# Evaluating Sender and Recipient Anonymity in a Structured Overlay *

Giuseppe Ciaccio
DISI, Università di Genova
via Dodecaneso 35
16146 Genova, Italy
ciaccio@disi.unige.it

**DISI Technical Report DISI-TR-05-13**

**3 October 2005**

## Abstract

*An open problem in structured overlay networks is related to the anonymity to be provided to recipients, namely, those nodes who respond to request messages. Such a feature is of main concerns when designing censorship-resistant distributed applications. In this paper it is shown that, in a chordal ring overlay, by enforcing a degree of imprecision in each peer's routing table we obtain better recipient anonymity while keeping the length of routing paths within logarithmic length. A suitable metrics for recipient anonymity is established, based on the amount of resources an adversary needs in order to break anonymity of recipients in the overlay. In terms of this metrics, it is shown that imprecise routing tables make it impossible for a "small" coalition of malicious peers to correlate overlay addresses to hosts for censorship or auditing purposes. As an aside, this approach is also shown to provide a "better" sender anonymity: good anonymity levels become more likely at the expenses of very low and very high ones.*

**Keywords**: structured overlay, anonymity, distributed hash table, censorship-resistance, peer to peer

## 1. Introduction and motivation

Peer to peer overlay networks have been receiving a lot of attention by the research community, as flexible and scalable low-level infrastructures for distributed applications of many kinds: network storage [17, 12, 37], naming [11], content publication [15, 10, 3, 35, 45, 39], multicast/anycast [33, 6, 28], and communication security [30, 46]. They have also been proposed as general networking infrastructures [16, 43, 19, 18], because of their potential ability to decouple network addresses from physical placements of cooperating hosts, an important feature for privacy and mobility.

The vast population of existing or proposed overlay systems can be broadly divided into two families, namely, unstructured overlays and structured overlays.

Structured overlays [32, 44, 13, 27, 21, 36, 26, 47] are receiving far more attention lately, because of performance guarantees they can in principle provide thanks to their regular topologies. Regular topologies allow routing algorithms to provably converge, and a careful choice of entries in routing tables can reduce the number of routing hops to even a constant quantity, independent of the overlay size [20, 25]. The most known example of a structured overlay is the chordal ring (Figure 1): $N$ peers are arranged in a ring, and each can route messages towards the *successor* in the ring as well as a small ($O(\log(N))$) number of other peers, called *fingers*, whose "distances" increase according to a geometric progression. With this organization, a message can be delivered in $O(\log(N))$ hops according to a so called "greedy" routing (Figure 2 and Section 3.2).

On the other hand, unstructured overlays like Freenet [10] and GNUnet [2] first leveraged techniques to enhance identity privacy or *anonymity* of participant entities.

Trivial indirection based on source rewriting and usual cryptographic machinery, or, even better, mixes chains [7, 4], can help hide the identity of a message *sender*. But there is another face of the coin, namely, *recipient* anonymity. This has to do with the ability,

*This research is supported by the Italian FIRB project *Webminds*.

or rather the difficulty, to correlate an overlay recipient address to the physical Internet address of the destination host. In order to break recipient anonymity, an adversary needs to build a *map of the overlay*, relating overlay addresses to host addresses. An overlay provides recipient anonymity whenever it makes it difficult or impossible for an adversary to build such a map.

Let us suppose that messages in the overlay do not convey any explicit information about the sender or recipient Internet addresses (anonymity would be hopeless otherwise). All of the information relating overlay addresses to Internet identities is kept into the peers' routing tables. A common instance of such information is the set of pairs $\langle overlay\ address, IP\ address \rangle$ that forms each peer's knowledge of remote peers. In order to map a given part of the overlay, an adversary needs to harvest sufficiently many routing tables in the system, either by attacking many honest nodes or, more probably, by deploying a large enough coalition of malicious nodes. Another precious source of information, however, is given by the shape of the overlay graph. In structured overlays, the presence of an arc between two nodes in the graph represents a mathematical relation between the overlay addresses of the two corresponding peers. For instance, arcs in a chordal ring represent distances on the overlay address space which conform to a geometric progression. Thus, in a structured overlay, the routing tables are more informative than in unstructured overlays. This improves the routing performance, but also helps the attacker wishing to map the overlay.

By contrast, the irregularity of unstructured overlays helps recipient anonymity. This is because unstructured networks lack of constraints between arcs in the graph and distances in the overlay address space, and this implies that the map between overlay address and Internet address of a recipient can be hidden to any other peer in the system, including neighbours. This is the reason for the success of unstructured networks as a support for censorship-resistant distributed storages [10], in which storing and retrieving are recipient roles. However, this feature is paid in terms of efficiency and availability.

Thus, there is an obvious interest to find trade offs between the efficiency of structured overlays and the privacy offered by unstructured ones, with the goal of improving recipient anonymity without seriously affecting routing performance or sender anonymity. This indeed is the motivation of the work accounted here.

It must be said, however, that recipient anonymity alone is not sufficient for censorship resistance: without the coexistence of sender anonymity, a document could be censored by exerting menace upon that document's readers (which play sender roles in a network storage systems, as they send write and read requests to storage hosts). Therefore, any compromise between performance and recipient anonymity should not impact too much on sender anonymity.

## 2. Gauging recipient anonymity

As pointed out at the end of last Section, a truly censorship-resistant system may only rely on both recipient and sender anonymity. At a first glance these two features are similar, but they actually do not share much with each other and demand different techniques, because of the different roles played by senders in comparison with recipients. Indeed, a sender is an active entity that issues requests, whereas a recipient may or may not issue replies. The fundamental difference is that requests make use of the routing algorithm to propagate, whereas replies usually do not (they often backtrack the same route of their corresponding requests [34], so that a return address is unnecessary and sender anonymity is preserved). Thus, although possibly observed, replies do not carry useful information related to the place where they have been issued. In our opinion, this is the reason why the various metrics for sender anonymity [8, 2, 5, 14, 40] have never been applied to the recipient case, although it is commonly believed it should be possible to do so. We rather argue that metrics of recipient anonymity must be quite different from metrics of sender anonymity.

Communication indirection is a common way to obtain both kinds of anonymity in an overlay network, but, as shown in Section 1, this is only sufficient in unstructured overlays, and is paid in terms of efficiency and availability.

Scarlata et al. [38] propose two recipient-oriented anonymizing recipes. One of them is based on IP multicast communication, but for its very nature it cannot be easily deployed on the current Internet. The other one is based on a proxy who publishes a recipient address decoupled from the recipient identity. Recipient anonymity is preserved, and performance too, but the system is not censorship-resistant, as the proxy identity is exposed to the adversary.

Serjantov [39] proposes a more sophisticated use of proxies. In that system, the public address of any given document is decoupled from the overlay placement of the document itself. The publisher divides its document into $M$ shares, $K$ of which are sufficient to rebuild the document. Then the publisher chooses $M$ overlay locations at random, called *forwarders*, to which the shares are sent. Each forwarder in turn chooses a peer,

called a *storer*, sends its own share of data to it, then forgets about the publisher's identity while remembering about the storer. The forwarders are thus the only places in the overlay where the indirection information needed to retrieve the actual document is kept. The public address of the document is made of the entire set of forwarders, plus other naming information. But, if at least $M-K+1$ forwarders are under adversarial control or under attack, the document becomes unreachable. The system thus requires that $M$ be far greater than $K$ in order for it to be reasonably censorship-resistant, and even so a powerful enough adversary can kick the document off.

Robust censorship-resistance is probably impossible to achieve without a significant sacrifice of efficiency. Indirection on unstructured overlays offers no performance or availability guarantees, structured overlays can always be mapped by sufficiently many colluders, and proxies can always be attacked or controlled by sufficiently powerful adversaries.

On the other hand, an adversary striving to break censorship-resistance, and recipient anonymity in particular, must deploy some resources: either malicious participants to a structured overlay, or servers for massive denial-of-service attacks.

Because of the above arguments, it is the author's opinion that recipient anonymity in a structured overlay, but also censorship-resistance in a generic distributed system, could be gauged in terms of the *amount of resources an adversary must control in the system* in order to achieve his goal. By dividing such amount of resources by the total amount of resources involved in the whole system, we get a measure that we call the *relative adversarial effort*. The larger such measure, the more resistant the system is to the adversarial actions. If $N$ is the system size (e.g., the number of peers in an overlay) and $K(N)$ is the size of the adversarial coalition (e.g., the number of malicious peers), then the relative effort is $E(N) = K(N)/N$.

On the ground of the above definition we propose the term *pretty good* (recipient) *anonymity* to denote anonymity that cannot be broken with asymptotically small effort. The effort $E(N)$ is termed *asymptotically small*, or "small" for short, when it approaches 0 when $N$ gets large. Our definition of pretty good anonymity, while recognizing that perfect anonymity is unachievable, formalizes the intuition that an anonymous system which could be defeated by a small minority of colluders cannot be regarded as a "good" one.

# 3. Preliminary assumptions

## 3.1. Anonymity metrics and attacker model

In this paper we focus on recipient anonymity rather than the weaker censorship-resistance, and use the relative adversarial effort as a metrics. When dealing with anonymity properties of the sender, however, we conform to other studies on structured overlays [24, 29, 41] by adopting the size of the anonymity set [8] as a metrics.

Our discussion on recipient anonymity assumes an adversarial model that, following [14], we term "internal, local, and passive"; that is, the adversary controls a number of peers in the system, each of which complies to the overlay protocol and does not generate malicious traffic, but can maliciously gather information from its internal routing tables as well as any messages it happens to forward. An "active" adversary poses much different security challenges, beyond our current scope. Global adversaries, either external (capable of observing possibly any message across the entire overlay) or internal (capable of controlling possibly any peer in the network) appear to be unrealistic in a large peer to peer system.

Moreover, we assume that the overlay protocol does not explicitly disclose the identity of any participant. This might seem an obvious requirement, but if we recall that the original protocol of Chord provides the IP address of the storer of required data as the result of the lookup of a desired key, we understand that this requirement is not so obvious. In an anonymous storage-retrieval system, a request for data returns the data themselves rather than the network identity of a storage server.

## 3.2. Generalized chordal rings

Although we believe that the entire work presented here could be adapted to any structured overlay, for practical purposes we had to choose a reference model of overlay network. In this paper we focus on the most widely used such model, namely, the aforementioned chordal ring. We present here a generalized version of the concepts originally introduced by [44]. Similar concepts are found in every scalable overlay.

Let us consider a set of peers logically organized into an overlay shaped as a ring. Each peer has a link to its own *successor* in the ring. The overlay supports the abstraction of a generic *address space*, consisting of the set of $2^k$ binary words of $k$ bits ordered as a circle modulo $2^k$. This space is mapped onto the ring of peers in

consecutive chunks or *address intervals*. For the purpose of our work, it is uninteresting to give meaning to the data possibly "stored" at each overlay address. In other words we choose an application-neutral standpoint, and therefore prefer the term "overlay network" to the more popular "distributed hash table".

If peer $P$ owns the address interval from $A_l$ to $A_u$, and peer $Q$ is the successor of $P$, then all addresses owned by $Q$ are greater than $A_u$ (modulo $2^k$).

In the ideal scenario in which the map of addresses onto peers is complete, that is, no "hole" is left between each peer and its successor, a message issued towards address $A$ can always reach the *recipient* (the peer which owns $A$ in its own local address interval) by traversing the successor chain starting from the sender. In a realistic scenario, however, a faulty or disconnected peer could break the successor chain and also create a "hole", a discontinuity in the address space. A degree of redundancy (backup locations) can help, but the system must quickly seal the successor chain and restore the address hole, or redundancy would eventually degrade. To this end, an easy solution is to allow each peer to know a *successor list*, rather that just the immediate successor. This allows a peer to talk directly to its successor's successor to seal the ring in case the successor has gone (the extension to the case of multiple adjacent faulty peers is straightforward).

Messages issued towards distant (in the overlay address space) recipients would take too long to reach their destinations through the successor chain. For distant recipients we need an alternate routing algorithm, so as to keep the routing path below an acceptable size. A common such mechanism is given by the *fingers*. A finger is an entry in the routing table of a peer, pointing to a distant peer in the overlay. The distance is measured between (one of the bounds of) the local address interval and (the corresponding bound of) the remote address interval. Each peer maintains its own list of fingers. Finger distances obey a mathematical requirement that we call the *distance rule*. The distance rule is often geometric on base 2. Given a bottom value $C$, called *cutoff*, the first finger has the largest possible distance $\leq C$ from local peer, the second finger has the largest possible distance $\leq 2C$, the third finger has the largest possible distance $\leq 4C$, and so on, up to spanning half of the address space. The finger at distance $C \cdot 2^m$ is said to have *magnitude* $m$; we will also call it the "finger $m$" for brevity. Clearly, each peer can have at most $O(\log(N))$ fingers. Figure 1 illustrates this concept.

The routing algorithm takes advantage of fingers in a so-called "greedy" way (Figure 2). When a peer $P$ gets an incoming message whose destination address is

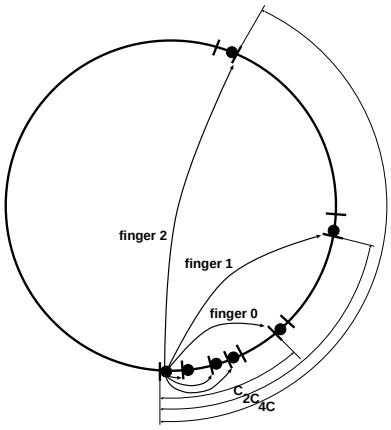


**Figure 1. An instance of a chordal ring.**

$A$, it acts as follows:

1. $P$ checks out if $A$ is locally owned; if so, the message has arrived and no routing is needed;

2. otherwise, $P$ computes the residual distance $D$ yet to be travelled by the message, as the difference between $A$ and (one of the bounds of) the locally owned address interval;

3. $P$ chooses the finger of largest magnitude whose distance does not exceed $D$, and forwards the message to it. If no such finger is found, $P$ forwards to successor.

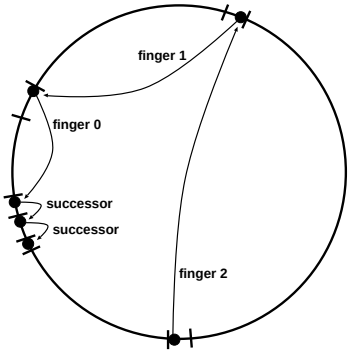


**Figure 2. "Greedy" routing in a chordal ring. With $N$ participants and complete routing tables, $O(\log(N))$ hops are sufficient.**

In a chordal ring with complete finger tables conforming to a geometric distance rule, a total travel distance of $D$ is covered in two phases, namely:

- $O(\log_2(D/C))$ finger hops until the residual distance falls shorter than the cutoff $C$; then
- an $O(1)$ number of short-range hops through the successor chain.

## 4. Imprecise routing information

From the point of view of recipient anonymity, having large routing tables at each peer is detrimental, because in such a case an adversary controlling a small number of peers could quickly map large portions of the overlay (Section 1). Large routing tables are detrimental to sender anonymity as well, since they would allow shorter routing paths, and the mixing techniques against traffic analysis are less effective when the mixex chains are short.

Thus, imposing severe constraints on the maximum size of routing tables is necessary to improve both kinds of anonymity. However this may not be sufficient. For instance, in a chordal ring with $N$ peers, routing tables as small as $O(\log(N))$ allow routing in $O(\log(N))$ hops but also allow an adversarial coalition of $O(N/\log(N))$ peers to map the entire overlay, with a relative effort $E(N) = O(1/\log(N))$. Following the definitions in Section 2, we say that in this case the anonymity of any recipient could be broken with "small" adversarial effort ($E(N)$ approaches 0 when $N$ grows).

In a chordal ring, the minimum size a coalition must have in order to map an entire overlay of $N$ peers is upper bounded by $N/S_r$, where $S_r$ is the size of the routing tables. The adversarial effort $E(N)$ is thus $1/S_r$. If $S_r$ was allowed to increase with $N$, the effort would be "small". If, however, $S_r$ was fixed, then it would be impossible to route in $O(\log(N))$ hops when the system is large, and thus the system would be unscalable. Apparently there is no way for recipient anonymity.

The above argument could be put against any structured overlay. Note, however, that we are implicitly assuming that routing tables carry the exact overlay addresses of the remote peers they point to, that is, each peers knows the exact distance to any peer listed in its own routing table. Things might become better by relaxing this constraint.

Our idea is that a routing table should be allowed to grow large enough to allow $O(\log(N))$ routing, but not larger, in order to preserve sender anonymity through sufficiently long routing paths. In addition, a routing table should only be allowed to contain a small and fixed amount of exact addressing information, whereas most of the information in the table should be *imprecise*. Indeed, only exact information can contribute to map the overlay, so if its amount is kept small and fixed in each routing table, a coalition of peer wishing to map the overlay could not be "small". This way, a pretty good recipient anonymity (Section 2) would be provided.

Having imprecise entries in the routing table raises questions concerning convergence and efficiency of the routing algorithm. The routing paths could become longer, or be unsuccessful. The introduction of imprecise information in the routing table must fullfill some constraints, which depend on the overlay graph, so that convergence is preserved and routing paths do not become excessively long. In the following we propose a way to manage imprecise entries in the routing tables of a chordal ring so that routing convergence is ensured, routing efficiency is preserved, but the routing tables themselves are of no help for an adversarial coalition to map the overlay.

## 5. Successors and their anonymity issues

Let us consider a set of peers organized into a chordal ring. We have already pointed out (Section 3.2) that a better resilience is obtained by having each peer manage a successor list rather than a single link to the immediate successor.

Note that, in order for the messages to follow the successor chain, no peer is required to know the address interval of any member in its own successor list, not even the immediate successor. The knowledge of the locally owned address interval is indeed sufficient to decide whether a message can be managed locally or has to proceed on. This is good news for recipient anonymity: the lesser a peer knows, the lesser an adversary can know.

On the other hand, a peer $P$ wishing to seal a hole immediately ahead of it must explore its own successor list, find the member $Q$ immediately after the hole, and ask $Q$ to reveal the lower bound of its address interval, so as to know the extent of the hole to be repaired. To ensure recipient anonymity, however, $Q$ should never disclose its own interval to anybody. We clearly need a compromise, and propose that $Q$ only reveals something to $P$ if $Q$ can verify the following two conditions:

- $Q$'s predecessor in the ring has gone;
- $P$ belongs to $Q$'s predecessor chain.

To sum up, recipient anonymity requires that each peer knows and manages a *predecessor list*, in addition to the successor list.

At this point, a legitimate question would be to what extent the management of successor and predecessor lists could be exploited by a coalition to map the

overlay and defeat recipient anonymity. The answer is that the occasional disclosure of addressing information to a predecessor cannot be exploited by an adversary, because more and more information could only be gained at the price of opening more and more holes in the address ring, perhaps killing more and more peers. On the other hand, the topological information found in each successor list is exact (and has to be kept so through periodical maintenance), but it contributes very little against recipient anonymity: each peer can estimate the address interval owned by its successor, but the overall degree of recipient anonymity is still "pretty good" (Section 2) because the coalition needed to harvest such information on the entire ring would size $O(N)$.

## 6. Fingers and their anonymity issues

In a chordal ring, the most efficient way to build and maintain a finger table takes advantage from the recursive nature of the geometric distance rule. To find the finger 0, $P$ sends a suitable request along its successor chain, perhaps in a recursive style [44], until the most distant peer still within cutoff distance $C$ is found. To find a finger of magnitude $m > 0$, $P$ asks its own current finger $m-1$ to be contacted by its finger $m-1$. [1] Such an *incremental procedure* minimizes the number of contacted peers, so it should be preferred when anonymity is of concern, because it can minimize the information leak towards potential adversaries.

However, the above (traditional, after Chord) definition of fingers poses two serious threats on recipient anonymity, namely:

1. If peer $P$ has peer $Q$ as its own finger of magnitude $m$, then $P$ knows that $Q$'s address interval is more or less at distance $C \cdot 2^m$ from itself. Thus, $Q$'s address interval is indirectly disclosed to $P$. In general, in a ring counting $N$ peers, each participant has $O(\log(N))$ fingers and thus can map the address intervals of as many other peers. A malicious coalition counting $O(N/\log(N))$ peers, which corresponds to a "small" relative effort for the adversary, can thus map the entire overlay, as already pointed out in Section 4.

2. When searching finger 0, peer $P$ exposes its own address interval to the whole successor chain up to the finger; this is bad for recipient anonymity, especially if the successor list is long.

[1] In case the address interval of $P$ spans the entire cutoff distance, the finger of magnitude 0 could not be found. In this case $P$ starts by directly searching its finger of magnitude $n$ along successor chain within distance $C \cdot (n+1)$, where $n$ is such that $C \cdot (n+1)$ is larger than the size of $P$'s address interval.

## 7. Improving anonymity with imprecise fingers

The two anonymity flaws outlined at the end of last Section are impossible to fix, because they are implied by the traditional definition of fingers. To improve recipient anonymity we must modify such definition, by making use of the concept of imprecise routing envisioned in Section 4.

Our goal is to obfuscate part of the topological information conveyed by traditional fingers, and to protect peers against excessive exposure when they search their fingers of magnitude 0.

In the following, $F_P$ is a secret random value generated by the generic peer $P$ with uniform probability over $[0, C/4[$ (the cutoff distance $C$ has been introduced in Section 3.2). $F_P$ thus cannot exceed $C/4$.

Let us suppose peer $P$ use the incremental procedure outlined in Section 6 to build its own finger table. The first step is to find the finger of magnitude 0. As already pointed out, finding finger 0 potentially exposes the address interval of $P$ to the whole successor chain, up to finger 0, because of the need for all successors to compute their distance from $P$. To fix such information leak, $P$ acts as follows: rather than sending (the lower bound of) its own address interval to the successor, $P$ alters the information by subtracting $F_P$, then sends the altered value $A$.

Let $Q$ be a generic peer in $P$'s successor chain. After receiving an altered value $A$ from its predecessor, it acts as follows:

1. $Q$ estimates (the lower bound of) its successor's address interval, $L$; this is trivial, as the successor is adjacent in the address space.

2. $Q$ computes the distance between $A - F_Q$ and $L$, namely, $L - A + F_Q$. This amounts to computing the distance between the original requestor, namely $P$, and $Q$'s successor, incremented by the sum of the two random secrets $F_P$ and $F_Q$; the computed distance is thus greater than the real one by an unknown quantity in $[0, C/2[$, since each random secret is in $[0, C/4[$.

3. If such distance is greater than $C$, then $Q$ contacts the original requestor $P$ and claims to be its finger 0.

4. Otherwise $Q$'s successor is a better candidate, so $Q$ forwards the value $A$ to the successor.

Since the request emitted by $P$ is initially altered by the random quantity $F_P$, $P$ is not disclosing its own addressing information to the successors. When $P$ is

eventually contacted by a peer $Q$ claiming to be its finger of magnitude 0, $P$ can conclude that the distance of $Q$ is the largest possible within an upper bound randomly distributed between $C-C/2 = C/2$ and $C$. This is a substantial departure from the traditional definition of fingers, according to which the distance of finger 0 is known with far greater precision (the largest distance not exceeding $C$).

The approximation introduced on finger 0 cumulates over fingers of greater magnitude. Using the incremental procedure, a finger of magnitude 1 is imprecise because its distance is close to a value between $2 \cdot (C/2) = C$ and $2C$, a finger of magnitude 2 is imprecise because its distance is close to a value between $4 \cdot (C/2) = 2C$ and $4C$, and so on. The generic finger $m$ is located at a distance which is the largest possible not exceeding an unknown random value between $C \cdot 2^{m-1}$ and $C \cdot 2^m$ (Figure 3).

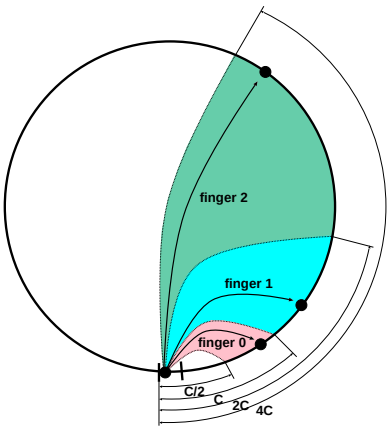


**Figure 3. A chordal ring (successors omitted) in which the fingers are affected by an unknown random error. The average error increases proportionally with the distance of the peer. This way, no peer can infer much about the overlay addresses of other peers, and this improves recipient anonymity. Yet, $O(log(N))$ hops are still sufficient to route messages towards an arbitrary destination.**

Such an amount of finger imprecision is a good device for recipient anonymity. The more distant a finger, the lesser the information about the overlay addresses it actually owns. A lot of indeterminacy can be put on even the closest of fingers, provided the cutoff $C$ be large compared to the average size of address intervals of peers. No matter how large, no adversarial coalition can gather sufficiently exact information from finger tables; on the other hand, the successor lists have already be shown to be poor of useful information for an adversary (see Section 5).

## 8. Imprecise fingers and routing performance

The traditional "greedy" routing algorithm converges even with the imprecise fingers defined in Section 7. The number of hops increases, of course, but not dramatically.

Traversing a traditional, *exact* finger of magnitude $m$ reduces the residual distance by no more than $C \cdot 2^m$, as we have seen in Section 3.2. With the imprecise fingers, however, a hop through a finger of magnitude $m$ accomplishes a distance which is shorter or equal compared to the exact fingers of same magnitude, but never null. We already know [44] that the "greedy" routing with exact fingers converges. In our case the fingers fall shorter, and the successor relation is preserved, so the same convergence arguments of [44] apply.

At worst, hopping through an imprecise finger of magnitude $m$ decreases the residual distance by no more than $C \cdot 2^{m-1}$. Therefore, to accomplish a distance decrease of $C \cdot 2^m$, it is necessary to traverse no more than two consecutive fingers of same magnitude $m$. We thus conclude that the use of imprecise fingers at most doubles the worst-case number of traversed fingers. In a system with complete finger tables, the routing paths thus remains $O(\log_2(D/C))$. [2]

In Section 9 we shall evaluate the actual increase of path length as observed in a number of simulations.

## 9. Simulation results

In order to validate the effectiveness of imprecise fingers, evaluate their impact on various aspects of system performance, and make a comparison with the traditional approach of exact fingers, we have built a simulator for a chordal ring over an address space made of 32-bit addresses.

In the simulated system with imprecise fingers, the cutoff distance $C$ covers at least 20 bits of address and at least 10 consecutive peers on average; fingers are built according to the incremental procedure outlined in Section 6. By contrast, the reference system with exact fingers has same number of peers and same address

[2] Actually, if the random secret $F_P$ owned by the generic peer $P$ was randomly distributed over an interval smaller than $[0, C/4[$, the routing paths would be shorter. We have chosen to let $F_P$ span a quarter of the cutoff distance because this way all the formulas were simpler, yet the generality of the result is preserved.

space, but the cutoff distance is set to 1 as with traditional chordal rings; fingers are computed explicitly rather than incrementally, in order to avoid that higher magnitude fingers could be affected by cumulated imprecision arising from fingers of lower magnitude.

After creating a sample ring with $N$ uniformly distributed peers, the simulator fills up each peer's successor list and finger table; fingers can be imprecise or exact, depending on a compile-time flag. The simulator then computes the standard deviation of finger distances across all peers as a function of the magnitude; for such a computation, the distances over the ring are normalized to $[0, N]$ so that the standard deviation is expressed in terms of (average) peers rather than distances on the overlay address space (this is made feasible by the uniform distribution of peers over the ring). At this point, in order to evaluate statistics of path lengths, the simulator generates all the routing paths from each peer to the peer owning the overlay address 0 (any destination addresses is equivalent to 0 modulo rotational transformation of the chordal ring). Finally, in order to evaluate the different impact of imprecise vs. exact fingers on sender anonymity, the simulator generates a number of adversarial configurations over the ring; the fraction $f$ of adversaries over the entire population is specified at runtime, and the simulator obtains each adversarial configuration by marking each peer as adversary with probability $f$. For each adversarial configuration, the simulator computes statistics of the anonymity sets of all "honest" peers, by processing the set of all routing paths as follows:

1. For each "honest" peer $P$, scan the routing path from $P$ to address 0 until the first adversarial peer is found. Let us call $last(P)$ the result of the scan. If the path does not meet adversaries, $last(P)$ is assigned the pair $\langle -1, -1 \rangle$; otherwise $last(P)$ is assigned the pair $\langle C, m \rangle$, where $C$ is the first adversarial peer found along the path, and $m$ is the magnitude of the finger which was the last hop in the path up to $C$, or -1 if such last hop was a successor link. The reason why we take the magnitude of last hop into account shall become clear at the next step.

2. For each adversarial peer $C$, count all "honest" peers $P$ such that $last(P) = \langle C, m \rangle$ with given $m$. Let us call $a(C, m)$ such count. $a(C, m)$ is the size of the anonymity set that the colluder $C$ can associate to a generic lookup for address 0 that it could intercept. The reason why this anonymity set depends on $m$ is that $C$ can indeed discriminate among possible originators of a lookup by looking at which incoming link the lookup has came from; intuitively, a lookup coming from the immediate predecessor may have a lot of possible originators, whereas a lookup coming from a link corresponding to the finger of greatest magnitude may only have one originator.

3. For each "honest" peer $P$, if $last(P) = \langle C, m \rangle = \langle -1, -1 \rangle$ then the anonymity set size from $P$'s point of view is equal to the total number of "honest" peers in the ring, namely, $f \cdot N$; otherwise, the anonymity set size is equal to $a(C, m)$. The case of unintercepted lookup is thus taken into account when estimating the average sender anonymity from the sender viewpoint. [3]

All the results have been collected by running the simulator over 500 sample rings of given size.

### 9.1. Finger imprecision and recipient anonymity

The standard deviation of finger distances is an indicator of the degree of imprecision of fingers themselves, provided the expectation of each finger distance is consistent with the overlay's distance rule (Section 3.2). If, in a population of sample rings, finger distances did not differ much from their mean values, any adversarial peer could deduce the address interval of each of its own fingers with good confidence, and thus a coalition could get a precise map of the overlay with high probability. This however would not be the case, was the standard deviation significant. So a high standard deviation of finger distances in a population of chordal rings is good news for recipient anonymity, and indeed the proposed algorithm for imprecise fingers (Section 7) aims at increasing such a statistics. We also expect greater standard deviation with greater finger magnitude, because of the incremental procedure by which imprecise fingers are built (Section 6).

Figure 4 represents the standard deviation of finger distances as a function of the magnitude, with different ring sizes. Both imprecise and exact fingers have been evaluated. In all cases, the expectation of each finger distance has been checked against the theoretical value as given by the distance rule (base-2 geometric progression starting with given cutoff distance) and no significant difference was found, so the standard deviation is a correct indicator of finger imprecision.

For an easier comparison among different ring sizes, and in order to help interpreting the results themselves, finger distances and their deviations have been normalized to the interval $[0, N]$ where $N$ is the number of

[3] Actually, the simulator does not require three distinct scans of the entire set of peers in order to accomplish the above three steps.

peers in the ring. This amounts to taking the average size of a peer's address interval as distance unit over the ring. With this normalization, a deviation of 100 means that the corresponding finger is likely to point up to 100 peers away from the correct position on the ring.

Due to the different sizes and cutoff distances of the involved chordal rings, fingers on one ring cannot be compared to fingers on another ring, not even if they have the same magnitude. This is the reason why the curves in Figure 4 are not aligned with one another. Nevertheless, the Figure clearly shows that traditional fingers deviate from their expectations by less than the average size of a peer's address interval, that is, they are practically exact as expected. By contrast, imprecise fingers are affected by a sharply larger deviation (as expected as well).

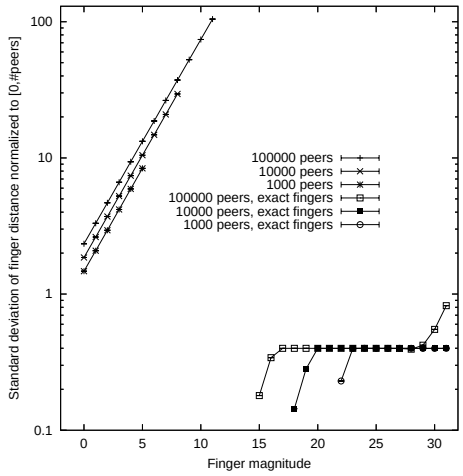


**Figure 4. Finger imprecision as represented by the standard deviation of finger distances. Imprecise fingers are indeed affected by a large deviation, especially when the magnitude is high. By contrast, exact fingers (the ones traditionally found in chordal rings) are affected by negligible imprecision no matter the magnitude. The address intervals of the rings are normalized to the number of peers, so that the finger deviations are expressed in terms of (average) peers.**

Assuming a deviation of 5 as a threshold between "sufficiently precise fingers" and "fingers too imprecise to be useful for mapping the overlay", we can deduce, for instance, that in a ring with 10000 peers equipped with imprecise fingers, only fingers of magnitude 0, 1, and 2 are useful for the adversary. If we also consider the immediate successor as a further piece of precise information, we deduce that each colluding peer can map at most 4 other peers in the overlay. We therefore conclude that the relative adversarial effort cannot be less than 1/4 in this case, that is, an adversarial coalition must count at least 25% of all participants in order to break anonymity of an arbitrary recipient. The same analysis in the case of exact fingers yields a relative adversarial effort of $1/\log_2(N)$, which roughly amounts to 0.075 with 10000 peers and 0.06 with 100000 peers.

So, imprecise fingers do improve recipient anonymity quite a lot.

Another advantage of imprecise fingers is that the amount of approximation, and the degree of recipient anonymity thereof, can be increased by choosing a larger cutoff distance. This however would be paid with longer routing paths.

### 9.2. Impact on path lengths

As already envisaged in Section 8, the use of imprecise fingers can lead to a greater number of hops along the route from sender to recipient. We now report the results provided by the simulator concerning this specific performance metrics.

As apparent from Figure 5, chordal rings with imprecise fingers have quite longer routing paths compared to rings with traditional fingers. If we compare the curves of routing path length with imprecise vs. traditional fingers, and focus on the worst cases points, we see the difference never exceed the factor of 2 predicted in Section 8. But, with traditional fingers, the number of hops has a great variability and is often much smaller that the worse cases; this does not happen with imprecise fingers, which exhibit a more regular profile. So the difference between the two families of curves appears more dramatic.

However, the two kinds of simulated chordal rings also have different cutoff distances, besides using different kinds of fingers. In order to isolate the impact of finger imprecision from the impact of cutoff distance, we ran the simulator on a "hybrid" overlay in which the cutoff distance is the same as in the system with imprecise fingers, but fingers are of traditional kind. The results with 1000 peers are reported in Figure 6. The curve yielded by the "hybrid" system has a highly variable shape, similar to the system with traditional fingers, but the span is much larger and reaches the curve of the system with imprecise fingers. We conclude that the two ingredients of our recipe for anonymity, namely, cutoff distance and imprecise fingers, have different costs: the cutoff distance impacts on the worse-case path length, whereas the finger imprecision smooths the profile of path length by driving the whole curve

towards the worse case.

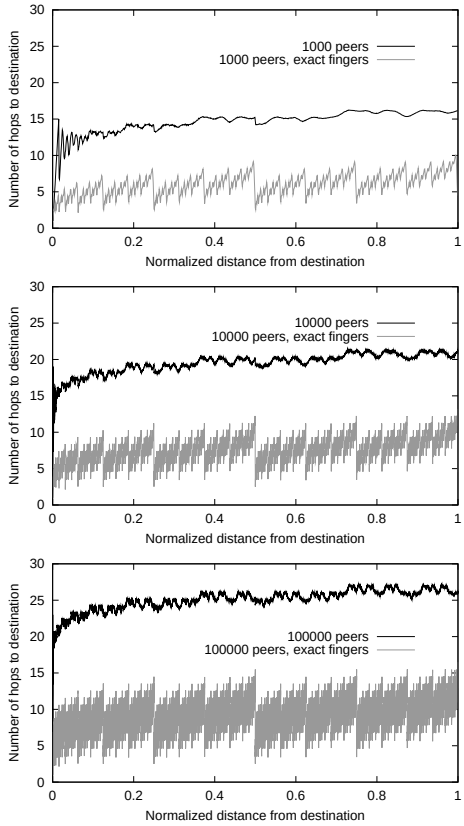


**Figure 5. Average length of routing path from each sender to destination 0 in a chordal ring with given number of participants. Each sender is represented by its normalized distance from destination in the address space. Both imprecise and traditional fingers have been evaluated.**

### 9.3. Impact on sender anonymity

Finally, we provide a comparison between rings with imprecise fingers and rings with traditional fingers with respect to the degree of sender anonymity (size of the anonymity set). Imprecise fingers are aimed at recipient anonymity, yet they would be impractical, should their use compromise senders.

The results have been obtained by running the simulator over 500 sample rings of given size, each with 100 sample adversarial configurations with given percentage of attackers.

Let us first discuss the average sender anonymity as a function of the distance between sender and recipient (this distance is normalized to the size of the complete address space). The overall result is that, with imprecise fingers, the average sender anonymity only decreases by a small amount compared to traditional fingers. This is shown by Figure 8, where chordal rings with both kinds of fingers are compared with one another with varying percentage of attackers.

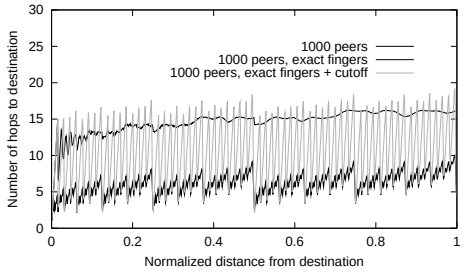


**Figure 6. Average length of routing path from each sender to destination 0 in a chordal ring with 1000 peers. Each sender is represented by its normalized distance from destination in the address space. In addition to imprecise and traditional fingers, a "hybrid" kind of chordal ring has been evaluated, in which fingers are traditional but the cutoff distance is the same as in the system with imprecise fingers.**

Another important insight is that, on average, the sender anonymity in both cases is quite high when the percentage of attackers is not overwhelming. It becomes very poor when this percentage raises 50%, but a system with so many attackers should be considered as highly compromised indeed.

It is also apparent that systems with more peers provide a better anonymity; however, they also appear to be more sensitive to the percentage of attackers (Figure 7).

Of course, the average sender anonymity alone is not informative enough. The variability around the average value must also be considered. Indeed, we can observe that such variability is always strong, regardless of the fingers being imprecise or not. Figure 9 shows the average frequency distribution of sender anonymity in chordal rings with 1000 peers and three different percentages of attackers, where the averaging concerns all peers in the overlay. It is apparent that the choice between imprecise or traditional fingers leads to deeply different distributions of sender anonymity: with imprecise fingers all distributions are more sparse and span a large interval of pretty good anonymity levels, as opposed to traditional fingers which only span too small or very large anonymity levels. In other words,

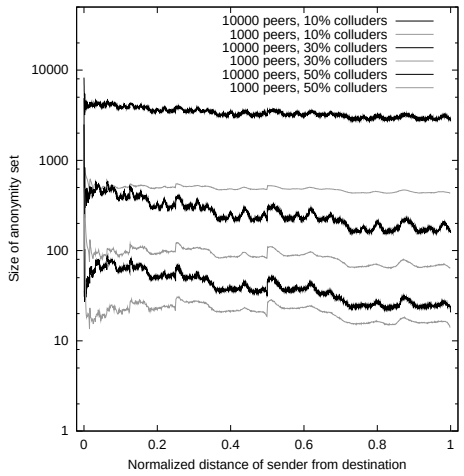


**Figure 7. Average sender anonymity in simulated chordal rings with 1000 and 10000 peers and imprecise fingers, with three different percentages of attackers. The order of captions reflects the order of the curves from top to bottom. Sender anonymity is better if the number of participants is high, but in such a case it also becomes more sensitive to the percentage of colluders.**

imprecise fingers increases the probability to get a fairly good sender anonymity. The same conclusions can be drawn for systems with 10000 peers (Figure 10). It is the author's opinion that such a better distribution largely compensates for the slightly lower average level of sender anonymity.

Finally, in order to isolate the impact of finger imprecision from the impact of cutoff distance, we ran the simulator on "hybrid" chordal rings in which the cutoff distance is the same as in the system with imprecise fingers, but fingers are of traditional kind. After simulating overlays with 1000 and 10000 peers, no significant differences have emerged with respect to traditional chordal rings of same size. We thus conclude that the changes of distribution of sender anonymity are entirely due to the finger imprecision.

## 10. Related work

Imprecise routing information is at the core of unstructured overlays. With Freenet [10], for example, a message directed to key $A$ is routed towards a node $P$ if $P$ has previously been able to route back responses from keys "similar" to $A$. Thus, a routing table entry that points to $P$ does not say anything about the keys actually stored at $P$, nor does it say much about the placement of $P$ in the overlay topology. GNUnet and MUTE follows a similar approach, with some more randomness [2, 35].

There are very few attempts to improve anonymity in structured overlays. Achord [23] is an enhancement of Chord[44] with anonymity features. Aiming at enforcing sender anonymity, Achord implements recursive-style [44] routing (because of the indirection) and forces each response to travel back to sender along the same route previously tracked by the corresponding request (so that the sender address need not be disclosed to the receiver). In addition, Achord restricts each peer's ability to know about other peers: each peer is allowed to know the IP addresses of at most $k*\log(N)$ other peers, where $k$ is the finger table size. The goal is similar to ours, namely, to enforce recipient anonymity by making it more difficult for an adversarial coalition to map the overlay. However the proposed solution is weak: the entire Achord ring could be mapped by a coalition of $N/k*\log(N) = O(N/\log(N))$ peers, which is "small" according to the definition given in Section 2 (the relative effort $E(N)$ amounts to $1/\log(N)$, which tends to 0 as $N$ grows). Our work strived to overcome such a limitation.

Other studies [24, 5, 29] only focus on sender anonymity in plain Chord, without considering recipient anonymity. According to [29], Chord provides a good amount of sender anonymity in terms of size of anonymity set. This is in apparent contradiction with [24]. Apparently, the difference between these two works is that the former considers the anonymity from the attacker point of view, whereas the latter chooses the point of view of the generic "honest" sender. In addition, the latter work shows an analytical mistake, since the event that a lookup is not intercepted by any adversary is overlooked in the anonymity evaluation. Such an event is not so unlikely, and its impact on the average anonymity set size makes a difference. As we have seen in Section 9, in order to estimate the sender anonymity of our system, we followed the approach suggested by [5] by choosing simulation rather than analytical tools. We too preferred the sender viewpoint when estimating sender anonymity, but did not forget about the weight of unintercepted messages, so our results look a lot better that the ones in [24].

Agyaat [42] provides a compromise between anonymity and efficiency by means of a two-level hybrid organization in which the Chord structured overlay works together with the Gnutella unstructured system. Gnutella-like "clouds" are connected with one another by means of a Chord ring. It is an interesting and very effective approach that deserves a deeper

anonymity analysis.

SkipNet [22] and Skip Graphs [1], both inspired to the Skip List data structure [31], and Symphony [26], are overlay networks that make use of somehow randomized routing entries. However, randomized routing tables are not enough to obtain imprecise routing. In the aforementioned systems, indeed, each peer is autonomous in choosing the distance each routing entry is to point at. By contrast, imprecise routing requires that no peer has precise knowledge of such distances.

## 11. Conclusions and open issues

The most important result reported in this paper concerns recipient anonymity. After defining a metrics for this elusive feature, we have proposed the use of some randomization on long-range connections in structured overlay networks as a mean for obtaining better recipient anonymity without sacrificing the nice properties of structured overlays (provable routing convergence and, to some extent, performance). The study has been conducted on chordal rings, where our idea takes the form of what we call *imprecise fingers*. We however believe a similar study can be carried out on other topologies.

The positive impact of imprecise fingers on recipient anonymity, as envisaged by theory, has been confirmed by simulations.

Our result can be summarized by saying that anonymous routing can be accomplished even in a chordal ring, and can be done in $O(\log(N))$ hops where $N$ is the number of peers in the overlay. If we liked slogans, we would say that anonymity can be efficient.

In practice, the impact of imprecise fingers on routing paths is not negligible. The number of hops increase quite much, although remaining within $O(\log(N))$. A substantial growth of the routing path length raises performance and availability concerns: latency-sensitive applications might suffer, and longer routing paths are also more sensitive to the failure probability of individual peers. The cutoff distance of the chordal ring has been found as one of the parameters that directly affects the path lengths; future investigations are thus in order, concerning the role of cutoff distance in the trade-offs between anonymity, efficiency, and availability.

Another important result concerns the impact of our randomization mechanism on sender anonymity. The proposed solution would have been impractical, was recipient anonymity obtained at the expenses of sender anonymity. However, the simulations have shown that the average sender anonymity decreases of a small amount, and this decrease is compensated by a better distribution of the sender anonymity levels: good levels become more likely at the expenses of very low and very high levels. As an aside, this work also presents a deep evaluation of sender anonymity of traditional chordal rings.

We have also managed to embody the mechanism of imprecise routing tables into NEBLO [9], a chordal ring overlay with anonymity features. NEBLO is still in beta development stage, yet it has already been released to the community (under the GNU General Public Licence).

An unexplored security concern is about the algorithm by which a new peer joins the overlay. In order to preserve anonymity, it is crucial that colluding peers be given no control on which position in the overlay they are going to occupy. The obvious, and widely adopted, rule based on the pair $\langle IPaddress, port \rangle$ of the newcomer, however, appears weak as long as the adversary is able to use an IP domain of choice. NEBLO implements a slightly better join algorithm which however does not solve the problem completely.

Another security concern is about the incremental procedure outlined in Section 6, that each peer should follow when building its own finger table. Let us suppose the generic peer $P$ wants to locate its own finger 0. It issues a request which travels along the successor chain until a valid candidate is met. But, if the request meets an adversarial peer, from then on the whole incremental procedure can be diverged towards the adversarial coalition. The finger 0 would be an adversary, and the same would occur with all fingers of greater magnitude, and thus the sender anonymity of $P$ would be entirely compromised. No variant of such a procedure can prevent this kind of opportunistic attack from occurring: any kind of search for fingers may possibly end up in a colluder, and from there on the search can be fully managed within the adversarial coalition. We thus conclude that sender anonymity is impossible to achieve as long as routing tables are built by running the routing protocol itself. Yet, one could wonder about algorithms for finger location that decrease the strike probability of this opportunistic attack.

## References


[1] J. Aspnes and G. Shah. Skip Graphs. In *Proc. of the 14th Annual ACM-SIAM Symp. on Discrete Algorithms (SODA '03)*, Jan. 2003.

[2] K. Bennett and C. Grothoff. GAP: Practical Anonymous Networking. In *Proc. of Workshop on Privacy Enhancing Technologies (PET 2003)*, Dresden, Germany, Mar. 2003.

[3] K. Bennett, C. Grothoff, T. Horozov, and I. Patrascu. Efficient Sharing of Encrypted Data. In *Proc.*

*of ACISP 2002*, pages 107–120. Springer-Verlag, July 2002.

[4] O. Berthold, H. Federrath, and S. Köpsell. Web MIXes: A System for Anonymous and Unobservable Internet Access. In H. Federrath, editor, *Proc. of Designing Privacy Enhancing Technologies: Workshop on Design Issues in Anonymity and Unobservability (PET)*, pages 115–129. Springer-Verlag, LNCS 2009, July 2000.

[5] N. Borisov and J. Waddle. Anonymity in Structured Peer-to-Peer Networks. gnunet.org/papers/borisov_waddle.pdf, Dec. 2003.

[6] M. Castro, P. Druschel, A. M. Kermarrec, and A. Rowstron. Scribe: A Large-scale and Decentralized Application-level Multicast Infrastructure. *IEEE Journal on Selected Areas in Communications, special issue on Network Support for Multicast Communications*, 20(8), Oct. 2002.

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

[8] D. Chaum. The Dining Cryptographers Problem: Unconditional Sender and Recipient Untraceability. *Journal of Cryptology*, 1(1):65–75, 1988.

[9] G. Ciaccio. The NEBLO homepage, http://www.disi.unige.it/project/neblo/.

[10] I. Clarke, O. Sandberg, B. Wiley, and T. W. Hong. Freenet: A Distributed Anonymous Information Storage and Retrieval System. In *Proc. of Designing Privacy Enhancing Technologies: Workshop on Design Issues in Anonymity and Unobservability (PET)*, pages 46–66, July 2000.

[11] R. Cox, A. Muthitacharoen, and R. Morris. Serving DNS using a Peer-to-Peer Lookup Service. In *Proc. of the 1st International Peer To Peer Systems Workshop (IPTPS02)*, Mar. 2002.

[12] F. Dabek, M. F. Kaashoek, D. Karger, R. Morris, and I. Stoica. Wide-area Cooperative Storage with CFS. In *Proc. of 18th ACM Symp. on Operating Systems Principles*, Oct. 2001.

[13] F. Dabek, J. Li, E. Sit, J. Robertson, M. F. Kaashoek, and R. Morris. Designing a DHT for low latency and high throughput. In *Proc. of the 1st USENIX Symposium on Networked Systems Design and Implementation (NSDI '04)*, San Francisco, CA, Mar. 2004.

[14] C. Díaz, S. Seys, J. Claessens, and B. Preneel. Towards measuring anonymity. In R. Dingledine and P. Syverson, editors, *Proc. of Privacy Enhancing Technologies Workshop (PET 2002)*. Springer-Verlag, LNCS 2482, Apr. 2002.

[15] R. Dingledine, M. J. Freedman, and D. Molnar. The Free Haven Project: Distributed Anonymous Storage Service. In H. Federrath, editor, *Proc. of Designing Privacy Enhancing Technologies: Workshop on Design Issues in Anonymity and Unobservability (PET)*. Springer-Verlag, LNCS 2009, July 2000.

[16] J. Eriksson, M. Faloutsos, and S. Krishnamurthy. PeerNet: Pushing Peer-to-Peer Down the Stack. In *Proc. of the 2nd International Workshop on Peer-to-Peer Systems (IPTPS03)*, Berkeley, CA, 2003.

[17] J. K. et al. Oceanstore: An Architecture for Global-scale Persistent Storage. In *Proc. of ACM ASPLOS*, Nov. 2000.

[18] M. J. Freedman and R. Morris. Tarzan: A peer-to-peer anonymizing network layer. In *Proc. of the 9th ACM Conference on Computer and Communications Security (CCS 2002)*, Washington, DC, Nov. 2002.

[19] I. Goldberg. *A Pseudonymous Communications Infrastructure for the Internet*. PhD thesis, UC Berkeley, Dec. 2000.

[20] A. Gupta, B. Liskov, and R. Rodrigues. One Hop Lookups for Peer-to-peer Overlays. In *Proc. of the 9th Workshop on Hot Topics in Operating Systems (HotOS-IX)*, Lihue, Hawaii, May 2003.

[21] I. Gupta, K. Birman, P. Linga, A. Demers, and R. van Renesse. Kelips: Building an Efficient and Stable P2P DHT Through Increased Memory and Background Overhead. In *Proc. of the 2nd International Workshop on Peer-to-Peer Systems (IPTPS03)*, Berkeley, CA, 2003.

[22] N. Harvey, M. B. Jones, S. Saroiu, M. Theimer, and A. Wolman. Skipnet: A Scalable Overlay Network with Practical Locality Properties. In *Proc. of the 4th USENIX Symposium on Internet Technologies and Systems (USITS '03)*, Mar. 2003.

[23] S. Hazel and B. Wiley. Achord: A Variant of the Chord Lookup Service for Use in Censorship Resistant Peer-to-Peer Publishing Systems. In *Proc. of the 1st International Peer To Peer Systems Workshop (IPTPS02)*, Mar. 2002.

[24] J. Kannan and M. Bansal. Anonymity in Chord. www.cs.berkeley.edu/ kjk/chord-anon.ps, Dec. 2002.

[25] B. Leong and J. Li. Achieving One-Hop DHT Lookup and Strong Stabilization by Passing Tokens. In *Proc. of the 12th International Conference on Networks (ICON)*, Nov. 2004.

[26] G. S. Manku, M. Bawa, and P. Raghavan. Symphony: Distributed Hashing in a Small World. In *Proc. of the fourth USENIX Symposium on Internet Technologies and Systems (USITS'03*, Seattle, WA, Mar. 2003.

[27] P. Maymounkov and D. Mazières. Kademlia: A Peer-to-peer Information System Based on the XOR Metric. In *Proc. of the 1st International Peer To Peer Systems Workshop (IPTPS02)*, Mar. 2002.

[28] A. Mislove, G. Oberoi, A. Post, C. Reis, P. Druschel, and D. S. Wallach. AP3: Cooperative, Decentralized Anonymous Communication. In *Proc. of 11th ACM SIGOPS European Workshop*, Leuven, Belgium, Sept. 2004.

[29] C. O'Donnell and V. Vaikuntanathan. Information Leak in the Chord Lookup Protocol. In *Proc. of the 4th IEEE Int.l Conf. on Peer-to-Peer Computing (P2P2004)*, Zurich, Switzerland, Aug. 2004.

[30] P. Perlegos. DoS Defense in Structured Peer-to-Peer Networks. Technical Report UCB-CSD-04-1309, U.C. Berkeley, Mar. 2004.

[31] W. Pugh. Skip Lists: a Probabilistic Alternative to Balanced Trees. *Comm. of ACM*, 33(6):668–676, 1990.

[32] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A Scalable Content-Addressable Network. In *Proc. of ACM SIGCOMM*, Aug. 2001.

[33] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker. Application-level Multicast Using Content-addressable Networks. In *Proc. of 3rd Int.l Workshop on Networked Group Communication*, Nov. 2001.

[34] M. Reed, P. Syverson, and D. Goldschlag. Anonymous Connections and Onion Routing. *IEEE Journal on Selected Areas in Communications, special issue on Copyright and Privacy Protection*, 1998.

[35] J. Rohrer. MUTE: Simple, Anonymous File Sharing. http://mute-net.sourceforge.net/.

[36] A. Rowstron and P. Druschel. Pastry: Scalable, Distributed Object Location and Routing for Large-scale Peer-to-peer Systems. In *Proc. of Int.l Conf. on Distributed System Platforms*, Nov. 2001.

[37] A. Rowstron and P. Druschel. Storage Management and Caching in PAST, a Large-scale, Persistent Peer-to-peer Storage Utility. In *Proc. of 18th ACM Symp. on Operating Systems Principles*, Oct. 2001.

[38] V. Scarlata, B. N. Levine, and C. Shields. Responder Anonymity and Anonymous Peer-to-Peer File Sharing. In *Proc. of IEEE International Conference on Network Protocols (ICNP)*, Nov. 2001.

[39] A. Serjantov. Anonymizing Censorship Resistant Systems. In *Proc. of the 1st International Peer To Peer Systems Workshop (IPTPS02)*, Mar. 2002.

[40] A. Serjantov and G. Danezis. Towards an information theoretic metric for anonymity. In R. Dingledine and P. Syverson, editors, *Proc. of Privacy Enhancing Technologies Workshop (PET 2002)*. Springer-Verlag, LNCS 2482, Apr. 2002.

[41] R. Sherwood, B. Bhattacharjee, and A. Srinivasan. P5: A protocol for scalable anonymous communication. In *Proc. of the 2002 IEEE Symposium on Security and Privacy*, May 2002.

[42] A. Singh and L. Liu. Agyaat: Providing Mutually Anonymous Services over Structured P2P Networks. Technical Report GIT-CERCS-04-12, Georgia Inst. of Tech. CERCS, 2004.

[43] I. Stoica, D. Adkins, S. Zhuang, S. Shenker, and S. Surana. Internet Indirection Infrastructure. In *Proc. of ACM SIGCOMM'02*, Aug. 2002.

[44] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: a Scalable Peer-to-peer Lookup Service for Internet Applications. In *Proc. of ACM SIGCOMM*, Aug. 2001.

[45] M. Waldman, A. Rubin, and L. Cranor. Publius: A Robust, Tamper-evident, Censorship-resistant and Source-anonymous Web Publishing System. In *Proc. of the 9th USENIX Security Symposium*, pages 59–72, Aug. 2000.

[46] J. Wang, L. Lu, and A. Chien. Tolerating Denial-of-Service Attacks Using Overlay Networks - Impact of Topology. In *Proc. of ACM Workshop on Survivable and Self-Regenerative Systems*, Oct. 2003.

[47] B. Y. Zhao, J. D. Kubiatowicz, and A. D. Joseph. Tapestry: An Infrastructure for Fault-resilient Wide-area Location and Routing. Technical Report UCB-CSD-01-1141, U.C. Berkeley, Apr. 2001.

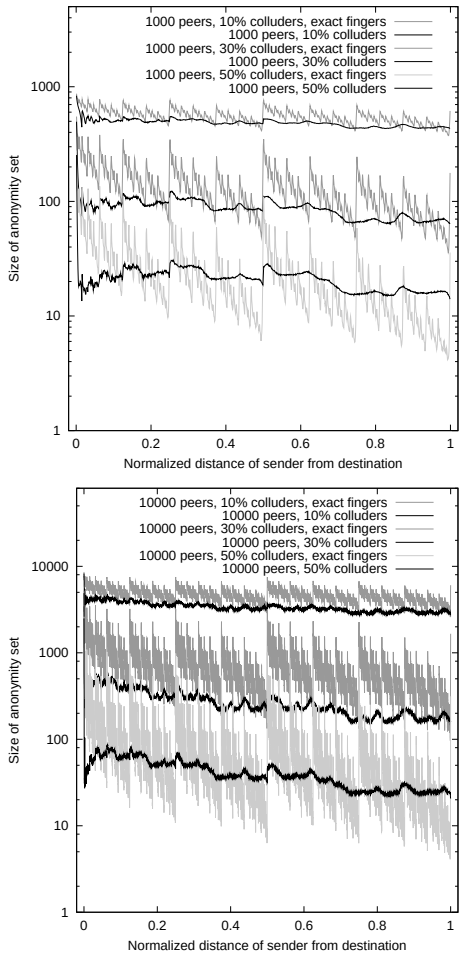


**Figure 8. Average sender anonymity in simulated chordal rings with 1000 and 10000 peers and three different percentages of attackers. Systems with imprecise fingers are compared to systems with exact fingers. The order of captions reflects the order of the curves from top to bottom. Imprecise fingers show a slightly lower sender anonymity compared to exact fingers, but the difference is not substantial.**

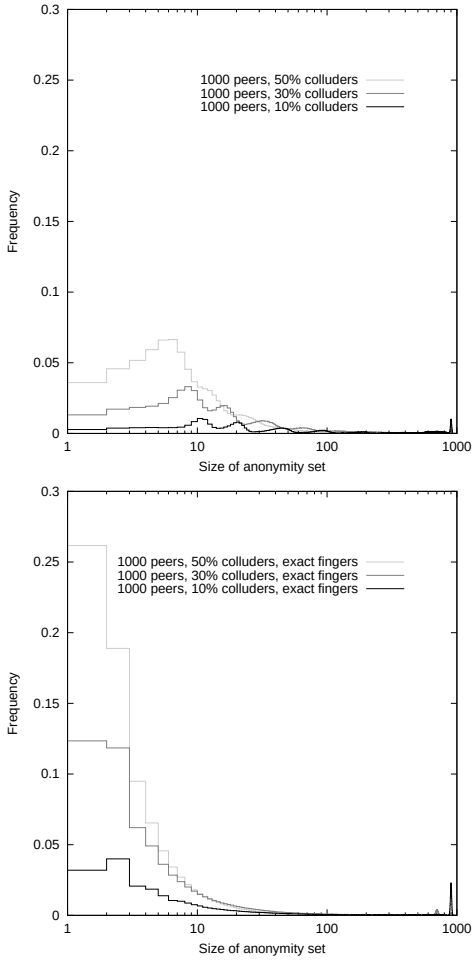


**Figure 9. Average frequency distribution of sender anonymity in simulated chordal rings with 1000 peers and both imprecise and traditional fingers, with three different percentages of attackers.**

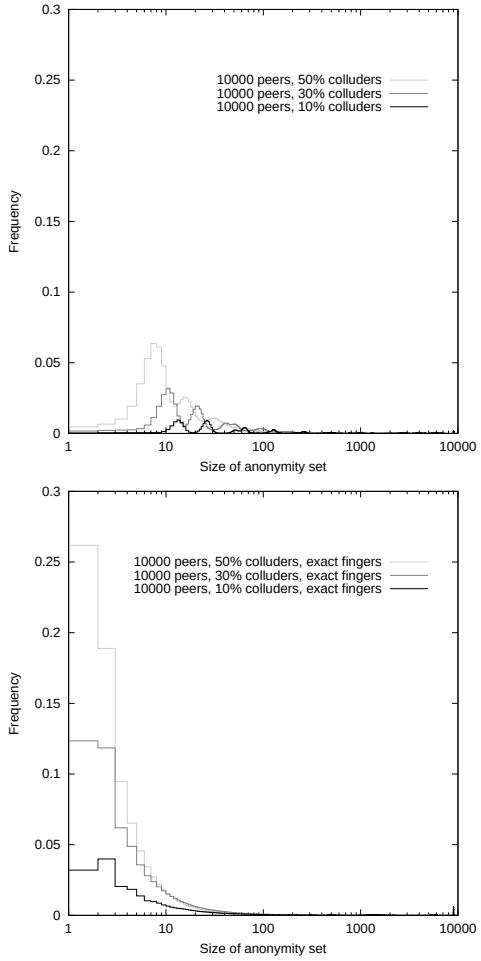


**Figure 10. Average frequency distribution of sender anonymity in simulated chordal rings with 10000 peers and both imprecise and traditional fingers, with three different percentages of attackers.**