realized. Finally an observable model was chosen since if one cannot be sure about what a potential observer might not perceive, then the worst should be assumed that he could perceive everything.

The model of the PROB-channel is the basis of our work considering anonymous message transmission techniques. Based on the results demonstrated in this paper future analysis will concentrate on active adversaries, which will probably require the usage of active channels. In the following in this chapter we will continue with the formal definition of the PROB-channel and introduce the adversary.

2.1 Description of the Environment

Let S denote the set of senders, R the set of recipients, and M the set of messages. Let $S(m_i)$ denote the sender of message m_i, $R(m_i)$ the recipient of message m_i, whereas $t_S(m_i)$ the time of sending of message m_i and $t_R(m_i)$ the time if delivery of message m_i. The system operates in continuous time, thus events cannot happen at the same time (no parallel entry into the channel). Time of transporting the message from the sender to the channel and from the channel to the recipient will not be considered. This simplification does not substantially affect the conclusions drawn.

2.2 Specification of the Channel

The channel delivers messages from senders to recipients. No messages are born inside the channel and messages won't be dropped by the channel. An incoming message from its sender will be delivered to its recipient after a delay with the following properties:

- the delay δ is a probability variable with a given $f(\delta)$ density function, $\delta = t_R - t_S$, where δ is message- and time-invariant;
- the channel will deliver all messages before a predefined, message- and time-invariant maximal delay $\delta_{\max}$ (time-to-live) and after a predefined, message- and time-invariant minimal delay $\delta_{\min}$, thus $\underset{m_i}{\forall}[\delta_{\min} < t_R(m_i) - t_S(m_i) < \delta_{\max}]$.

Therefore channel C can be characterized by the parameters $f(\delta)$, $\delta_{\min}$, $\delta_{\max}$.

2.3 Message sending

In the following we assume that sender $s_a \in S$, $s_a = S(m_i)$ sends a message $m_i \in M$ to recipient $r_b \in R$, $r_b = R(m_i)$. Message m_i and the recipient's ID enter the channel in the encrypted form $\alpha_i := E_S(r_b, m_i)$ at time $t_S(m_i) = t_S(\alpha_i)$, whereas m_i will be delivered to the recipient in the form $\beta_i := E_R(m_i)$, encrypted with a different key, at time $t_R(m_i) = t_R(\beta_i)$ (see Fig. 1).

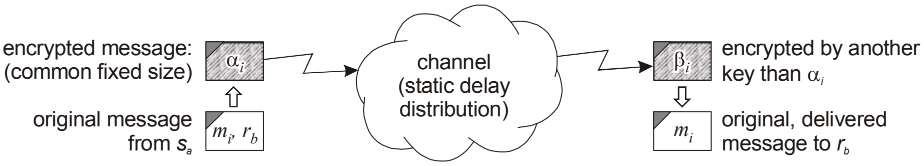
encrypted message:
(common fixed size)
α_i
original message from s_a
m_i, r_b
channel
(static delay distribution)
β_i
encrypted by another key than α_i
m_i
original, delivered message to r_b

Fig. 1. Message sending through the PROB-channel

We assume furthermore that the adversary cannot break the applied encryption, thus he can decode nether α_i nor β_i. This could be achieved for example if at startup of the system each sender and recipient agreed a symmetric key with the channel (e.g. with the help of Diffie-Hellman protocol). Afterwards the sender s_a would use his key to encrypt the address of r_b together with the message m_i to form α_i. The channel would decrypt this packet, re-encrypt the message m_i with the recipient's key (thus create β_i) and forward it after the delay to the recipient, who could finally decrypt it with his key. Of course this simple scenario implies that the channel gains access to the contents of the plain message. However using security protocols (e.g. SSL, TLS) over the services offered by the PROB-channel would eliminate this problem.

2.4 The Observer

Let us now state what are the capabilities of a passive observer in this model. Such an observer can only eavesdrop encrypted messages, he cannot decrypt them (unless sent to him) nor can he modify, delete, replay or delay[1] messages. The aim of the observer is to match delivered messages (β_i) with their senders — or at least guess the link with good probability — and so get information about who communicates with whom.

We assume that the observer can eavesdrop all ends of the channel, this way he knows all encrypted messages sent and their time of sending, all delivered encrypted messages and their time of receipt. He also knows the parameters and the environment of the channel. Thus he is in possession of the following information:

- environment (S, R) and parameters ($f(\delta)$, $\delta_{\min}$, $\delta_{\max}$) of the channel;
- $\varepsilon_S := \{\alpha_i := E_S[S(m_i), m_i]\}$, $\vartheta_S := \{t_S(\alpha_i)\}$ — sent messages and their time of sending;
- $\varepsilon_R := \{\beta_i := E_R[m_i]\}$, $\vartheta_R := \{t_R(\beta_i)\}$ — received messages and their time of receipt.

[1] Note that besides the traditional manipulation techniques an attacker can also delay messages in order to compromise anonymity.

This could be summarized as a passive adversary with knowledge of the system parameters.

Let Ψ denote the history of the system given by the following parameters: C, S, R, ε_S, ε_R, ϑ_S, ϑ_R. In the following we assume that the observer knows the full history Ψ of the system and he can perceive all the observable properties during the whole operation of the system. As we will see, the probability that a delivered message can be traced back to its sender, can even in this case be limited.

3 Confidence of the Observer

Let a specific history of the system be $\Psi^* := (C^*, S^*, R^*, \varepsilon_S^*, \varepsilon_R^*, \vartheta_S^*, \vartheta_R^*)$. In order to evaluate, which sender sent which message, for each delivered message β_k^* and for each sender s_l^* a probability $P_{\beta_k^*, s_l^*, \Psi^*}$ can be determined. If the observer knows the history Ψ^* of the system, he can conclude that β_k^* was sent by s_l^* with the probability $P_{\beta_k^*, s_l^*, \Psi^*}$:

$$P_{\beta_k^*, s_l^*, \Psi^*} = P[S(\beta_k^*) = s_l^* \mid \Psi = \Psi^*] \qquad \textbf{(1)}$$

The observer naturally looks for the most probable source where $P_{\beta_k^*, \Psi^*} := \max_{s_l^*} P_{\beta_k^*, s_l^*, \Psi^*}$.

In order to trace back the messages to their senders the observer calculates the probabilities (1) and marks the most probable sender as the potential real sender of the message in question.

Equation (1) only formulates the aim of the observer, how the respective values could be calculated is not yet defined. In the following sections we are going to show two techniques (global and local back-tracing) that specify, how the adversary might calculate the numerical values from the history Ψ^* of the system.

The following sets need to be defined for simplifying upcoming equations. Let $\mu_{\beta_k^*, \Psi^*}$ denote the set of encrypted sent messages α_j^*, which might have left the channel as β_k^* (2) considering the properties of $f^*(\delta)$. Furthermore let $\eta_{\beta_k^*, s_l^*, \Psi^*}$ denote the set of α_j^* in $\mu_{\beta_k^*, \Psi^*}$, which were sent by s_l^* (3).

$$\mu_{\beta_k^*, \Psi^*} := \{\alpha_j^* \mid [t_R(\beta_k^*) - \delta_{\max}^*] < t_S(\alpha_j^*) < [t_R(\beta_k^*) - \delta_{\min}^*]\} \qquad \textbf{(2)}$$

$$\eta_{\beta_k^*, s_l^*, \Psi^*} := \{\alpha_j^* \mid [\alpha_j^* \in \mu_{\beta_k^*, \Psi^*}] \wedge [S(\alpha_j^*) = s_l^*]\} \qquad \textbf{(3)}$$

3.1 Global Back-tracing

In order to compute the probabilities in (1) the obvious and optimal solution would be to perform global back-tracing, thus the observer would try all possibilities and choose the most probable one.

In order to do this, one has to generate all possible match combinations (the g_i-s) of sent and received messages (4). A match $g_i := \left\langle g_i^1, g_i^2, \ldots, g_i^{|M^*|} \right\rangle$ means that the delivered encrypted message β_k^* entered the channel as $g_i^k = \alpha_j^* \in \varepsilon_S^*$.

$$G_{\Psi^*} := \{g_i := \left\langle g_i^1, g_i^2, \ldots, g_i^{|M^*|} \right\rangle \mid [g_i \in \bigtimes_{1 \le k \le |M^*|} \mu_{\beta_k^*, \Psi^*}] \wedge [\underset{1 \le j \le |M^*|}{\forall} \; \underset{\substack{(1 \le k \le |M^*|) \wedge \\ (j \ne k)}}{\forall} (g_i^j \ne g_i^k)]\} \tag{4}$$

After having all match combinations G_{Ψ^*}, based upon their probabilities the observer can calculate (1) as follows:

$$P_{\beta_k^*, s_l^*, \Psi^*} = \sum_{S(g_i^k) = s_l^*} P(g_i \mid \Psi = \Psi^*) \tag{5}$$

In order to get the values for (1), the probability of the matches (g_i-s)— which state that the delivered message (β_k^*) entered the channel from the respective sender s_l^*— need to be added up.

As it will be shown in section 6, uniformly distributed delay provides system optimum. In this case each g_i is equally probable, thus the probabilities can be calculated as follows:

$$P(g_i \mid \Psi = \Psi^*) = \frac{1}{|G_{\Psi^*}|} \tag{6}$$

Unfortunately global back-tracing is exponential by the number of sent messages and thus ineffective for practical use.

3.2 Local Back-tracing

If the observer performs the delivered message → sender matching for each delivered message independently, then equation (7) gives the probability that s_l^* is the sender of β_k^* if history Ψ^* is known—a possible algorithm for (1).

$$P_{\beta_k^*, s_l^*, \Psi^*} = \frac{\sum\limits_{\substack{\forall [\alpha_i^* \in \eta_{\beta_k^*, s_l^*, \Psi^*}] \\ \alpha_i^*}} f^*[t_R(\beta_k^*) - t_S(\alpha_i^*)]}{\sum\limits_{\substack{\forall [\alpha_j^* \in \mu_{\beta_k^*, \Psi^*}] \\ \alpha_j^*}} f^*[t_R(\beta_k^*) - t_S(\alpha_j^*)]} \tag{7}$$

Equation (7) gives the probability as a quotient of the sums of the delay density function's values: in the numerator summation is done on the set of messages sent by the particular sender (i.e. any of his sent messages could have become this particular delivered message) and in the denominator summation is done on all sent messages in the respective time interval (i.e. this sum is constant for all possible senders for a particular delivered message).

Unfortunately local back-tracing has a great disadvantage. Originating from its local aspect even in a very simple scenario it can produce false results. Assume the following: two senders (s_0 and s_1) send messages to two recipients (r_0 and r_1) through the channel ($\delta_{\min} = 1$ and $\delta_{\max} = 4$ with uniform distribution). Messages are sent and delivered as follows:

Table 1. Message distribution example

Sent message	Sender	Time of sending	Delivered message	Recipient	Time of receipt
α_0	s_0	1.0	β_0	r_1	3.0
α_1	s_1	2.1	β_1	r_1	4.9
α_2	s_1	4.0	β_2	r_0	6.0
α_3	s_0	5.1	β_3	r_0	7.9

This example message distribution is shown on Fig. 2. It is obvious that β_0 can only originate from α_0, which implies that α_0 could not become β_1 and so on. However local back-tracing cannot handle this condition and considers α_0 for the calculations for β_1 and comes to an inadequate result: only β_0 and β_2 would be guessed correctly despite all messages in this scenario being tracable.

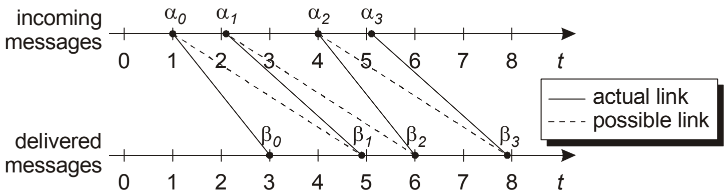

Fig. 2. Example message distribution

In this example the observer performing local back-tracing is only able to compromize the anonymity of the delivered messages β_0 and β_2. However for β_1 and β_2 both senders appear as potential subjects with an equal probability.

This example clearly illustrates the weakness of local back-tracing. It is to note that global back-tracing would have successfully linked incoming and delivered messages. However since only local back-tracing is feasible especially for larger sets of messages, in our work we will use locally back-tracing techniques for the drawn conclusions.

It has to be emphasized that a defense against a locally back-tracing observer is not guaranteed to work against an adversary performing global back-tracing. Our assumption is however that under special circumstances (see the MIN/MAX property with uniform message delay distribution in section 6) the history of the system can become resilient against both kinds of adversaries. On the other hand if the senders don't produce enough messages, in degenerate cases local back-tracing might not detect the real matching (e.g. if the sets $\mu_{\beta_k^*,\Psi^*}$ are small).

4 Source- and Destination-Hiding Property

History $\Psi = (C, S, R, \varepsilon_S, \varepsilon_R, \vartheta_S, \vartheta_R)$ of a system is *source-hiding* with parameter Θ if the observer cannot assign a sender to any delivered message β_k with a probability greater than Θ:

$$\underset{\beta_k \in \varepsilon_R}{\forall} P_{\beta_k,\Psi} \leq \Theta \qquad \textbf{(8)}$$

Pfitzmann and Köhntopp [3] defined in their paper the term *sender anonymity*. Translated to the model of the PROB-channel this would mean that delivered messages are not linkable to a sender. Thus the source hiding property can be seen as a numerical measure for the sender anonymity. The aim of this measure is the same as in [6,11]—to quantify the quite elusive notion anonymity, however instead of the entropy of the probability distribution we use the maximal probability for our quantitative analysis. We have chosen this new measure for a simple reason: it is much more intuitive and does not disregard the important aspects of a practical measure.

Respectively also *recipient anonymity* was also defined. In our model this would mean that sent messages are not linkable to a recipient. For this purpose the destination-hiding property can be introduced.

Similarly to (1) the probability $P_{\alpha_j^*,r_l^*,\Psi^*}$ can be defined for each sent message α_j^* and for each recipient r_l^*. If the observer knows the history Ψ^* of the system, he can conclude that α_j^* was received by r_l^* with the probability $P_{\alpha_j^*,r_l^*,\Psi^*}$:

$$P_{\alpha_j^*,r_l^*,\Psi^*} = P[R(\alpha_j^*) = r_l^* \mid \Psi = \Psi^*] \qquad \textbf{(9)}$$

The observer naturally looks for the most probable destination where $P_{\alpha_j^*,\Psi^*} := \max_{r_l^*} P_{\alpha_j^*,r_l^*,\Psi^*}$.

Finally definition of the destination-hiding property is as follows: history $\Psi = (C, S, R, \varepsilon_S, \varepsilon_R, \vartheta_S, \vartheta_R)$ of a system is *destination-hiding* with parameter Ω if the observer cannot assign a recipient to any sent message α_j with a probability greater than Ω:

$$\underset{\alpha_j \in \varepsilon_S}{\forall} P_{\alpha_j,\Psi} \leq \Omega \qquad \textbf{(10)}$$

Naturally the observer can apply similar global and local back-tracing methods in order to compromise recipient anonymity as those defined in sections 3.1 and 3.2.

5 MIN/MAX Property

In order to be able to limit the possible value of equation (7) influencing the source-hiding property even in the worst case, restrictions have to be applied for the intervals between message sendings:

- First, summation in the numerator has to be performed on the smallest possible set. In order to achieve this, senders should not be allowed send more than one message in a given time interval.
- Second, summation in the denominator has to be performed on the greatest possible set. In order to achieve this, senders should be obliged to send at least one message in a given time interval. (If it is otherwise not achievable, senders should send *dummy* messages to randomly chosen recipients.)

The effect of dummy messages on anonymity has been analyzed by Berthold and Langos [9]. As it has been evaluated thoroughly, we do not handle requirements for contents, here only the frequency range for sending such messages is analyzed.

Considering the above limitations, history $\Psi = (C, S, R, \varepsilon_S, \varepsilon_R, \vartheta_S, \vartheta_R)$ of a system possesses the MIN/MAX property with parameters $\tau_{\min}$, $\tau_{\max}$ ($\tau_{\min} \leq \tau_{\max}$), if it holds that no sender sends more than one message within a time interval $\tau_{\min}$ (11) and all senders send at least one message in a time interval $\tau_{\max}$ (12).

$$\underset{s_l \in S}{\forall} \; \underset{\alpha_j | S(\alpha_j) = s_l}{\forall} \; \xi_{s_l,\alpha_j} = \varnothing \qquad \textbf{(11)}$$

$$\underset{s_l \in S}{\forall} \; \underset{\alpha_j | S(\alpha_j) = s_l}{\forall} \; \neg(\zeta_{s_l,\alpha_j} = \varnothing) \qquad \textbf{(12)}$$

Where ξ_{s_l,α_j} is the set of sent encrypted messages, which were sent by sender s_l maximal $\tau_{\min}$ after sending α_j (13) and ζ_{s_l,α_j} is the set of sent encrypted messages, which were sent by s_l maximal $\tau_{\max}$ after sending α_j (14).

$$\xi_{s_l,\alpha_j} := \{\alpha_i \mid (S(\alpha_i) = s_l) \wedge (t_S(\alpha_i) > t_S(\alpha_j)) \wedge ([t_S(\alpha_i) - t_S(\alpha_j)] < \tau_{\min})\} \qquad \textbf{(13)}$$

$$\zeta_{s_l,\alpha_j} := \{\alpha_i \mid (S(\alpha_i) = s_l) \wedge (t_S(\alpha_i) > t_S(\alpha_j)) \wedge ([t_S(\alpha_i) - t_S(\alpha_j)] < \tau_{\max})\} \qquad \textbf{(14)}$$

If these conditions hold, for the probabilities (1) assigned to any delivered encrypted message and sender, a message-invariant upper limit $\hat{P}_\Psi$ can be given (15), and thus the source-hiding property can be guaranteed (assuming $\tau_{\max} \leq [\delta_{\max} - \delta_{\min}]$):

$$P_{\beta_k,\Psi} \leq \Theta = \hat{P}_\Psi = \frac{\sum_{i=1}^{\Delta_{\min}} \max_{(i-1)\cdot\tau_{\min} \leq q < i\cdot\tau_{\min}} f(q)}{|S| \cdot \sum_{i=1}^{\Delta_{\max}} \min_{(i-1)\cdot\tau_{\max} \leq q < i\cdot\tau_{\max}} f(q)} \qquad \textbf{(15)}$$

Where $\Delta_{\max} = \left\lfloor \frac{\delta_{\max} - \delta_{\min}}{\tau_{\max}} \right\rfloor$ and $\Delta_{\min} = \left\lceil \frac{\delta_{\max} - \delta_{\min}}{\tau_{\min}} \right\rceil$.

Unfortunately the same approach does not work for the destination-hiding property. The frequency of sending messages may be specified for the senders but the frequency of receipt cannot be specified for the recipients. Either the senders have to send messages uniformly distributed to all the recipients or the channel has to create dummy messages in order to ensure that each recipient receives the same amount of messages (with the same distribution). However coordinating the senders in a distributed environment seems to be difficult. On the other hand the option of dummy messages created by the channel moves us into the category of active channels, which is not the scope of this paper. Ultimately we have to realize that with the limitations of the PROB-channel the destination-hiding property cannot be realized efficiently.

6 Optimum — Uniformly Distributed Delay

Coming back to the source-hiding property, in the best case—while doing the local back-tracing—the observer can only pick randomly for a delivered message from those who sent a message in the relevant time frame ($\delta_{\min} - \delta_{\max}$).

Is the distribution $f(\delta)$ of the delay in a channel uniform (between $\delta_{\min}$ and $\delta_{\max}$ $f(\delta) = f_{\max}$, otherwise zero), then with a history $\Psi = (C, S, R, \varepsilon_S, \varepsilon_R, \vartheta_S, \vartheta_R)$ for all delivered encrypted messages β_k we get:

$$P_{\beta_k,\Psi} = \frac{\max\limits_{s_l} \left| \eta_{\beta_k, s_l, \Psi} \right|}{\left| \mu_{\beta_k,\Psi} \right|} \tag{16}$$

If Ψ has the MIN/MAX property with parameters $\tau_{\min}$, $\tau_{\max}$ ($\tau_{\max} \leq [\delta_{\max} - \delta_{\min}]$), then the upper limit in (15) can be brought into a simpler form:

$$P_{\beta_k,\Psi} \leq \Theta = \hat{P}_\Psi = \frac{\Delta_{\min}}{|S| \cdot \Delta_{\max}} \approx \frac{\tau_{\max}}{|S| \cdot \tau_{\min}} \tag{17}$$

If Ψ also fulfills the condition $\tau_{\min} = \tau_{\max}$ ($\tau_{\max} \leq [\delta_{\max} - \delta_{\min}]$), meaning that each sender sends messages with a period exactly $\tau_{\min} = \tau_{\max}$, then the history of the system reaches the global optimum and the observer has to pick the sender for each delivered encrypted message randomly from all senders (from S):

$$P_{\beta_k,\Psi} \leq \Theta = \hat{P}_\Psi \approx \frac{1}{|S|} \tag{18}$$

Interpreting these we can formulate that with uniformly distributed delay the observer does not achieve anything by eavesdropping, he has to pick randomly from the senders who sent a message in the relevant time frame. If the senders satisfy the MIN/MAX conditions as well, then the level of anonymity can be controlled exactly.

7 Simulation Results

In this section simulation results will be introduced. Basically the following two aspects will be illustrated:

- difference between *general* (see section 7.1) and *MIN/MAX* (see section 7.2) message sending and
- difference between *non-uniform* (triangle, see Fig. 3) and *uniform* distribution.

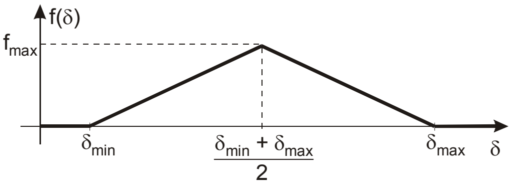

Fig. 3. Triangle distribution

According to the categories above, four simulation scenarios can be defined. For each scenario, the following parameters were the same:

- there were 20 senders and 20 receivers;
- $\delta_{\text{min}} = 1$ and $\delta_{\text{max}} = 4$;
- simulation duration T = 2000: each sender was sending messages between time index 0 and T.

Each simulation scenario was repeated 20 times and the average of the results weighted with the total number of messages in the actual run are discussed in the following.

While the observer was performing local back-tracing three variables were maintained after the calculations for each message. Before each run these three variables were initialized to 0.

- *sure*—if the observer could successfully link the delivered message to its actual sender[2] then *sure* is increased by 1;
- *maybe*—if there were q senders with the same probability of sending the specified message then *maybe* is increased by q^{-1} and *failed* is increased by $1-q^{-1}$;
- *failed*—if the observer could not link the delivered message to its actual sender (i.e. he linked the message to the wrong sender) then *failed* is increased by 1.

Note. In the following *Sure*, *Maybe* and *Failed* measure are the weighted averages of *sure*, *maybe* and *failed* divided by the average number of messages. On the diagrams below the quantity *Maybe* + *Sure* (the ratio with which the observer linked successfully messages to their real senders) is shown. As *Failed* = 1 – (*Maybe* + *Sure*), it is not shown on the diagrams.

[2] In order to check, whether the observer linked the right sender to a delivered message in the simulation there was an entity that knew the real sender of each message and this entity decided, whether the observer was successful or not.

7.1 General message sending

This section analyses the difference between uniform and another (in this case the triangle) distribution. For the scenarios *general* senders were used. Behavior of such senders is characterized by the parameter U:

- at initialization each general sender generates its own maximal delay U_{max}, which is a random number in the interval 0...U;
- then the sender repeatedly generates a random number in the interval $0...U_{max}$, waits this amount of time and then sends a message to a randomly chosen receiver;
- message sending stops if a message sending happened after time T.

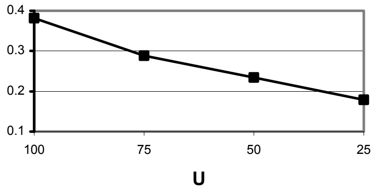

Fig. 4. General senders with triangle distribution

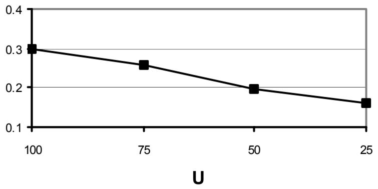

Fig. 5. General senders with uniform distribution

For numerical values, see section A.1 of the Appendix. It can clearly be seen on the diagrams above (Fig. 4 and 5) that uniform distribution reduces the chances of the observer significantly in contrast to another (in this case triangle) distribution.

7.2 MIN/MAX message sending

MIN/MAX sending was performed with parameter τ_{min} = 0.9. Value of τ_{max} was chosen to be 1.0, 1.5, 2.0, 2.5 and 2.95. MIN/MAX senders were sending messages to randomly chosen recipients with random intervals between the message sending according to the appropriate τ_{min}, τ_{max} restrictions.

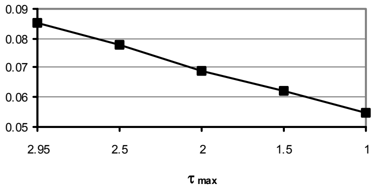

Fig. 6. MIN/MAX senders with triangle distribution

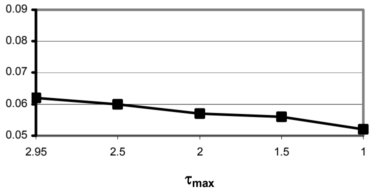

Fig. 7. MIN/MAX senders with uniform distribution

For numerical values, see section A.2 of the Appendix.

An increase in the number of messages could be observed, which implied basically the greater *Failed* ratio. Although uniform distribution (Fig. 7) is still better than the triangle one (Fig. 6), the absolute difference is not that substantial any more—the relative difference is still significant.

It can clearly be seen that the uniform distribution's $\hat{P}_{\psi}$ guarantees strong source-hiding property. Also note that the theoretical minimum of 0.05 for the certainty of the observer with 20 senders is almost achieved with uniform distribution with τ_{min} = 0.9 and τ_{max} = 1.0 (the actual value was 0.067, see section A.3 of the Appendix for more details).

It should also be mentioned that originating from the form of the triangle distribution equation (15) could not give usable upper limit for the certainty of the observer, thus source-hiding property in that case could not be guaranteed.

8 Conclusion

In this paper the formal model of the PROB-channel was introduced. Assuming a passive observer, we analyze what conclusions could be drawn for a non-adaptive, real-time relaying node based solely on observation of the timing of events and the parameters of the channel. With the help of a numerical measure of sender anonymity—the source hiding property—we show that the MIN/MAX approach combined with the optimal uniformly distributed delay, successfully prevents a locally back-tracing observer breaking the security of the PROB-channel. On the other hand global back-tracing (while having an exponentional computational complexity) could achieve much better results against our system. Our assumption however is that under the circumstances pointed out in chapter 6 (e.g. MIN/MAX property and uniform distribution) the difference between global and local back-tracing is not substantial. Future analysis needs to prove this assumption.

Further research is also required to find out how our model has to be altered in order to guarantee recipient anonymity efficiently. It should be evaluated, how cascading such nodes can improve the resistance of the anonymity system against active attackers. Finally it should be investigated what an active attacker can accomplish against the model described here and how the PROB-channel should be extended to successfully protect against an active opponent. Probably an active channel is required that would dynamically react to the distribution of actual message arrivals.

Acknowledgements

The authors would like to thank George Danezis for his helpful comments and the anonymous referees for useful suggestions.

References

1. Chaum, D.: Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. In *Communications of the ACM*, volume 24, number 2, pp. 84-88, 1981
2. Reed, M., Syverson, P., Goldschlag, D.: Anonymous Connections and Onion Routing. In *IEEE Journal on Selected Areas in Communication Special Issue on Copyright and Privacy Protection*, pp. 482-494, 1998
3. Pfitzman, A., Kohntopp, M.: Anonymity, Unobservability, and Pseudonymity – A Proposal for Terminology. In *Designing Privacy Enhancing Technologies*, Federrath, H. (editor), Springer-Verlag LNCS 2009, pp. 1–9, 2001
4. Back, A., Möller, U., Stiglic, A.: Traffic Analysis Attacks and Trade-Offs in Anonymity Providing Systems. In *Information Hiding: 4th International Workshop*, Moskowitz, I.S. (editor), Springer-Verlag LNCS 2137, pp. 245-257, 2001
5. Serjantov, A., Sewell, P.: Passive Attack Analysis for Connection-Based Anonymity Systems. In *ESORICS 2003*, Snekkens, E., Gollmann., D. (editors), Springer-Verlag LNCS 2808, pp. 116-131, 2003
6. Serjantov, A., Danezis, G.: Towards an Information Theoretic Metric for Anonymity, in *Privacy Enhancing Technologies (PET2002)*, Syverson, P., Dingledine, R. (editors), Springer-Verlag LNCS 2482, pp. 41-53, 2002
7. Reiter, M. K., Rubin, A. D.: Crowds: Anonymity for Web Transactions, *ACM Transactions on Information and System Security*, volume 1, number 1, pp. 66-92, 1998
8. Berthold, O., Federrath, H., Köhntopp, M.: Project "Anonymity and Unobservability in the Internet", *Conference on Freedom and Privacy 2000 (CFP 2000), Workshop on Freedom and Privacy by Design*, 2000
9. Berthold, O., Langos, H.: Dummy traffic against long term intersection attacks, in *Privacy Enhancing Technologies (PET2002)*, Syverson, P., Dingledine, R. (editors), Springer-Verlag LNCS 2482, pp. 110-128, 2002
10. Diaz, C., Seys, S., Claessens, J., Preneel, B.: Towards measuring anonymity, *in Privacy Enhancing Technologies (PET2002)*, Syverson, P., Dingledine, R. (editors), Springer-Verlag LNCS 2482, pp 54-68, 2002
11. Newman, R. E., Moskowitz, I. S., Syverson, P., Serjantov, A.: Metrics for Traffic Analysis Prevention. In *Privacy Enhancing Technologies (PET2003)*, Dingledine, R. (editor), Springer Verlag LNCS, 2003
12. Wright, M., Adler, M., Levine, B. N., Shields, C.: Defending Anonymous Communications Against Passive Logging Attacks. In *ISOC Symposium on Network and Distributed System Security*. 2002
13. Rennhard, M., Plattner, B.: Introducing Morphmix: Peer-to-Peer based Anonymous Internet Usage with Collusion Detection. In *Workshop on Privacy and Electronic Society (WPES)*, 2002
14. Anonymizer.com—Online Privacy Services, `http://www.anonymizer.com`
15. Möller, U., Cottrell, L., Palfrader, P., Sassaman, L.: Mixmaster Protocol — Version 2. Draft, 2002
16. Kesdogan, D., Egner, J., Büschkes, R.: Stop–and–Go MIXes: Providing Probabilistic Anonymity in an Open System. In *Information Hiding Workshop (IH1998)*, Springer Verlag, LNCS 1525, 1998
17. Boucher, P., Shostack, A., Goldberg, I.: Freedom Systems 2.0 Architecture. White paper, Zero Knowledge Systems Inc., 2000

Appendix A

In the Appendix numerical values of the simulations will be given.

A.1 Values for General Message Sending

In this section numerical values of the simulation of general senders (see section 9.1) will be given. General senders don't follow constraints to increase their anonymity, they send messages randomly. Table 2 shows the results with triangle distribution, whereas values for the uniform distribution can be found in Table 3.

Table 2. Results for general senders with triangle distribution

U	*Sure*	*Maybe*	*Failed*	Number of Messages
100	0.382	0	0.618	2811
75	0.288	0	0.712	4134
50	0.234	0	0.766	6560
25	0.179	0	0.821	12313

Table 3. Results for general senders with uniform distribution

U	*Sure*	*Maybe*	*Failed*	Number of Messages
100	0.141	0.156	0.703	2735
75	0.115	0.141	0.744	3622
50	0.096	0.103	0.801	5625
25	0.089	0.073	0.838	9000

It can clearly be seen that with triangle distribution the observer can always choose exactly one sender (*Maybe* is always 0). With uniform distribution the certainty of the observer is lower (overall *Failed* increases) and in several occasions he cannot choose between different senders (*Maybe* is not 0).

A.2 Values for MIN/MAX Message Sending

In this section numerical values of the simulation of MIN/MAX senders (see section 9.2) will be given. MIN/MAX senders enforce the MIN/MAX property for the distribution of sent messages, thus they greatly improve anonymity with the help of dummy messages. Table 4 shows the results with triangle distribution, whereas values for the uniform distribution can be found in Table 5.

Table 4. Results for MIN/MAX senders with triangle distribution

$\tau_{\max}$	*Sure*	*Maybe*	*Failed*	Number of Messages
1.0	0.055	0	0.945	42094
1.5	0.062	0	0.938	33320
2.0	0.069	0	0.931	27569
2.5	0.078	0	0.922	23525
2.95	0.085	0	0.915	20767

Table 5. Results for MIN/MAX senders with uniform distribution

$\tau_{\max}$	*Sure*	*Maybe*	*Failed*	Number of Messages
1.0	<0.001	0.052	0.948	42107
1.5	0.002	0.054	0.944	33317
2.0	0.002	0.055	0.943	27617
2.5	0.002	0.058	0.940	23473
2.95	0.002	0.060	0.938	20818

Comparing the results of MIN/MAX senders with general senders (see previous section), one can observe that the number of sent messages increased (dummy messages were introduced) and this resulted in a greater *Failed* ratio.

A.3 Upper Limit for the Confidence of the Observer

For MIN/MAX senders equation (15) gives a limit ($\hat{P}_{\Psi}$) for the certainty of the observer. In the simulation actual values were the following (Table 6):

Table 6. $\hat{P}_{\Psi}$ limits for MIN/MAX senders

$\tau_{\max}$	$\hat{P}_{\Psi} = \Theta$ for triangle distribution	$\hat{P}_{\Psi} = \Theta$ for uniform distribution	
	according to (15)	according to (15)	according to (17)
1.0	0.27	0.067	0.056
1.5	∞	0.1	0.083
2.0	∞	0.2	0.111
2.5	∞	0.2	0.139
2.95	∞	0.2	0.164