How to Bypass Two Anonymity Revocation Schemes

George Danezis[1] and Len Sassaman[2]

[1] Microsoft Research,
Cambridge, UK
gdane@microsoft.com
[2] K.U. Leuven, ESAT/COSIC,
Kasteelpark Arenberg 10,
B-3001 Leuven-Heverlee, Belgium.
Len.Sassaman@esat.kuleuven.be

Abstract. In recent years, there have been several proposals for anonymous communication systems that provide intentional weaknesses to allow anonymity to be circumvented in special cases. These anonymity revocation schemes attempt to retain the properties of strong anonymity systems while granting a special class of people the ability to selectively break through their protections. We evaluate the two dominant classes of anonymity revocation systems, and identify fundamental flaws in their architecture, leading to a failure to ensure proper anonymity revocation, as well as introducing additional weaknesses for users not targeted for anonymity revocation.

1 Introduction

Anonymous communication systems have been studied extensively since David Chaum introduced the mix in 1981 [5]. Their principal aim is to hide the fact that Alice is communicating with Bob from network adversaries or corrupt nodes in the anonymity-providing system. Practical anonymous communication systems have been proposed and fielded for email [25, 13] and web-browsing [2, 32]. They are based on intermediate nodes relaying the communication and hiding the correspondences between their inputs and outputs to obscure who is talking with whom. An extensive survey of anonymous communication channels and their properties is provided in [12].

Many approaches have also been proposed to mitigate the potential for abuse of anonymous communications. These approaches fall into two main classes. The first one, based on blacklisting [21], is respectful of users' anonymity and empowers service providers to block abusive users without ever finding their true identity. This approach is similar to the blacklisting of anonymous credentials [33, 4]. Another form of blacklisting is used by Mixmaster; as senders of abusive content cannot be identified, recipients of abusive content who do not wish to receive mail from the anonymous remailer network can submit their email addresses to be blocked, so that they will not receive unwanted communication in the future.

This technique has been implemented in Mixmaster on a per-remailer basis with support for the network-wide "Remailer Abuse Blacklist" (RAB), which ensures the silencing of abusive messages regardless of the remailer used (as long as it is a participant in the RAB.)

The second approach is based on anonymity revocation or anonymity escrow, and allows a collection of authorities to revoke the anonymity of a user associated with a particular communication. Revocation has fundamentally different aims from blacklisting, and can be applied to tracing arbitrary messages between (even consenting) users to prevent covert communication.

Two lines of research have been developing in engineering revocation mechanisms into anonymity systems. The first family of systems is by Díaz and Preneel [16, 8, 9] (DP), and the second and latest by Köpsell, Wendolsky and Federrath [23] (KWF). Given the similarity in their approach, we will examine in detail the latest KWF system [23], and show in the discussion how our results are applicable to the first set of systems.

The key feature of the DP and KWF anonymity revocation mechanisms is that they "wrap-around" any anonymity system without modifying its internal functioning. Decoupling revocation from the anonymity channel is a wise design choice. It recognizes that building anonymous communication systems requires a careful balance between engineering and security, and adding more requirements into the core of the designs may lead to unsafe systems. This approach also adds generality, since, in theory at least, it would allow any secure anonymity system to be easily modified to include a revocation mechanism. A secondary design aim of the revocation mechanisms is to retain the same set of trust assumptions and security properties as the underlying anonymity systems.

In this work, we demonstrate that both revocation mechanisms are not as effective as believed, and some forms of anonymous communication are always possible despite them. The scheme's independence from any particular anonymous channel turns into a weakness: as we show there exists no concrete practical channel to instantiate it securely. Even a single party within the anonymization infrastructure, adverse to the revocation protocol, is sufficient to help senders bypass it to achieve anonymity without revocation, and without a significant reduction in the quality of anonymity.

Furthermore, for onion-routing systems, the proposed architectures could lead to a reduction in security even when the revocation mechanism is not exercised. In most cases, the grafting of the revocation mechanism opens systems to Denial of Service attacks and heightens the risk of censorship.

This paper is organized as follows: the basic architecture of the KWF system is outlined in Section 2, and the key techniques necessary to bypass it in Section 3. Section 4 enumerates specific instances of KWF with concrete anonymous channels and describes how to bypass revocation in each case. Concerns about weakening the security of the anonymous channels by adding KWF to them are presented in Section 5, and questions about the desirability of revocation systems are discussed in Section 6. KWF and DP are contrasted in Section 7, and conclusions on our work are offered in Section 8.

2 The Internals of the KWF Scheme

The KWF [23] mechanism is a generic construction adding revocation capabilities to any anonymous communication channel. The aim of the KWF scheme is to not interfere with any of the security properties provided by the anonymous channel unless the anonymity of the communication is to be revoked. In such cases, the revocation authorities should reliably learn the identity of the sender or initiator of the revoked anonymous communication. This property should hold under the same security assumptions guaranteeing anonymity made by the underlying channel.

The KWF scheme implements mechanisms for revocation by requiring users to perform special steps before sending an anonymous message, as well as examining all messages output from the anonymous channel to the world. It relies on threshold group signatures for its security; using those, a member of a group can sign a message identifying himself as a member of the group without leaking any additional information about his identity. However, a quorum of group managers can invoke a revocation procedure to uncover a user's identity if some abuse is detected.

The KWF scheme also includes special features that allow operators not to learn any information about the identity of the traced user. These are, to a large extent, irrelevant to our attacks. Therefore, we shall not examine them in detail. We refer the reader to the full scheme [23] for further details.

Aside from the parties taking part in the anonymous communications, as well as the parties facilitating anonymization, the KWF scheme relies on some additional entities. Since the scheme uses group signatures, an entity is designated to be the *group manager*, that has the power to trace a group signature to a specific pseudonym. A *third party* is trusted to check sender's real identities and correctly package cryptographic tokens based on them. A *verifier* is entrusted with verifying signatures and censoring invalid messages. Finally, some abstract *authorities* are authorized to learn the senders of revoked messages – this party possess by design (as opposed to being due to an accidental weakness in the system) the ability to remove any user's anonymity, as well as ensure that node operators comply with the protocol.

The skeleton of the KWF protocol proceeds as follows:

1. *Login.* A user wishing to send an anonymous message first logs in to a third party and acquires a signed 'revocation token'. This token is a ciphertext of his real identity (this may be a strong identity, derived from a public-key certificate, or simply the IP address of the user) encrypted using a threshold crypto-system. The user becomes a member of the group that is allowed to send messages through the channel. The third party gives the user the secret key to prove membership to the group, using a (revocable) group signature scheme.
2. *Sending.* The user signs his message using his group signature key and packages it cryptographically, as appropriate for the specific anonymous channel. He then sends the message, or performs whatever action is necessary to execute the anonymous communication channel protocol.

3. *Checking.* Once the message is output from the anonymous channel, it is given to a verifier. The verifier checks the group signature on the message; if it is not valid, the message is discarded. If the signature is valid, the message is forwarded to the intended recipient of the message.
4. *Revocation.* In case the message offends some policy, the revocation procedure is set in motion. The group signature associated with the message is provided to the group manager that traces it to a particular pseudonym. The pseudonym is used to retrieve the 'revocation token', and the real identity of the sender is retrieved by threshold decryption, performed by some third parties, and given to the authorities.

An important objective of the KWF scheme is to not modify the trust model of the anonymous channel. For this reason, the third parties necessary to perform the threshold decryption and provide the identity of the user to the authorities are chosen to be the same third parties that facilitate the anonymization of messages, i.e., the anonymization infrastructure itself. The stated aim is for the revocation protocol to be secure under the same conditions as anonymity is secured: when a threshold of honest servers exists in the network.

Our attacks against the KWF scheme, and the closely-related DP scheme, show that the protocols do not meet this objective. It is possible to bypass the revocation mechanisms and achieve strong anonymity if even a single participant in the anonymity infrastructure is unwilling to follow the revocation protocol. For some common choices of anonymous channels, it is even possible to bypass the revocation mechanisms without the help of any insider.

3 Outline of the Bypass Attacks

The key assumption on which the KWF and DP schemes base their security is that there can be no leakage of information from inside the channel to the world unless it passes through the verification step. In the KWF design, the anonymous channel is presented as a pipe with a clear entry and exit point, while in the DP design, the mixes are assumed to be unable to misbehave.

In practice, anonymous channels are complex multi-party protocols involving many often-untrusted participants who are in a position to learn a lot of information about the messages in transit. Engineering anonymous channels devoid of covert channels has never been a core objective of designers. The idea that the verifier is able to 'catch' all message flows from the network to the outside world is particularly hard to implement when the sender is *intentionally* trying to leak information through an accomplice that is part of the infrastructure.

Our attack only modifies the *Sending* step of the KWF protocol. A user correctly logs into the third party and acquires the appropriate credentials to use the anonymity system. However, he does not sign the message that he wishes to send. Instead, he packages it in such a way as to take advantage of a single accomplice in the infrastructure that will leak the message to the world (or to co-conspirators) without first presenting it to the verifier. We shall examine in

detail, in the next section, how this can be done in the most common anonymizing channels.

Why does the attack work in general? Anonymous channels have been designed to be *incentives compatible*. They rely on the parties that will benefit from the anonymity properties, the senders in our case, to package their messages in such a way as to leak no information about their content or destinations. With the exception of anonymous channels designed for elections,[3] there is no mechanism preventing users from packaging their messages in a way that reveals their contents to arbitrary third parties.

The KWF design provides incentives for users to bypass the verifier. It is trivial in almost all anonymity designs to make use of a corrupt insider (and often even an observer) to leak their messages out of the channel without being subject to the verifier's scrutiny.

It is important to understand the role of the insider that enables non-revocable anonymous communications: the only service they provide is leaking the message to the outside world without vetting it through the verifier. As such, insiders only facilitate a covert channel, but are not required to provide any anonymity: the use of the anonymous channel, and the otherwise honest participants, already provides this. Therefore the insiders do not need to act as anonymizing relays, but merely as exits from the channel.

It is not necessary for the corrupt insider to have any details about the real identity of the sending user, and it is impossible to obtain any additional information by observing any of its internal state. Therefore a compromise of the insider nodes does not lead to a compromise of the sender's identities.

4 Bypassing specific KWF-* mechanisms

To illustrate our attacks, we will show how a sender can use unintended covert channels in most anonymity systems to leak messages to others without being subject to the verifier's censorship.

We will have to show in all cases that (1) the message benefits from the anonymity properties of the channel (without the corrupt insiders contributing to the anonymity); (2) that a single corrupt insider is sufficient to bypass the system; (3) that the message can be leaked in a way that does not arouse suspicion. We shall denote the instances of the KWF systems as KWF-*, where the '*' denotes the specific anonymous channel used by the system.

KWF-cascades. The KWF is first presented in terms of mix cascades [2, 18], so we should start by demonstrating that a single dishonest member of the mix cascade can bypass the verifier.

Mix cascades anonymize messages by relaying them through a predetermined and fixed set of intermediary nodes. The messages are encoded in multiple

[3] It is important that election systems provide a method for the voter to verify that her vote is counted, but prevent the voter from proving how she voted to a third party, to achieve coercion resistance.

layers of encryption, and each intermediary strips a layer before passing the message along to the next mix. With n mixes in the cascade, the message leaving the sender should be encrypted under the public keys of all mixes and look like:

$$\mathrm{M}' = E_{K_1} E_{K_2} \dots E_{K_n}(A, M) \tag{1}$$

Where $E_k(\cdot)$ denotes encryption under the key k, A the final address of the message and M the message itself. M' is sent to the first node N_1, where it is decrypted and forwarded to the next node.

Assume that the single node N_j and the sender are collaborating to bypass the revocation mechanism. The sender simply packages the message as:

$$\mathrm{M}' = E_{K_1} E_{K_2} \dots E_{K_j} E_{K_{\text{secret}}}(A, M) \tag{2}$$

The key K_{secret} can be a shared key between the sender and node N_j. The message will be correctly relayed until node N_j. At this point, it will appear in the clear to node N_j, which can leak it to any third party.[4] The node N_j should then forward the ciphertext $E_{K_{\text{secret}}}(A, M)$ along, to make its observable operation indistinguishable from an honest node. The message arriving at the final node will be indistinguishable from a random plaintext, and will be discarded by the verifier as not having a valid signature.

The double encryption of the message received by N_j provides compulsion resistance. The mix is able to follow the protocol unaltered and decrypt the message first with its public key K_j. Then it can covertly check whether the message is to be leaked, by checking on whether it decrypts correctly with the key K_{secret}. Yet if an adversary captures the node, and compels it to reveal its secrets, there is no way to prove that any key exist beyond the first one. To achieve this messages encrypted under K_{secret} should be indistinguishable from those destined to the next stage of mixing – a property that is simple to implement.

Compulsion resistance protects the collaborating mix from reprisal, but is not necessary to maintain anonymity. In case both keys N_j and K_{secret} are leaked the message still benefits from the anonymity provided by the mixes $N_1, \dots, N_{j-1}$.

Since the message contains no signature to revoke, it is not possible to trace it back when the verifier receives it. Furthermore, if the message has gone through at least a single honest node before reaching the node N_j, it has benefited from the anonymity of the channel without being traceable. It is also clear that a single N_j is sufficient to leak the messages, and that that sender has a high bandwidth channel to leak and anonymize messages. (The bandwidth is at least as high as if it were using the legitimate system.)

We conclude that for the KWF-cascades system, the security goal that the revocation mechanisms should work if there is a threshold of honest users does not hold.

[4] Of course, node N_j may very well be the intended recipient, with no further dissemination of the message necessary.

KWF-mix. Mix systems [5] are sets of routers that decrypt and forward messages to a designated address. Multiple mixes are chained together to form paths, over which messages are relayed. As with cascades, the cryptographic format of messages upon injection into the mix network is:

$$\mathrm{M} = E_{K_1}(N_2, E_{K_2}(\ldots(N_n, E_{K_n}(A, M)))) \tag{3}$$

Messages are encrypted using multiple layers with the public keys of intermediate mix nodes. Unlike the formatting of messages for mix cascades, the address of the next node is included in the encrypted envelope to facilitate routing.

Mix systems can trivially be used to implement the bypass attack by including in the path a single corrupt mix that will leak the messages to its final destination. A message can be formatted as:

$$\mathrm{M} = E_{K_1}(N_2, E_{K_2}(\ldots(N_j, E_{K_j}E_{K_{\mathrm{secret}}}(A, M)))) \tag{4}$$

Node N_j is dishonest, decrypts the message and forwards it to its final destination without checking its signature or mediating the communication through the verifier. Indistinguishability from honest behaviour can still be achieved. If node N_j comes under compulsion, it can reveal its private key, but without revealing K_{secret}, it is impossible to distinguish its operation from the honest nodes. Messages can still contain valid routing information for a subsequent path [11], making it impossible for an adversary to distinguish the node that leaks messages.

As with cascades, a single dishonest node is able to bypass the revocation mechanism and allow anonymous communication to take place. Even if node N_j is under passive surveillance, the message has benefited from the anonymity offered by nodes $N_1 \ldots N_{j-1}$. Therefore, for general mix systems, the KWF revocation mechanism does not meet its security goals.

Other considerations. Other covert channels are available in some mix systems, allowing for covert communication even without the need for a corrupt party. Proposals to make mix networks robust assume that inputs and outputs of relayed messages are published on a public bulletin board [20, 26]. In such designs, Alice and Bob can communicate covertly by sharing a key and encoding the message so that it exits some mix "in the clear".

Requiring mix systems not to publish any information would make universal verifiability of delivery impossible to implement using efficient techniques, and would make such networks insecure against denial-of-service attacks [3].

KFW-or/tor. Onion routing architectures [32, 17] employ layered encryption and paths over networks of routers, and are architecturally very similar to mix networks. As a result, the same techniques can be used to route the stream through a dishonest node that leaks information to the outside world without checking signatures or presenting them to the verifier.

While architecturally related to mixing, onion routing defends against a very different threat model, and it is likely that the verifier will be able to mount de-anonymization attacks if it relays and checks all streams of traffic. This

is due to the onion routing being susceptible to passive attacks, while mix networks should be secure against a global passive adversary. We discuss this further in section 5.

KFW-buses. Buses [1] is a broadcast anonymization protocol. Nodes arrange themselves in one or multiple paths, over which 'buses' travel. Buses are bit-strings containing multiple messages that are encrypted and re-encrypted as the 'bus' is relayed by each node, making it impossible to tell at what node they were introduced or removed. Messages are encrypted under the secret key of the final recipient, which detects them by trial decryption.

The peer-to-peer nature of buses makes it difficult to implement the KWF scheme, since only the final recipient gets to see and decrypt the message, not any other intermediary. Therefore, the KWF architecture cannot be applied to buses, since covert channels to the final recipients are intrinsic to the security of the scheme.

The obvious modification to accommodate KWF would be to address all messages to the verifier, who would then forward them to their respective receivers. This architecture would be equivalent to using the verifier as a single-hop proxy, since all the messages would simply be encrypted under its public key. This falls short of the original security properties of buses, which offer perfect receiver anonymity.

KWF-PIR. Recently, proposals for the use of Information Theoretic Private Information Retrieval [7] in anonymous communications have been made [30, 19]. In Information Theoretic PIR, a set of databases (all containing an identical data set) are queried such that it is impossible for an adversary to determine what information is being requested by the user making the query unless all databases being queried are in collusion. In the proposed anonymity systems built on this form of PIR, the PIR database is used to store messages for pseudonyms. The KWF-* system is ill-suited for the architecture thus far proposed, though one can imagine a KWF design for such systems. There exists a significant problem with such a system, however: while a KWF-PIR solution could force the database operators to collude and reveal the queries made by a user so that a known IP address is linked to an unknown pseudonym, the reverse would require stripping the anonymity of the system's users until a match was found, violating the privacy of many legitimate users in the process.

Furthermore, anonymous covert channels exist in systems where large amounts of random data are passed between different entities [28], and a KWF-PIR solution would not prevent anonymous communications if a communicating party operated part of the infrastructure.

There are two protocols in the literature, Crowds [29] and Dining Cryptographers' networks (DC-nets) [6], for which the KWF scheme would be applicable.

DC-nets are information-theoretically secure and devoid of covert channels. It is therefore possible to imagine all users contributing their inputs to the centralized verifier server that combines, shares, and outputs the final message only if it is valid. This centralized architecture would be secure, but would scale poorly.

Scaling problems, as well as sensitivity to denial of service, are so prevalent in DC-nets that they are not used in practice.

Crowds [29] is a simple pass-the-parcel mechanism, in which nodes probabilistically relay messages in the clear before forwarding them to their final destination. A naive implementation of the KWF mechanism could be bypassed: the sender simply does not include a ring signature into the sent messages. Given the limited control the sender has over the routing of messages, delivery is only possible if the message reaches a collaborating node. Therefore the attack has to rely on a large fraction of collaborating nodes being part of the Crowds network – a much stronger assumption than those made so far.

The KWF mechanism could also be easily modified to be robust even against those attacks. The KWF protocol would have to be augmented to ensure that all honest Crowds nodes check the validity of the signature of all messages they relay and report those nodes that forward messages without valid signatures. This is possible in Crowds since relayed messages are visible to nodes in clear. In practice, Crowds is not in use because of the weak anonymity properties it provides, which also enable checking the KWF signatures at each step.

5 Unintentionally Introduced Weaknesses in the KWF Scheme

The KWF scheme aims to "allow for deanonymisation without weakening the general trust model of an anonymity service." Yet through the introduction of a new infrastructure component, "the verifier", users of the system become more susceptible to attacks by an adversary not in collusion with the operators of the revocation system.

End-to-end traffic analysis. The verifier greatly reduces the difficulty of performing *end-to-end traffic analysis attacks*, by serving as a convenient single end-point in a fixed location. The fact that all traffic leaving the anonymity system can be observed, and timed, at the verifier facilitates multiple attacks described in the literature.

Many of the statistical disclosure attacks are made easier in this architecture [14, 24], since the message recipients are readily available to the verifier. Onion-routing security is greatly affected, since the verifier can collaborate with any entry node to correlate the low-latency streams of traffic to trace them [34, 10]. In general, the KWF architecture interacts poorly with anonymity systems providing security against a partial adversary only, since it requires the verifier to act as a centralized global observer. Classic active attacks such as the $(n-1)$ attack [31], in which an adversary injects a single message in a mix along with many of their own messages, are much easier to perform against the whole network by the verifier.

Furthermore, the effectiveness of dummy traffic is greatly reduced. Network-generated cover traffic cannot contain valid ring signatures to get past the

verifier and reach the final recipients. Therefore such dummy messages, injected by mixes to thwart traffic analysis, are easily distinguishable from user generated messages exiting the network, rendering them useless.

Denial of service. The verifier acts as a computational and communication bottleneck, since it has to inspect every message. That further exposes the system to denial-of-service attacks. Even a single rogue node, or client, can create a near-limitless number of messages with syntactically-correct, semantically-invalid signatures, to force the verifier to perform a verification of the signature – an expensive public-key operation. The anonymity of this node would be protected by the operation of the network, and it would be difficult to uncover and stop it.

Censorship. The verifier can easily be turned into a censor. Instead of attempting to trace messages violating an arbitrary policy, it can simply silently drop them. This feature of the revocation protocol goes against the latest attempts to make anonymous communications more robust against censorship [22]. The login servers could also act as censors by not providing signature keys to selected individuals.

This form of denial-of-service in anonymity systems does not simply impact availability, but can also be used as a tool to decrease anonymity. Therefore, the ability of the verifier to drop messages allows an adversary to increase its chance of tracing a message [3].

Anonymity-set reduction. Multiple anonymity revocation orders may be issued against the same anonymity set, thus weakening the anonymity provided to "legitimate" users. This is a concern that any user of such a system could justifiably hold, for they have no control over the identity or actions of the other members of their anonymity set (which in most systems is a random set of users highly correlated to the time in which the system is used.) If a significant portion of the traffic is deanonymized by the revocation mechanisms, the legitimate users will be put at risk of identification as well.

6 Additional Concerns

The KWF scheme takes great pains to ensure that multiple parties (the anonymization node operators) are involved in revoking anonymity. The intent is to demonstrate that the individual node operators all cooperate in providing the revocation of the anonymity their services provide; however, since participation in such a system is presumed to be compulsory, the nodes participating in such a system must be considered "in collusion", or at best "under compulsion", from the standpoint of the existing anonymity threat models. This is especially true in the KWF system, where a mechanism is employed to hide the identity of the traced user from the operators, making it impossible to judge the legitimacy or proportionality of any revocation order before complying.

The discussion of jurisdiction issues with regard to the system presented in the KWF and DP papers is also missing. Backdoored anonymity systems are

unlikely to be preferred in jurisdictions where they are not mandatory, thus resulting in separate networks for different jurisdictions, eliminating many of the benefits of jurisdictional arbitrage. This leaves many unanswered questions regarding which law enforcement agencies and judges are empowered to order the revocation of a user's anonymity, and presents difficult problems with regard to maintaining a single cohesive anonymity-set.

A system which has a known weakness as part of its design is likely to reduce the overall trust of the system—and for good reasons, as we have demonstrated. From a deployment point of view, it is unclear why users would use a revocable anonymity system instead of the easily accessible global anonymity solutions that are free of backdoors. An attempt could be made to impose such restrictions on the Internet as a whole, or put up country-level censorship systems (such as those used in China and Iran [15]) to prevent the use of true anonymity systems, but both of those approaches present their own problems beyond the scope of this paper.

The designers of the KWF scheme indicate that the system is to be used to revoke the identities of users sending to "suspicious addresses" or visiting "suspicious websites." Besides the obvious question of "what makes a recipient suspicious?" there can be legitimate concerns that "suspicious" websites may be created for the purpose of entrapment, to learn the identity of a user because of legitimate past communications. An attack whereby a user is tricked into visiting a given "suspicious website" could easily be implemented through the use of hidden frames in an unrelated site. E.g., if an attacker knows that website A is "suspicious", the attacker could embed an element of website A in an innocuous website B or in an HTML email, causing the user to unwittingly visit the "suspicious" website. This could lead to the targeted deanonymization of honest users for reasons other than those stated in an official warrant, and even allows for corruption of this process by non-governmental agents.

7 Discussion of the DP Design

The DP class of revocable anonymity solutions has its origins in the APES European Union project. Part of the project is intended to further develop anonymous communications, and also includes methods for the "control" of anonymous communications [9].

We refer to the DP solutions as a "class" of revocation technologies, because there are subtle differences between proposals. We focus on the most recent proposal published by Díaz and Preneel [16]. The DP scheme is harder to evaluate than the KWF scheme, because only a high-level design is proposed. Many difficult questions of security emerge when the gaps in the design are filled in.

The DP scheme shares some elements in common with the KWF scheme: a *certification authority* that holds users' identities in escrow, *judges* who are empowered to revoke the system, and a *credential* that is associated with any communication sent through the network. There are some basic problems with the DP approach not found in the KWF-* system, following from the fact that

the identity credential is not cryptographically bound to the data being communicated. Hence, a mix can "frame" a user by associating the credential of a "suspicious" user with a legitimate user's traffic. This may be done specifically to harm that user, or by a rogue mix wishing to associate a valid credential with communication from a user other than the owner of that credential, to conceal the identity of the non-credentialed user.

DP's key difference from the KWF scheme is that the verifier is not centralized; instead, all exit nodes from the anonymity network are entrusted to check signatures for validity before forwarding messages. The Bypass attack can successfully be applied, and becomes trivial if even a single exit node is corrupt, as for the KWF bypass attacks described in Section 3. Other security issues relating to the centralization of the verifier in KWF, such as denial of service or traffic analysis attacks, are not so severe due to this decentralized approach.

8 Conclusions

We find the two proposals for "conditional anonymity" to be a significant departure from the strength and protection assurances in traditional, non-backdoored anonymity systems. The systems studied are ineffective at providing revocation against users wishing to engage in covert communications, and additionally introduce weaknesses that may compromise legitimate users.

From a technical point of view, no practical, deployed, anonymous communication channel fulfils the goals of the KWF or DP schemes, to provide mechanism-independent anonymity revocation. Instead, they allow single insiders, or even eavesdroppers, to leak anonymous messages out of the network, allowing unrevocable communications to take place. Furthermore, the KWF-* mechanisms have the potential to reduce anonymity through the introduction of a single verifier entity that mediates all output messages or streams from the network. The verifier can mount traffic analysis attacks, be subject to denial of service, or censor messages to impact availability or facilitate tracing.

From a deployment as well as a policy point of view, we believe revocation mechanisms to be ill-conceived: it is unclear why any operator or user would choose to use them when alternatives are available, and the proposed schemes systematically fail to ensure any security against the abuse of the revocation interfaces. Ironically, by introducing a strong framework for node collusion to achieve revocation, the amount of trust a given user is likely to place in the system as a whole is reduced. Yet, these issues go beyond the strictly technical focus of this paper.

Acknowledgments

We would like to thank David Chaum, Roger Dingledine, Nick Mathewson, and Steven Murdoch for their input on this subject, and Meredith L. Patterson for feedback on an early draft of this paper. Apu Kapadia spent considerable time

and effort providing suggestions that greatly improved the style and essence of this work.

The work of Len Sassaman was supported in part by the Concerted Research Action (GOA) Ambiorics 2005/11 of the Flemish Government, by the IBBT (Flemish Government), by the IAP Programme P6/26 BCRYPT of the Belgian State (Belgian Science Policy), and by the EU within the PRIME Project under contract IST-2002-507591.

References

1. Amos Beimel and Shlomi Dolev. Buses for anonymous message delivery. *Journal of Cryptology*, 16(1):25–39, 2003.
2. Oliver Berthold, Hannes Federrath, and Stefan Köpsell. Web MIXes: A system for anonymous and unobservable Internet access. In H. Federrath, editor, *Proceedings of Designing Privacy Enhancing Technologies: Workshop on Design Issues in Anonymity and Unobservability*, pages 115–129. Springer-Verlag, LNCS 2009, July 2000.
3. Nikita Borisov, George Danezis, Prateek Mittal, and Parisa Tabriz. Denial of service or denial of security? In Ning et al. [27], pages 92–102.
4. Stefan Brands, Liesje Demuynck, and Bart De Decker. A practical system for globally revoking the unlinkable pseudonyms of unknown users. In Josef Pieprzyk, Hossein Ghodosi, and Ed Dawson, editors, *ACISP*, volume 4586 of *Lecture Notes in Computer Science*, pages 400–415. Springer, 2007.
5. David Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM*, 24(2):84–88, 1981.
6. David Chaum. The dining cryptographers problem: Unconditional sender and recipient untraceability. *Journal of Cryptology*, 1:65–75, 1988.
7. Benny Chor, Oded Goldreich, Eyal Kushilevitz, and Madhu Sudan. Private information retrieval. In *Proceedings of the IEEE Symposium on Foundations of Computer Science*, pages 41–50, 1995.
8. Joris Claessens, Claudia Díaz, Caroline Goemans, Bart Preneel, Joos Vandewalle, and Jos Dumortier. Revocable anonymous access to the internet. *Journal of Internet Research*, 13(4):242–258, 2003.
9. Joris Claessens, Claudia Díaz, Svetla Nikova, Bart De Win, Caroline Goemans, Mieke Loncke, Vincent Naessens, Stefaan Seys, Bart De Decker, Jos Dumortier, and Bart Preneel. Technologies for controlled anonymity. APES deliverable D10, Katholieke Universiteit Leuven, 2003.
10. George Danezis. The traffic analysis of continuous-time mixes. In *Proceedings of Privacy Enhancing Technologies workshop (PET 2004)*, volume 3424 of *LNCS*, pages 35–50, May 2004.
11. George Danezis and Jolyon Clulow. Compulsion resistant anonymous communications. In Mauro Barni, Jordi Herrera-Joancomartí, Stefan Katzenbeisser, and Fernando Pérez-González, editors, *Information Hiding*, volume 3727 of *Lecture Notes in Computer Science*, pages 11–25. Springer, 2005.
12. George Danezis and Claudia Diaz. A survey of anonymous communication channels. Technical Report MSR-TR-2008-35, Microsoft Research, January 2008.
13. George Danezis, Roger Dingledine, and Nick Mathewson. Mixminion: Design of a Type III anonymous remailer protocol. In *IEEE Symposium on Security and Privacy*, pages 2–15. IEEE Computer Society, 2003.

14. George Danezis and Andrei Serjantov. Statistical disclosure or intersection attacks on anonymity systems. In Jessica J. Fridrich, editor, *Information Hiding*, volume 3200 of *Lecture Notes in Computer Science*, pages 293–308. Springer, 2004.
15. Ronald J. Deibert, John G. Palfrey, Rafal Rohozinski, and Jonathan Zittrain, editors. *Access Denied: The Practice and Policy of Global Internet Filtering.* The MIT Press, 2008.
16. Claudia Díaz and Bart Preneel. Accountable anonymous communication. Chapter in: Security, privacy and trust in modern data management. springer, 2006.
17. Roger Dingledine, Nick Mathewson, and Paul F. Syverson. Tor: The second-generation onion router. In *USENIX Security Symposium*, pages 303–320. USENIX, 2004.
18. Roger Dingledine, Vitaly Shmatikov, and Paul Syverson. Synchronous batching: From cascades to free routes. In *Proceedings of Privacy Enhancing Technologies workshop (PET 2004)*, volume 3424 of *LNCS*, pages 186–206, May 2004.
19. Ian Goldberg. Improving the robustness of private information retrieval. In *IEEE Symposium on Security and Privacy*, pages 131–148. IEEE Computer Society, 2007.
20. Markus Jakobsson, Ari Juels, and Ronald L. Rivest. Making mix nets robust for electronic voting by randomized partial checking. In Dan Boneh, editor, *USENIX Security Symposium*, pages 339–353. USENIX, 2002.
21. Peter C. Johnson, Apu Kapadia, Patrick P. Tsang, and Sean W. Smith. Nymble: Anonymous IP-address blocking. In Nikita Borisov and Philippe Golle, editors, *Privacy Enhancing Technologies*, volume 4776 of *Lecture Notes in Computer Science*, pages 113–133. Springer, 2007.
22. Stefan Köpsell and Ulf Hillig. How to achieve blocking resistance for existing systems enabling anonymous web surfing. In Vijay Atluri, Paul F. Syverson, and Sabrina De Capitani di Vimercati, editors, *WPES*, pages 47–58. ACM, 2004.
23. Stefan Köpsell, Rolf Wendolsky, and Hannes Federrath. Revocable anonymity. In Günter Müller, editor, *ETRICS*, volume 3995 of *Lecture Notes in Computer Science*, pages 206–220. Springer, 2006.
24. Nick Mathewson and Roger Dingledine. Practical traffic analysis: Extending and resisting statistical disclosure. In David Martin and Andrei Serjantov, editors, *Privacy Enhancing Technologies*, volume 3424 of *Lecture Notes in Computer Science*, pages 17–34. Springer, 2004.
25. Ulf Möller, Lance Cottrell, Peter Palfrader, and Len Sassaman. Mixmaster Protocol — Version 2. IETF Internet Draft, July 2003.
26. C. Andrew Neff. A verifiable secret shuffle and its application to e-voting. In P. Samarati, editor, *Proceedings of the 8th ACM Conference on Computer and Communications Security (CCS 2001)*, pages 116–125. ACM Press, November 2001.
27. Peng Ning, Sabrina De Capitani di Vimercati, and Paul F. Syverson, editors. *Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, Alexandria, Virginia, USA, October 28-31, 2007.* ACM, 2007.
28. Meredith L. Patterson and Len Sassaman. Subliminal channels in the private information retrieval protocols. In *Proceedings of the 28th Symposium on Information Theory in the Benelux*, Enschede, NL, 2007. Werkgemeenschap voor Informatie- en Communicatietheorie.
29. Michael K. Reiter and Aviel D. Rubin. Anonymous web transactions with crowds. *Commun. ACM*, 42(2):32–38, 1999.
30. Len Sassaman, Bram Cohen, and Nick Mathewson. The pynchon gate: a secure method of pseudonymous mail retrieval. In Vijay Atluri, Sabrina De Capitani di Vimercati, and Roger Dingledine, editors, *WPES*, pages 1–9. ACM, 2005.

31. Andrei Serjantov, Roger Dingledine, and Paul F. Syverson. From a trickle to a flood: Active attacks on several mix types. In Fabien A. P. Petitcolas, editor, *Information Hiding*, volume 2578 of *Lecture Notes in Computer Science*, pages 36–52. Springer, 2002.
32. Paul Syverson, Gene Tsudik, Michael Reed, and Carl Landwehr. Towards an Analysis of Onion Routing Security. In H. Federrath, editor, *Proceedings of Designing Privacy Enhancing Technologies: Workshop on Design Issues in Anonymity and Unobservability*, pages 96–114. Springer-Verlag, LNCS 2009, July 2000.
33. Patrick P. Tsang, Man Ho Au, Apu Kapadia, and Sean W. Smith. Blacklistable anonymous credentials: blocking misbehaving users without TTPs. In Ning et al. [27], pages 72–81.
34. Matthew Wright, Micah Adler, Brian Neil Levine, and Clay Shields. Defending anonymous communication against passive logging attacks. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, pages 28–43, May 2003.