As appeared in: Proceedings of IEEE Global Communications Conference (GLOBECOM), St. Louis, MO, November 2005.

Anonymity Analysis of Mix Networks Against Flow-Correlation Attacks

Ye Zhu, Xinwen Fu, Riccardo Bettati, and Wei Zhao
Department of Computer Science, Texas A&M University
College Station, Texas, U.S.A.
Email: zhuye@tamu.edu, {xinwenfu, bettati, zhao}@cs.tamu.edu

***Abstract*— Mix networks are designed to provide anonymity for users in a variety of applications, including anonymous web browsing and numerous E-commerce systems. Such networks have been shown to be susceptible to flow correlation attacks empirically. In this paper, we model the effectiveness of flow correlation attacks. Our results illustrate the quantitative relationship among system parameters such as sample size, noise level, payload flow rate, and detection rate. Our analysis quantitatively predicts how existing flow-based anonymous systems would fail under flow-correlation attacks, thus providing useful guidelines for the design of future anonymous systems.**

I. Introduction

Anonymity has become necessary and legitimate in many scenarios, such as anonymous web browsing, E-Voting, E-Banking, E-Commerce, and E-Auctions. In each of these scenarios, encryption alone cannot achieve the anonymity required by participants [12].

So-called mix networks have been developed to support anonymity in distributed systems, where a third party can observe the traffic flowing between participants [4], [8]. We focus our study on a particular type of statistical flow-based timing attack against mix networks, the so-called *flow-correlation attack* [13]. In this attack, an adversary attempts to *reconstruct* the path of the communication flow through the mix network from the sender to the receiver. The basic building block of the flow correlation attack can be described as follows: given timing data of a flow at the input of a mix and timing data of aggregated traffic leaving the mix at each outgoing link, what is the link taken by the flow? The effectiveness of flow correlation attacks has been illustrated before for the case of single mixes [3], [7], [13]. We will show in Section IV-A that it is very effective for mix networks as well.

If sufficient data is available from all the links in the mix network, this path reconstruction can be done on a mix-by-mix basis all the way to the suspected receiver(s). If data is available from a subset of links only, the corresponding mixes can be clustered into *supermixes*, and the path construction can be done at the supermix level.

In this paper, we propose a general modeling framework for mix networks, based on the detection rate, i.e., the probability that an adversary correctly identifies the path taken by a flow at the output of a mix (or mix network). Under this framework, we can accurately predict detection rates given the configuration of the mix network and the amount of available data to an adversary. Our theories show that given enough data, the adversary can achieve arbitrarily high detection rates, which follow from the observations made in [13].

This paper gives an overview of the analysis of mix networks. The details are described in the companion technical report [14]. The remainder of this paper is organized as follows: Section II outlines our mix network model and threat model. In Section III, after formally defining the flow-correlation attack problem, we describe how to estimate the effectiveness of the attack. In Section IV, we use simulation experiments to validate the accuracy of our analytical results. We conclude this paper in Section V and discuss the future work.

II. Models

A mix [1] is a relay device for anonymous communication. Figure 1 shows hosts communicating with each other by way of a mix network. A single-mix network can achieve a certain level of communication anonymity: The sender of a message attaches the receiver address to a packet and encrypts it using the mix's public key. Upon receiving a packet, the mix decodes the packet using its private key. Different from an ordinary router, a mix usually will not relay the received packet immediately. Rather, it collects several packets and then sends them out in a *batch*. The order of packets may be altered as well. Both batching and reordering are needed in order to prevent timing-based attacks. Without, a simple timing correlation of packets collected at the input and output links may break the anonymity that the mix tries to maintain. As shown in Figure 1, mixes are often deployed as *mix networks*, which are connected through overlays. Mix networks have the benefit that they generally continue providing some level of anonymity even in the presence of compromised mixes.

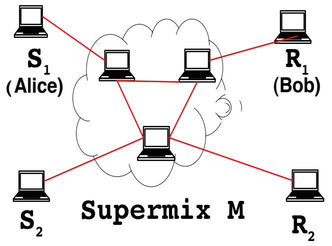


Fig. 1. Mix Network

Batching strategies are designed to prevent not only simple timing analysis attacks, but also powerful trickle attacks, flood attacks, and many other forms of attacks [2], [3], [7], [9]. In general, the transmission of a batch of packets can be triggered by different events, e.g., queue length reaching a pre-defined threshold, a timer having a time out, or some combination of these two. In this paper, we focus on two batching strategies: simple proxy and timed mix. They are denoted as S_I and S_{II} respectively in this paper. Simple proxies are used in Tor [4] and do not batch packets. Timed mixes fire cumulated packets every t seconds. The theorems deducted in this paper can also be applied to all the other batching strategies by just employing different queuing models.

We assume a global passive adversary who has knowledge of the mix's infrastructure. The adversary cannot correlate a packet on an input link to another packet on an output link based on content or on size. The former is prevented by encryption and the latter by packet padding, respectively. We evaluate how well the attacker can statistically correlate flows based on timing despite batching in the mix. In this paper we do not consider link padding with dummy packets, but rely on naturally occurring cross traffic instead. This follows the practice of existing mix networks such as Tor [4]. We assume that the specific objective of the adversary is to identify the output link of a traffic flow that appears on an input link.

III. Flow Correlation Attacks and Detection Rate Analysis

A. Flow Correlation Attacks

Define a traffic flow as a series of packets exchanged between a sender (Alice) and a receiver (Bob) in the network. For the attacker who reconstructs the path of a flow, a fundamental question must be answered: *given a flow, f, into a mix or mix network, which output link does the flow use?* For example, consider the network in Figure 2 where f', c_1' and c_2' are output flows of input flows f, c_1 and c_2, respectively. The goal of the adversary is to determine whether input flow f, after passing through the mix, goes through $link_{M\to R_1}$ (link from mix M to R_1) or $link_{M\to R_2}$.

Flow f is not alone in the mix network: a significant amount of cross traffic either naturally exists, or is generated by the mix network. We therefore assume that (i) there is noisy cross traffic (for example, c_1 and c_2 in Figure 2) interfering with the correlation analysis, and (ii) traffic average rates on all the output links (for example, $link_{M\to R_1}$ and $link_{M\to R_2}$ in Figure 2) are the same. The second assumption in particular renders simple statistical attacks, such as average traffic rate based attacks in [10], invalid. In this section, we will always use the setup of Figure 2 as an example to demonstrate our analysis technique.

1) Flow-Correlation Attack Algorithm: To determine which output link the input flow f uses, an adversary has to collect information and make a determination based on some statistical analysis. In this paper, we consider that the adversary adopts a method based on mutual information [13] of the input flow and output link aggregated flows, (i.e., the flow presumably embedded in the cross traffic) and chooses the output link whose aggregated flow has the biggest mutual information with the input flow. Using Figure 2 as the example, the adversary will collect a traffic sample from both input and output links. Then, she calculates mutual information $I(f, l_{M\to R_1})$ and $I(f, l_{M\to R_2})$, where $l_{M\to R_1} = f' + c_1'$ is the aggregated flow on $link_{M\to R_1}$ and $l_{M\to R_2} = c_2'$ is the aggregated flow on $link_{M\to R_2}$. Mutual information measures the dependence between flows. A decision will then be made in the following way: if $I(f, l_{M\to R_1}) > I(f, l_{M\to R_2})$, the adversary will declare $link_{M\to R_1}$ as f's output link. Otherwise, $link_{M\to R_2}$ will be chosen.

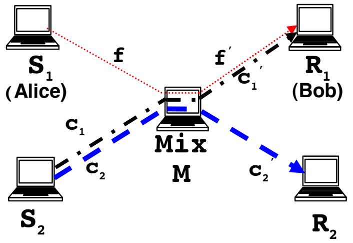


Fig. 2. Mix Setup and Flow Configuration

2) Mutual Information Estimation: From the discussion above, we can see that an accurate estimation of mutual information of input and output traffic is critical in flow-correlation attacks.

We assume that the adversary uses the following packet counting scheme to estimate the mutual information between the input flow f and any aggregated flow l on an output link.

First, the adversary collects (by, say, sniffing) a sample of traffic traces of the input flow f and the aggregated output flow l. Next, each traffic trace is divided into segments. The length of the segments is equal to T seconds, which is denoted as *sampling interval*. The number of segments in a trace is denoted N and is called *sample size*. Then, the number of packets in each segment of both traces is counted. Let a and b represent the random variables of the numbers of packets in a segment of traffic trace from an input flow and output link aggregated flow, respectively. The *Input flow packet rate time series* $f_T = \{a_1, \cdots, a_N\}$ is obtained, where a_i is the number of packets in the i^{th} segment of the input traffic flow trace. Note that $a_i \in \{0, \cdots, r\}$, where $r = \max(a)$. Similarly, the *Output link aggregated flow packet rate time series* $l_T = \{b_1, \cdots, b_N\}$ is obtained, where b_i is the number of packets in the i^{th} segment of the output link traffic flow trace. Note that $b_i \in \{0, \cdots, s\}$, where $s = \max(b)$. Then, the joint time series $J_T = \{(a_1, b_1), \cdots, (a_N, b_N)\}$ is derived, where a_i and b_i are elements in the time series f_T and l_T, respectively. Finally, the mutual information of the input flow and the output link flow can be *estimated* by the following formula:

$$\hat{I}(f, l) \approx \sum_{a=0}^{r} \sum_{b=0}^{s} \hat{p}(a, b) \log \frac{\hat{p}(a, b)}{\hat{p}(a)\hat{p}(b)} \qquad (1)$$

where $\hat{p}_a$, $\hat{p}_b$, and $\hat{p}_{a,b}$ are the *frequencies* of a, b, and (a, b) within f_T, l_T, and J_T, respectively.

B. Derivation of the Detection Rate

The *detection rate* v is defined as the probability that the adversary correctly recognizes the output link of the input flow

f. Without loss of generality, we assume that the input flow f goes through a mix's output link $link_{M\to R_1}$. Based on the algorithm described in Section III-A.1, the general formula of detection rate is as follows:

$$\begin{aligned} v &= Pr\Big(\hat{I}(f, l_{M\to R_1}) > \hat{I}(f, l_{M\to R_2}), \\ &\qquad \cdots, \hat{I}(f, l_{M\to R_1}) > \hat{I}(f, l_{M\to R_n})\Big) \end{aligned} \qquad (2)$$

1) Distribution of the Mutual Information: To calculate the detection rate by using (2), we need to obtain the probability distribution function of the *mutual information estimation* $\hat{I}(f,l)$ in (1). According to [5], for a sufficiently large sample size N, $\hat{I}(f,l)$ should satisfy a normal distribution. To obtain the distribution function, we therefore only need to estimate $\hat{I}(f,l)$'s mean and variance, which are given in Lemma 1 and 2, respectively. Their proofs can be found in [14].

Lemma 1: The mean of the mutual information estimation $\hat{I}(f,l)$ is given by

$$E(\hat{I}(f,l)) \approx I(f,l) + (r-1)(s-1)/N \qquad (3)$$

where $I(f,l)$ is the *original mutual information,* and r and s are as defined in Section III-A.2

Lemma 2: The variance of the mutual information estimation $\hat{I}(f,l)$ is given by $var(\hat{I}(f,l)) \approx \frac{C_{f,l}}{N}$, where $C_{f,l}$ is a constant and is defined as follows

$$\begin{aligned} C_{f,l} &= \sum_{a,b} p(a,b)\left(\log\frac{p(a,b)}{p(a)p(b)}\right)^2 \\ &\quad - \left(\sum_{a,b} p(a,b)\log\frac{p(a,b)}{p(a)p(b)}\right)^2 \end{aligned} \qquad (4)$$

and $p(a,b)$ is the *original probability distribution* of (a,b).

2) Detection Rate Theorem: Based on the distribution function of estimated mutual information, we can calculate the detection rate by the following theorem. Its proof can be found in [14].

Theorem 1: For a mix with any number of output links, the detection rate, v, is given by

$$\begin{aligned} v &\approx 1 - \sqrt{\frac{C_{f,l_{M\to R_1}}}{N}} \\ &\quad \times \int_{-\infty}^{-I(f,l_{M\to R_1})\sqrt{\frac{N}{C_{f,l_{M\to R_1}}}}} N(0,1)dx \end{aligned} \qquad (5)$$

where N is the sample size, $I(f, l_{M\to R_1})$ is the mutual information of the input flow f and its corresponding output link aggregated flow $l_{M\to R_1}$, $N(0,1)$ is the density function of the standard normal distribution, and $C_{f,l_{M\to R_1}}$ is a constant. We note that Theorem 1 is very general. In particular, no assumptions are made in Formula (5) about the batching strategy of the mix or about the network topology. Theorem 1 is therefore valid for mix networks with arbitrary topology. Similarly, no assumption is made about the type of traffic or about the amount of cross traffic. Clearly, the detection rate is an increasing function of sample size N. Thus, when sample size N increases, the detection rate approaches 100%. This formally proves the intuitive fact that any mix network will fail and cannot maintain anonymity if the adversary has access to a sufficiently large amount of traffic data.

3) Joint Distribution of (a,b): In Theorem 1, both constant C and the original mutual information $I(f,l)$ depend on the joint distribution function $p(a,b)$, which in turn is defined by the strategy of the mix network and the type and amount of traffic in the network. It can be estimated by two methods:

- *Direct Estimation.* That is, we can estimate $p(a,b)$ directly from the time series J_T defined in Section III-A.2. Specifically, from J_T, a frequency distribution of (a,b) can be established. Then, we can use standard statistical techniques to obtain an estimation of $p(a,b)$. See [11] for details.
- *Estimation based on Poisson Assumption.* The joint distribution $p(a,b)$ can be calculated as $p(a,b) = p(b|a)p(a)$. To calculate the conditional probability $p(b|a)$, we need to apply proper queuing models in accordance to mixing strategies. For example, if the input flow is assumed to be a Poisson process, for a simple proxy S_I, a M/D/1 queuing model should be used. For timed mix S_{II}, we should use an embedded Markov chain. Please refer [14] for a detailed derivation of the probability from the models.

IV. Evaluation

In this section, we assess the accuracy of methods we developed to estimate detection rate and to evaluate the performance of mix networks that are under flow-correlation attacks. We use the popular ns-2 network simulator for all the experimental evaluations.

A. Failure of Mix Network

Before we proceed to evaluate the accuracy of our predictive models for single mixes, we provide data to validate the claim made in Theorem 1: for any size of mix network (in fact any network), given sufficiently long data, flow correlation attack will ultimately achieve a detection rate arbitrary close to 100%.

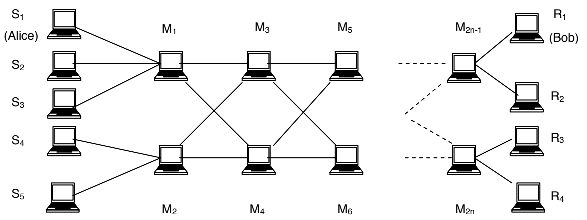


Fig. 3. Topology of Mix Network

The network topology for this experiment is shown in Figure 3: The senders and receivers are connected by a stratified cascade of 2n mixes. Each flow traverses n mixes to reach its receivers. Each link between mixes has a bandwidth of 10Mbit/s and propagation delay of 10ms. The senders and receivers are connected to the mix network via links with bandwidth of 100Mbit/s and propagation delay of 1ms. There are five flows in the network: flow $S_1 \rightarrow R_1$, flow $S_2 \rightarrow R_1$, flow $S_3 \rightarrow R_3$, flow $S_4 \rightarrow R_2$, and flow $S_5 \rightarrow R_4$ respectively. Flow $S_1 \rightarrow R_1$ and $S_2 \rightarrow R_1$ traverse odd-numbered mixes only, flow $S_5 \rightarrow R_4$ traverses even-numbered mixes only, flow $S_3 \rightarrow R_3$ and $S_4 \rightarrow R_2$ take the zigzag path between the two horizontal lines of the mixes, and flow $S_1 \rightarrow R_1$ is the flow of interest to us. To ease the control of noise traffic rate, only flow $S_1 \rightarrow R_1$ is TCP and other flows are UDP with Poisson arrival. The average traffic rate to all the receivers are adjusted to roughly five times the average rate of flow $S_1 \rightarrow R_1$. The mixes in network are all timed mixes with a batch interval of 10ms.

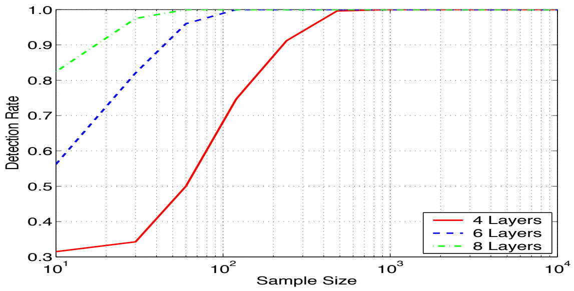


Fig. 4. Effectiveness of Flow Correlation Attack vs Size of the Mix Network

Figure 4 shows the detection rates of a flow correlation attack for different numbers of mixes in the network. The length of sampling segments is set to be 10ms. We note that, as stated in Theorem 1, the flow correlation attack remains effective even for large mix networks. In fact, the flow correlation attack achieves higher detection rates for larger mix networks! The reason is the loop-control mechanism of TCP: the more mixes on the path, the larger burstiness of the TCP flow from Alice to Bob. In turn, this makes Alice's flow more recognizable compared with the background noise traffic.

B. Estimation Error of Detection Rate

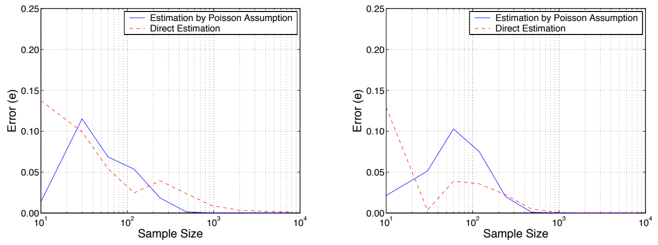
Formula (5) for the detection rate is an estimated one due to, at least, two reasons: First, the computation of mutual information is estimated by a truncated Taylor expansion, which introduces a certain error due to the limited number of terms. Second, the methods to estimate $p(a, b)$ will contribute some error to the estimation of detection rate.

In this subsection, we would like to examine the accuracy of our estimation in order to ensure the performance data we derive in this paper are practicably meaningful. We use the one-mix network setup in Figure 2.

We define e, the estimation error of the detection rate, as follows:

$$e = \frac{|\text{approximated detection rate} - \text{exact detection rate}|}{\text{exact detection rate}} \quad (6)$$

We obtain the exact detection rate in (6) by simulation. In all our experiments mentioned earlier, to prevent attacks based on analyzing average traffic rates, traffic average rates on all output links are assumed to be the same. The traffic type of payload flow can be either UDP or TCP, with traffic rates of 100 KBps and 80KBps bps, respectively. Compounded with noise traffic, each output link has an aggregated traffic rate 500 KBps. T, the length of sampling segments is set to be 10ms. Due to the space limitation, in this paper, we will only report our findings for mix networks using batching strategy S_I or S_{II}. Evaluation of other batching strategies with a variety of traffic conditions result in similar observations, which are reported in [14].



(a) No Batching, TCP Traffic

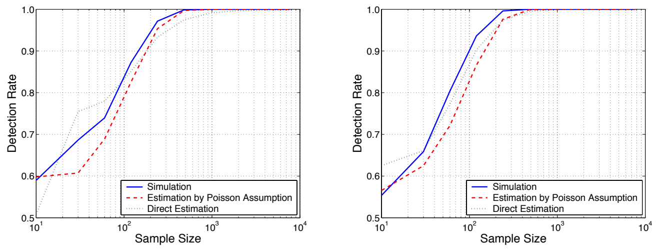
(b) Batching, TCP Traffic

Fig. 5. Estimation Error of Detection Rate

Figure 5 depicts the estimation error in terms of sample size. It shows that for all the traffic types and batching strategies, if the sample size is small (say, less than 100), the estimation error may be more than 5%. The estimation error diminishes and eventually approaches to zero for increasing sample sizes. This observation suggests our estimation methods will be quite useful in practical situations. The direct estimation method results in smaller error than the estimation by Poisson assumption. This is to be expected, as the traffic on the Internet is not inherently Poisson.

In comparison with the networks using different batching strategies, the estimation errors appear to be similar. However, when we compare networks with different traffic types, UDP traffic seems to result in less error. This is, perhaps, due to the difficulty in statistical modeling of TCP traffic.

C. Detection Rate



(a) No Batching, TCP Traffic

(b) Batching, TCP Traffic

Fig. 6. Detection Rate

Figure 6 shows the detection rate in terms of sample size. In all cases, the detection rate approaches 100% when the sample size is sufficiently large. This demonstrates the challenges posed by flow-correlation attacks and validates the claim made in Section III-B.2.

The implication of the above two observations is serious: a mix network would fail to provide anonymity under the flow-correlation attacks if the adversary is allowed to collect its sample for a time period of sufficient length. Note that, by using our formulae, a system designer can relatively precisely predict the situations where the failure may occur and invoke other countermeasures (such as shortening the flow life time, utilizing channel hoping in wireless networks, etc).

D. Minimum Sample Size

As mentioned earlier, one way to provide a countermeasure against flow-correlation attacks is to reduce the flow life time to prevent the adversary from obtaining a sample that is sufficiently large. To provide some guidelines in this matter, we performed some measurements to establish the minimum sample size needed in order for the adversary to achieve a given detection rate.

Table I compares minimum sample sizes for different traffic types and batching strategies. As expected, in all cases the minimum sample size increases with increasing detection rate. For example, for the case of UDP traffic, the minimum sample size increases from about 130 to almost 200 when the detection rate requirement increases from 95% to 99%. While this observation is expected, our formulae can provide useful guidelines for system parameter selection here.

detection rate	Poisson Traffic		TCP Traffic	
	Batching	No Batching	Batching	No Batching
95%	130	120	195	135
99%	195	175	290	215

TABLE I
MINIMUM SAMPLE SIZE

For UDP traffic, it appears that batching is not particularly effective in terms of the minimum sample size. However, effectiveness of batching to be much more interesting for TCP traffic: We observe that the minimum sample size actually *decreases* when we add batching.

This is somehow against intuition: if sample size is a measure of the level of difficulty for an adversary, our data show that the adversary has more difficulty to achieve the required detection rate in a network without batching than that with batching. This phenomenon actually can be explained. When batching is performed, the TCP traffic may start oscillating, due to the feedback mechanisms inherent to TCP. Consequently, this oscillation seems to provide a much better signature for the adversary to make a recognition by correlating the traffic on input and output links. (The data indicates that this happens to a lesser extent for UDB as well. Again this can be explained by the additional patterns added by batching – this time without amplification through feedback.) We believe this is an important discovery which justifies the necessity of our modeling and evaluation in this paper.

V. CONCLUSION

We have analyzed the anonymity of mix networks under flow correlation attacks. We present a formal model of the adversary and derived the detection rate as a performance measure of the system. Our theory discloses the underlying principle of flow-correlation attacks. As such, our results are the first to illustrate the quantitative relationship among system parameters, such as sample size, noise level, payload flow rate, and detection rate. Our analysis quantitatively reveals that flow-correlation attacks (by performing correlation of flows into and out of a mix) can seriously degrade anonymity in mix networks. Consequently, our results also provide useful guidelines for the design of future anonymous systems where additional countermeasures must be taken.

Future studies are needed on more effective countermeasures against flow-correlation attacks. Possible candidates are control of flow life time, multi-path routing, and camouflaging. Results in this paper and others [6] have repeatedly demonstrated that in many cases, simple and intuitive countermeasures in cyber security may not work as expected. A general theory should be developed to help system designers to quantify the security performance of the system and make proper design choices.

REFERENCES

[1] D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 4(2), February 1981.

[2] G. Danezis, R. Dingledine, and N. Mathewson. Mixminion: Design of a Type III Anonymous Remailer Protocol. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, May 2003.

[3] G. Danezis. The Traffic Analysis of Continuous-Time Mixes. In Proceedings of the 2004 Workshop on Privacy Enhancing Technologies, 2004.

[4] R. Dingledine, N. Mathewson, and P. Syverson. Tor: The second-generation onion router. In *Proceedings of the 13th USENIX Security Symposium*, 2004.

[5] M. Hutter. Distribution of mutual information. In *Advances in Neural Information Processing Systems 14*, pages 399–406, Cambridge, MA, 2002. MIT Press.

[6] H. Kargupta, S. Datta, Q. Wang, , and K. Sivakumar. Random data perturbation techniques and privacy preserving data mining. In *Proceedings of International Conference on Data Mining*, 2003.

[7] B.N. Levine, M.K. Reiter, C. Wang, and M.K. Wright Timing Attacks in Low-Latency Mix-Based Systems. In *Proceedings of Financial Cryptography (FC '04)*, February 2004.

[8] M. Reiter and A. Rubin. Crowds: Anonymity for web transactions. *ACM Transactions on Information and System Security*, 1(1), 1998.

[9] A. Serjantov, R. Dingledine, and P. Syverson. From a trickle to a flood: active attacks on several mix types. In *Proceedings of Information Hiding Workshop*, 2002.

[10] A. Serjantov and P. Sewell. Passive attack analysis for connection-based anonymity systems. In *Proceedings of European Symposium on Research in Computer Security (ESORICS)*, 2003.

[11] B. W. Silverman. *Density estimation for statistics and data analysis*. Chapman and Hall, London, New York, 1986.

[12] Q. Sun, D. R. Simon, Y. Wang, W. Russell, V. N. Padmanabhan, and L. Qiu. Statistical identification of encrypted web browsing traffic. In *Proceedings of IEEE Symposium on Security and Privacy*, 2002.

[13] Y. Zhu, X. Fu, B. Graham, R. Bettati, and W. Zhao. On flow correlation attacks and countermeasures in mix networks. In *Proceedings of Privacy Enhancing Technologies workshop (PET 2004)*, May 2004.

[14] Y. Zhu, X. Fu, B. Graham, R. Bettati, and W. Zhao. Theoretical analysis of flow based traffic analysis attacks in anonymous communication systems. Technical Report 2005-2-1, Texas A&M University, Computer Science Department, February 2005.