A Survey of Anonymous Communication Channels

George Danezis* Claudia Diaz†

January 2008

Abstract

We present an overview of the field of anonymous communications, from its establishment in 1981 from David Chaum to today. Key systems are presented categorized according to their underlying principles: semi-trusted relays, mix systems, remailers, onion routing, and systems to provide robust mixing. We include extended discussions of the threat models and usage models that different schemes provide, and the trade-offs between the security properties offered and the communication characteristics different systems support.

Contents

*Microsoft Research, Cambridge, UK.
†COSIC, ESAT, K.U.Leuven, Belgium.

1 Introducing Anonymous Communications

Research on anonymous communications started in 1981 with Chaum's seminal paper "Untraceable electronic mail, return addresses, and digital pseudonyms" (Chaum 1981). Since then, a body of research has concentrated on building, analyzing and attacking anonymous communication systems. In this survey we look at the definition of anonymous communications and the major anonymous communication systems grouped in families according to the key design decisions they are based on.

Data communication networks use addresses to perform routing which are, as a rule, visible to anyone observing the network. Often addresses (such as IP addresses, or Ethernet MACs) are a unique identifier which appear in all communication of a user, linking of all the user's transactions. Furthermore these persistent addresses can be linked to physical persons, seriously compromising their privacy.

Anonymizing the communication layer is thus a necessary measure to protect the privacy of users, and protect computer systems against traffic analysis. They also support anonymization techniques at the application layer, such as anonymous credentials, elections and anonymous cash.

In the remaining of this introductory section, we introduce the terminology of anonymity properties, we present the existing models for quantifying the anonymity, and we explain some limits on anonymity imposed by black box attacks. Section 2 presents anonymity systems based on (centralized) trusted and semi-trusted relays, and introduces the link between anonymous communications and censorship resistance. Mix-based anonymous communication systems are extensively described in Section 3. Section 4 introduces other proposals for anonymous communication, and Section 5 presents the conclusions of this survey.

1.1 Terminology

Prior to the quantification of anonymity, a set of working definitions for *anonymity* and other related concepts, such as *unlinkability* or *unobservability* were needed.

Pfitzmann & Hansen (2001) proposed a set of working definitions for anonymity, unlinkability, unobservability and pseudonymity. These definitions have since

been adopted in most of the anonymity literature. Their authors continue releasing regular updates on the document addressing feedback from the research community[1].

Anonymity. To enable the anonymity of a subject, there always has to be an appropriate set of subjects with potentially the same attributes. Anonymity is thus defined as *the state of being not identifiable within a set of subjects, the anonymity set.*

The *anonymity set* is the set of all possible subjects. With respect to acting entities, the anonymity set consists of the subjects who might cause an action. With respect to addressees, the anonymity set consists of the subjects who might be addressed. Both anonymity sets may be disjoint, be the same, or they may overlap. The anonymity sets may vary over time.

According to the Pfitzmann-Hansen definition of anonymity, the subjects who may be related to an anonymous transaction constitute the *anonymity set* for that particular transaction. A subject carries on the transaction *anonymously* if he cannot be distinguished (by an adversary) from other subjects. This definition of anonymity captures the probabilistic information often obtained by adversaries trying to identify anonymous subjects.

Unlinkablity. The [ISO15408 1999] defines unlinkability as follows:

"[Unlinkability] ensures that a user may make multiple uses of resources or services without others being able to link these uses together. [...] Unlinkability requires that users and/or subjects are unable to determine whether the same user caused certain specific operations in the system."

We may differentiate between "absolute unlinkability" (as in the given ISO definition above; i.e., "no determination of a link between uses") and "relative unlinkability" (i.e., "no change of knowledge about a link between uses"), where "relative unlinkability" could be defined as follows:

Unlinkability of two or more Items Of Interest (IOIs; e.g., subjects, messages, events, actions, ...) means that within the system (comprising these and possibly other items), from the attackers perspective, these items of interest are no more and no less related after his observation than they were related concerning his a-priori knowledge.

This means that the probability of those items being related from the attackers perspective stays the same before (a-priori knowledge) and after the attackers observation (a-posteriori knowledge of the attacker). Roughly speaking, providing relative unlinkability of items means that the ability of the attacker to relate these items does not increase by observing the system.

Unobservability. In contrast to anonymity and unlinkability, where not the IOI, but only its relationship to IDs or other IOIs is protected, for unobservability, the IOIs are protected as such. *Unobservability is the state of items of interest (IOIs) being indistinguishable from any IOI (of the same type) at all.*

[1] `http://dud.inf.tu-dresden.de/Anon_Terminology.shtml`

This means that messages are not discernible from random noise. As we had anonymity sets of subjects with respect to anonymity, we have unobservability sets of subjects with respect to unobservability. Sender unobservability then means that it is not noticeable whether any sender within the unobservability set sends. Recipient unobservability then means that it is not noticeable whether any recipient within the unobservability set receives. Relationship unobservability then means that it is not noticeable whether anything is sent out of a set of could-be senders to a set of could-be recipients. In other words, it is not noticeable whether within the relationship unobservability set of all possible sender-recipient-pairs, a message is exchanged in any relationship.

Pseudonymity. Pseudonyms are identifiers of subjects. We can generalize pseudonyms to be identifiers of sets of subjects. The subject which the pseudonym refers to is the holder of the pseudonym.

Being pseudonymous is the state of using a pseudonym as ID.

We assume that each pseudonym refers to exactly one holder, invariant over time, being not transferred to other subjects. Specific kinds of pseudonyms may extend this setting: a group pseudonym refers to a set of holders; a transferable pseudonym can be transferred from one holder to another subject becoming its holder. Such a group pseudonym may induce an anonymity set: Using the information provided by the pseudonym only, an attacker cannot decide whether an action was performed by a specific person within the set.

Defining the process of preparing for the use of pseudonyms, e.g., by establishing certain rules how to identify holders of pseudonyms, leads to the more general notion of pseudonymity:

Pseudonymity is the use of pseudonyms as IDs.

An advantage of pseudonymity technologies is that accountability for misbehavior can be enforced. Also, persistent pseudonyms allow their owners to build a pseudonymous reputation over time.

1.2 Anonymity Metrics

Most attacks on anonymous communication networks provide the adversary with probabilistic information on the identity of the entities communicating with each other. This is the reason why information theoretical anonymity metrics (Serjantov & Danezis 2002, Diaz, Seys, Claessens & Preneel 2002) have been widely adopted to quantify the anonymity provided by a variety of designs.

But before information theoretic anonymity metrics were proposed, there had been some attempts to quantify anonymity in communication networks.

Reiter & Rubin (1998) define the *degree of anonymity* as a probability $1-p$, where p is the probability assigned by an attacker to potential senders. In this model, users are more anonymous as they appear (towards a certain adversary) to be less likely of having sent a message. This metric considers users separately, and therefore does not capture anonymity properties very well. Consider a first system with 2 users. The first user u_1 appears to be the sender of a message with probability 1/10, and the second user u_2 with probability 9/10. Now

consider a second system with 1000 users. User u_1 appears as the sender with probability 1/10, while all the other users are assigned probabilities of having sent the message below 0.001. According to the definition of Reiter and Rubin, the *degree of anonymity* of u_1 would be the same in both systems ($d = 0.9$). However, in the second system, u_1 looks much more likely to be the sender than any other user, while in the first he is the less likely candidate of being so.

Berthold, Pfitzmann & Standtke (2000) define the *degree of anonymity* as $A = log_2(N)$, where N is the number of users of the system. This metric only depends on the number of users of the system, and therefore does not express the anonymity properties of different systems. The total number N of users may not even be known. Moreover, adversaries may be able to obtain probabilistic information on the set of potential senders, which is not taken into account in this metric.

Information theoretic anonymity metrics were independently proposed in two papers presented at the *2nd Workshop on Privacy Enhancing Technologies*. The basic principle of both metrics is the same. The metric proposed by citeSerj02 uses entropy as measure of the *effective anonymity set size*. The metric proposed by Diaz et al. (2002) goes one step further, normalizing the entropy to obtain a *degree of anonymity* in the scale 0..1.

The quantification of anonymity is dependent on the adversary considered. The adversary has certain capabilities and deploys attacks in order to gain information and find links between subjects and items of interest. Most of these attacks lead to a distribution of probabilities that assign subjects a certain probability of being linked to the items of interest. In this respect, a clear and detailed formulation of the attack model considered is a required step to measure the anonymity provided towards that attacker.

The information theoretic concept of entropy (Shannon 1948) provides a measure of the uncertainty of a random variable. Let X be the discrete random variable with probability mass function $p_i = Pr(X = i)$, where i represents each possible value that X may take with probability $p_i > 0$. In this case, each i corresponds to a subject of the anonymity set; i.e., p_i is the probability of subject i being linked to the item of interest.

The entropy describes thus the information (measured in bits) contained in the probability distribution that describes the links between a set of subjects (the anonymity set) and an item of interest. In Serjantov & Danezis (2002), the entropy is proposed as a measure of the effective anonymity set size. If the entropy is normalized by the maximum the system could provide (if it was perfect and leaked no information) for a given number of users, we obtain a degree of anonymity (Diaz et al. 2002) that gives a measure of the anonymity provider's performance.

1.3 Limits and black box attacks

No matter how good the anonymity performed by the network, persistent communication between two users will eventually be detected just by observing the edges of the network and correlating the activity at the two ends of the com-

munication. Such *intersection attacks*, are presented in Berthold, Pfitzmann & Standtke (2000), who try to extract the sender of a stream of messages by intersecting the sender anonymity sets of consecutive messages sent by a user. This attack model is also considered in (Kesdogan, Agrawal & Penz 2002, Agrawal, Kesdogan & Penz 2003) and (Wright, Adler, Levine & Shields 2003). The statistical variant of this attack is the statistical disclosure attack presented in (Danezis 2003*b*, Danezis & Serjantov 2004, Mathewson & Dingledine 2004).

In Wright, Adler, Levine & Shields (2002) the authors present a set of attacks that can be performed by a group of subverted network nodes. Against mix networks, they calculate the number of routes to be chosen between a sender and a receiver before the full path has been entirely populated by subverted nodes. They also examine the effect that fixed or variable length paths have on this probability. Similar results are found for Crowds and DC-nets. In Wright et al. (2003) they extend their analysis to considering a subset of network nodes that simply log all traffic, and provide bounds on how quickly an intersection attack can be performed. Despite these studies being set in the frame of particular systems, like DC-nets and Crowds, they in fact explore fundamental limitations for any systems that select trusted parties at random from a larger set of potentially corrupt nodes to provide security.

2 Trusted and semi-trusted relays

We start presenting anonymous communications by introducing systems that rely on one central trusted node to provide security. We will see that they provide a varying, but usually low, degree of anonymity protection against traffic analysis and active attacks.

2.1 The Anon.penet.fi relay

Johan Helsingius started running a trusted mail relay, `anon.penet.fi`, providing anonymous and pseudonymous email accounts in 1993. The technical principle behind the service was a table of correspondences between real email addresses and pseudonymous addresses, kept by the server. Email to a pseudonym would be forwarded to the real user. Email from a pseudonym was stripped of all identifying information and forwarded to the recipient. While users receiving or sending email to a pseudonym would not be able to find out the real email address of their anonymous correspondent, it would be trivial for a local passive attacker or the service itself to uncover the correspondence by correlating the timing of incoming and outgoing email traffic.

While protecting from a very weak threat model, the service was finally forced to close down through legal attacks. In 1996 the "Church of Spiritual Technology, Religious Technology Center and New Era Publications International Spa" reported that a user of `anon.penet.fi` sent a message to a newsgroup infringing their copyright. Johan Helsingius, the administrator of `anon.penet.fi`, was asked to reveal the identity of the user concerned. The

details of the case, that put enormous strain on the service, can be found in the press releases of the 23 September 1996 (Helsingius 1996*b*, Helsingius 1996*c*) or the information center set up around this case (Newman 1997). Reputation attacks were also experienced, when unfounded reports appeared in mainstream newspapers about the service being used to disseminate child pornography (Helsingius 1996*a*).

The service finally closed in August 1996 since it could no longer guarantee the anonymity of its users. The closure was quite significant for the privacy and anonymity research community. In the initial judgment the judge had ruled that "a witness cannot refrain from revealing his information in a trial" (Helsingius 1996*c*), even though an appeal was lodged on the grounds of privacy rights protected by the Finnish constitution, and the fact that the information might be privileged, as is the case for journalistic material.

The concept that even non-malicious relay operators could be forced under legal or other compulsion, to reveal any information they have access to, provided a new twist to the conventional threat models. Honest relays or trusted nodes could under some circumstances be forced to reveal any information they held concerning the origin or destination of a particular communication. Minimizing the information held by trusted parties is therefore not just protecting their users, but also the services themselves.

2.2 Anonymizer & SafeWeb

Anonymizer[2] is a company set up by Lance Cottrell, also author of the Mixmaster remailer software, that provides anonymous web browsing for subscribed users. The Anonymizer product acts as a web proxy through which all web requests and replies are relayed. The web servers accessed, should therefore not be able to extract any information about the address of the requesting user. Special care is taken to filter out any "active" content, such as javascript or Java applets, that could execute code on the user's machine, and then signal back identifying information.

As for `anon.penet.fi`, the anonymity provided depends critically on the integrity of the Anonymizer company and its staff. The service is less vulnerable to legal compulsion attacks, since no long-term logs are required to be kept, that could link users with resources accessed. Unlike email, users always initiate web requests, and receive the replies, and all records can be deleted after the request and reply have been processed. Records can be made unavailable to seize just a few seconds after the provision of the anonymous service to the user.

SafeWeb was a company that provided a very similar service to Anonymizer. The two main differences in their initial products, was that SafeWeb allowed the traffic on the link from SafeWeb to the user to be encrypted using SSL (Dierks & Allen 1999), and "made safe" active content in pages by using special wrapper functions. Unfortunately their system of wrappers did not resist a set of attacks devised by Martin & Schulman (2002). Simple javascript attacks turned out to

[2] `http://www.anonymizer.com/`

be able to extract identifying information from the users.

In the absence of any padding or mixing, a passive attacker observing the service would also be able to trivially link users with pages accessed, despite the use of SSL. This vulnerability was studied in (danezis n.d., Bissias, Liberatore, & Levine 2005, Sun, Simon, Wang, Russell, Padmanabhan & Qiu 2002, Cheng & Avnur n.d., Hintz 2002). This line of research established that an adversary is capable of compiling a library of 'traffic signatures' for all interesting web-pages that may be accessed. The signatures can then be compared with the traffic characteristics of the encrypted SSL connection to infer the page that was accessed.

The key weaknesses come down to the shape of traffic, which is inadequately padded and concealed. Browsers request resources, often HTML pages, that are also associated with additional resources (images, style sheets, ...). These are downloaded through an encrypted link, yet their size is apparent to an observer, and can be used to infer which pages are accessed. There are many variants of this attack: some attempt to build a profile of the web-site pages and guess for that which pages are being accessed while others use these techniques to beat naive anonymizing SSL proxies. In the latter case, the attacker has access to the cleartext input streams and he tries to match them to encrypted connections made to the proxy.

Note that latent structure and contextual knowledge are again of great use to extract information from traffic analysis: in danezis (n.d.), it is assumed that users will mostly follow links between different web resources. A Hidden Markov Model is then used to trace the most likely browsing paths a user may have taken given only the lengths of the resources that can be observed. This provides much faster and more reliable results than considering users that browse at random, or web-sites that have no structure at all.

2.2.1 Censorship resistance

The threat model that SafeWeb wished to protect against was also very peculiar. The company was partly funded by the United States Central Intelligence Agency (CIA), and attempted to secure funding from The Voice of America and the Internet Broadcasting Bureau in order to "help Chinese and other foreign Web users get information banned in their own company *(sic)*"[3] (Singer 2001). This claim explicitly links anonymous communications with the ability to provide censorship resistance properties. The link has since then become popular, and often anonymity properties are seen as a pre-requisite for allowing censorship resistant publishing and access to resources. No meticulous requirements engineering study has even been performed that proves (or disproves) that claim, and no cost benefit analysis has ever been performed to judge if the technical cost of an anonymity system would justify the benefits in the eyes of those interested in protecting themselves against censorship. Furthermore no details

[3]The Freudian slip confusing "country" with "company", and the way this goal can be understood in two opposing ways might be interpreted as quite telling of the nature of the CIA's interest in this product.

were ever provided, besides hearsay claims, about groups using this particular technology in a hostile environment, and their experiences with it. The latter would be particularly interesting given the known vulnerabilities of the product at the time.

The first paper to make a clear connection between censorship resistant storage and distribution is Anderson's Eternity service (Anderson 1996). Serjantov has also done some interesting work on how to use strong anonymous communications to provide censorship resistance (Serjantov 2002). The system presented is, for good technical and security reasons, very expensive in terms of communication costs and latency. Peer-to-peer storage and retrieval systems such as Freenet (Clarke, Sandberg, Wiley & Hong 2000), FreeHaven (Dingledine, Freedman & Molnar 2000) and, more recently, GnuNet (Bennett & Grothoff 2003) also claimed to provide anonymous communications. Attacks against some anonymity properties provided by GnuNet have been found (Kügler 2003). Since the design of the three mentioned systems changes frequently it is very difficult to assess the security, or the anonymity they provide at any time. Feamster, Balazinska, Harfst, Balakrishnan & Karger (2002) and Feamster, Balazinska, Wang, Balakrishnan & Karger (2003) have looked at different aspects of web censorship resistance by making use of steganography to send high volumes of data, and what they called "URL hopping" to transmit requests. Finally, aChord (Hazel & Wiley 2002) presents concisely the requirements of a censorship resistant system, and attempts to build one based on a distributed hash table primitive.

Aside from complete systems, many isolated mechanisms have been proposed to bypass restrictive firewalls that attempt to prevent access to an anonymous communication system. The Tor (Dingledine, Mathewson & Syverson 2004*a*) system, provides a mode that tunnels everything over TCP Port 80 which is often not filtered since it is usually reserved for HTTP (Web) traffic. Anonymizer relies on providing people behind national firewalls (notoriously in China and Iran) with network addresses unknown to the firewall that are not being filtered (Leyden 2006). This results in an arms race between the providers of fresh addresses, that extensively rely on spam for distribution, and the authorities that seek to detect them and block them. A similar architecture (Köpsell & Hilling 2004), that relies on volunteers donating their non-blocked network address to help those behind filtering firewalls, has been described and implemented by the JAP[4] project.

Other studies have looked at censorship and Internet filtering in China (Walton 2001), and the specific capabilities of the national firewall (Clayton, Murdoch & Watson 2006). It was discovered that it simply sends TCP resets to force communicating parties, with compliant TCP/IP stacks, to drop their connection. Modified clients that ignore such resets were able to carry on communicating. Finally, two studies, one German (Dornseif 2003) and one British (Clayton n.d.), have looked at the effectiveness of Internet Service Providers filtering out web sites that are known to contain child pornography. In their

[4] `http://anon.inf.tu-dresden.de/`

study of the live BT content blocking system *cleanfeed* (Clayton n.d.) they discovered that forbidden content could be trivially accessed. Furthermore, the blocking mechanism had the unwanted feature that it could be used as an oracle for interested parties to discover sites with illegal material.

2.3 Type I "Cypherpunk" remailers

Type I remailers, first developed by Eric Hughes and Hal Finney (Parekh 1996), are nodes that relay electronic mail, after stripping all identifying information and decrypting it with their private key. The first code-base was posted to the cypherpunks mailing list, which gave the remailers their nickname. The encryption was performed using the Pretty Good Privacy (PGP) public key encryption functions. The encoding scheme was also designed to be performed manually, using standard text and email editing tools. Many remailers could be chained together, in order for users not to rely on a single remailer to protect their anonymity.

Reply blocks were supported to allow for anonymous reply addresses. The email address of the user would be encrypted using the remailer's public key, and inserted in a special header. If a user wished to reply to an anonymous email, the remailer would decrypt it and forward the contents.

The type I remailers offer better resistance to attacks than the simple `anon.penet.fi` relay. No database that links real user identities with pseudonyms is kept. The addressing information required to reply to messages is included in the messages themselves, in an encrypted form.

The encryption used when the messages are forwarded through the network prevents the most trivial passive attacks based on observing the exact bit patterns of incoming messages and linking them to outgoing ones. However it leaks information, such as the size of the messages. Since PGP, beyond compressing the messages, does not make any further attempts to hide their size, it is trivial to follow a message in the network just by observing its length. The reply blocks provided are also a source of insecurity. They can be used many times and an attacker could encode an arbitrary number of messages in order to mount an attack to find their destination. Since all relies encoded with the same reply block would contain an identical sequence of bytes, this attack is even easier than the statistical disclosure attacks (Danezis 2003*b*, Danezis & Serjantov 2004). The attack can then be repeated to trace any number of hops.

Despite these drawbacks, type I remailers became very popular. This is due to the fact that their reply capabilities allowed the use of Nym Servers. Their reply block feature, that is not present in the later type II Mixmaster software, is both essential to build nym servers, but also insecure even against passive adversaries. This has prompted the design of Mixminion, a type III remailer, that is extremely resistant to traffic analysis, and provides secure single-use reply blocks.

2.4 Crowds

Crowds was developed by Reiter & Rubin (1998) at the AT&T Laboratories. It aims to provide a privacy preserving way of accessing the web, without web sites being able to recognize who is browsing. Each user contacts a central server and receives the list of participants, the "crowd". A user then relays her web requests by passing it to another randomly selected node in the crowd. Upon receiving a request each node tosses a biased coin and decides if it should relay it further through the crowd or send it to the final recipient. Finally, the reply is sent back to the user via the route established as the request was being forwarded through the crowd.

Crowds is a landmark in anonymity research since its security relies on the adversary not being able to observe the links. Instead, the adversary is assumed to only control a fraction of nodes in each crowd, and the ultimate web server. Although this threat model was initially believed to be unrealistic and fragile, it was later realized that it can be achieved using simple link encryption and padding.

A system which also relies for its security on the inability of the adversary to intercept all communication channels was presented by Katti, Katabi & Puchala (2005). They conceptually 'slice' each message into many parts, using a secret sharing scheme, and send them out in the network using different channels to an intermediary: if the adversary fails to observe one of the channels they cannot reconstruct the message or the final address to which it is destined. The scheme can be simplified, by considering the secret sharing scheme over a partially secure set of channels, as a primitive encryption mechanism, and the intermediary as a trusted relay that is to decrypt the message and forward it.

Crowds is one of the first papers to address quantitatively how colluding nodes would affect the anonymity provided by the system. It is clear that after the first dishonest node in the path of a request no further anonymity is provided, since the clear text of all requests and replies is available to intermediate nodes. Therefore, given a certain fraction of colluding attacker nodes it is possible to measure the anonymity that will be provided (Diaz et al. 2002).

Crowds also introduces the concept of *initiator anonymity*: a node that receives a request cannot know if the previous node was the actual requester or was just passing the request along. This property is quite weak and two independent groups have found attacks that identify the originator of requests (Wright et al. 2002, Shmatikov 2002). They discovered that if a client repeatedly requests a particular resource, they can eventually be linked: The attack relies on the intuition that the true initiator of the repeated request will be the predecessor of a corrupt node more often than a random node in the crowd. Therefore, for each resource accessed it is sufficient to count how many times each node is seen to be accessing it, and select the node corresponding to the most requests as the most likely initiator. This attack sensitized the anonymity community to the problem of protecting *persistent* relationships instead of simple single message or request exchanges.

Despite the difficulty of securing initiator anonymity, a lot of subsequent

systems such as achord (Hazel & Wiley 2002) and MorphMix (Rennhard & Plattner 2002), try to achieve it.

2.5 Nym servers

Nym servers (Mazières & Kaashoek 1998) store an anonymous reply block, and map it to a pseudonymous email address. When a message is received for this address it is not stored, but immediately forwarded anonymously using the reply block to the owner of the pseudonym. In other words, Nym Servers act as a gateway between the world of conventional email and the world of anonymous remailers. Since they hold no identifying information, and are simply using anonymous reply blocks for routing purposes, they do not require users to trust them in order to safeguard their anonymity. Over the years, special software has been developed to support complex operations such as encoding anonymous mail to go through many remailers, and managing Nym Server accounts. Mathewson (2005) presents a contemporary design for a Nym server called *Underhill* that uses the state of the art in remailer technology, and *NymBaron* is its current implementation (Fousse & Reinhard 2006).

Nym servers are also associated with pseudonymous communications. Since the pseudonymous identity of a user is relatively persistent it is possible to implement reputation systems, or other abuse prevention measures. For example, a nym user might at first only be allowed to send out a small quantity of email messages, that increases over time, as long as abuse reports are not received by the nym server operator. Nym servers and pseudonymous communications offer some hope of combining anonymity and accountability.

At the same time, it is questionable how long the true identity of a pseudonymous user can be hidden. If all messages sent by a user are linked between them by the same pseudonym, one can try to apply author identification techniques to uncover the real identity of the user. Rao & Rohatgi (2000) in their paper entitled "Can Pseudonymity Really Guarantee Privacy?" show that the frequency of function words[5] in the English language can be used in the long term to identify users. A similar analysis could be performed using the sets of correspondents of each nym, to extract information about the user. Mathewson & Dingledine (2004) have noted that statistical disclosure attacks are very effective at linking pseudonyms with their corresponding users, when those are based on remailer systems.

The shortcomings of remailer based systems have prompted a line of research that looks at alternative techniques to provide receiver anonymity. Techniques from Private Information Retrieval (PIR) have been suggested. PIR is concerned with a family of techniques that allow clients to query a database, without the database or any third party being able to infer which record was retrieved. PIR in the context of receiver anonymity can be implemented either using secure hardware (Asonov & Freytag n.d., Kesdogan, Borning & Schmeink 2002), or

[5] Function words are specific English words used to convey ideas, yet their usage is believed to be independent of the ideas being conveyed. For example: *a, enough, how, if, our, the, ...*

distributed servers as in Sassaman, Cohen & Mathewson (2005) and Kissner, Oprea, Reiter, Song & Yang (2004). Private Search, a simplified PIR construction (Ostrovsky & III 2005) was made more efficient by Bethencourt, Song & Waters (2006) and Danezis & Diaz (2006), and has the potential to be used in efficient receiver anonymity systems.

Interestingly, Ishai, Kushilevitz, Ostrovsky & Sahai (2006) also show that given a strong anonymous channel, one can construct an efficient Private Information Retrieval system.

3 Mix systems

The type I remailer, presented in section 2.3, is the insecure version of a whole body of research that we shall call mix systems and mix networks. This section presents secure constructions based on these ideas.

3.1 Chaum's original mix

The first, and most influential, paper in the field of anonymous communications was presented in (Chaum 1981). Chaum introduced the concept of a "mix" node that hides the correspondences between its input messages and its output messages in a cryptographically strong way.

The work was done in the late seventies, when RSA public key encryption was relatively new. For this reason the paper might surprise today's reader by its use of raw RSA, the direct application of modular exponentiation for encryption and decryption, along with an ad-hoc randomisation scheme. Nonces are appended to the plaintext before encryption in order to make two different encryptions output different ciphertext.

The principal idea is that messages to be anonymized are relayed through a node, called a mix. The mix has a well-known RSA public key, and messages are divided into blocks and encrypted using this key. The first few blocks are conceptually the "header" of the message, and contain the address of the next mix. Upon receiving a message, a mix decrypts all the blocks, strips out the first block that contains the address of the recipient, and appends a block of random bits (the junk) at the end of the message. The length of the junk is chosen to make messages size invariant. The most important property that the decryption and the padding aim to achieve is *bitwise unlinkability*. An observer, or an active attacker, should not be able to find the link between the bit pattern of the encoded messages arriving at the mix and the decoded messages departing from the mix. The usage of the word encoded and decoded instead of encrypted and decrypted serves to highlight that the former operations are only used to achieve unlinkability, and not confidentiality, as may be understood to be the aim of encryption. Indeed, modifying RSA or any other encryption and decryption functions to provide unlinkability against passive or active attackers is a problem studied in depth in the context of the design of Mixminion.

Pfitzmann & Pfitzmann (1990) show that Chaum's scheme does not provide the necessary unlinkability properties. The RSA mathematical structure can be subject to active attacks that leak enough information during decryption to link ciphertexts with their respective plaintexts. Further *tagging attacks* are possible, since the encrypted blocks, using RSA are not in any way dependent on each other, and blocks can be duplicated or simply substituted by known ciphertexts. The output message would then contain two blocks with the same plaintext or a block with a known plaintext, respectively. Once again, the use of RSA in the context of a hybrid cryptosystem, in which only the keys are encrypted using the public key operations, and the body of the message using a symmetric cipher were not very well studied at the time.

A further weakness of Chaum's scheme is its direct application of RSA decryption, which is also used as a signature primitive. An active attacker could substitute a block to be signed in the message and obtain a signature on it. Even if the signature has to have a special form, such as padding, that could be detected, a blinding technique could be used to hide this structure from the mix. It would be unfair to blame this shortcoming on Chaum, since he himself invented RSA blind signatures only a few years later (Chaum 1983).

The second function of a mix is to actually mix together many messages, to make it difficult for an adversary to follow messages through it, on a first-in, first-out basis. Therefore a mix batches a certain number of messages together, decodes them as a batch, reorders them in lexicographic order and then sends them all out. Conceptually, while bitwise unlinkability makes sure that the contents of the messages do not allow them to be traced, mixing makes sure that the output order of the messages does not leak any linking information.

In order to make the task of the attacker even more difficult, dummy messages are proposed. Dummy messages are generated either by the original senders of messages or by mixes themselves. As far as the attacker is concerned, they are indistinguishable in length and content to normal messages, which increases the difficulty in tracing the genuine messages. We will call the actual mixing strategy, namely the batching and the number of dummy messages included in the inputs or outputs, the *dynamic aspects* of mixing.

Chaum notes that relying on just one mix would not be resilient against subverted nodes, so the function of mixing should distributed. Many mixes can be chained to make sure that even if just one of them remains honest some anonymity would be provided. The first way proposed to chain mixes is the *cascade*. Each message goes through all the mixes in the network, in a specific order. The second way proposed to chain mixes is by arranging them in a fully connected *network*, and allowing users to pick arbitrary routes through the network. Berthold, Pfitzmann, and Standtke argue in Berthold, Pfitzmann & Standtke (2000) that mix networks do not offer some properties that cascades offer. They illustrate a number of attacks to show that if only one mix is honest in the network the anonymity of the messages going through it can be compromised. These attacks rely on compromised mixes that exploit the knowledge of their position in the chain; or multiple messages using the same sequence of mixes through the network.

Along with the ability for a sender to send messages anonymously to a receiver, Chaum presents a scheme by which one can receive messages anonymously. A user that wishes to receive anonymous email constructs an *anonymous return address*, using the same encoding as the header of the normal messages. She creates blocks containing a path back to herself, and recursively encrypts the blocks using the keys of the intermediate mixes. The user can then include a return address in the body of a message sent anonymously. The receiver simply includes the return address as the header of his own message and sends it through the network. The message is routed through the network as if it was a normal message.

The reply scheme proposed has an important feature. It makes replies in the network indistinguishable from normal messages. In order to securely achieve this, it is important that both the encoding and the decoding operation provide bitwise unlinkability between inputs and outputs. This is necessary, because replies are in fact encoded when processed by the mix. The resulting message, after it has been processed by all the mixes in the chain specified by the return address, is then decoded with the keys distributed to the mixes in the chain. Both the requirement for decryption to be as secure as encryption, and for the final mix to know the decryption keys to recover the message, means that raw RSA cannot be used. Therefore, a hybrid scheme is proposed that simply encrypts a symmetric key in the header along with the address of the next mix in the chain, that can be used to encrypt or decrypt the message. Since the keys are encoded in the return address by the user, they can be remembered by the creator of the reply block and used to decrypt the messages that are routed using them. Return addresses were also discussed in the Babel system (citebabel) and implemented in the cypherpunk type I remailers. Unfortunately, other deployed systems like Mixmaster did not support them at all.

Chaum's suggestion that a receipt system should be in place to make sure that each mix processes correctly messages, has become a branch of anonymity research in itself, namely mix systems with verifiable properties. We will give an overview of these systems in section 3.9. A system was also proposed to support pseudonymous identities that was partly implemented as the Nym Server described in section 2.3.

3.2 ISDN mixes, Real Time mixes and Web mixes

Pfitzmann, Pfitzmann & Waidner (1991) designed a system to anonymize ISDN telephone conversations. This design could be considered practical, from an engineering point of view, since it met the requirements and constraints of the ISDN network. Later the design was generalized to provide a framework for real-time, low-latency, mixed communications in (Jerichow, Müller, Pfitzmann, Pfitzmann & Waidner 1998). Finally, many of the design ideas from both ISDN and Real Time mixes were adapted for anonymous web browsing and called Web Mixes (Berthold, Federrath & Köpsell 2000). Part of the design has been

implemented as a web anonymizing proxy, JAP[6]. All three designs were the product of what could be informally called the Dresden anonymity community (although early research started in Karlsruhe), and the main ideas on which these systems are based are better illustrated by presenting them together.

A major trend in all three papers is the willingness to secure anonymous communication, even in the presence of a very powerful adversary. It is assumed that this adversary would be able to observe all communications on the network (global passive), modify the communications on the links by delaying, injecting or deleting messages, and control all but one of the mixes. While other designs, such as Mixmaster and Babel (that will be presented next), opted for a free route network topology, ISDN, Real Time and Web mixes always use cascades of mixes, making sure that each message is processed by all mixes in the same order. This removes the need for routing information to be passed along with the messages, and also protects the system from a whole set of intersection attacks presented in Berthold, Pfitzmann & Standtke (2000). The debate between the pros and cons of cascade topologies has continued throughout the years, with debates (such as (Díaz, Danezis, Grothoff, Pfitzmann & Syverson 2004)) as well as work exploring the advantages of different topologies (Danezis 2003*a*, Dingledine, Shmatikov & Syverson 2004).

The designs try never to compromise on security, and attempt to be efficient. For this reason, they make use of techniques that provide bitwise unlinkability with very small bandwidth overheads and few asymmetric cryptographic operations. *Hybrid encryption with minimal length* encrypts the header, and as much as possible of the plaintext in the asymmetrically encrypted part of the message. A stream cipher is then used to encrypt the rest of the message. This must be performed for each mix that relays the message.

Furthermore, it is understood that some protection has to be provided against active tagging attacks on the asymmetrically encrypted header. A block cipher with a globally known key is used to transform the plaintext before any encryption operation. This technique allows the hybrid encryption of long messages with very little overhead. It is interesting to notice that while the header is protected against tagging attacks, by using a known random permutation, there is no discussion about protecting the rest of the message encrypted using the stream cipher. Attacks in depth could be used, by which a partially known part of the message is XORed with some known text, in order to tag the message in a way that is recognizable when the message is decrypted. As we will see Mixmaster protects against this using a hash, while Mixminion makes sure that if modified, the tagged decoded message will contain no useful information for the attacker.

From the point of view of the dynamic aspects of mixing, ISDN, Real Time and Web mixes also introduce some novel techniques. First the route setup messages are separated from the actual data traveling in the network. In ISDN mixes, the signaling channel is used to transmit the onion encoded message that contains the session keys for each intermediary mix. Each mix then recognizes

[6] `http://anon.inf.tu-dresden.de/`

the messages belonging to the same stream, and uses the session key to prime the stream cipher and decode the messages. It is important to stress that that both "data" messages and "route setup" messages are mixed with other similar messages. It was recognized that all observable aspects of the system such as route setup and end, have to be mixed.

In order to provide anonymity for both the initiator and the receiver of a call, rendezvous points were defined. An initiator could use an anonymous label attached to an ISDN switch in order to be anonymously connected with the actual receiver. This service is perhaps the circuit equivalent of a Nym server that can be used by message-based systems. It was also recognized that special cases, such as connection establishment, disconnection and busy lines could be used by an active attacker to gain information about the communicating party. Therefore a scheme of *time slice* channels was established to synchronize such events, making them unobservable to an adversary. Call establishment, as well as call ending have to happen at particular times, and are mixed with, hopefully many, other such events. In order to create the illusion that such events happen at particular times, real or cover traffic should be sent by the users' phones through the cascade for the full duration of the time slice. An even more expensive scheme requires users to send cover traffic through the cascade back to themselves all the time. This would make call initiation, call tear-down and even the line status unobservable. While it might be possible to justify such a scheme for ISDN networks where the lines between the local exchange and the users are not shared with any other parties, it is a very expensive strategy to implement over the Internet in the case of Web mixes.

Overall, the importance of this body of work is the careful extension of mixes to a setting of high-volume streams of data. The extension was done with careful consideration for preserving the security features in the original idea, such as the unlinkability of inputs and outputs and mixing all the relevant information. Unfortunately, while the ideas are practical in the context of telecommunication networks, where the mix network is intimately linked with the infrastructure, they are less so for widely deployed modern IP networks. The idea that constant traffic can be present on the lines, and that the anonymity can be guaranteed, but be relatively low, is not practical in such contexts. Onion routing, presented in section 3.6, provides a more flexible approach that can be used as an overlay network, but it is at the same time open to more attacks. These techniques may become increasingly relevant if fixed rate traffic, such as streaming data and VoIP, require anonymization.

3.3 Babel and Mixmaster

Babel (Gülcü & Tsudik 1996) and Mixmaster ((Möller, Cottrell, Palfrader & Sassaman 2003)) were designed in the mid-nineties, and the latter has become the most widely deployed remailer. They both follow a message-based approach, namely they support sending single messages, usually email, though a fully connected mix network.

Babel offers sender anonymity, called the "forward path" and receiver anonymity,

through replies traveling over the "return path". The forward part is constructed by the sender of an anonymous message by wrapping a message in layers of encryption. The message can also include a return address to be used to route the replies. The system supports bidirectional anonymity by allowing messages to use a forward path, to protect the anonymity of the sender, and for the second half of the journey they are routed by the return address so as to hide the identity of the receiver.

While the security of the forward path is as good as in the secured original mix network proposals, the security of the return path is slightly weaker. The integrity of the message cannot be protected, thereby allowing tagging attacks, since no information in the reply address, which is effectively the only information available to intermediate nodes, can contain the hash of the message body. The reason for this is that the message is only known to the person replying using the return address. This dichotomy will guide the design of Mixminion, since not protecting the integrity of the message could open a system to trivial tagging attacks. Babel reply addresses and messages can also be used more than once, while messages in the forward path contain a unique identifier and a time-stamp that makes detecting and discarding duplicate messages efficient.

Babel also proposes a system of intermix detours. Messages to be mixed could be "repackaged" by intermediary mixes, and sent along a random route through the network. It is worth observing that even the sender of the messages, who knows all the symmetric encryption keys used to encode and decode the message, cannot recognize it in the network when this is done.

Mixmaster has been an evolving system since 1995 (Möller et al. 2003). It is the most widely deployed and used remailer system.

Mixmaster supports only sender anonymity, or in the terminology used by Babel, only anonymizes the forward path. Messages are made bitwise unlinkable by hybrid RSA and EDE 3DES encryption, while the message size is kept constant by appending random noise at the end of the message. In version two, the integrity of the RSA encrypted header is protected by a hash, making tagging attacks on the header impossible. In version three, the noise to be appended is generated using a secret shared between the remailer, and the sender of the message, included in the header. Since the noise is predictable to the sender, it is possible to include in the header a hash of the whole message therefore protecting the integrity of the header and body of the message. This trick makes replies impossible to construct: since the body of the message would not be known to the creator of the anonymous address block, it is not possible to compute in the hash.

Beyond the security features, Mixmaster provides quite a few usability features. It allows large messages to be divided in smaller chunks and sent independently through the network. If all the parts end up at a common mix, then reconstruction happens transparently in the network. So large emails can be sent to users without requiring special software. Recognizing that building robust remailer networks could be difficult (and indeed the first versions of the Mixmaster server software were notoriously unreliable) it also allowed messages to be sent multiple times, using different paths. It is worth noting that no

analysis of the impact of these features on anonymity has ever been performed.

Mixmaster also realizes that reputation attacks, by users abusing the remailer network, could discredit the system. For this reason messages are clearly labeled as coming from a remailer and black lists are kept up-to-date with email addresses that do not wish to receive anonymous email. While not filtering out any content, for example not preventing death threats being transmitted, at least these mechanisms are useful to make the network less attractive to email spammers.

3.4 Mixminion: the Type III Remailer

Mixminion (Danezis, Dingledine & Mathewson 2003) is the state of the art anonymous remailer. It allows for a fixed size message, of about 28 kbytes, to be anonymously transported over a set of remailers, with high latency. Mixminion supports sender anonymity, receiver anonymity via single-use reply blocks (SURBs), and bi-directional anonymity by composing the two mechanisms. This is achieved by mixing the message through a string of intermediate Mixminion remailers. These intermediate remailers do not know their position on the path of the message, or the total length of the path (avoiding partitioning attacks as described by (Berthold, Pfitzmann & Standtke 2000)). Intermediate remailers cannot distinguish between messages that benefit from sender anonymity and anonymous replies.

Mixminion's first key contribution concerns the cryptographic packet format. Transported messages are divided into two main headers and a body. Each main header is further divided into sub-headers encrypted under the public keys of intermediary mixes. The main objective of the cryptographic transforms is to protect messages from tagging (Pfitzmann & Pfitzmann 1990, Pfitzmann 1994): an active adversary or corrupt node may modify a message, in the hope that they will be able to detect the modification after the message has been mixed. This would allow an adversary to trace the message and compromise anonymity. Mixmaster solves this problem by including an integrity check in the header read by each intermediary: if tampering is detected the message is dropped at the fist honest mix. Mixminion cannot use a similar mechanism, because of the need to support indistinguishable routing of anonymous replies. Instead, it relies on an all-or-nothing encryption of the second header and the body of the message, which is very fragile. Tampering cryptographically results in the address of the final receiver and the message being destroyed. The cryptographic format was designed to be well understood, and as a result it is quite conservative and inefficient.

The Minx packet format aims to provide the same properties as Mixminion at a lower computational cost and overhead (Danezis & Laurie 2004). It relies on a single pass of encryption in IGE mode, that propagates ciphertext errors forward. As a result, modifying the message results again in all information about the final receiver and the message being destroyed. Since all messages look random, no partial information is ever leaked through tampering.

Mixminion uses a TCP based transport mechanism, that can accommodate

link padding. Messages are transferred between remailers using a TLS protected tunnel, with an Ephemeral Diffie-Hellman based key exchange to provide forward security. This renders any material gathered by a passive adversary useless, since it cannot be presented to a remailer operator for decryption after the ephemeral keys are deleted. It also detects active adversaries that try to corrupt data traveling on the network. Therefore an adversary must be running malicious nodes to attack the network.

Two proposals have been put forward to strengthen the forward security and compulsion resistance of Mixminion mixing. The first, in Danezis (2002), assumes that any communication leaves a trail of keys on intermediate mixes that can be used to decrypt future communications. Once a key is used, it is deleted or updated using a one-way function. Since subsequent messages may be dependent on previous messages for decoding, a mix that honestly deletes keys cannot decrypt intercepted messages upon request. Furthermore, an adversary needs to intercept and request the decryption of many messages in order to retrieve the key material necessary to decode any particular target message. The second technique (Danezis & Clulow 2005) relies on the fact that the genuine receiver of an anonymous reply can pretend to be a relay, and pass the message to another pre-determined node. This assumes a peer-to-peer remailer system, and may be an incentive to run a Mixminion server.

The implementation of Mixminion brought to the surface many practical questions. Since the transport of Mixminion messages is unreliable, it is important to implement mechanisms for retransmissions and forward error correction. Such mechanisms are not trivial to implement and may lead to traffic analysis attacks. In order to be secure, all clients must know the full network of remailers. This has proved to be a scalability problem, and a distributed directory service had to be specified in order to distribute this information. Robust mechanisms for vetting nodes, and ensuring their honest behavior are still elusive. Practical deployment and integration into familiar clients has also been a challenge.

3.5 Foiling flooding attacks

As we saw above, Babel and Mixmaster implement a traditional mix network model. They also both extend the original idea of mixing batches of messages together to feeding back messages in a pool, in the case of Mixmaster, or to delaying a fraction of messages an additional round, in the case of Babel. Such mix strategies, along with others, are susceptible to an $(n-1)$ attack, in which the adversary sends one message to be traced to an empty mix, followed by adversary messages. When the mix flushes, the only message that cannot be recognized is the one to be traced, which compromises anonymity.

In Serjantov, Dingledine & Syverson (2002) the attack is explained as acting in two phases: a trickle and a flood. In the first instance the adversary tries to flush all genuine messages from the mix, before injecting the target message and flooding it. A more rigorous analysis of how many different mix strategies are susceptible to such an attack is provided by O'Connor (2005).

The simple strategy proposed to counter such attacks is admission control,

through authentication and ticketing systems (Berthold & Langos 2002). If each user is properly authenticated when sending a message, flooding can be detected and foiled. This solution is not fully satisfactory though, since corrupt mixes may also inject messages. Having to authenticate may also reduce the perception of security offered by the system.

Diaz & Serjantov (2003) introduced a model for representing the mixing strategies of pool mixes. This model allows for easy computation of the anonymity provided by the mixing strategy towards active and passive adversaries. It was noted that $n-1$ attacks on pool mixes were favored by the deterministic dependency of the number of messages forwarded in any round and the number of messages kept in the pool for future rounds. The adversary could use this knowledge to optimize his efforts in terms of time and number of messages generated and have 100% certainty on the detection of the target at the output of the mix. In order to increase the effort and the uncertainty of the attacker, they propose randomizing the number of messages forwarded, as a binomial distribution of the number of messages contained in the pool. The randomization can be done almost for free: at the time of forwarding, the mix, instead of choosing a fix number of random messages from the pool, flips a biased coin for each message.

The first effect of the randomization is that the attacker succeeds only probabilistically, and the effort of the attacker increases as he tries to increase his probability of success. Díaz & Preneel (2004) analyzes the robustness of various combinations of mixing and dummy generation strategies towards $n-1$ attacks. It is shown that the combination of binomial mixing and randomized dummy generation strategies sets a lower bound on the anonymity of the target message. The adversary is able to significantly reduce the anonymity set of the message but he does not uniquely identify the message at the output of the mix. The protection offered to the message is proportional to the amount of dummies generated by the mix. A detailed analysis of the result and costs of deploying $n-1$ attacks is presented in Serjantov (2004).

Stop-and-Go mixes (Kesdogan, Egner & Büschkes 1998) (sg-mix) present a mixing strategy, that is not based on batches but delays. It aims at minimizing the potential for $(n-1)$ attacks. Each packet to be processed by an sg-mix contains a delay and a time window. The delay is chosen according to an exponential distribution by the original sender, and the time windows can be calculated given all the delays. Each sg-mix receiving a message, checks that it has been received within the time window, delays the message for the specified amount of time, and then forwards it to the next mix or final recipient. If the message was received outside the specified time window it is discarded. This security feature was, however, not implemented in the practical implementation of sg-mixes Reliable, which inter-operated with the pool mixes of the Mixmaster network. A practical comparison on the anonymity provided by both the pool and sg nodes of the Mixmaster network towards passive adversaries is presented in Díaz, Sassaman & Dewitte (2004). This paper shows that, even in very low traffic conditions, the pool nodes provide a high anonymity set to the messages they route at the expense of longer delays. The Reliable node, which does not

adapt the delay to the traffic load, provides no anonymity in extreme cases.

A very important feature of sg-mixes is the mathematical analysis of the anonymity they provide. Assuming that the messages arriving to the mix follow a Poisson distribution, it is observed that each mix can be modeled as a $M/M/\infty$ queue, and a number of messages waiting inside it follow the Poisson distribution. The delays can therefore be adjusted to provide the necessary anonymity set size.

The time window is used in order to detect and prevent $(n-1)$ attacks. It is observed that an adversary needs to flush the sg-mix of all messages, then let the message to be traced through and observe where it goes. This requires the attacker to hold the target message for a certain time, necessary for the mix to send out all the messages it contains and become empty. The average time that the message needs to be delayed can be estimated, and the appropriate time window can be specified to make such a delayed message be rejected by the mix.

A different solution to the $(n-1)$ attack, the rgb-mix by Danezis & Sassaman (2003), is based on a controlled level of cover traffic. In their scheme, each mix in the network sends 'red' heartbeat messages back to itself through the mix network. If at some point such messages stop arriving it may mean that the mix is subject the the first phase of the $(n-1)$ attack. The mix then responds by injecting 'green' cover traffic to confuse the adversary. The key property that makes this scheme secure is the inability of the adversary to tell apart genuine messages, to be blocked, and heartbeat messages that need to be let through for the mix not to introduce additional cover traffic. Under normal operating conditions the traffic overhead of this scheme is minimal, since additional traffic is only introduced as a response to attack.

3.6 Onion routing

Onion routing (Goldschlag, Reed & Syverson 1996, Reed, Syverson & Goldschlag 1998, Goldschlag, Reed & Syverson 1999, Syverson, Tsudik, Reed & Landwehr 2000) is the equivalent of mix networks, but in the context of circuit-based routing. Instead of routing each anonymous packet separately, the first message opens a circuit through the network, by labeling a route. Each message having a particular label is then routed on this predetermined path. Finally, a message can be sent to close the path. Often, we refer to the information traveling in each of these labeled circuits as an anonymous stream.

The objective of onion routing is to make traffic analysis harder for an adversary. It aims first at protecting the unlinkability of two participants who know each other from third parties, and secondly, at protecting the identities of the two communicating parties from each other. Furthermore, onion routing notes that ISDN mixes are not easily implementable over the Internet, and aims to distribute the anonymous network and adapt it to run on top of TCP/IP.

The first message sent through the network is encrypted in layers, that can only be decrypted by a chain of onion routers using their respective private keys. This first message contains key material shared between the original sender and the routers, as well as labels and addressing information about the next node.

As with Chaum's mixes, care is taken to provide bitwise unlinkability, so that the path that the first message takes is not trivial to follow just by observing the bit patterns of messages. Loose routing is also proposed, according to which routers relay streams through paths that are not directly specified in the original path opening message. The hope was that such a scheme would increase the anonymity provided.

Data traveling in an established circuit is encrypted using the symmetric keys distributed to the routers. Labels are used to indicate which circuit each packet belongs to. Different labels are used on different links, to ensure bitwise unlinkability, and the labels on the links are encrypted using a secret shared key between pairs of onion routers. This prevents a passive observer from knowing which packets belong to the same anonymous stream, but does not hide this information from a subverted onion router.

Onion routing admits to being susceptible to a range of attacks. It has become clear that in the absence of heavy amounts of cover traffic, patterns of traffic are present that could allow an attacker to follow a stream in the network and identify the communicating parties. Such attacks have been called timing attacks. While they are often cited in the literature (Raymond 2000), details of how they work and how effective they are have only been presented relatively recently.

Unlike ISDN mixes, onion routing does not perform mixing on the requests for opening or closing channels. While it might be plausible that enough data would be available to mix properly, it is very unlikely that the anonymity of circuit-setup messages can be maintained. Therefore, an attacker could follow such messages and compromise the anonymity of the correspondents. Furthermore, very little mixing is done in the system generally, because of the real-time performance that is assumed to be needed. Onion routing aims at providing anonymous web browsing, and therefore would become too slow if proper mixing was to be implemented. Therefore, a mixing strategy that is very close to first-in first-out for each stream is implemented. This provides only minimal mixing, and as a result a lot of attacks against onion routing focus on its weak dynamic features.

In order to make deployment easier, it was recognized that some onion routers might wish to only serve particular clients. The concept of *exit policies* was developed to encapsulate this, allowing routers to advertise which section of the network they were configured to serve. Onion routers are also free to peer with only a subset of other routers, with which they maintain long standing connections.

Zero Knowledge, a Canadian company, designed the Freedom network that follows quite closely the architecture of onion routing. The principal architect of the network was Ian Goldberg (Goldberg 2000) who published with others a series of technical papers describing the system at various levels of detail (Boucher, Shostack & Goldberg 2000, Back, Goldberg & Shostack 2001).

3.7 Tor: The new generation onion router

The onion routing project was revived in 2004, with the design and implementation of a second generation onion router called Tor (Dingledine, Mathewson & Syverson 2004*b*). Tor relays arbitrary TCP streams over a network of relays, and is particularly well tuned to work for web traffic, with the help of the Privoxy[7] content sanitizer.

Tor uses a traditional network architecture: a list of volunteer servers is downloaded from a directory service. Then, clients can create paths by choosing three random nodes, over which their communication is relayed. Instead of an 'onion' being sent to distribute the cryptographic material, Tor uses an iterative mechanism. The client connects to the first node, then it request this node to connect to the next one. The bi-directional channel is used at each stage to perform an authenticated Diffie-Hellman key exchange. This guarantees forward secrecy and compulsion resistance: only short term encryption keys are ever needed. This mechanism was first described in Cebolla (Brown 2002), and is not covered by the Onion Routing patent (Reed, Syverson & Goldschlag 2001).

One notable difference between Tor and previous attempts at anonymizing streams of traffic, is that it does not claim to offer security against even passive global observers. A set of traffic analysis techniques (Danezis 2004, Levine, Reiter, Wang & Wright 2004, Serjantov & Sewell 2003, Zhu, Fu, Graham, Bettati & Zhao 2004, Wang, Chen & Jajodia 2005, Zhu & Bettati 2005) have been developed throughout the years to trace streams of continuous traffic traveling in a low latency network. A separate but related thread of research has been developed in the intrusion detection community, that tries to uncover machines used as stepping stones for attack (Wang & Reeves 2003, Blum, Song & Venkataraman 2004). These attacks have been shown difficult to foil, unless the latency of the traffic is high, or a lot of cover traffic is injected – both of which are very expensive. Tor instead opts for getting security though being highly usable and cheap to operate (Back, Möller & Stiglic 2001, Dingledine & Mathewson 2005). As a result, an adversary who can observe a stream at two different points, can trivially realize it is the same traffic.

Its vulnerability against passive adversaries has made Tor fragile against previously unexplored attacks. First, a realistic assessment of the probability a single party can observe multiple points on the path is necessary. It turns out that the topology of the Internet is such that many, seemingly unrelated networks, are interconnected through hubs, or long distance links that can be observed cheaply by a single ISP entity (Feamster & Dingledine 2004). A second possible path for attack in Murdoch & Danezis (2005) uses indirect network measurements to perform traffic analysis, and does away with the assumption that a passive adversary needs local access to the communication to perform traffic analysis. An attacker relays traffic over all routers, and measures their latency: this latency is affected by the other streams transported over the router. Long term correlations between known signals injected by a malicious server and the measurements are possible. This allows an adversary to trace a connection

[7] `http://www.privoxy.org/`

up to the first router used to anonymize it.

Tor also provides a mechanism for 'hidden servers'. A hidden server opens an anonymous connection and uses it to advertise a contact point. A client that wants to contact the server, goes to the contact point and negotiates a separate anonymous channel used to relay the actual communication. An attack against this early architecture was demonstrated by Øverlier & Syverson (2006). The intuition behind this attack is that an adversary can open multiple connections to the hidden server, sequentially or in parallel, and control the flow of traffic towards the server. The adversary needs to control one corrupt router, and wait until for one of the connections his router is chosen by the server as the first node for the fresh anonymous path. Then the adversary effectively controls two nodes on the anonymous path, one of which is next to the real server – and the anonymity provided to the server is completely compromised. The idea of consistently using a more trusted 'valet' router as the first node into the Tor network was proposed as a countermeasure against this attack by verlier & Syverson (2006).

3.8 Peer-to-peer mix networks

In Chaum's original work it is assumed that if each participant in the mix network also acts as a mix for others, this would improve the overall security of the network. Recent interest in peer-to-peer networking has influenced some researchers to further examine such networks with large, but transient, numbers of mixes.

Freedman & Morris (2002) designed *Tarzan*, a peer-to-peer network in which every node is a mix. A peer initiating the transport of a stream through the network would create an encrypted tunnel to another node, and ask that node to connect the stream to another peer. By repeating this process a few times, it is possible to have an onion encrypted connection, relayed through a sequence of intermediate nodes.

An interesting feature of Tarzan is that the network topology is somewhat restricted. Each node maintains persistent connections with a small set of other nodes, forming a structure called a *mimics*. Routes of anonymous messages are selected in such a way that they will go through mimics and between mimics in order to avoid links with insufficient traffic. A weakness of the mimics scheme is that the selection of neighboring nodes is done on the basis of a network identifier or address which, unfortunately, is easy to spoof in real-world networks.

The original Tarzan design only required each node to know a random subset of other nodes in the network. This is clearly desirable due to the very dynamic nature of peer-to-peer networks, and the volatility of nodes. On the other hand, Danezis & Clayton (2006) found some attacks against this strategy in a preliminary version of Tarzan (Freedman, Sit, Cates & Morris 2002). The attack relies on the fact that the network is very large, and nodes have a high churn rate. As a result any single node only knows a small subset of other nodes. An adversary node, included on the anonymous path, can tell that the originator of the connection knew three nodes: the corrupt node itself, its successor and

predecessor. It turns out that those three nodes identify uniquely the originator of the stream with very high probability. The final version of Tarzan requires each node to know all others in order to fix this attack, which is clearly less practical.

Rennhard & Plattner (2002) introduced *MorphMix*, which shares a very similar architecture and threat model with Tarzan. A crucial difference is that the route through the network is not specified by the source but chosen by intermediate nodes, observed by witnesses specified and trusted by the user. While the attack by Danezis and Clayton does not apply to route selection, variants might apply to the choice of witness nodes.

MorphMix realises that leaving the intermediate nodes to choose the route through the network might lead to *route capture* or, in other words, the first subverted mix on the path choosing only other subverted mixes. For this reason, MorphMix includes a *collusion detection* mechanism that monitors for any cliques in the selection of nodes in the path. This prevents subverted nodes from routinely running attacks on the network but does not provide security in every case. Tabriz & Borisov (2006) presented an attack on the collusion resistance mechanism of MorphMix.

3.9 Robust & verifiable mix constructions

Chaum's original mix network design included a system of signed receipts to assure senders that their messages have been properly processed by the network. A whole body of research was inspired by this property and has attempted to create mix systems which are robust against subverted servers denying service, and that could offer a proof of their correct functioning alongside the mixing. Such systems have been closely associated with voting, where both universal verifiability of vote delivery and privacy are of great importance.

Most of the proposed schemes use the idea of a mix cascade. For this reason, no information is usually communicated between the sender of a message and intermediate mixes in the cascade. It is assumed that routing information is not necessary, since mixes process messages in a fixed order. The first scheme to take advantage of this was the *efficient anonymous channel and all/nothing election scheme* proposed by Park, Itoh & Kurosawa (1993). In this system, messages are an El Gamal ciphertext of fixed length, independently of the number of mixes they go through. Furthermore, the scheme uses a *cut and choose* strategy, which makes it all-or-nothing, meaning that if any of the ciphertexts is removed, then no result at all is output. This property assures that partial results do not affect a re-election. Birgit Pfitzmann found two attacks against this proposal (Pfitzmann 1994). The first attack is very similar to Pfitzmann & Pfitzmann (1990), and makes use of characteristics that are invariant at the different stages of mixing because of the El Gamal cryptosystem. An active attack is also found, where the input El Gamal ciphertext is blinded, by being raised to a power, which results in the final output also being raised to this power. This is a chosen ciphertext attack with which a lot of systems will struggle, and eventually fail to eliminate. Birgit Pfitzmann also notes that the

threat model assumed is somehow weaker than the one proposed by Chaum. A dishonest sender is capable of disrupting the whole network, which is worse than a single mix, as it is the case in Chaum's paper. Birgit did not propose any practical countermeasures to these attacks, since any straightforward fix would compromise some of the interesting features of the system.

In parallel with Birgit Pfitzmann's work, Kilian & Sako (1995) proposed a *receipt-free mix-type voting scheme*. They attempt to add universal verifiability to Park et al. (1993), which means that all senders will be able to verify that all votes were taken into account, not simply their own. They also highlight that many verifiable mix schemes provide at the end of mixing a receipt, that could be used to sell or coerce one's vote, and attempt to make their system *receipt-free*. They do this by forcing each mix to commit to their inputs and outputs, and prove in zero knowledge that they performed the decryption and shuffle correctly. Unfortunately, Michels & Horster (1996) show that the scheme is not receipt-free if a sender collaborates with a mix, and that the active attacks based on blinding proposed by Birgit Pfitzmann could be used to link inputs to outputs.

In order to avoid disruption of the system if a subset of mixes is subverted, Ogata, Kurosawa, Sako & Takatani (1997) proposed a *fault tolerant anonymous channel*. This uses a threshold cryptosystem to make sure that a majority of mixes can decode messages even if a minority does not collaborate. Two systems are proposed, one based on El Gamal and the other based on the r^{th} residue problem. A zero knowledge proof of correct shuffling is also proposed for the r^{th} residue problem.

In 1998, Abe (1998) presented a mix system that provided universal verifiability and was efficient, in the sense that the verification work was independent from the number of mix servers. This scheme shows an attack on Kilian & Sako (1995), that uses the side information output for the verification to break the privacy of the system. It then presents a mix system that works in two phases, El Gamal re-encryption and then threshold decryption. The first phase is proved to be correct before the second can proceed, and then a proof of correct decryption is output at the end of the second stage.

The systems that provide universal verifiability based on proofs of permutations, and zero knowledge proofs are computationally very expensive. Jakobsson (1998) designs the Practical Mix, and tries to reduce the number of expensive operations. In order to prove the correctness of the shuffle, novel techniques called *repetition robustness* and *blinded destructive robustness* are introduced. The network works in two phases: first, the ciphertexts are El Gamal blinded, and then, the list of inputs is replicated. Each of the replicated lists is decoded by all mixes, which results in lists of blinded plaintexts. The resulting lists are sorted and compared. If all elements are present in all lists then no mix has tampered with the system and the unblinding and further mixing can proceed. Otherwise, the sub-protocol for cheater detection is run. While being very efficient, the Practical Mix has not proved to be very secure, as shown by Desmedt & Kurosawa (2000). They show that one subverted mix in the practical mix can change ciphertexts, and still not be detected. They then introduce a new

mix design, in which verification is performed by subsets of mixes. The subsets are generated in such a way that at least one is guaranteed not to contain any subverted mixes.

In an attempt to further reduce the cost of mixing, Jakobsson (1999) introduced the Flash Mix, that uses re-encryption instead of blinding to keep the number of exponentiations down. As in the practical mix, mixing operates in many phases, and uses *repetition robustness* to detect tampering. Furthermore, two dummy messages are included in the input, that are de-anonymized after all mixes have committed to their outputs, in order to make sure that attacks such as Desmedt & Kurosawa (2000) do not work. An attack against Flash mixing was found in Mitomo & Kurosawa (2000) and fixed by changing the unblinding protocol.

A breakthrough occurred when Furukawa & Sako (2001) and Neff (2001) proposed efficient general techniques to universally verify the correctness of a shuffle of El Gamal ciphertexts. The first provides proof that the matrix used was a permutation matrix, and the second uses verifiable secret exponent multiplication to improve its efficiency.

Even though the above techniques are more efficient than any other previously known, they are still not efficient enough to scale for elections, with millions of participants. For this reason, Golle, Zhong, Boneh, Jakobsson & Juels (2002) proposed optimistic mixing, a mix that works quickly if there is no attack detected, but provides no result if an error occurs. In this case, it provides a fall back mechanism for a more robust technique such as Neff (2001) to be used. Each mix in the chain outputs a "proof" of permutation, that could be faked by tampering with the ciphertexts. This is detected by making the encryption plaintext-aware. The second decryption, revealing the votes, is only performed if all outputs of mixing are well-formed. A series of attacks were found against this scheme (Wikström 2003*b*, Wikström 2002). The first two attacks are closely related to Pfitzmann (1994) and can break the anonymity of any user. The second attack is related to Desmedt & Kurosawa (2000) and can break the anonymity of all users and compromise the robustness. Finally, attacks based on improperly checking the El Gamal elements are also applicable, and further explored in Wikström (2003*a*).

A serious drawback of traditional robust mix cascades is that each mix has to wait for the output of the previous mix before processing messages. This means that the latency increases with the number of mixes, and that most of the time mixes perform no computations. Golle & Juels (2004*b*) present a technique that allows for universally verifiable parallel mixing in four steps of communication and the equivalent of two steps of mixing. Their techniques drastically reduce the latency of mixing, but Borisov (2005) shows that when multiple input messages are known to the adversary, the anonymity provided by this technique is far from optimal.

A hopeful line of research looks at extending robust cascades into general mix networks. These may use non deterministic decoding and routing protocols, possibly implemented using the new universal re-encryption primitive suggested in Golle, Jakobsson, Juels & Syverson (2004), and improved for space efficiency

by Fairbrother (2004). Sadly, its mathematical properties make universal re-encryption malleable and many systems that use it (Klonowski, Kutylowski & Zagrski 2005,, Gomulkiewicz, Klonowski & Kutylowski 2004, Klonowski, Kutylowski, Lauks & Zagorski 2004, Lu, Fang, Sun & Guo 2005, Lu, Fang, Sun & Cheng 2005) were found to be vulnerable to tagging attacks in Danezis (2006).

Finally, a fundamentally new way of looking at robust mixing is presented in Adida & Wikström (2005): mixing is seen as a computation to be outsourced to a third party. Yet, this third party should gain no information about the actual shuffle. Two protocols that implement such an algorithm are presented, based on Paillier and BGN homomorphic encryption. The third party accepts a set of ciphertexts, and performs in the obfuscated mixing algorithm to produce a re-encrypted and shuffled set of outputs. Despite only public keys being used, neither the third party, nor any observer, can link inputs and outputs.

3.10 Provable security for mix-networks

While most designs for robust mix nets use pure El Gamal encryption, some provide solutions for hybrid schemes. Ohkubo & Abe (2000) present a hybrid mix without ciphertext expansion. Jakobsson & Juels (2001) also present a scheme that is resistant to any minority coalition of servers. Möller (2003) proves the security properties of a mix packet format. Further work in proving packet formats correct was presented in Camenisch & Lysyanskaya (2005). Other packet formats attempt to provide specialized properties: Golle (2004) allows a mix to prove that a particular output was an input to the mix, clearing the operator from any suspicion that they injected the message.

Reputation based schemes have also been used to increase the reliability of mix networks in Dingledine, Freedman, Hopwood & Molnar (2001), and mix cascades in Dingledine & Syverson (2002). Both these papers present how *statistics pages* compiled in the Mixmaster system using *pingers* (Palfrader n.d.) can be replaced with a more robust system to determine which nodes are reliable and which are not. Users can then choose reliable nodes, or the system can exclude unreliable ones from directories.

An option that universal verifiability to be implemented on generic mix networks, is the randomized partial checking presented in Jakobsson, Juels & Rivest (2002). In this scheme, all mixes commit to their inputs and outputs and then they are required to disclose half of all correspondences. This assures that if a mix is dropping messages it will be quickly detected. Privacy is maintained by pairing mixes, and making sure that the message is still going through enough secret permutations. For safety, it was proposed that mixes are paired, and when one in a pair is required to disclose the correspondences the other keeps it secret, in order to ensure that enough mixing is performed for each message. In Gomulkiewicz, Klonowski & Kutylowski (2003) it is shown, using path coupling tools and graph theory, that such caution is not necessary, since messages will mix with high probability after $\log N$ steps, even if correspondences are revealed at random.

A separate line of research attempts to prove the mixing, and hence privacy

properties instead of robustness. Such systems are usually expensive, since they rely on extensive amounts of cover traffic to provably ensure that no information about the actual traffic patterns is leaked. Systems in this tradition are Rackoff & Simon (1993) and Berman, Fiat & Ta-Shma (2004). The latter proves that in a random network of communications, one could embed a very large number of possible sub-networks of a certain butterfly-like form, and show that, at each step, messages are mixed together with high probability. Interestingly, Klonowski & Kutylowski (2005) prove that traditional mix networks mix all input messages after $\log N$ rounds, despite the presence of adversaries that reveal to each other the path of messages.

Some mix strategies are designed on purpose to be fragile. If a single message gets deanonymized through compulsion, then all the messages get deanonymized (Reiter & Wang 2004), or a secret of the mix operator can easily be inferred (Golle, Wang, Jakobsson & Tsow 2006). This provides operators with incentives to resist compulsion, and is meant to make it disproportionate (in jurisdictions where this is a concern) to request even one message to be traced.

4 Other systems

A number of other anonymous communication systems have been proposed through the years. Chaum (1988) presents the dining cryptographers' network, a multi-party computation that allows a set of participants to have perfect (information theoretic) anonymity. The scheme is very secure but impractical, since it requires a few broadcasts for each message sent and is easy to disrupt for dishonest users. A modification of the protocol in Waidner & Pfitzmann (1989), guarantees availability against disrupting nodes. Herbivore (Goel, Robson, Polte & Sirer 2003) uses DC-nets as part of a two-level anonymity system: users form small cliques that communicate within them using DC-nets. Finally, in Golle & Juels (2004*a*) asymmetric techniques are described that make DC-nets robust against disruption.

P5, by Sherwood, Bhattacharjee & Srinivasan (2002), uses broadcast-trees to achieve anonymity. Buses by Beimel & Dolev (2003) use the metaphor of a bus route that travels over all nodes carrying messages. This is in fact a broadcast, and trade-offs between longer routes and more routes are discussed from an anonymity and latency perspective.

Traffic Analysis Prevention (TAP) systems, attempt to provide third party anonymity, given a collaborating set of senders, receivers and relays. Timmerman (1999) describes adaptive traffic masking techniques, and a security model to achieve *traffic flow confidentiality* (Timmerman 1997). The information theoretic approach to analysing TAP systems is presented by Newman, Moskowitz, Syverson & Serjantov (2003). They study how much protection is offered overall to the traffic going through a TAP system, by creating a rigorous mathematical model of traffic analysis, rerouting and cover traffic. This builds on their previous work in Venkatraman & Newman-Wolfe (1994). The research group at the Texas A&M University has a long-term interest in traffic analysis prevention

of real time traffic (Guan, Fu, Xuan, Shenoy, Bettati & Zhao 2001). Similarly, Jiang, Vaidya & Zhao (2000) present TAP techniques to protect wireless packet radio traffic.

5 Conclusions

Anonymous communications, despite being first proposed over 25 years ago, has become since 2000 an extremely active field of research. It is also increasingly relevant since systems that are the direct result of this research, like Tor, JAP and Mixminion, are being deployed and used to protect the privacy of thousands of people.

Anonymous communications research has also matured to the point that new systems must imperatively take into account the existing literature and ensure that they are not weak under known attacks and models. The aims of this survey has been to present a road map of the most important systems-concepts, and the key refinements they have been subject to.

As in any mature field new designs will inevitably have to mix and match from elements already present, in older systems, to best match their environment. Designs tailored to peer-to-peer systems or telephony are a prime example of this. Those systems are also a prime example of the care that researcher must exert then mixing and matching ideas: anonymous communications are fragile, and even simple modifications may lead to traffic analysis attacks.

A set of key observations must be in the minds of designers and researchers looking at anonymous communications in the future.

The *concepts of anonymity* in communication networks is a well understood problem. Definitions and metrics that express the anonymity properties of communications are available, and used to evaluate systems. Despite all security efforts, an upper limit on the anonymity that a system can provide is given by black box attacks: no matter how good the anonymity system is, effective attacks can be deployed in the long term by observing the edges of the anonymous communication network. As a result we say that the use of anonymous communication can be secured only tactically (for short periods) and not strategically or in the long term.

Concerning *trust models* the earliest anonymous communication systems relied on one central trusted server. This model has proven weak against compulsion, denial of service, traffic analysis carried by a local eavesdropper, or maliciousness of the central node. Centralized trust models have been abandoned in favor of models where trust is distributed over a set of servers. Trusting a set of unknown, random nodes presents some serious limitations as well, particularly against attackers able to introduce a large number of corrupted nodes in the system (sybil attacks).

Solutions for *high-latency applications* such as email have significantly evolved since the first schemes were proposed. The loose delay requirements allow for the design of secure solutions, providing a reasonably high resistance to attacks and anonymity level.

On the other hand, *low-latency* constraints seriously limit the anonymity that can be provided against powerful adversaries. Currently deployed solutions are vulnerable against attackers who have access to both ends of the communication. In particular, the variability of HTTP traffic makes it hard to conceal the correlation of input and output at the edges of the network using black box attacks.

There has been a link between anonymous communications and *censorship resistance* research, as solutions for one problem have been applied to the other. More research is needed to determine whether anonymity is the best tactic to distribute content that may be censored, or whether it adds cost that may be limiting the distribution even further.

Finally, anonymous communication networks can be subject to a wide range of attacks. The most popular attacker models is the global attacker (with access to all communication lines, passive or active); and attackers capable of controlling a subset of the nodes in the network (Crowds model). The attacks against which anonymity networks are most vulnerable include traffic analysis, flooding, compulsion, and attacks on the cryptographic protocols (such as tagging attacks).

Know-how in attacking anonymous communication grows at the same, or even faster rate, as our ability to design secure systems. As more systems are deployed further attacks are uncovered, making use of the implementation environment and the actual usage of the anonymity systems. Anonymity design has proved to be a non trivial problem, but so far we have only scraped the surface of the anonymity engineering, deployment and operations problems.

References

Abe, M. (1998), Universally verifiable MIX with verification work independent of the number of MIX servers, *in* K. Nyberg, ed., 'Advances in Cryptology (Eurocrypt'98)', Vol. 1403 of *LNCS*, Springer-Verlag, Helsinki, Finland, pp. 437–447.

Adida, B. & Wikström, D. (2005), 'Obfuscated ciphertext mixing', Cryptology ePrint Archive, Report 2005/394.

Agrawal, D., Kesdogan, D. & Penz, S. (2003), Probabilistic treatment of mixes to hamper traffic analysis, *in* 'IEEE Symposium on Security and Privacy', IEEE Computer Society, Berkeley, CA, USA, pp. 16–27.

Anderson, R. (1996), The eternity service, *in* '1st International Conference on the Theory and Applications of Cryptology (Pragocrypt '96)', Czech Technical University Publishing House, Prague, Czech Republic, pp. 242–252.

Asonov, D. & Freytag, J. (n.d.), 'Almost optimal private information retrieval', *Privacy Enhancing Technology 2002* .

Back, A., Goldberg, I. & Shostack, A. (2001), Freedom systems 2.1 security issues and analysis, White paper, Zero Knowledge Systems, Inc.

Back, A., Möller, U. & Stiglic, A. (2001), Traffic analysis attacks and trade-offs in anonymity providing systems, *in* I. S. Moskowitz, ed., 'Information Hiding workshop (IH 2001)', Vol. 2137 of *LNCS*, Springer-Verlag, pp. 245–257.

Beimel, A. & Dolev, S. (2003), 'Buses for anonymous message delivery', *Journal of Cryptology* **16**(1), 25–39.

Bennett, K. & Grothoff, C. (2003), GAP – practical anonymous networking, *in* R. Dingledine, ed., 'Privacy Enhancing Technologies workshop (PET 2003)', Vol. 2760 of *LNCS*, Springer-Verlag, pp. 141–160.

Berman, R., Fiat, A. & Ta-Shma, A. (2004), Provable unlinkability against traffic analysis, *in* A. Juels, ed., 'Proceedings of Financial Cryptography (FC '04)', Springer-Verlag, LNCS 3110.

Berthold, O., Federrath, H. & Köpsell, S. (2000), Web MIXes: A system for anonymous and unobservable Internet access, *in* H. Federrath, ed., 'Designing Privacy Enhancing Technologies', Vol. 2009 of *LNCS*, Springer-Verlag, pp. 115–129.

Berthold, O. & Langos, H. (2002), Dummy traffic against long term intersection attacks, *in* 'Designing Privacy Enhancing Technologies, Proceedings of PET'02', Springer-Verlag, LNCS 2482, pp. 110–128.

Berthold, O., Pfitzmann, A. & Standtke, R. (2000), The disadvantages of free MIX routes and how to overcome them, *in* H. Federrath, ed., 'Designing Privacy Enhancing Technologies', Vol. 2009 of *LNCS*, Springer-Verlag, pp. 30–45.

Bethencourt, J., Song, D. X. & Waters, B. (2006), New constructions and practical applications for private stream searching (extended abstract)., *in* 'S&P', IEEE Computer Society, pp. 132–139.

Bissias, G. D., Liberatore, M., & Levine, B. N. (2005), Privacy vulnerabilities in encrypted HTTP streams, *in* '5th Workshop on Privacy Enhancing Technologies (PET2005)'.

Blum, A., Song, D. X. & Venkataraman, S. (2004), Detection of interactive stepping stones: Algorithms and confidence bounds., *in* E. Jonsson, A. Valdes & M. Almgren, eds, 'RAID', Vol. 3224 of *Lecture Notes in Computer Science*, Springer, pp. 258–277.

Borisov, N. (2005), An analysis of parallel mixing with attacker-controlled inputs, *in* 'Proceedings of Privacy Enhancing Technologies workshop (PET 2005)'.

Boucher, P., Shostack, A. & Goldberg, I. (2000), Freedom systems 2.0 architecture, White paper, Zero Knowledge Systems, Inc.

Brown, Z. (2002), Cebolla – pragmatic IP anonymity, *in* 'Ottowa Linux Symposium'.

Camenisch, J. & Lysyanskaya, A. (2005), A formal treatment of onion routing, *in* V. Shoup, ed., 'Proceedings of CRYPTO 2005', Springer-Verlag, LNCS 3621, pp. 169–187.

Chaum, D. (1981), 'Untraceable electronic mail, return addresses, and digital pseudonyms', *Communications of the ACM* **24**(2), 84–88.

Chaum, D. (1983), Blind signatures for untraceable payments, *in* D. Chaum, R. L. Rivest & A. T. Sherman, eds, 'Advances in Cryptology (Crypto'82)', Plenum Press, New York and London, pp. 199–203.

Chaum, D. (1988), 'The dining cryptographers problem: Unconditional sender and recipient untraceability', *Journal of Cryptology* **1**, 65–75.

Cheng, H. & Avnur, R. (n.d.), 'Traffic analysis of ssl encrypted web browsing', `http://citeseer.ist.psu.edu/656522.html`.

Clarke, I., Sandberg, O., Wiley, B. & Hong, T. W. (2000), Freenet: A distributed anonymous information storage and retrieval system, *in* H. Federrath, ed., 'Designing Privacy Enhancing Technologies', number 2009 *in* 'LNCS', Springer-Verlag, Berkeley, CA, USA, pp. 46–66.

Clayton, R. (n.d.), 'Failures in a Hybrid Content Blocking System'.

Clayton, R., Murdoch, S. J. & Watson, R. N. M. (2006), Ignoring the great firewall of china, *in* G. Danezis & P. Golle, eds, 'Privacy Enhancing Technologies workshop (PET 2006)', LNCS, Springer-Verlag.

Danezis, G. (2002), Forward secure mixes, *in* J. Fisher-Hubner, ed., 'Nordic workshop on Secure IT Systems (Norsec 2002)', Karlstad, Sweden, pp. 195–207.

Danezis, G. (2003*a*), Mix-networks with restricted routes, *in* R. Dingledine, ed., 'Privacy Enhancing Technologies workshop (PET 2003)', Vol. 2760 of *LNCS*, Springer-Verlag, Dresden, Germany, pp. 1–17.

Danezis, G. (2003*b*), Statistical disclosure attacks, *in* Gritzalis, Vimercati, Samarati & Katsikas, eds, 'Security and Privacy in the Age of Uncertainty, (SEC2003)', IFIP TC11, Kluwer, Athens, pp. 421–426.

Danezis, G. (2004), The traffic analysis of continuous-time mixes, *in* 'Proceedings of Privacy Enhancing Technologies workshop (PET 2004)', LNCS.

Danezis, G. (2006), Breaking four mix-related schemes based on universal re-encryption, *in* 'Proceedings of Information Security Conference 2006', Springer-Verlag.

danezis, G. (n.d.), Traffic analysis of the http protocol over tls. `http://www.cl.cam.ac.uk/~gd216/TLSanon.pdf`.

Danezis, G. & Clayton, R. (2006), Route fingerprinting in anonymous communications, *in* 'IEEE International Conference on Peer-to-Peer Computing (P2P2006)', IEEE.

Danezis, G. & Clulow, J. (2005), Compulsion resistant anonymous communications., *in* M. Barni, J. Herrera-Joancomartí, S. Katzenbeisser & F. Pérez-González, eds, 'Information Hiding', Vol. 3727 of *Lecture Notes in Computer Science*, Springer, pp. 11–25.

Danezis, G. & Diaz, C. (2006), 'Improving the decoding efficiency of private search', Cryptology ePrint Archive, Report 2006/024. `http://eprint.iacr.org/`.

Danezis, G., Dingledine, R. & Mathewson, N. (2003), Mixminion: Design of a Type III Anonymous Remailer Protocol, *in* 'IEEE Symposium on Security and Privacy', Berkeley, CA.

Danezis, G. & Laurie, B. (2004), Minx: A simple and efficient anonymous packet format, *in* 'Workshop on Privacy in the Electronic Society (WPES 2004)', ACM.

Danezis, G. & Sassaman, L. (2003), Heartbeat traffic to counter $(n-1)$ attacks, *in* 'workshop on Privacy in the Electronic Society (WPES 2003)', Washington, DC, USA.

Danezis, G. & Serjantov, A. (2004), Statistical disclosure or intersection attacks on anonymity systems., *in* J. J. Fridrich, ed., 'Information Hiding', Vol. 3200 of *Lecture Notes in Computer Science*, Springer, pp. 293–308.

Desmedt, Y. & Kurosawa, K. (2000), How to break a practical mix and design a new one, *in* B. Preneel, ed., 'Advances in Cryptology (Eurocrypt 2000)', Vol. 1807 of *LNCS*, Springer-Verlag, Bruges, Belgium, pp. 557–572.

Díaz, C., Danezis, G., Grothoff, C., Pfitzmann, A. & Syverson, P. F. (2004), Panel discussion - mix cascades versus peer-to-peer: Is one concept superior?, *in* Martin & Serjantov (2005), pp. 242–242.

Díaz, C. & Preneel, B. (2004), Reasoning about the anonymity provided by pool mixes that generate dummy traffic, *in* 'Proceedings of 6th Information Hiding Workshop (IH'04)', Springer, LNCS 3200, pp. 309–325.

Díaz, C., Sassaman, L. & Dewitte, E. (2004), Comparison between two practical mix designs, *in* 'Proceedings of 9th European Symposiumon Research in Computer Security (ESORICS'04)', Springer-Verlag, LNCS 3193, pp. 141–159.

Diaz, C. & Serjantov, A. (2003), Generalising mixes, *in* R. Dingledine, ed., 'Privacy Enhancing Technologies workshop (PET 2003)', Vol. 2760 of *LNCS*, Springer-Verlag, Dresden, Germany, pp. 18–31.

Diaz, C., Seys, S., Claessens, J. & Preneel, B. (2002), Towards measuring anonymity, *in* R. Dingledine & P. Syverson, eds, 'Privacy Enhancing Technologies workshop (PET 2002)', Vol. 2482 of *LNCS*, Springer-Verlag, San Francisco, CA, USA, pp. 54–68.

Dierks, T. & Allen, C. (1999), 'Rfc 2246: The tls protocol version 1.0', `http://www.ietf.org/rfc/rfc2246.txt`.

Dingledine, R., Freedman, M. J., Hopwood, D. & Molnar, D. (2001), A Reputation System to Increase MIX-net Reliability, *in* I. S. Moskowitz, ed., 'Information Hiding workshop (IH 2001)', Vol. 2137 of *LNCS*, Springer-Verlag, pp. 126–141.

Dingledine, R., Freedman, M. J. & Molnar, D. (2000), The free haven project: Distributed anonymous storage service, *in* H. Federrath, ed., 'Designing Privacy Enhancing Technologies', Vol. 2009 of *LNCS*, Springer-Verlag, Berkeley, CA, USA, pp. 67–95.

Dingledine, R. & Mathewson, N. (2005), 'Anonymity loves company: Usability and the network effect', In "Security and Usability", an O'Reilly Media book.

Dingledine, R., Mathewson, N. & Syverson, P. (2004*a*), Tor: The second-generation onion router, *in* 'Proceedings of the 13th USENIX Security Symposium'.

Dingledine, R., Mathewson, N. & Syverson, P. (2004*b*), Tor: The second-generation onion router, *in* 'Proceedings of the 13th USENIX Security Symposium'.

Dingledine, R., Shmatikov, V. & Syverson, P. F. (2004), Synchronous batching: From cascades to free routes., *in* Martin & Serjantov (2005), pp. 186–206.

Dingledine, R. & Syverson, P. (2002), Reliable MIX Cascade Networks through Reputation, *in* M. Blaze, ed., 'Financial Cryptography (FC '02)', Vol. 2357 of *LNCS*, Springer-Verlag.

Dingledine, R. & Syverson, P. F., eds (2003), *Privacy Enhancing Technologies, Second International Workshop, PET 2002, San Francisco, CA, USA, April 14-15, 2002, Revised Papers*, Vol. 2482 of *Lecture Notes in Computer Science*, Springer.

Dornseif, M. (2003), Government mandated blocking of foreign web content, *in* 'von Knop, J., Haverkamp, W., Jessen, E. (eds.): Security, E-Learning, E-Services: Proceedings of the 17. DFN-Arbeitstagung uber Kommunikationsnetze, Dusseldorf'.

Fairbrother, P. (2004), An improved construction for universal re-encryption., *in* Martin & Serjantov (2005), pp. 79–87.

Feamster, N., Balazinska, M., Harfst, G., Balakrishnan, H. & Karger, D. (2002), Infranet: Circumventing web censorship and surveillance, *in* D. Boneh, ed., 'USENIX Security Symposium', San Francisco, CA, pp. 247–262.

Feamster, N., Balazinska, M., Wang, W., Balakrishnan, H. & Karger, D. (2003), Thwarting web censorship with untrusted messenger discovery, *in* R. Dingledine, ed., 'Privacy Enhancing Technologies workshop (PET 2003)', Vol. 2760 of *LNCS*, Springer-Verlag, Dresden, Germany, pp. 125–140.

Feamster, N. & Dingledine, R. (2004), Location diversity in anonymity networks, *in* 'Proceedings of the Workshop on Privacy in the Electronic Society (WPES 2004)', Washington, DC, USA.

Fousse, L. & Reinhard, J.-R. (2006), 'Nymbaron: A type iii nymserver', On-line. `http://www.komite.net/laurent/soft/nymbaron/`.

Freedman, M. J. & Morris, R. (2002), Tarzan: A peer-to-peer anonymizing network layer, *in* V. Atluri, ed., 'ACM Conference on Computer and Communications Security (CCS 2002)', ACM, Washington, DC, pp. 193–206.

Freedman, M. J., Sit, E., Cates, J. & Morris, R. (2002), Introducing tarzan, a peer-to-peer anonymizing network layer, *in* P. Druschel, M. F. Kaashoek & A. I. T. Rowstron, eds, 'International workshop on Peer-to-Peer Systems (IPTPS)', Vol. 2429 of *LNCS*, Springer-Verlag, Cambridge, MA, pp. 121–129.

Furukawa, J. & Sako, K. (2001), An efficient scheme for proving a shuffle, *in* J. Kilian, ed., 'Advances in Cryptology (Crypto 2001)', Vol. 2139 of *LNCS*, Springer-Verlag, Santa Barbara, CA, USA, pp. 368–387.

Goel, S., Robson, M., Polte, M. & Sirer, E. G. (2003), Herbivore: A Scalable and Efficient Protocol for Anonymous Communication, Technical Report 2003-1890, Cornell University, Ithaca, NY.

Goldberg, I. (2000), A Pseudonymous Communications Infrastructure for the Internet, PhD thesis, UC Berkeley.

Goldschlag, D. M., Reed, M. G. & Syverson, P. F. (1996), Hiding routing information, *in* R. J. Anderson, ed., 'Information Hiding', Vol. 1174 of *LNCS*, Springer-Verlag, Cambridge, U.K., pp. 137–150.

Goldschlag, D. M., Reed, M. G. & Syverson, P. F. (1999), 'Onion routing', *Communications of the ACM* **42**(2), 39–41.

Golle, P. (2004), Reputable mix networks, *in* 'Proceedings of Privacy Enhancing Technologies workshop (PET 2004)', Vol. 3424 of *LNCS*.

Golle, P., Jakobsson, M., Juels, A. & Syverson, P. (2004), Universal re-encryption for mixnets, *in* 'Proceedings of the 2004 RSA Conference, Cryptographer's track', San Francisco, USA.

Golle, P. & Juels, A. (2004*a*), Dining cryptographers revisited, *in* 'Proceedings of Eurocrypt 2004'.

Golle, P. & Juels, A. (2004*b*), Parallel mixing, *in* 'Proceedings of the 11th ACM Conference on Computer and Communications Security (CCS 2004)', ACM Press.

Golle, P., Wang, X., Jakobsson, M. & Tsow, A. (2006), Deterring voluntary trace disclosure in re-encryption mix networks, *in* 'Proceedings of the 2006 IEEE Symposium on Security and Privacy', IEEE CS, Oakland, CA, pp. 121–131.

Golle, P., Zhong, S., Boneh, D., Jakobsson, M. & Juels, A. (2002), Optimistic mixing for exit-polls, *in* Y. Zheng, ed., 'Advances in Cryptology (Asiacrypt 2002)', Vol. 2501 of *LNCS*, Springer-Verlag, Queenstown, New Zealand, pp. 451–465.

Gomulkiewicz, M., Klonowski, M. & Kutylowski, M. (2003), Rapid mixing and security of chaum's visual electronic voting, *in* 'Proceedings of ESORICS 2003'.

Gomulkiewicz, M., Klonowski, M. & Kutylowski, M. (2004), Onions based on universal re-encryption – anonymous communication immune against repetitive attack, *in* C. H. Lim & M. Yung, eds, 'Information Security Applications, 5th International Workshop, WISA 2004', Vol. 3325 of *Lecture Notes in Computer Science*, Springer, Jeju Island, Korea, pp. 400–410.

Guan, Y., Fu, X., Xuan, D., Shenoy, P. U., Bettati, R. & Zhao, W. (2001), 'Netcamo: camouflaging network traffic for qos-guaranteed mission critical applications', *IEEE Transactions on Systems, Man, and Cybernetics* **Part A 31**(4), 253–265.

Gülcü, C. & Tsudik, G. (1996), Mixing E-mail with Babel, *in* 'Network and Distributed Security Symposium — NDSS '96', IEEE, San Diego, California, pp. 2–16.

Hazel, S. & Wiley, B. (2002), Achord: A variant of the chord lookup service for use in censorship resistant peer-to-peer publishing systems, *in* P. Druschel, M. F. Kaashoek & A. I. T. Rowstron, eds, 'Peer-to-Peer Systems, First International workshop, IPTPS 2002', Vol. 2429 of *LNCS*, Springer-Verlag, Cambridge, MA, USA.

Helsingius, J. (1996*a*), 'Johan helsingius closes his internet remailer', `http://www.penet.fi/press-english.html`.

Helsingius, J. (1996*b*), 'Johan helsingius gets injunction in scientology case privacy protection of anonymous messages still unclear', `http://www.penet.fi/injunc.html`.

Helsingius, J. (1996*c*), 'Temporary injunction in the anonymous remailer case', `http://www.penet.fi/injuncl.html`.

Hintz, A. (2002), Fingerprinting websites using traffic analysis., *in* Dingledine & Syverson (2003), pp. 171–178.

Ishai, Y., Kushilevitz, E., Ostrovsky, R. & Sahai, A. (2006), 'Cryptography from anonymity', Cryptology ePrint Archive, Report 2006/084. `http://eprint.iacr.org/`.

Jakobsson, M. (1998), A practical mix, *in* K. Nyberg, ed., 'Advances in Cryptology - EUROCRYPT '98', Vol. 1403 of *LNCS*, Springer-Verlag, Espoo, Finland, pp. 448–461.

Jakobsson, M. (1999), Flash Mixing, *in* 'Principles of Distributed Computing - PODC '99', ACM Press.

Jakobsson, M. & Juels, A. (2001), An optimally robust hybrid mix network, *in* 'Principles of Distributed Computing (PODC 2001)', ACM, Newport, Rhode Island, USA, pp. 284–292.

Jakobsson, M., Juels, A. & Rivest, R. L. (2002), Making mix nets robust for electronic voting by randomized partial checking, *in* D. Boneh, ed., 'USENIX Security Symposium', USENIX, San Francisco, CA, USA, pp. 339–353.

Jerichow, A., Müller, J., Pfitzmann, A., Pfitzmann, B. & Waidner, M. (1998), 'Real-Time MIXes: A Bandwidth-Efficient Anonymity Protocol', *IEEE Journal on Selected Areas in Communications* **16**(4), 495–509.

Jiang, S., Vaidya, N. H. & Zhao, W. (2000), Routing in packet radio networks to prevent traffic analsyis, *in* 'IEEE Information Assurance and Security Workshop'.

Katti, S., Katabi, D. & Puchala, K. (2005), Slicing the onion: Anonymous communication without pki, *in* 'HotNets'.

Kesdogan, D., Agrawal, D. & Penz, S. (2002), Limits of anonymity in open environments, *in* F. A. P. Petitcolas, ed., 'Information Hiding workshop (IH 2002)', Vol. 2578 of *LNCS*, Springer-Verlag, Noordwijkerhout, The Netherlands, pp. 53–69.

Kesdogan, D., Borning, M. & Schmeink, M. (2002), Unobservable surfing on the world wide web: Is private information retrieval an alternative to the mix based approach?, *in* Dingledine & Syverson (2003), pp. 224–238.

Kesdogan, D., Egner, J. & Büschkes, R. (1998), Stop-and-Go MIXes: Providing probabilistic anonymity in an open system, *in* D. Aucsmith, ed., 'Information Hiding workshop (IH 1998)', Vol. 1525 of *LNCS*, Springer-Verlag, Portland, Oregon, USA, pp. 83–98.

Kilian, J. & Sako, K. (1995), Receipt-free MIX-type voting scheme — a practical solution to the implementation of a voting booth, *in* L. C. Guillou & J.-J. Quisquater, eds, 'Advances in Cryptology (Eurocrypt 1995)', Vol. 921 of *LNCS*, Springer-Verlag, Saint-Malo, France, pp. 393–403.

Kissner, L., Oprea, A., Reiter, M. K., Song, D. X. & Yang, K. (2004), Private keyword-based push and pull with applications to anonymous communication., *in* M. Jakobsson, M. Yung & J. Zhou, eds, 'ACNS', Vol. 3089 of *Lecture Notes in Computer Science*, Springer, pp. 16–30.

Klonowski, M. & Kutylowski, M. (2005), Provable anonymity for networks of mixes, *in* 'Proceedings of Information Hiding Workshop (IH 2005)'.

Klonowski, M., Kutylowski, M., Lauks, A. & Zagorski, F. (2004), Universal reencryption of signatures and controlling anonymous information flow, *in* 'WARTACRYPT '04 Conference on Cryptology', Bedlewo/Poznan.

Klonowski, M., Kutylowski, M. & Zagrski, F. (2005,), Anonymous communication with on-line and off-line onion encoding, *in* P. Vojts, M. Bielikov, B. Charron-Bost & O. Skora, eds, 'SOFSEM 2005: Theory and Practice of Computer Science, 31st Conference on Current Trends in Theory and Practice of Computer Science', Lecture Notes in Computer Science, 3381, Liptovsk Jn, Slovakia, pp. 229–238.

Köpsell, S. & Hilling, U. (2004), How to achieve blocking resistance for existing systems enabling anonymous web surfing, *in* 'Proceedings of the Workshop on Privacy in the Electronic Society (WPES 2004)', Washington, DC, USA.

Kügler, D. (2003), An Analysis of GNUnet and the Implications for Anonymous, Censorship-Resistant Networks, *in* R. Dingledine, ed., 'Privacy Enhancing Technologies workshop (PET 2003)', Vol. 2760 of *LNCS*, Springer-Verlag, Dresden, Germany, pp. 161–176.

Levine, B. N., Reiter, M. K., Wang, C. & Wright, M. K. (2004), Timing attacks in low-latency mix-based systems, *in* A. Juels, ed., 'Proceedings of Financial Cryptography (FC '04)', Springer-Verlag, LNCS 3110.

Leyden, J. (2006), 'Anonymizer looks for gaps in great firewall of china', The Register.

Lu, T., Fang, B., Sun, Y. & Cheng, X. (2005), Performance analysis of wongoo system., *in* 'CIT', IEEE Computer Society, pp. 716–723.

Lu, T., Fang, B., Sun, Y. & Guo, L. (2005), Some remarks on universal re-encryption and a novel practical anonymous tunnel., *in* X. Lu & W. Zhao, eds, 'ICCNMC', Vol. 3619 of *Lecture Notes in Computer Science*, Springer, pp. 853–862.

Martin, D. & Schulman, A. (2002), Deanonymizing users of the safeweb anonymizing service, Technical Report 2002-003, Boston University Computer Science Department.

Martin, D. & Serjantov, A., eds (2005), *Privacy Enhancing Technologies, 4th International Workshop, PET 2004, Toronto, Canada, May 26-28, 2004, Revised Selected Papers*, Vol. 3424 of *Lecture Notes in Computer Science*, Springer.

Mathewson, N. (2005), 'Underhill: A proposed type 3 nymserver protocol specification', On-line. `http://svn.conuropsis.org/nym3/trunk/doc/nym-spec.txt`.

Mathewson, N. & Dingledine, R. (2004), Practical traffic analysis: Extending and resisting statistical disclosure, *in* 'Proceedings of Privacy Enhancing Technologies workshop (PET 2004)', LNCS.

Mazières, D. & Kaashoek, M. F. (1998), The Design, Implementation and Operation of an Email Pseudonym Server, *in* 'ACM Conference on Computer and Communications Security (CCS'98)', ACM Press, San Francisco, CA, USA, pp. 27–36.

Michels, M. & Horster, P. (1996), Some remarks on a receipt-free and universally verifiable mix-type voting scheme, *in* K. Kim & T. Matsumoto, eds, 'Advances in Cryptology (Asiacrypt '96)', Vol. 1163 of *LNCS*, Springer-Verlag, Kyongju, Korea, pp. 125–132.

Mitomo, M. & Kurosawa, K. (2000), Attack for flash MIX, *in* T. Okamoto, ed., 'Advances in Cryptology (Asiacrypt 2000)', Vol. 1976 of *LNCS*, Springer-Verlag, Kyoto, Japan, pp. 192–204.

Möller, B. (2003), Provably secure public-key encryption for length-preserving chaumian mixes, *in* M. Joye, ed., 'Topics in Cryptology CT-RSA 2003', Vol. 2612 of *LNCS*, Springer-Verlag, San Francisco, CA, USA, pp. 244–262.

Möller, U., Cottrell, L., Palfrader, P. & Sassaman, L. (2003), 'Mixmaster Protocol — Version 2', Draft.

Murdoch, S. J. & Danezis, G. (2005), Low-cost traffic analysis of Tor, *in* 'Proceedings of the 2005 IEEE Symposium on Security and Privacy', IEEE CS.

Neff, C. A. (2001), A verifiable secret shuffle and its application to e-voting, *in* P. Samarati, ed., 'ACM Conference on Computer and Communications Security (CCS 2002)', ACM Press, pp. 116–125.

Newman, R. (1997), 'The Church of Scientology vs. `anon.penet.fi`'. `http://www.xs4all.nl/~kspaink/cos/rnewman/anon/penet.html`.

Newman, R. E., Moskowitz, I. S., Syverson, P. & Serjantov, A. (2003), Metrics for traffic analysis prevention, *in* 'Privacy Enhancing Technologies Workshop', Dresden, Gernamy.

O'Connor, L. (2005), On blending attacks for mixes with memory, *in* 'Proceedings of Information Hiding Workshop (IH 2005)'.

Ogata, W., Kurosawa, K., Sako, K. & Takatani, K. (1997), Fault tolerant anonymous channel, *in* Y. Han, T. Okamoto & S. Qing, eds, 'Information and Communication Security, First International Conference (ICICS'97)', Vol. 1334 of *LNCS*, Springer-Verlag, Beijing, China, pp. 440–444.

Ohkubo, M. & Abe, M. (2000), A Length-Invariant Hybrid MIX, *in* T. Okamoto, ed., 'Advances in Cryptology (Asiacrypt 2000)', Vol. 1976 of *LNCS*, Springer-Verlag, Kyoto, Japan, pp. 178–191.

Ostrovsky, R. & III, W. E. S. (2005), Private searching on streaming data., *in* V. Shoup, ed., 'CRYPTO', Vol. 3621 of *Lecture Notes in Computer Science*, Springer, pp. 223–240.

Øverlier, L. & Syverson, P. (2006), Locating hidden servers, *in* 'Proceedings of the 2006 IEEE Symposium on Security and Privacy', IEEE CS.

Palfrader, P. (n.d.), 'Echolot: a pinger for anonymous remailers'. `http://www.palfrader.org/echolot/`.

Parekh, S. (1996), 'Prospects for remailers: where is anonymity heading on the internet?', *First Monday* **1**(2). On-line journal `http://www.firstmonday.dk/issues/issue2/remailers/`.

Park, C., Itoh, K. & Kurosawa, K. (1993), Efficient anonymous channel and all/nothing election scheme, *in* T. Helleseth, ed., 'Advances in Cryptology (Eurocrypt '93)', Vol. 765 of *LNCS*, Springer-Verlag, Lofthus, Norway, pp. 248–259.

Pfitzmann, A. & Hansen, M. (2001), Anonymity, Unobservability and Pseudonymity – A Proposal for Terminology, *in* H. Federrath, ed., 'Designing Privacy Enhancing Technologies', Springer-Verlag, LNCS 2009, pp. 1–9.

Pfitzmann, A., Pfitzmann, B. & Waidner, M. (1991), ISDN-mixes: Untraceable communication with very small bandwidth overhead, *in* W. Effelsberg, H. W. Meuer & G. Müller, eds, 'GI/ITG Conference on Communication in Distributed Systems', Vol. 267 of *Informatik-Fachberichte*, Springer-Verlag, pp. 451–463.

Pfitzmann, B. (1994), Breaking efficient anonymous channel, *in* A. D. Santis, ed., 'Advances in Cryptology (Eurocrypt '94)', Vol. 950 of *LNCS*, Springer-Verlag, Perugia, Italy, pp. 332–340.

Pfitzmann, B. & Pfitzmann, A. (1990), How to break the direct RSA-implementation of MIXes, *in* J.-J. Quisquater & J. Vandewalle, eds, 'Advances in Cryptology (Eurocrypt '89)', Vol. 434 of *LNCS*, Springer-Verlag, Houthalen, Belgium, pp. 373–381.

Rackoff, C. & Simon, D. R. (1993), Cryptographic defense against traffic analysis, *in* 'ACM Symposium on Theory of Computing (STOC'93)', ACM, pp. 672–681.

Rao, J. R. & Rohatgi, P. (2000), Can pseudonymity really guarantee privacy?, *in* 'Proceedings of the 9th USENIX Security Symposium', USENIX, pp. 85–96.

Raymond, J.-F. (2000), Traffic Analysis: Protocols, Attacks, Design Issues, and Open Problems, *in* H. Federrath, ed., 'Designing Privacy Enhancing Technologies', Vol. 2009 of *LNCS*, Springer-Verlag, pp. 10–29.

Reed, M. G., Syverson, P. F. & Goldschlag, D. M. (1998), 'Anonymous connections and onion routing', *IEEE Journal on Selected Areas in Communications* **16**(4), 482–494.

Reed, M. G., Syverson, P. F. & Goldschlag, D. M. (2001), 'Onion routing network for securely moving data through communication networks', United States Patent 6,266,704.

Reiter, M. & Rubin, A. (1998), 'Crowds: Anonymity for web transactions', *ACM Transactions on Information and System Security (TISSEC)* **1**(1), 66–92.

Reiter, M. & Wang, X. (2004), Fragile mixing, *in* 'Proceedings of the 11th ACM Conference on Computer and Communications Security (CCS 2004)', ACM Press.

Rennhard, M. & Plattner, B. (2002), Introducing MorphMix: Peer-to-Peer based Anonymous Internet Usage with Collusion Detection, *in* 'Workshop on Privacy in the Electronic Society (WPES 2002)', Washington, DC, USA.

Sassaman, L., Cohen, B. & Mathewson, N. (2005), 'The pynchon gate: a secure method of pseudonymous mail retrieval', *Proceedings of the 2005 ACM workshop on Privacy in the electronic society* pp. 1–9.

Serjantov, A. (2002), Anonymizing censorship resistant systems, *in* P. Druschel, M. F. Kaashoek & A. I. T. Rowstron, eds, 'International Peer-To-Peer Systems workshop (IPTPS 2002)', Vol. 2429 of *LNCS*, Springer-Verlag, Cambridge, MA, USA, pp. 111–120.

Serjantov, A. (2004), On the Anonymity of Anonymity Systems, PhD thesis, University of Cambridge.

Serjantov, A. & Danezis, G. (2002), Towards an information theoretic metric for anonymity, *in* R. Dingledine & P. Syverson, eds, 'Privacy Enhancing Technologies workshop (PET 2002)', Vol. 2482 of *LNCS*, Springer-Verlag, San Francisco, CA, USA, pp. 41–53.

Serjantov, A., Dingledine, R. & Syverson, P. (2002), From a trickle to a flood: Active attacks on several mix types, *in* F. A. P. Petitcolas, ed., 'Information Hiding workshop (IH 2002)', Vol. 2578 of *LNCS*, Springer-Verlag, Noordwijkerhout, The Netherlands, pp. 36–52.

Serjantov, A. & Sewell, P. (2003), Passive attack analysis for connection-based anonymity systems, *in* 'European Symposium on Research in Computer Security (ESORICS 2003)', Gjovik, Norway.

Shannon, C. (1948), 'A mathematical theory of communication', *The Bell System Technical Journal* **27:379–423**, 623–656.

Sherwood, R., Bhattacharjee, B. & Srinivasan, A. (2002), P5: A protocol for scalable anonymous communication, *in* 'IEEE Symposium on Security and Privacy', IEEE Computer Society, Berkeley, California, USA, p. 58.

Shmatikov, V. (2002), Probabilistic analysis of anonymity, *in* 'Computer Security Foundations workshop (CSFW-15 2002)', IEEE Computer Society, Cape Breton, Nova Scotia, Canada, pp. 119–128.

Singer, M. (2001), 'CIA Funded SafeWeb Shuts Down', `http://siliconvalley.internet.com/news/article.php/3531_926921`.

Sun, Q., Simon, D. R., Wang, Y.-M., Russell, W., Padmanabhan, V. N. & Qiu, L. (2002), Statistical identification of encrypted web browsing traffic., *in* 'IEEE Symposium on Security and Privacy', pp. 19–30.

Syverson, P. F., Tsudik, G., Reed, M. G. & Landwehr, C. E. (2000), Towards an analysis of onion routing security, *in* H. Federrath, ed., 'Designing Privacy Enhancing Technologies', Vol. 2009 of *LNCS*, Springer-Verlag, Berkeley, CA, USA, pp. 96–114.

Tabriz, P. & Borisov, N. (2006), Breaking the collusion detection mechanism of morphmix, *in* G. Danezis & P. Golle, eds, 'Privacy Enhancing Technologies workshop (PET 2006)', LNCS, Springer-Verlag.

Timmerman, B. (1997), A security model for dynamic adaptive traffic masking, *in* 'New Security Paradigms Workshop', ACM, Langdale, Cumbria, UK, pp. 107–115.

Timmerman, B. (1999), Secure dynamic adaptive traffic masking, *in* 'New Security Paradigms Workshop', ACM, Ontario, Canada, pp. 13–24.

Venkatraman, B. R. & Newman-Wolfe, R. E. (1994), Performance analysis of a method for high level prevention of traffic analysis using measurements from a campus network, *in* 'Proceeding of the IEEE/ACM Tenth Annual Computer Security Applications Conference', IEEE CS Press, Orlando, FL, pp. 288–297.

verlier, L. & Syverson, P. (2006), Valet services: Improving hidden servers with a personal touch, *in* G. Danezis & P. Golle, eds, 'Privacy Enhancing Technologies workshop (PET 2006)', LNCS, Springer-Verlag.

Waidner, M. & Pfitzmann, B. (1989), The dining cryptographers in the disco — underconditional sender and recipient untraceability with computationally secure serviceability, *in* J.-J. Quisquater & J. Vandewalle, eds, 'Advances in Cryptology (Eurocrypt '89)', Vol. 434 of *LNCS*, Springer-Verlag, Houthalen, Belgium, p. 690.

Walton, G. (2001), 'Chinas golden shield: corporations and the development of surveillance technology in the Peoples Republic of China', *Montreal: International Centre for Human Rights and Democratic Development, URL (consulted 29 October 2001): http://www. ichrdd. ca/frame. iphtml* .

Wang, X., Chen, S. & Jajodia, S. (2005), Tracking anonymous peer-to-peer voip calls on the internet, *in* 'Proceedings of the ACM Conference on Computer and Communications Security', pp. 81–91.

Wang, X. & Reeves, D. S. (2003), Robust correlation of encrypted attack traffic through stepping stones by manipulation of interpacket delays., *in* S. Jajodia, V. Atluri & T. Jaeger, eds, 'ACM Conference on Computer and Communications Security', ACM, pp. 20–29.

Wikström, D. (2002), How to break, fix, and optimize "optimistic mix for exit-polls", Technical Report T2002-24, Swedish Institute of Computer Science, SICS, Box 1263, SE-164 29 Kista, SWEDEN.

Wikström, D. (2003*a*), Elements in $Z_p^* \backslash G_q$ are dangerous, Technical Report T2003-05, Swedish Institute of Computer Science, SICS, Box 1263, SE-164 29 Kista, SWEDEN.

Wikström, D. (2003*b*), Four practical attacks for "optimistic mixing for exit-polls", Technical Report T2003-04, Swedish Institute of Computer Science, SICS, Box 1263, SE-164 29 Kista, SWEDEN.

Wright, M., Adler, M., Levine, B. N. & Shields, C. (2002), An analysis of the degradation of anonymous protocols, *in* 'Network and Distributed Security Symposium (NDSS '02)', San Diego, California.

Wright, M., Adler, M., Levine, B. N. & Shields, C. (2003), Defending anonymous communication against passive logging attacks, *in* 'IEEE Symposium on Security and Privacy', IEEE Computer Society, Berkeley, CA, USA, p. 28.

Zhu, Y. & Bettati, R. (2005), Un-mixing mix traffic, *in* 'Proceedings of Privacy Enhancing Technologies workshop (PET 2005)'.

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