# Information Leaks in Structured Peer-to-Peer Anonymous Communication Systems

Prateek Mittal Nikita Borisov

Department of Electrical and Computer Engineering
University of Illinois at Urbana–Champaign
{mittal2,nikita}@illinois.edu

## ABSTRACT

We analyze information leaks in the lookup mechanisms of structured peer-to-peer anonymous communication systems and how these leaks can be used to compromise anonymity. We show that the techniques that are used to combat active attacks on the lookup mechanism dramatically increase information leaks and increase the efficacy of passive attacks. Thus there is a trade-off between robustness to active and passive attacks.

We study this trade-off in two P2P anonymous systems, Salsa and AP3. In both cases, we find that, by combining both passive and active attacks, anonymity can be compromised much more effectively than previously thought, rendering these systems insecure for most proposed uses. Our results hold even if security parameters are changed or other improvements to the systems are considered. Our study therefore motivates the search for new approaches to P2P anonymous communication.

## Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: General—*Security and protection*; C.2.4 [**Computer-Communication Networks**]: Distributed Systems

## General Terms

Security

## Keywords

Anonymity, attacks, information-leaks, peer-to-peer

## 1. INTRODUCTION

Anonymous communication hides the identity of communication partners from third parties, or hides user identity from the remote party. The Tor network [16], deployed in 2003, now serves hundreds of thousands of users and carries terabytes of traffic a day [35]. Originally an experimental network used by privacy enthusiasts, it is now entering mainstream use; for example, several consulates were found to be using it to evade observation by their host country [22].

The capacity of Tor is already strained, and to support a growing population a peer-to-peer approach will likely be necessary, as P2P networks allow the network capacity to scale with the number of users. Indeed, several proposals for peer-to-peer anonymous communication have been put forward [28, 34, 21, 39]. However, P2P networks present new challenges to anonymity, one of which is the ability to locate relays for anonymous traffic.

In Tor, clients use a directory to retrieve a list of all the running routers. Such a directory will not scale as the number of routers grows, since the traffic to update the directory would become prohibitively expensive. Instead, a peer-to-peer lookup is needed to locate an appropriate relay. Such a lookup, however, can be subject to attack: malicious nodes can misdirect it to find relays that are colluding and violate the anonymity of the entire system. All of the P2P anonymous communication designs therefore incorporate some defense against such attacks; e.g. AP3 [28] uses secure routing techniques developed by Castro et al [7], and Salsa uses redundant routing with bounds checks [34].

These defenses, however, come at a cost. They operate by performing extra checks to detect incorrect results returned by malicious nodes. These checks cause many messages to be exchanged between nodes in the network, some of which might be observed by attackers. As a result, a relatively small fraction of attackers can make observations about a large fraction of lookups that occur in the P2P network, acting as a near-global passive adversary. As most modern anonymity systems assume that a global passive adversary is too costly, they are not designed to resist such attacks. Therefore, this small fraction of attackers can successfully attack anonymity of the system.

We examine this problem through a case study of two P2P anonymous communication systems: Salsa and AP3. In both systems, defenses against active attacks create new opportunities for passive attacks. Salsa makes heavy use of redundancy to address active attacks, rendering it vulnerable to passive information leak attacks. Further, increasing the levels of redundancy will improve passive attack performance, and often make the system weaker overall. We find that even in the best case, Salsa is much less secure than previously considered. Salsa was designed to tolerate up to 20% of compromised nodes; however, our analysis shows that in this case, over one quarter of all circuits will be compromised by using information leaks. Similarly, conventional analysis

*CCS'08,* October 27–31, 2008, Alexandria, Virginia, USA.
Copyright 2008 ACM 978-1-59593-810-7/08/10 ...$5.00.

of AP3 suggests that it provides probable innocence when up to 33% of nodes are compromised, and can tolerate up to 50% of compromised nodes by increasing the path length. However, our analysis puts these numbers at 5% and 10%, respectively.

We studied potential improvements to Salsa that can be achieved by increasing the path length or introducing a public key infrastructure (PKI). We found that these tools offer only a limited defense against our attacks, and the system is still not secure for practical purposes. Our results demonstrate that information leaks are an important part of anonymity analysis of a system and that new advances in the state of the art of P2P anonymous communication are needed.

The rest of the paper is organized as follows. In Section 2 we present the state of art in low-latency anonymous communication. We discuss information leaks from lookups in Section 3 and show the trade-off between security and anonymity. In Sections 4 and 5, we present attacks based on information leaks from lookups on AP3 and Salsa. Section 6 contains the related work and we conclude in Section 7.

# 2. BACKGROUND

In this section, we present a brief overview of anonymous communication. We motivate the need for decentralized and scalable solutions, and discuss why structured peer-to-peer systems have strong potential. We also describe our adversarial threat model.

## 2.1 Low-Latency Anonymous Communication Systems

Anonymous communication systems can be classified into low-latency and high-latency systems. High latency anonymous communication systems like Mixminion [12] and Mixmaster [29] are designed to be secure even against a powerful global passive adversary; however, the message transmission times for such systems are typically on the order of several hours. This makes them unsuitable for use in applications involving interactive traffic like web browsing and instant messaging. The focus of this paper is on low-latency anonymous communication systems.

Tor [16] is a popular low-latency anonymous communication system. Users (clients) download a list of servers from central directory authorities and build anonymous paths using onion routing [45]. There are several problems with Tor's architecture. First, the reliance on central directory authorities makes them an attractive target for the attackers. Second, Tor serves hundreds of thousands of users and the use of a relatively small number of servers to build anonymous paths becomes a performance bottleneck. Finally, Tor requires all users to maintain a global view of all the servers. As the number of servers increases, maintaining a global view of the system becomes costly, since churn will cause frequent updates and a large bandwidth overhead. In order to address these problems, a peer-to-peer architecture will likely be necessary. However, peer-to-peer networks present new challenges to anonymity, one of which is the ability to locate relays for anonymous traffic.

Several designs for peer-to-peer low-latency anonymous communication have been proposed. Tarzan [21] replaced the centralized directory authority with a gossip protocol that was used to distribute knowledge of all peers to all other peers. While decentralized, the requirement that each node maintain an up-to-date global view of the system means that the system could scale only to about 10,000 nodes. MorphMix [39] was designed to scale to much larger network sizes. It built an unstructured peer-to-peer overlay between all the relays and created paths along this overlay to forward anonymous communications. In MorphMix, a node along the path is queried for its neighbors in order to choose the next hop. To prevent the node from providing malicious results, a scheme using witness nodes and a collusion detection mechanism is used. However, the collusion detection mechanism can be circumvented by a set of colluding adversaries who model the internal state of each node, thus violating anonymity guarantees [46].

Several other designs have used so-called structured peer-to-peer topologies [34, 28], also known as distributed hash tables (DHTs), as a foundation for anonymous peer-to-peer communication. Structured topologies assign neighbor relationships using a pseudorandom but deterministic mathematical formula based on the IP addresses or public keys of nodes. This allows the relationships to be verified externally, presenting fewer opportunities for attacks. AP3 [28] used a secure lookup mechanism [7] in the Pastry DHT [40] to select random forwarders and used them to build an anonymous communication path. The secure lookup techniques are based on a PKI, and thus do not achieve a truly decentralized security model. The lookup was also not designed to be anonymous, a property that we will show to have important consequences for the security of AP3.

Salsa [34] aimed to offer secure P2P anonymous communication in a system without a PKI. It designed a custom DHT structure and a custom secure lookup mechanism specifically tailored for the purposes of anonymous communication. Its secure lookup and path construction mechanisms rely heavily on redundancy to detect potential attacks. As we will show, such redundancy creates information leaks, and presents a trade-off between resisting active attacks and presenting more opportunities for passive attacks.

## 2.2 Threat Model

Low-latency anonymous communication systems are not designed to to be secure against a global passive adversary. We consider a partial adversary who controls a fraction $f$ of all the nodes in the network. This set of malicious nodes colludes and can launch both passive and active attacks. We consider the set of colluding nodes is static and the adversary cannot compromise nodes at will. In terms of the standard terminology introduced by Raymond [37], our adversary is internal, active and static.

Even in networks with large numbers of nodes, $f$ can be a significant fraction of the network size. Both Salsa and AP3 use mechanisms to prevent Sybil attacks [18], which would allow an adversary to attain an $f$ arbitrarily close to 1. However, powerful adversaries, such as governments or large organizations, can potentially deploy enough nodes to gain a significant fraction of the network. Similarly, botnets, whose average size has grown in excess of 20,000 nodes [36], present a very real threat to anonymity.

# 3. INFORMATION LEAKS VIA SECURE LOOKUPS

It has been recognized that unprotected DHTs are extremely vulnerable to attacks on the lookup mechanism. First of all, malicious nodes can perform a Sybil attack [18] and join the network many times, increasing the fraction $f$. Second, they can intercept lookup requests and return incorrect results by listing a colluding malicious node as the closest node to a key, increasing the fraction of lookups that return malicious nodes. Finally, they can interfere with the routing table maintenance and cause the routing tables of honest nodes to contain a larger fraction of malicious nodes; this will increase the chance that a lookup can be intercepted and the result can be subverted.

## 3.1 Castro et al.'s secure lookup

Castro et al. [7] designed a suite of mechanisms to counter these attacks. We discuss their mechanisms in context of Pastry [40], a structured peer-to-peer overlay network, though they are applicable to other DHTs. They proposed:

- *Secure node identifier assignment:* Each node is issued a certificate by a trusted authority, which binds the node identifier with a public key. The authority limits the number of certificates and prevents Sybil attacks.

- *Secure routing table maintenance:* Even with secure node ID assignment, attackers can maliciously influence routing table construction. The Pastry routing algorithms allow flexibility in selecting a neighbor for each slot, which is used for optimizing latency or other metrics. Attackers can exploit this flexibility by suggesting malicious choices for these slots. Secure routing table maintenance eliminates this flexibility by creating a parallel, constrained routing table where each slot can have only a single possible node, as verified by secure lookup. This solution ensures that, on average, only a fraction $f$ of a node's neighbors will be malicious.

- *Secure lookups (secure message forwarding):* For secure lookups, a two-phase approach is employed. The message is routed via the normal routing table (optimized for latency) and a routing failure test is applied. If the test detects a failure, redundant routing is used and all messages are forwarded according to the constrained routing table. The failure test makes use of the observation that the density of honest nodes is greater than the density of malicious nodes. The idea behind redundant routing is to ensure that multiple copies of messages are sent to the key root via diverse routes. Note that Castro et al. consider the problem of securely routing to the entire replica set, for which a neighbor anycast mechanism is also used. We refer the reader to [7] for a detailed explanation of the techniques.

Used together, these techniques are quite effective at ensuring that a lookup returns the actual closest node to the randomly chosen identifier, which in turn suggests that it is malicious with probability $f$. However, the secure lookup mechanism generates many extra messages: the routing failure tests involves contacting the entire root set of a node ($L$ immediate neighbors in the node ID space), and redundant routing sends a request across several paths. These messages let attackers detect when a lookup has been performed between two honest nodes with high probability. The probability of detecting the lookup initiator can be approximated as $1-(1-f)^{L+\log_{2^b} N}$, which is quite high for the typical values of $L = 16$ and $b = 4$. In Figure 1(a), we plot the probability of detection of the lookup initiator as a function of the fraction of compromised nodes $f$. We can see that a small fraction of 5% compromised nodes can detect the lookup initiator more than 60% of the time. Moreover, when the fraction of compromised nodes is about 10%, the lookup initiator is revealed 90% of the time.

This shows the fundamental tension that is encountered by a DHT lookup. The default Pastry mechanisms provide little defense against active adversaries who try to disrupt the lookup process, dramatically increasing the probability that a lookup returns a compromised node. Castro et al.'s mechanisms solve this problem, but introduce another, as the lookup is no longer anonymous and can be observed by malicious nodes. A relatively small fraction of malicious nodes can, therefore, act as a near-global passive adversary and compromise the security of anonymous communication systems. The secure lookup exposes nodes to increased surveillance; we note that this may have consequences for protocols other than anonymous communication that are built on top of secure lookup.

## 3.2 Salsa secure lookup

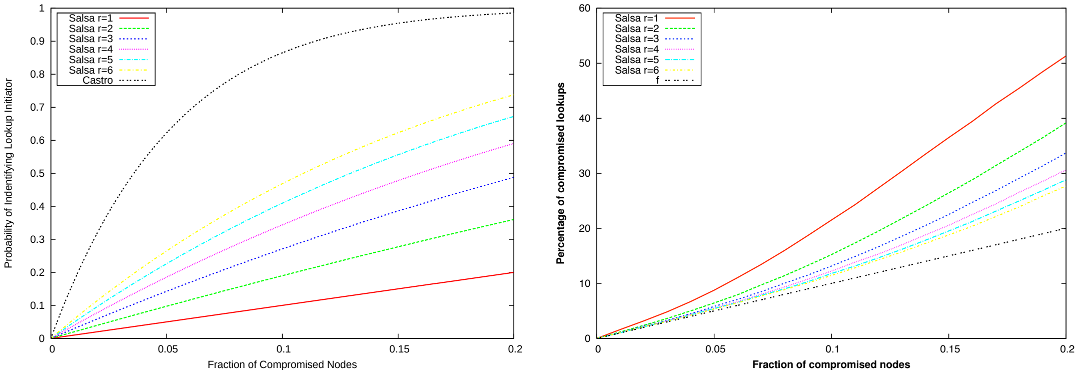
Salsa [34] is based on a custom-built DHT that maps nodes to a point in an ID space corresponding to the hash of their IP address. The ID space in Salsa is divided into groups, organized into a binary tree structure. Each node knows all the nodes in its group (local contacts), and a small number of nodes nodes in other groups (global contacts).

Similar to Pastry, nodes must rely on other nodes to perform a recursive lookup. A malicious node who intercepts the request could return the identity of a collaborating attacker node. Salsa makes use of redundant routing and bounds checks to reduce the lookup bias. The Salsa architecture is designed to ensure that redundant paths have very few common nodes between them (unlike Pastry or Chord [44]). This reduces the likelihood that a few nodes will be able to modify the results for all the redundant requests. A lookup initiator asks $r$ local contacts (chosen at random) to perform a lookup for a random key. The returned value that is closest to the key is selected and a bounds check is performed. If the distance between the prospective owner and the key is greater than a threshold distance $b$, it is rejected, reasoning once again that malicious nodes are less dense than honest ones and thus will fail the bounds check much more frequently. If the bounds check test fails, the result of the lookup is discarded and another lookup for a new random key is performed. Redundant routing and the bounds check work together: an attacker would need to both intercept all of the redundant lookups and have a malicious node that is close enough to avoid the bounds check.

Salsa is resistant to conventional attacks that target the lookup mechanism as long as the fraction of malicious nodes in the system is less that 20%. Since Salsa does not provide adequate security for higher values of $f$, we shall limit our analysis to low values.

In Figure 1(b), we study the effect of varying redundancy on the lookup bias. To compute our results, we used a simulator developed by the authors of Salsa [33].[1] The simula-

[1] Our results differ slightly from those shown in [34] because of a bug in the simulator. We have communicated the bug to the authors and it has been accepted.



(a) Information leak from secure lookups

(b) Percentage of compromised lookups

**Figure 1: Salsa lookup mechanism.**

tor was configured to simulate 1000 topologies, and in each topology, results were averaged over 1000 random lookups. The lookup bias is sensitive to the average lookup path length, which in turn is about $\log_2 |G|$, where $|G|$ is the number of groups. This is because longer path lengths give attackers more opportunities to intercept the lookup and subvert the result. We therefore used 128 groups, which would be a typical number in a large network, and 1000 nodes in our simulation. We can see that increasing $r$ clearly reduces the fraction of compromised lookups, thus increasing security. For $f = 0.2$, the fraction of compromised lookups drops from 39% to 27% when $r$ is increased from 2 to 6.

The initiator of a lookup can be identified by the attackers if any of the local contacts used for redundant lookups are compromised. The probability of detecting the lookup initiator is $1 - (1 - f)^r$, as depicted in Figure 1(a). Clearly, increasing $r$ increases the chance that a lookup initiator is detected. This illustrates the trade-off between security and anonymity of a lookup.

In this section, we observed that secure lookups leak information about the lookup initiator. Furthermore, we observed a trade-off between the security and anonymity of a lookup. A relatively small fraction of compromised nodes are able to observe a large fraction of lookups. Next, we shall use this to break the anonymity of AP3 and Salsa.

# 4. AP3

AP3 [28] is an anonymous communication system built on top of Pastry [40]. The essence of AP3 operation is similar to Crowds [38], where a random walk over all of the nodes in the system is used to forward requests while concealing the initiator's identity. In both AP3 and Crowds, a node $A$ who wants to send a message to a node $B$ first picks a random relay $F_1$ to forward the message. $F_1$ then flips a weighted coin, and with probability $p$ it chooses another relay, $F_2$, and forwards the request there. With probability $1 - p$, $F_1$ delivers the message directly to the recipient $B$.

Therefore, a message is forwarded through a path of nodes, all of which are selected randomly. The path length follows a geometric distribution, with the expected length being $\frac{1}{1-p}$. We can assume that some of the relays will be malicious and will try to guess the identity of the initiator. However, due to the stochastic nature of the forwarding, such relays will have a hard time telling whether they received a message from the initiator directly, or from another relay. Reiter and Rubin first analyzed the probability that the initiator is correctly identified [38]; we review the terminology used in their analysis here, as we will extend it in later sections.

Let $H_k$ denote the event that the first attacker in the forwarding path occupies the $k$th position, where the initiator is at the 0th. Let $H_{k+} = H_k \vee H_{k+1} \vee H_{k+2} \vee ...$ and let $I$ denote the event that attackers identified the initiator correctly. Then, given that an attacker intercepts a message, the chance that the initiator guessed correctly is $P(I|H_{1+})$. This can be further decomposed as:

$$\begin{aligned} P(I|H_{1+}) &= \frac{P(I \wedge H_{1+})}{P(H_{1+})} \\ &= \frac{P(H_1)P(I|H_1) + P(H_{2+})P(I|H_{2+})}{P(H_{1+})} \end{aligned} \quad (1)$$

Note that $P(I|H_1) = 1$, since in this case the initiator is identified correctly, and $P(I|H_{2+}) = 0$. If we let $f$ represent the fraction of nodes that are compromised, then:

$$P(I|H_{1+}) = \frac{P(H_1)}{P(H_{1+})} = \frac{f}{\sum_{i=1}^{\infty} (p(1-f))^{i-1} f}$$

Reiter and Rubin proposed the notion of *probable innocence* as happening whenever the true initiator is identified with a probability less than 1/2. By solving $P(I|H_{1+}) < 1/2$ for $f$, we can see that as long as $f < 1 - \frac{1}{2p}$, probable innocence will be assured. For example, with $p = 0.75$, up to 33% nodes can be malicious without compromising probable innocence. By increasing $p$, even larger fractions of compromised nodes can be tolerated, up to the limit of 50% when $p = 1$. (Of course, larger $p$ results in longer paths.)

## 4.1 The $E_1$ Attack

The chief difference between AP3 and Crowds is the manner in which the relays are chosen. Both aim to pick a relay at random out of all the nodes in the system, but Crowds assumes that all nodes know about all other nodes, which does not scale. AP3 uses the secure lookup due to Castro et al. to locate relays. To pick a relay, a node performs a secure lookup in the Pastry DHT for a random key. This, in turn, can be used to break probable innocence. In addition to the base observation—node $A$ used malicious node $B$ as a

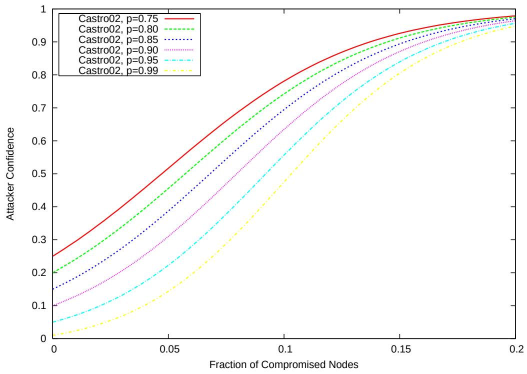


**Figure 3:** $P(I|E_1)$

relay—the malicious nodes have an extra observation point: whether any other node has performed a lookup for node $A$. We will define the event $E_1$ as the case when no lookups for $A$ have been detected. ($E_1$ implies $H_{1+}$.) We can then calculate the probability $P(I|E_1)$:

$$P(I|E_1) = \frac{P(I \wedge E_1)}{P(E_1)}$$

To calculate $P(E_1)$, we need to consider two cases: either $A$ is, in fact, the initiator ($H_1$), or some other node, $Q$, forwarded the request to $A$ ($H_{2+}$). In the former case, $E_1$ will be true unless there is another spurious lookup (false positive) for $A$ due to another request that is detected by the attackers. We call the spurious lookup event $FP$. In the latter scenario, we need two things to happen: first, no spurious lookup has happened, and second, the lookup from $Q$ to $A$ was not detected. We call this second event $Q$. Figure 2 represents the analysis of the two cases.

Therefore, we can express $E_1$ as:

$$E_1 \equiv (H_1 \wedge \neg FP) \vee (H_{2+} \wedge Q \wedge \neg FP)$$

Because $H_1$ and $H_{2+}$ are exclusive, and $FP$ and $Q$ are independent from $H_1$, $H_{2+}$, and each other, we can write:

$$P(E_1) = P(H_1)P(\neg FP) + P(H_{2+})P(\neg FP)P(Q)$$

Therefore,

$$\begin{aligned} P(I|E_1) &= \frac{P(H_1)P(\neg FP)}{P(H_1)P(\neg FP) + P(H_{2+})P(\neg FP)P(Q)} \\ &= \frac{P(H_1)}{P(H_1) + P(H_{2+})P(Q)} \qquad (2) \end{aligned}$$

Note that $P(I|E_1)$ can be computed independently of $P(FP)$; this is because we are conditioning on $E_1$, which implies that no spurious lookups have occurred. Note also that as $P(Q)$ grows smaller, the fraction approaches closer to 1. As we noted in the Section 3.1, with the secure lookup due to Castro et al., $P(Q)$ is quite small, even for small $f$.

Figure 3 shows the attacker confidence as a function of the fraction of the nodes that are compromised for varying $p$, using $N = 1000, b = 4, L = 16$. Our calculations show that to achieve $P(I|E_1) < 1/2$, we require that $f < 0.05$, which is much smaller than the previously computed limit of $f < 0.33$. Furthermore, the theoretical limit for the fraction of attackers that AP3 can tolerate can be computed by letting $p \to 1$, which is approximately 10% attackers. Again, this limit is much smaller than the conventional figure of 50%. This shows the fundamental tension that is encountered by AP3. The default Pastry mechanisms provide little defense against active adversaries who will try to disrupt the lookup process, dramatically increasing $P(H_1)$ and thus $P(I|H_{1+})$. Castro et al. suggested mechanisms solve this problem, but introduced another, as the lookup is no longer anonymous and can be observed by malicious nodes.

## 4.2 The $E_i$ attack

In addition to $E_1$, the can use the observation that if there is a chain of lookups leading to the predecessor node, then the first node in the chain is more likely to be the initiator than any other node. For instance, we can define $E_2$ as the case when attackers observe a lookup by some node Q of the previous hop (P), but do not detect a lookup for Q. Furthermore, the previous hop (P) should not have looked up any other nodes. We now compute $P(I|E_2)$. Depending on the probabilities of $P(E_2 \wedge H_1)$ and $P(E_2 \wedge H_2)$, the attacker may guess that $P$ or $Q$ is the initiator of the path.

These probabilities will depend on the chance of a false positive lookup detection, which in turn depends on the amount of lookup traffic elsewhere in the network. We define $x$ to be the number of paths that are being constructed (by all nodes) at the same time as this one. A reasonable number for $x$ is $N/100$, which means that during this path construction, 1% of all nodes also performed a concurrent path construction. A number much larger than this (e.g. $N/10$) would mean that nodes are spending a significant fraction of their time (10%) constructing paths, rather than using them for anonymous communication. Also, if any nodes in the network are not in active use, this will decrease $x$.

Given $x$, we can compute the false positive probability $\alpha$ using the following equation:

$$\alpha = 1 - \left(\frac{N-1}{N}\right)^{x\left(1-(1-f)^{L+\log_{2^b} N}\right)}$$

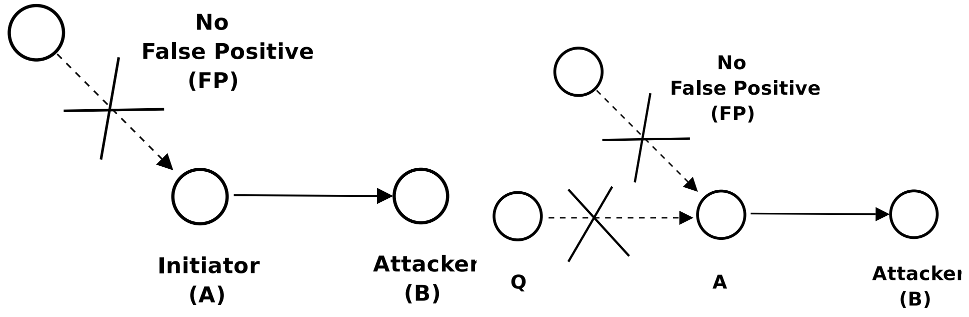
It is easy to see that as long as the false positive detection probability is small, $P(E_2 \wedge H_1) \ll P(E_2 \wedge H_2)$. Therefore, the attacker strategy here would be to guess the node (Q) looking up the previous hop to be the initiator. Therefore $P(I|E_2 \wedge H_1) = 0$ and $P(I|E_2 \wedge H_{3+}) = 0$.

$$P(I|E_2) = \frac{P(I|E_2 \wedge H_2)P(E_2 \wedge H_2)}{P(E_2 \wedge H_1) + P(E_2 \wedge H_2) + P(E_2 \wedge H_{3+})} \qquad (3)$$

Figure 4 plots $P(I|E_2)$ as a function of $f$ for varying $p$. The trend for $P(I|E_2)$ is very similar to our analysis of $P(I|E_1)$. Again, we can see that for $p = 0.75$, the maximum fraction of attackers that AP3 can handle while maintaining $P(I|E_2) < 1/2$ is only 5%. Due to lack of space, we have limited our analysis to only $P(I|E_1)$ and $P(I|E_2)$. In this sense, ours is a conservative analysis and the attackers can utilize many more observation points. For instance, one could define a general event $E_i$ analogous to $E_2$. If the false positives are small, $P(I|E_i)$ can be approximated as:

$$P(I|E_i) = \frac{P(H_i)}{P(H_i) + P(H_{(i+1)+})P(Q)}$$

The above formulation neglects false positives and is only an approximation. However, in practice, the approximation works quite well. In Figure 4, we can see that the results of



(a) $E_1 \wedge H_1$ (b) $E_1 \wedge H_{2+}$

**Figure 2: Information leak in AP3.**

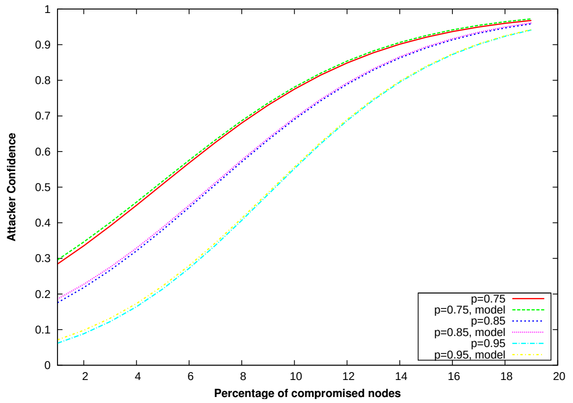


**Figure 4:** $P(I|E_2)$

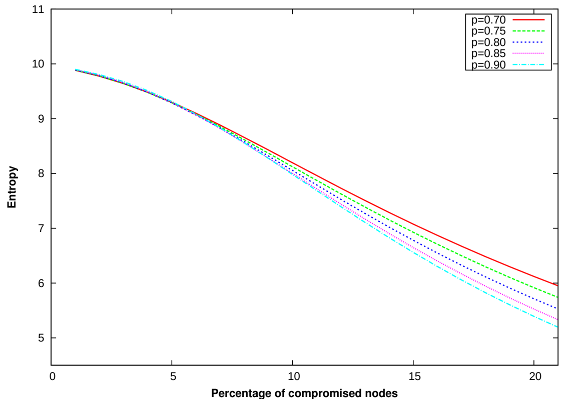


**Figure 5: Entropy as a function of $f$.**

the approximate model are quite close to the actual formulation that takes false positives into account.

Note that the metrics $P(I|E_1)$ and $P(I|E_2)$ are only indicative of the attacker confidence in identifying the initiator *given* the observations $E_1$ and $E_2$. They do not consider the probabilities of the attackers observing $E_1$ and $E_2$. We use the entropy metric of anonymity [15, 41] to take this into account. The metric relies on computing the entropy of the distribution of possible initiators of a path. In the case of $E_i$, the probability that the identified node is the initiator is $P(I|E_i)$, and the probability assigned to any other node is $\frac{1-P(I|E_i)}{N-1}$.[2] Let $H(E_i)$ be the entropy of the system under the observation $E_i$. Then, the average entropy can be computed as follows:

$$\begin{aligned} H = & P(E_1)H(E_1) + P(E_2)H(E_2) \\ & + (1 - P(E_1) - P(E_2)) \log_2 N \end{aligned}$$

Figure 5 plots the entropy as a function of $f$, for varying $p$, using $N = 1000$. Note that higher values of $p$ have *lower* entropy, and are thus considered to provide worse anonymity under the entropy metric. This is because with higher path lengths, the observation $E_2$ (and $E_3, E_4, \ldots$) is more frequent, even though each observation has lower confidence. The latter effect dominates, highlighting one of the open questions in anonymity analysis: is it better to have an anonymity system that allows weak attacks frequently, or strong attacks rarely?

# 5. SALSA

We shall now analyze Salsa's path building mechanism. For anonymous communication, a path is built between the initiator and the recipient via proxy routers (nodes). Layered encryption ensures that each node knows only its previous and next hop in the path. The nodes used for the paths are randomly selected from the global pool of nodes, even though each node has only local knowledge of a small subset of the network.

## 5.1 Salsa Path Building

To build a circuit, the initiator chooses $r$ random IDs ([34] sets $r = 3$) and redundantly looks up the corresponding nodes (called the first set/stage of nodes). Keys are established with each of these nodes. Each of the first set of nodes does a single lookup for $r$ additional nodes (second set of nodes). A circuit is built to each of the nodes in the second group, relayed through one of the nodes in the first group. Again, the initiator instructs the second set of nodes (via the circuits) to do a lookup for a final node. One of the paths created between the first and the second set of nodes is selected and the final node is added to the circuit. We use the parameter $l$ to refer to the number of stages in the circuit ([34] sets $l = 3$). Figure 7(a) depicts the Salsa

[2] This is a slight simplification; the entropy metric can take into account that, for example, in the case of $E_2$, $P$ is more likely to be the initiator than a random node.

path building mechanism for $r = 3$ and $l = 3$. Note that redundant lookups are used only to look up the nodes in the first stage; later lookups rely on the redundancy in the path building mechanism itself.

## 5.2 Active Path Compromise Attacks on Salsa

Active attacks on the lookup mechanism can bias the probability that nodes involved in Salsa's path building mechanism are compromised. Borisov et al. [6] noted that Salsa path building is also subject to a public key modification attack.[3] If all the nodes in a particular stage are compromised, they can modify the public keys of the next set of nodes being looked up. This attack defeats Salsa's bounds check algorithm that ensures the IP address is within the right range, since it cannot detect an incorrect public key. Also, since the traffic toward the node whose public key has been modified is forwarded via corrupt nodes, the attackers are guaranteed to intercept the messages. They can then complete the path building process by emulating all remaining stages (and hence, the last node). The public key modification attack and attacks on Salsa lookup mechanism are active attacks. Now, by end-to-end timing analysis, the path will be compromised if the first and last nodes in the circuit are compromised. Conventional analysis of anonymous communication typically focuses on minimizing the chance of path compromise attacks. By increasing the redundancy in the path building mechanism, this chance can be minimized. This is because increasing $r$ decreases the chance of both active attacks on lookups as well as public key modification attacks.

We now describe three types of passive information leak attacks on Salsa. We shall also show that increasing redundancy increases the effectiveness of the information leak attacks, resulting in a trade-off between robustness against active attacks and passive information leak attacks.

## 5.3 Conventional Continuous Stage Attack

A path in Salsa can be compromised if there is at least one attacker node in every stage of the path. Suppose that there are attacker nodes $A_1, A_2, A_3$ in the three stages respectively. In the path building mechanism, a node performs a lookup for all $r$ nodes in the following stage implying that $A_1$ would have looked up $A_2$ and $A_2$ would have looked up $A_3$. Hence the attacker can easily (passively) bridge the first and last stages, thereby compromising the anonymity of the system. This attack was mentioned in [34]. Note that if we increase redundancy as per conventional analysis, the effectiveness of the continuous stage attack also increases. This is because increasing redundancy increases the chance that attackers are present in each stage (which is $1-(1-f)^r$), giving them more opportunities to launch this attack. Next, we shall describe two new bridging attacks also based on information leaks from lookups.

## 5.4 Bridging an Honest First Stage

This attack is based on the observation that initiator performs redundant lookups for the nodes in the first stage. If the adversary can deduce the identities of the nodes in the first stage (they need not be compromised), and detect any of initiator's redundant lookups for nodes in the first stage, the anonymity of the system is compromised. Consider the Figure 7(a); malicious nodes are depicted in black. The first stage $(A_1, B_1, C_1)$ is comprised solely of honest nodes, the second stage $(A_2, B_2, C_2)$ has all malicious nodes and the third stage node $A_3$ is also compromised. The attackers know the identities of $A_1, B_1, C_1$ because of key establishment with them. Now if they detect a node performing a lookup for either $A_1, B_1$, or $C_1$, they can identify that node as the initiator. Since the initiator performs 9 lookups for the first stage nodes, the probability of detecting this initiator is $1-(1-f)^9$, which translates into a probability of 0.87 for $f = 0.2$. A similar attack strategy is applicable when only 2 or even one node in the second stage is compromised. In the latter scenario, the second stage knows the identity of only a single node in the first stage, and if the initiator is detected looking up that node, then the path is compromised. This occurs with probability $1-(1-f)^3$; which is 0.49 for $f = 0.2$. Similar to the continuous stage attack, notice that an increase in $r$ increases the probability that attackers can detect a lookup by the initiator for the first node.

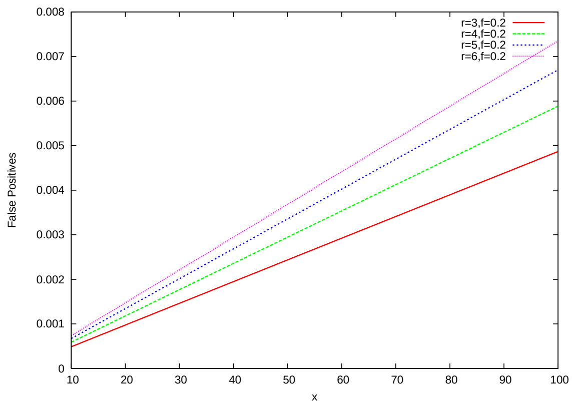


**Figure 6: False positives in bridging an honest first stage.**

It is important to note that there are some false positives in the attack. The false positives occur when a node (say $A_1$) in the first stage is involved in building more than one path. In such a scenario, more than one node will lookup $A_1$, and the attackers may detect a lookup for $A_1$ not done by the actual initiator. Using the variable $x$ to model the amount of lookup traffic by other nodes, as in Section 4.2, we can compute the false positives as:

$$1-\left(\frac{N-1}{N}\right)^{x(1-(1-f)^r)}$$

. Figure 6 depicts the false positives for varying $r$ using $f = 0.2, N = 1000$. Note that for $x < \frac{N}{100}$, the false positives are less than 0.1%.

## 5.5 Bridging an Honest Stage

Salsa is also vulnerable to a bridging attack where attacker nodes separated by a stage with all honest nodes are able to deduce that they are on the same path. Consider the arrangement of nodes depicted in Figure 7(b). The first stage has one malicious node $A_1$, the second stage consists solely of honest nodes, and the last node $A_3$ is compromised. $A_1$ knows the identities of all three nodes in the second stage; as it has performed a lookup for them. Also, as part of the path building mechanism, one of the nodes in the second stage will establish a key with the compromised third stage

[3] Their analysis did not take into account the lookup bias.

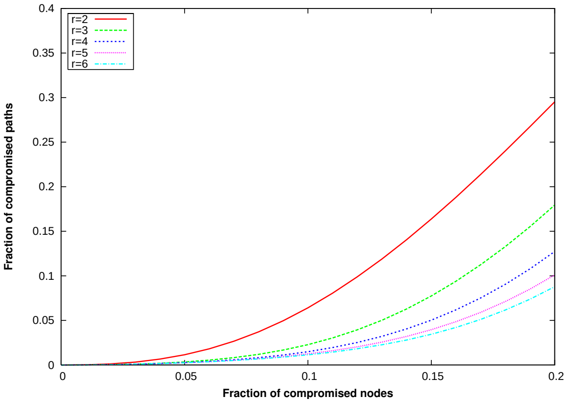


**Figure 8: Conventional path compromise attacks: Increasing redundancy counters active attacks.**

node $A_3$. In such a scenario, $A_1$ and $A_3$ can deduce that they are part of the same path as they both observe a common honest node. Similarly, if any of the nodes in the first stage are compromised and the last node is compromised, the path is compromised. In such an attack the compromised nodes in the first stage need not be selected as relays. Again, recall that increasing $r$ increases the chance of an attacker being present in a stage, resulting in a higher probability of bridging an honest stage. The probability of false positives in this scenario can be analyzed as $1 - (\frac{N-1}{N})^x$, which for $x = N/100$ and $N = 1000$ is less than 1%.

## 5.6 Results

We now present experimental results for active path compromise attacks and information leak attacks on Salsa. Our results have been computed by modeling the Salsa path building mechanism as a stochastic activity network in the Möbius framework [9]. For a fixed $f$ and $r$, the input to the model is the lookup bias, which was computed using the Salsa simulator [33], with simulation parameters $N = 1000, |G| = 128$.

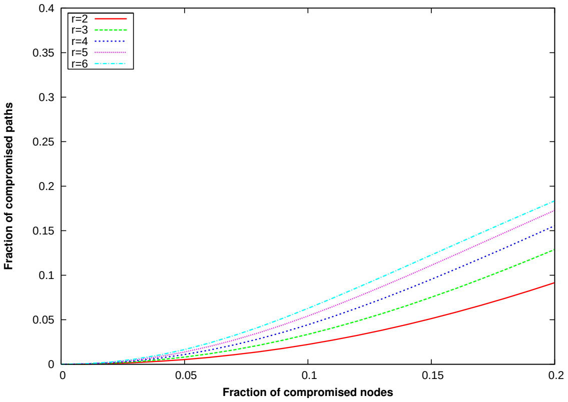


**Figure 9: Information leak attacks: Increasing redundancy makes the passive adversary stronger.**

Figure 8 shows the chance of active path compromise attacks on Salsa for varying levels of redundancy. It is easy to see that increasing $r$ reduces the fraction of compromised paths. For instance, at $f = 0.2$, 17% paths are compromised

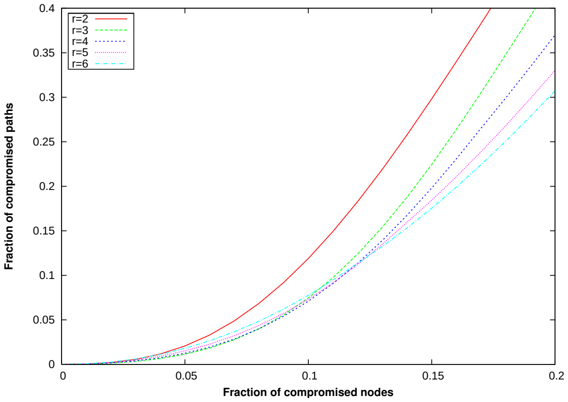


**Figure 10: All conventional and information leak attacks: For maximal anonymity, $r = 3$ is optimal for small $f$. Note that there is a crossover point at $f = 0.1$ when $r = 6$ becomes optimal.**

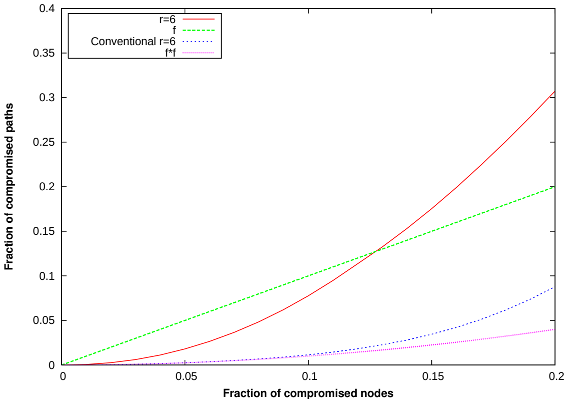


**Figure 11: Comparison of all attacks with conventional active attacks: Note that for $f > 0.12$, fraction of compromised paths is greater than $f$**

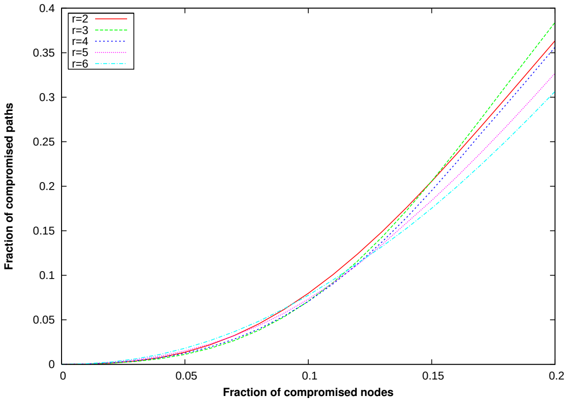
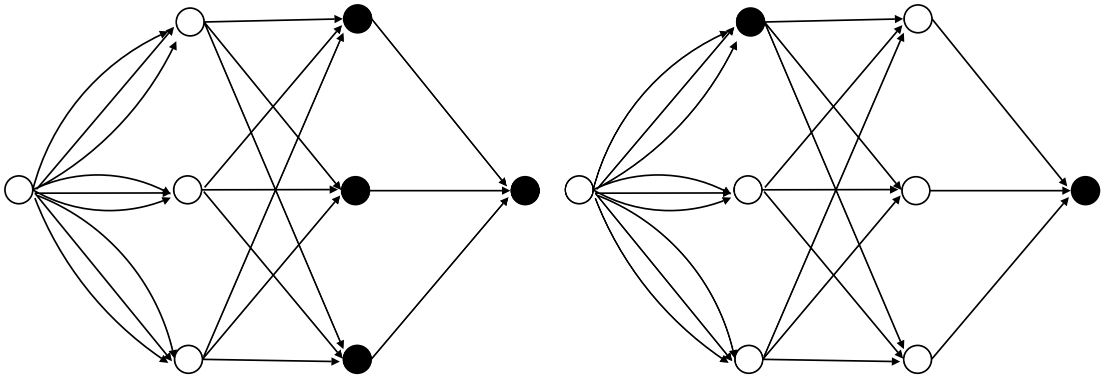


**Figure 12: Salsa with a PKI—All conventional and information leak attacks. Even with a PKI, the security of Salsa is much worse as compared to conventional analysis.**



(a) Bridging an honest first stage

(b) Bridging an honest stage

**Figure 7: Information leak attacks on Salsa.**

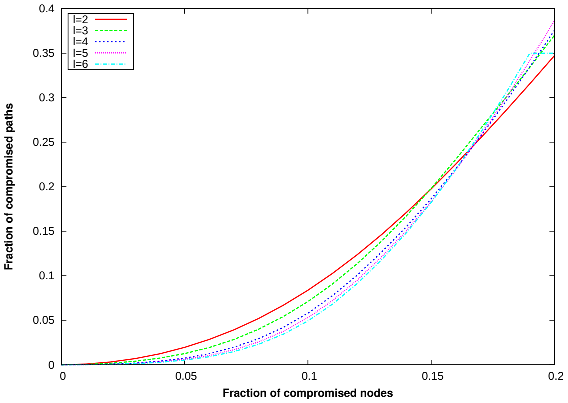


**Figure 13: Effect of varying the path length: Note that there is only limited benefit of increasing path length.**

using $r = 3$. The corresponding value for $r = 6$ is approximately 8%. This is not surprising, as increasing $r$ reduces the chance of both active attacks on lookups and attacks involving public key modification.

The continuous stage attack and both our bridging attacks are examples of passive attacks. Figure 9 shows the fraction of compromised paths under the passive attacks. We can see that an increase in $r$ increases the effectiveness of the passive attacks, and is detrimental to anonymity. For 20% attackers, even for a small value of $r = 3$, the initiator can be identified with probability 0.125. Higher values of $r$ can increase the probability of identifying the initiator to over 0.15. Note also that the bridging attack significantly improves upon the previous attacks on Salsa: using only the continuous stage attack, for $r = 3, f = 0.2$, anonymity is broken with a probability of only 0.048, less than half of what is possible with bridging.

The active path compromise attacks can be combined with passive information leak attacks. Figure 10 shows the fraction of compromised paths for all passive and active attacks. An interesting trend is observed where increasing redundancy (beyond $r = 2$) is detrimental to security for small values of $f$. This is in sharp contrast to conventional analysis; the inclusion of information leak attacks have made the effect of passive attacks more dominant over the effect of active attacks. There is a crossover point at about 10% malicious nodes, after which increasing $r$ reduces to probability of path compromise. This is because active attacks are dominant for higher values of $f$. Note that $r = 2$ results in significantly worse security because of poor resilience to both lookup attacks and public key modification attacks.

This shows the tension between the passive and active attacks. There is an inherent redundancy in Salsa path building mechanism to counter active attacks. However, the redundancy makes the passive adversary stronger and provides more opportunities for attack. From Figure 11 we can see that by conventional analysis, security provided by Salsa is close to that of Tor ($f^2$). With our information leak attacks taken into account, for $f > 0.12$, the security provided by Salsa is even worse than $f$.

## 5.7 Improvements to Salsa

We next consider whether simple changes to Salsa's mechanisms would provide a defense against our attacks. First, we consider Salsa using a PKI, as in AP3. The public key modification attack would no longer work; however, other active attacks on the lookup mechanism and our passive information leak attacks would still apply. Figure 12 depicts the probability of identifying the initiator under all active and passive attacks in Salsa with PKI. Again, we can see the tension between active and passive attacks. Increasing redundancy (beyond $r = 2$) is detrimental to security for small values of $f$, because of the dominance of our information leak attacks. There is a crossover point, after which active attacks become dominant, and increasing $r$ increases security. With the public key modification attack gone, $r = 2$ becomes a more reasonable parameter, but even with a PKI, the fraction of compromised paths increases from 8% under conventional active attacks to more than 30% with our information leak attacks taken into account.

Finally, we explore the effect of increasing the path length ($l$) on the anonymity of Salsa. Figure 13 depicts the probability of identifying the initiator for varying values of $l$. There is an interesting trade-off in increasing the path length. On one hand, increasing $l$ reduces the chance of information leak attacks, because the attacker needs to bridge all stages. On the other hand increasing $l$ gives attackers more opportunities to launch active attacks, thereby increasing the probability that last node is compromised, which in turn gives attackers more observation points. This is basically a cascading effect: the presence of a malicious node in each stage increases the probability of presence of malicious nodes in the next stage. For small values of $f$, passive attacks are stronger, therefore increasing $l$ increases security, but for higher $f$, the active attacks and the cascading are dominant, therefore increasing $l$ decreases security.

We have proposed passive bridging attacks on Salsa that are based on information leaks from lookups, and can be launched by a partial adversary. Moreover, we have shown a trade-off between defenses against active and passive attacks. Even at the optimal point in the trade-off, the anonymity provided by the system in significantly worse than what was previously thought. This trade-off is present even in Salsa with a PKI. Moreover, increasing path length in Salsa has only a limited benefit on the user anonymity.

## 6. RELATED WORK

Secure routing in peer-to-peer networks has been the subject of a lot of research [43, 47, 7, 34, 25]. We studied lookup mechanisms proposed by Castro et al. [7] and Nambiar and Wright [34], focusing on the information leak from lookups, and observed a trade-off between security and anonymity of a lookup. Kapadia and Triandopoulos recently proposed Halo [25], which is also based on redundant routing, and exhibits a similar trade-off. Moreover, it uses very high redundancy levels as compared to Salsa, and would make our information leak attacks more effective. There have been some attempts to add anonymity to a lookup. Borisov [5] proposed an anonymous DHT based on Koorde [24], which performs a randomized routing phase before an actual lookup. Ciaccio [8] proposed the use of imprecise routing in DHTs to improve sender anonymity. These lookups were designed to be anonymous, but not secure: an active adversary could easily subvert the path of the lookup. As such, neither lookup mechanism can be used to build anonymous circuits.

Danezis and Clayton [11] studied attacks on peer discovery and route set up in anonymous peer-to-peer networks. They show that if the attacker learns the subset of nodes known to the initiator (by observing lookups, for example), its routes can be fingerprinted unless the initiator knows about the vast majority of the network. Danezis and Syverson [14] extend this work to observe that an attacker who learns that certain nodes are *unknown* to the initiator can carry out attacks as well and separate traffic going through a relay node. These attacks are similar in spirit to the ones we propose, but rather than absolute knowledge of the initiator's routing state, we use probabilistic inferences based on observed lookups. Recently, Bauer et al. [2] proposed a bridging attack in Tor where attacker nodes sandwiching an honest node can correlate the path even before a packet is sent. This attack is similar to our bridging attack on Salsa, except that we also utilize information leaks from lookups, and consider the issue of false positives.

Reiter and Rubin [38] proposed the predecessor attack, which was later extended by Wright et al. [48, 49, 50]. In this attack, an attacker tracks an identifiable stream of communication over multiple communication rounds and logs the preceding node on the path. To identify the initiator, the attacker uses the observation that initiator is more likely to be the predecessor than any other node in the network. For peer-to-peer anonymous communication systems like Salsa, the number of rounds required by predecessor attack to identify the initiator with high probability is inversely proportional to the probability of success of end-to-end timing analysis. This means that defenses that minimize the probability that both the first and last nodes are attackers also increase resilience against predecessor attacks.

Similar to predecessor attacks, there is a thread of research that deals with degradation of anonymity over a period of time. Berthold et al. [4] and Raymond [37] propose intersection attacks that aim to compromise sender anonymity by intersecting sets of users that were active at the time the intercepted message was sent, over multiple communication rounds. Similarly, Kesdogan et al. [26] use intersection to find recipients of a given users message. A statistical version of this attack was proposed by Danezis [10] and later extended by Mathewson and Dingledine [27]. These attacks typically require an adversary to observe a significant fraction of the network. Information leaks in peer-to-peer systems, however, can allow even a partial adversary to make observations about a large fraction of lookups and path building, and can therefore form a basis of effective statistical intersection and disclosure attacks.

An important point of our paper is that, when building anonymous systems, it is important not to abstract away the properties of the system that can affect anonymity. Our analysis of AP3 is an example of how composition of two designs that are secure individually [38, 7] creates new vulnerabilities. Similar in spirit to ours, a lot of recent research has focused on details abstracted away by conventional analysis models to break the anonymity of the system. Such details include congestion and interference [31, 1], clock skew [30], heterogeneous path latency [23, 1], the ability to monitor Internet exchanges [32], and reliability [6]. Due to lack of space, we only briefly discuss the last two attacks. Conventional anonymity models of Tor view a connection from a client to a server as point to point link, and abstract away the fact that this connection passes through the internet routers. Murdoch and Zieliński [32] showed that Internet exchange-level adversaries were capable of observing a vast majority of this traffic, and could degrade user anonymity by performing end-to-end timing analysis. Borisov et al. [6] proposed a selective-DoS attack on anonymous communication, and show that attackers could selectively affect the reliability of the system in states that are hardest to compromise. The Selective-DoS attack affects peer-to-peer anonymous communication the most, because of the added complexity of knowing only a subset of the nodes in the network.

## 7. CONCLUSION

We have analyzed information leaks in the lookup mechanisms of peer-to-peer anonymous communications systems. Existing defenses against active attacks typically use redundant messages, which enables a relatively small fraction of attackers to observe a large number of lookups in the network. Attackers are thus able to act as a near global passive adversary and use this to break the anonymity of the system.

We have shown how attacks based on information leaks from lookups can be used to break the probable innocence guarantees in AP3. We computed the limit on the number of attackers that AP3 can handle while providing probable innocence as only 5% in the typical case, while the theoretical limit with increased path lengths is 10%; this is in contrast to the conventional analysis, which puts these figures at 33% and 50% respectively. A small fraction of malicious nodes can therefore compromise the security of AP3.

An important lesson learned from the AP3 analysis is that the composition of a secure DHT lookup mechanism with an anonymous communication protocol (as has been considered in other work [42]) should be carefully analyzed, as it is likely to introduce additional vulnerabilities.

We have also analyzed the security of Salsa under both active and passive attacks. We have demonstrated the tension that exists between defending against both active and passive adversaries. Defending against active adversaries requires higher redundancy, which increases the threat of passive attacks. Salsa was previously reported to tolerate up to 20% compromised nodes, but our results show that, with information leaks taken into account, over a quarter of all tunnels are compromised. Moreover, we show that the tension between active and passive attacks exists even if Salsa were to use a PKI. Also, increasing path lengths to counter our passive attacks only has a limited benefit, and in some cases, it even reduces anonymity.

Our results demonstrate that information leaks are an important part of anonymity analysis of a system and that new advances in the state of art of P2P anonymous communication are needed.

## Acknowledgments

We would like to thank Arjun Nambiar, Nayantara Malesh, and Matthew Wright for providing us with the Salsa simulator and for helpful discussions about the Salsa algorithms. We are also grateful to George Danezis, Matthew Wright, and the anonymous reviewers for their comments on earlier versions of this paper.

## 8. REFERENCES

[1] A. Back, U. Möller, and A. Stiglic. Traffic analysis attacks and trade-offs in anonymity providing systems. In I. S. Moskowitz, editor, *Information Hiding Workshop*, volume 2137 of *Lecture Notes in Computer Science*, pages 245–247. Springer-Verlag, Apr. 2001.

[2] K. Bauer, D. McCoy, D. Grunwald, T. Kohno, and D. Sicker. Low-resource routing attacks against Tor. In T. Yu, editor, *Workshop on Privacy in Electronic Society*, pages 11–20, November 2007.

[3] S. M. Bellovin and D. A. Wagner, editors. *IEEE Symposium on Security and Privacy*, May 2003.

[4] O. Berthold, H. Federrath, and M. Köhntopp. Project "anonymity and unobservability in the Internet". In L. Cranor, editor, *Tenth conference on Computers, Freedom and Privacy*, pages 57–65, New York, NY, USA, 2000. ACM.

[5] N. Borisov. *Anonymous Routing in Structured Peer-to-Peer Overlays*. PhD thesis, UC Berkeley, May 2005.

[6] N. Borisov, G. Danezis, P. Mittal, and P. Tabriz. Denial of service or denial of security? How attacks on reliability can compromise anonymity. In Wright and Syverson [52], pages 92–102.

[7] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. S. Wallach. Secure routing for structured peer-to-peer overlay networks. In D. Culler and P. Druschel, editors, *Symposium on Operating Systems Design and Implementation*, pages 299–314. USENIX, Dec. 2002.

[8] G. Ciaccio. Improving sender anonymity in a structured overlay with imprecise routing. In Danezis and Golle [13], pages 190–207.

[9] D. Daly, D. D. Deavours, J. M. Doyle, P. G. Webster, and W. H. Sanders. Möbius: An extensible tool for performance and dependability modeling. In B. R. Haverkort, H. C. Bohnenkamp, and C. U. Smith, editors, *Computer Performance Evaluation: Modelling Techniques and Tools*, volume 1786, pages 332–336, Schaumburg, IL, Mar. 2000. Springer.

[10] G. Danezis. Statistical disclosure attacks: Traffic confirmation in open environments. In Gritzalis, Vimercati, Samarati, and Katsikas, editors, *Proceedings of Security and Privacy in the Age of Uncertainty, (SEC2003)*, pages 421–426, Athens, May 2003. IFIP TC11, Kluwer.

[11] G. Danezis and R. Clayton. Route fingerprinting in anonymous communications. In *IEEE Conference on Peer-to-Peer Computing*, pages 69–72. IEEE Computer Society, Sept. 2006.

[12] G. Danezis, R. Dingledine, and N. Mathewson. Mixminion: Design of a Type III anonymous remailer protocol. In Bellovin and Wagner [3], pages 2–15.

[13] G. Danezis and P. Golle, editors. *Sixth Workshop on Privacy Enhancing Technologies*, volume 4258 of *Lecture Notes in Computer Science*, Cambridge, UK, June 2006. Springer.

[14] G. Danezis and P. Syverson. Bridging and fingerprinting: Epistemic attacks on route selection. In N. Borisov and I. Goldberg, editors, *Privacy Enhancing Technologies Symposium*, volume 5134 of *Lecture Notes in Computer Science*, pages 151–166. Springer, July 2007.

[15] C. Diaz, S. Seys, J. Claessens, and B. Preneel. Towards measuring anonymity. In Dingledine and Syverson [17], pages 184–188.

[16] R. Dingledine, N. Mathewson, and P. Syverson. Tor: The second-generation onion router. In M. Blaze, editor, *USENIX Security Symposium*, pages 303–320. USENIX Association, Aug. 2004.

[17] R. Dingledine and P. Syverson, editors. *Privacy Enhancing Technologies Workshop*, volume 2482 of *Lecture Notes in Computer Science*. Springer, April 2002.

[18] J. Douceur. The Sybil Attack. In Druschel et al. [19], pages 251–260.

[19] P. Druschel, F. Kaashoek, and A. Rowstron, editors. *International Workshop on Peer-to-Peer Systems (IPTPS)*, volume 2429 of *Lecture Notes in Computer Science*. Springer, Mar. 2002.

[20] H. Federrath, editor. *International Workshop on Design Issues in Anonymity and Unobservability*, volume 2009 of *Lecture Notes in Computer Science*. Springer, July 2000.

[21] M. J. Freedman and R. Morris. Tarzan: A peer-to-peer anonymizing network layer. In R. Sandhu, editor, *ACM Conference on Computer and Communications Security*, pages 193–206, New York, NY, USA, 2002. ACM.

[22] D. Goodin. Tor at heart of embassy passwords leak. *The Register*, September 10 2007.

[23] N. Hopper, E. Y. Vasserman, and E. Chan-Tin. How

much anonymity does network latency leak? In Wright and Syverson [52], pages 82–91.

[24] M. F. Kaashoek and D. R. Karger. Koorde: A simple degree-optimal distributed hash table. In F. Kaashoek and I. Stoica, editors, *International Workshop on Peer-to-Peer Systems (IPTPS)*, volume 2735 of *Lecture Notes in Computer Science*, pages 98–107. Springer, Feb. 2003.

[25] A. Kapadia and N. Triandopoulos. Halo: High-assurance locate for distributed hash tables. In C. Cowan and G. Vigna, editors, *Network and Distributed System Security Symposium*, pages 61–79, Feb. 2008.

[26] D. Kesdogan, D. Agrawal, and S. Penz. Limits of anonymity in open environments. In F. A. Petitcolas, editor, *Information Hiding Workshop*, volume 2578 of *Lecture Notes in Computer Science*, pages 53–69. Springer-Verlag, October 2002.

[27] N. Mathewson and R. Dingledine. Practical traffic analysis: Extending and resisting statistical disclosure. In D. Martin and A. Serjantov, editors, *Workshop on Privacy Enhancing Technologies*, volume 3424 of *Lecture Notes in Computer Science*, pages 17–24. Springer, May 2004.

[28] A. Mislove, G. Oberoi, A. Post, C. Reis, P. Druschel, and D. S. Wallach. AP3: Cooperative, decentralized anonymous communication. In M. Castro, editor, *ACM SIGOPS European Workshop*. ACM, 2004.

[29] U. Möller, L. Cottrell, P. Palfrader, and L. Sassaman. Mixmaster Protocol—Version 2. IETF Internet Draft, July 2003.

[30] S. J. Murdoch. Hot or not: Revealing hidden services by their clock skew. In Wright and di Vimercati [51], pages 27–36.

[31] S. J. Murdoch and G. Danezis. Low-cost traffic analysis of Tor. In V. Paxson and M. Waidner, editors, *IEEE Symposium on Security and Privacy*, pages 183–195, May 2005.

[32] S. J. Murdoch and P. Zieliński. Sampled traffic analysis by Internet-exchange-level adversaries. In N. Borisov and P. Golle, editors, *Privacy Enhancing Technologies Symposium*, volume 4776 of *Lecture Notes in Computer Science*, pages 167–183. Springer, June 2007.

[33] A. Nambiar and M. Wright. The Salsa simulator. `http://ranger.uta.edu/~mwright/code/salsa-sims.zip`.

[34] A. Nambiar and M. Wright. Salsa: a structured approach to large-scale anonymity. In Wright and di Vimercati [51], pages 17–26.

[35] P. Palfrader. Number of running Tor routers.

[36] L. D. Paulson. News briefs. *IEEE Computer*, 39(4):17–19, April 2006.

[37] J.-F. Raymond. Traffic analysis: Protocols, attacks, design issues, and open problems. In Federrath [20], pages 10–29.

[38] M. Reiter and A. Rubin. Crowds: Anonymity for web transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, June 1998.

[39] M. Rennhard and B. Plattner. Introducing MorphMix: Peer-to-peer based anonymous Internet usage with collusion detection. In *Workshop on Privacy in Electronic Society*, pages 91–102, Washington, DC, USA, November 2002.

[40] A. Rowstron and P. Druschel. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In G. Goos, J. Hartmanis, and J. van Leeuwen, editors, *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, volume 2218 of *Lecture Notes in Computer Science*, pages 329–350. Springer, Nov. 2001.

[41] A. Serjantov and G. Danezis. Towards an information theoretic metric for anonymity. In Dingledine and Syverson [17].

[42] M. Sherr, B. T. Loo, and M. Blaze. Towards application-aware anonymous routing. In T. Jaeger, editor, *Workshop on Hot Topics in Security*. USENIX Association, Aug. 2007.

[43] E. Sit and R. Morris. Security considerations for peer-to-peer distributed hash tables. In Druschel et al. [19], pages 261–269.

[44] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup protocol for Internet applications. *IEEE/ACM Transactions on Networking*, 11(1):17–32, 2003.

[45] P. Syverson, G. Tsudik, M. Reed, and C. Landwehr. Towards an analysis of onion routing security. In Federrath [20], pages 96–114.

[46] P. Tabriz and N. Borisov. Breaking the collusion detection mechanism of MorphMix. In Danezis and Golle [13], pages 368–383.

[47] D. Wallach. A survey of peer-to-peer security issues. In M. Okada, B. Pierce, A. Scedrov, H. Tokuda, and A. Yonezawa, editors, *International Symposium on Software Security*, volume 2609 of *Lecture Notes in Computer Science*, pages 253–258. Springer, 2002.

[48] M. Wright, M. Adler, B. N. Levine, and C. Shields. An analysis of the degradation of anonymous protocols. In P. van oorschot and V. Gligor, editors, *Network and Distributed System Security Symposium*, pages 39–50, Feb. 2002.

[49] M. Wright, M. Adler, B. N. Levine, and C. Shields. Defending anonymous communication against passive logging attacks. In Bellovin and Wagner [3], pages 28–41.

[50] M. Wright, M. Adler, B. N. Levine, and C. Shields. The predecessor attack: An analysis of a threat to anonymous communications systems. *ACM Transactions on Information and System Security*, 4(7):489–522, November 2004.

[51] R. Wright and S. D. C. di Vimercati, editors. *The 13th ACM Conference on Computer and Communications Security*, New York, NY, USA, Oct. 2006. ACM.

[52] R. Wright and P. Syverson, editors. *The 14th ACM Conference on Computer and Communications Security*, New York, NY, USA, 2007. ACM.