# Anonymous Communication in the Digital World*

Andriy Panchenko

Interdisciplinary Centre for Security, Reliability and Trust
University of Luxembourg
http://www.securityandtrust.lu
andriy.panchenko@uni.lu http://lorre.uni.lu/~andriy/

## Abstract

Privacy on the Internet is becoming a concern as an already significant and ever growing part of our daily activities is carried out online. While cryptography can be used to protect the integrity and confidentiality of contents of communication, everyone along the route on which a packet is traveling can still observe the addresses of the respective communication parties. This often is enough to uniquely identify persons participating in a communication. Anonymous communication is used to hide relationships between the communicating parties. These relationships as well as patterns of communication can often be as revealing as their content. Hence, anonymity is a key technology needed to retain privacy in communications.

This paper provides a very brief overview of my doctoral dissertation *"Anonymous Communication in the Age of the Internet"* [2] and then concisely focuses on one randomly selected aspect, namely, the attack on the anonymization concept called *Crowds*.

**Keywords and phrases** Security, privacy, anonymity, anonymous communication, confidentiality

**Digital Object Identifier** 10.4230/OASIcs.KiVS.2011.221

## 1 Introduction: Brief Thesis Overview

The goal of the thesis *"Anonymous Communication in the Age of the Internet"* [2] is to enhance the state-of-the-art in the field of anonymous communication and to contribute to a broader understanding of the topic and its primitives within the community of researchers as well as to create solid fundamentals for future designs of the systems empowering users with tools for strengthening their privacy protection on the Internet.

We first propose a practical attacker model for risk analysis in anonymous communication. We justify the applicability of the model by an analysis of the strength of anonymizing techniques compared to each other as well as some widely known attacks on them. We then present the design and evaluation of a novel lightweight anonymization protocol based purely on open standards. It significantly outperforms other existing approaches for anonymization while only slightly sacrificing the level of provided protection. We next propose and analyze two innovative approaches for scalable distribution of information about anonymization networks. They have security properties similar to a centralized directory, but scale gracefully and do not require trust in any third party. We use analytical models and simulations to validate our approaches. We also consider performance issues of anonymization networks. To this end we develop and evaluate path selection metrics for performance-improved onion routing. The results show that applying our methods, users can obtain a significant increase in performance without harming their anonymity. Alternatively, users can get a dramatic performance boost with little sacrifice in anonymity. We provide a practical approach to empirically analyze the strength of anonymity different methods of path selection provide

* This work was conducted at the RWTH Aachen University, Germany.

17th GI/ITG Conference on Communication in Distributed Systems (KiVS'11).
Editors: Norbert Luttenberger, Hagen Peters; pp. 221–226
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

in comparison to each other. Finally, we investigate several attacks against anonymous communication systems. Most notably, we present a traffic analysis attack against encrypted HTTP streams sent through different anonymizers with surprising results showing the effectiveness and ease of website identification in encrypted channels transferred through the commonly used anonymizers. Moreover, we show under which conditions and how innocent-looking application layer data can be used to speed up traditional attacks that are targeted at the network layer identification of a user's communication partners. We also propose and analyze different countermeasures hampering these attacks.

## 2 Crowds

Crowds [5] is a peer-to-peer system for anonymous web browsing. It is based on a simple randomized routing protocol, in which all participants forward messages on behalf of other users as well as their own. Crowds provides security by means of increased path length and was meant to be a trade-off between performance and security. The main idea of Crowds is to hide each user's communications by routing them randomly within a group of similar users (*"blending into a crowd"*). When a user requests a web page, he sends the request to another (randomly chosen) crowd member (called a *jondo*). Upon receiving such a request, this jondo (and, if any, all consecutive) decides whether to forward the message to the final destination or to another randomly chosen jondo by making a *biased coin toss*. More formally, the message is forwarded to its final destination with probability $1 - p_f$, or to some other random participant with probability $p_f$. Communication between jondos is encrypted, however each of them sees the content of passing messages, including the address of the final destination. The final request to the server is usually sent in plain.

## 3 Predecessor Attack on Crowds

Due to the fact that the message initiator always forwards messages to a randomly chosen node, but all consecutive nodes only with a certain probability, there exist an information leakage in the system: the one who forwards a message to a colluding jondo is more suspicious to be the message initiator than any other honest jondo. In the following we will show how to make use of this information leakage in order to deanonymize the users.

Let $n + c$ be the size of the crowd, $c$ is the number of colluding jondos and $p_f$ is the probability of forwarding in the system as defined in [5]. $n + c$ and $p_f$ are system parameters known to everyone, while $c$ is known only to the adversary.

Let $p_h$ denote the probability that a colluding jondo receives a message directly from the path initiator[1]. Let $p_l$ denote the probability for one honest jondo, which is not the initiator, to forward the message to a colluding jondo (note that $p_h + (n - 1) \cdot p_l = 1$). The interested reader is referred to [2, 3] for the details how to calculate the probabilities $p_h$ and $p_l$.

Because of the information leakage mentioned above, $p_h > p_l$ for all admissible parameter values [2]. This fact can be used by an attacker to make precise statements about the users' peers. We assume that the attacker participates in the crowd with $c$ jondos ($c \geq 1$) and, without loss of generality, only stores the passing communication to a single external entity, namely Bob. In this section we will show how to estimate the amount of communication that each honest user initiated with Bob. The estimation can be performed with an arbitrary precision [2].

[1] $h$ stands for "high", $l$ for "low"

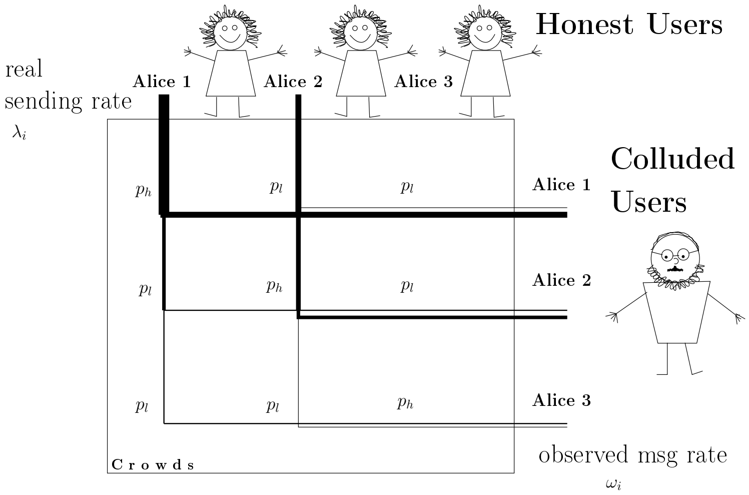


**Figure 1** Information flow for the attacker: matrix M

From the adversary's point of view, there are $n$ jondos that are sending messages to Bob, each with its own average rate $\lambda_i \geq 0$ per time interval. We model these as Poisson processes $A_i = \mathcal{P}_{\lambda_i}$ for $i \in 1 \dots n$. This kind of arrival distribution is in our opinion a fair trade-off between the analytical complexity and realism. Herewith we want to provide a first approximation which can be further refined by modeling arrivals in a more sophisticated manner. The colluding entities observe on average the following number of messages to Bob per time interval from the $i$-th system member:

$$E[\text{msg to Bob from } i] = p_h \cdot \lambda_i + \frac{1-p_h}{n-1} \cdot \sum_{j=1, j\neq i}^{n} \lambda_j \tag{1}$$

With the help of the following matrix $M$, we can model the number of messages $O_i$ arriving at the attacker from jondo $i$ to Bob:

$$M = \begin{pmatrix} p_h & p_l & \cdots & p_l \\ p_l & p_h & \cdots & p_l \\ \vdots & \vdots & & \vdots \\ p_l & p_l & \cdots & p_h \end{pmatrix} \tag{2}$$

$$O_i = \textstyle\sum_{j=1}^{n} \mathcal{P}_{m_{i,j}\lambda_j} \tag{3}$$

The observations can be seen as a vector of messages (each element is the number of messages received from the corresponding jondo by colluding members and addressed to Bob), which is a product of the vector of actually sent messages with the matrix $M$ (see Figure 1). The thickness of the line corresponds to the sending rate of the corresponding user (the thicker it is, the higher is the sending rate). Note that the missing line from one of the users on the right-hand side means that he does not send own messages, but rather only forwards some on behalf of the others (which corresponds to the thin line pointing to colluding users). Since $O_i$ is a sum of the Poisson processes, it is also a Poisson process with the following properties:

$$O_i = \mathcal{P}_{\omega_i} \quad \text{for} \quad \omega_i = \sum_{j=1}^{n} m_{i,j}\lambda_j \tag{4}$$

$$E[O_i] = V[O_i] = \omega_i \tag{5}$$

Because the cardinality of $\omega_i$ depends on the value of the $\lambda_i$, from the observations of $O_i$ it is possible to draw conclusions about the $\lambda_i$ (for $t \to \infty$):

$$(\omega_1, \ldots, \omega_n) = M(\lambda_1, \ldots, \lambda_n) \tag{6}$$

$$\iff \quad (\lambda_1, \ldots, \lambda_n) = M^{-1}(\omega_1, \ldots, \omega_n) \tag{7}$$

Indeed, the matrix $M$ is invertible to $M^{-1}$ because Crowds is unable to provide perfect security ($p_h \neq p_l$), thus:

$$M^{-1} = \begin{pmatrix} \tilde{p_h} & \tilde{p_l} & \cdots & \tilde{p_l} \\ \tilde{p_l} & \tilde{p_h} & \cdots & \tilde{p_l} \\ \vdots & \vdots & & \vdots \\ \tilde{p_l} & \tilde{p_l} & \cdots & \tilde{p_h} \end{pmatrix} \tag{8}$$

$$\tilde{p_h} = \frac{n+p_h-2}{np_h-1} \tag{9}$$

$$\tilde{p_l} = \frac{p_h-1}{np_h-1} \tag{10}$$

Under the common values for $p_f$, $n$, and $c$ ($0 < p_f < 1,\ n \gg c$) the following inequalities hold: $\tilde{p_h} > 0$ and $\tilde{p_l} < 0$.

Having observed $\omega_i$, it is thus possible to calculate an estimation for the $\lambda_i$, namely $\tilde{\lambda_i}$, as follows:

$$\tilde{\lambda_i} = \sum_{j=1}^{n} {m^{-1}}_{i,j}\omega_j \tag{11}$$

Using Chernoff style bound for the probability that a Poisson process deviates from its mean, it is even possible to calculate the number of observations needed in order to estimate sending rates $\lambda_i$ with any desired precision. The interested reader is referred to [2] for further details.

## 4 Predecessor Attack Speed-Up: Cross-Layer Attack

The classical predecessor attack presented above aims to identify a user's peer partners at the network layer only. We will show in this section that finding a user's peer partners by a predecessor attack can be sped up by building an extensive user profile on the application layer in parallel, i.e., identifying values of different communication attributes.

While a user typically communicates with many arbitrary peers, his application layer profile (set of accepted languages, browser version, etc.) remains usually the same. The required number of observations to confirm (identify) with arbitrary precision user $A_i$'s peer partner $B$ proportionally depends on the message rate from $A_i$ to $B$ observed by colluding users [2]. The same is valid for the application layer profile. Therefore, an attacker will discover the application layer profile of his victim much faster than the communication profile. If the attacker is using a statistical attack on the network layer, he can then use information from the application layer to bias the input for the network layer attack, e.g., by filtering out improbable combinations or in general, messages that do not fit to the victim's profile.

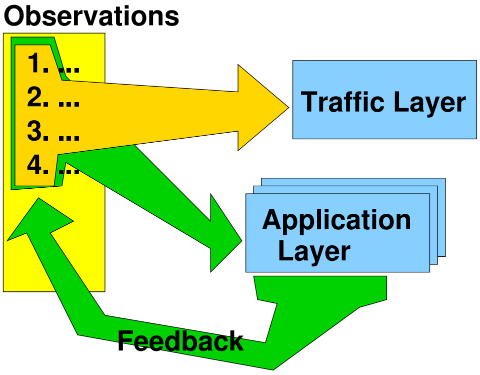


**Figure 2** Cross-layer information flow

Figure 2 illustrates how the attack works: at first a user's application layer profile is built applying the classical predecessor attack – but rather on application layer data. This information is further used in order to refine the classical attack on the network layer: the feedback is given to improve the observations which are used as an input to the attack on the traffic layer [2, 4].

After the profiles of users are built, it is possible to use the same data in order to identify the user's peer partners. Actually, even during the process of building the application layer profile, the calculations for the network layer can already be adapted on the fly. We propose two attacks using application layer data that work as follows:

**combined attack:** the classical predecessor attack (as described in the previous section) is applied only for messages matching the profile;

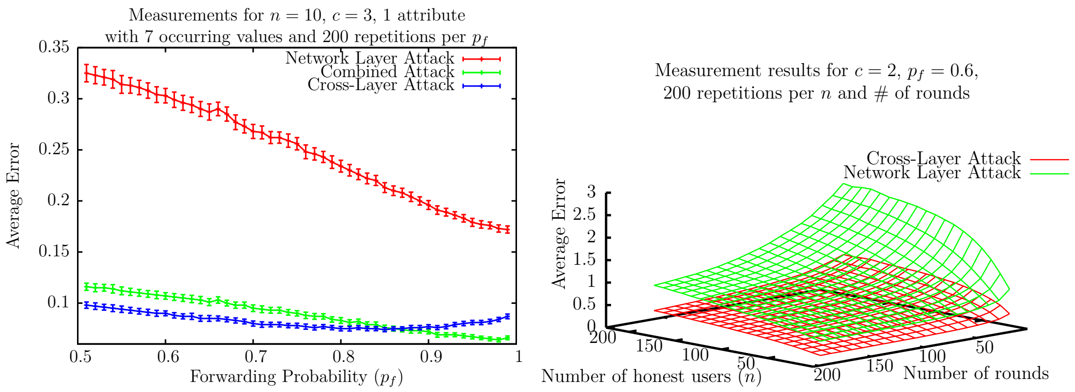
**cross-layer attack:** same as above, but additionally $p_h^{new}$ and $p_l^{new}$ are used instead of $p_h$ and $p_l$. $p_h^{new}$ is the probability that $A_i$ is the originator of the message, given that the message is received from $A_i$ by an attacker and that it has a corresponding application layer attribute. Similarly, $p_l^{new} = \frac{1-p_h^{new}}{n-1}$.

In order to evaluate the effectiveness of the new attacks we have performed simulations. The simulations are done as follows: every honest user has exactly two random communication peers. These two peers do not change during the whole simulation run. In each round every honest user sends between 0 and 3 messages to each of his communication peers. On the application layer, we used a single attribute with 7 possible values of the attribute in the communication profile (e.g., accepted languages in an HTTP request).

Figure 3 shows the simulation results including 95% confidence intervals in the 2D plot. Both of the introduced attacks are significantly more precise than the original network layer attack. We found out that the advantage of our attacks is higher for networks with a lower fraction of colluding members. For the considered scenario the accuracy is improved by a factor of 3. An interesting finding is that up to the forwarding probability of 0.86 the cross-layer attack is more precise than the combined one. However, for $p_f \geq 0.86$ the combined attack is more accurate. These results can be observed in Figure 3a.

## 5 Discussion and Conclusion

We showed how the information leakage in Crowds can be used in order to deanonymize its users. We also showed that enriching network layer information with innocent-looking



**(a)** Different values of forwarding probability

**(b)** Growing no. of observation rounds

**Figure 3** Cross-layer attack: simulation results

application layer data (e.g., browser strings in HTTP requests) can be used to significantly increase accuracy and speed up traditional attacks targeted at the network layer only. It is naïve to think that an attacker would not make use of all the information which is available. Hence, information from both network and application layer has to be considered in future research in order to provide usable and realistic metrics for anonymity.

The attack speed-up can be circumvented if the filtering system on the application layer would substitute the identifying information from the browser with the known statistical distribution of the browser data like, e.g., [1]. To be even more unidentifiable, one could observe the communication of the others as an attacker does, and calculate this statistic on his own. This is due to the reason that distribution of identifying information among the users in anonymizing networks may be different from those on the Internet in general.

The network layer attack itself, however, cannot be circumvented completely. Crowds leaks routing information under any admissible parameter values [2]. The number of observations needed to determine the sending rate of the users in the crowd precisely enough can be relatively small [2]. This rises a reasonable doubt on the possibility of using the system for strong anonymity in an open environment.

For more information about this and other topics around anonymous communication, the interested reader is referred to [2].

## References

1 Browser statistics. http://www.w3schools.com/browsers/browsers_stats.asp, January 2011.

2 Andriy Panchenko. *Anonymous Communication in the Age of the Internet.* PhD thesis, Department of Computer Science, RWTH Aachen University, 2010. Wissenschaftsverlag Mainz, Aachen.

3 Andriy Panchenko and Lexi Pimenidis. Crowds revisited: Practically effective predecessor attack. In *Proceedings of the 12th Nordic Workshop on Secure IT-Systems (NordSec 2007)*, Reykjavik, Iceland, October 2007.

4 Andriy Panchenko and Lexi Pimenidis. Cross-Layer Attack on Anonymizing Networks. In *Proceedings of the 15th International Conference on Telecommunications (ICT 2008)*, St. Petersburg, Russia, June 2008. IEEE Xplore.

5 Michael K. Reiter and Aviel D. Rubin. Crowds: Anonymity for Web Transactions. *ACM Transactions on Information and System Security*, pages 66 – 92, April 1998.