Journal of Computer and Communications, 2014, 2, 66-76
Published Online March 2014 in SciRes. http://www.scirp.org/journal/jcc
http://dx.doi.org/10.4236/jcc.2014.24010

The SMS Chaum Mix

Matthew Rothmeyer, Dale R. Thompson, Matthew Moccaro

University of Arkansas, JBHT-CSCE, Fayetteville, Arkansas, USA
Email: mprothme@gmail.com, D.R.Thompson@IEEE.org, mmoccaro@uark.edu

Received November 2013

Abstract

Mobile devices such as smartphones are prime candidates for the application of mixing techniques to provide anonymity for small groups of individuals numbering 30 to 40 members. In this work, a Chaum mix inspired, smartphone based network that uses the Short Message Service (SMS) is proposed. This system leverages both techniques used by current anonymity networks as well as knowledge gained from current and past research to make messages private and untraceable. Previously published attacks to anonymous systems are addressed as well as mitigation techniques.

Keywords

Chaum Mix; Distributed System; Mobile Security; Anonymity; Information Privacy

1. Introduction

Much research has been performed to provide anonymity and secrecy in communication systems. This research takes the form of protocols and rule sets applied to specific classes or modes of communication. Many of these protocols, taking their cues from Chaum's initial research into mixes [1], are formed of loosely connected computational nodes capable of interacting in a fashion similar to a peer-to-peer network. These nodes are usually distributed and rely on sets of volunteered hardware instead of some overarching organization to provide the network backbone. While Chaum's focus on mixes was initially directed at electronic mailing networks, it has seen applications in many areas ranging from hiding web traffic in the onion router [2] to Telephony [3]. With an understanding of Chaum Mixing, one could deduce that almost any network of devices capable of meeting the constraints defined above could achieve anonymity through an application of the concepts proposed by Chaum and the research inspired by them.

Mobile devices are prime candidates for the application of mixing techniques to provide anonymous communication. These devices fit the organizational requirements and capabilities of a Chaum-type network. However, they are currently vulnerable to snooping by both individuals and cellphone service providers. Mobile devices such as phones are incredibly pervasive; a recent report by the International Telecommunication Union (ITU) predicts that by 2014 the number of cellphones will exceed the population of our planet [4]. All of these devices share one common form of communication, the Short Message Service (SMS). An anonymity framework that uses SMS, as ours proposes to do, would bring the ability to communicate anonymously to groups using a common mobile platform, regardless of access to a data connected network.

How to cite this paper: Rothmeyer, M., Thompson, D.R. and Moccaro, M. (2014) The SMS Chaum Mix. *Journal of Computer and Communications*, **2**, 66-76. http://dx.doi.org/10.4236/jcc.2014.24010

The need for anonymity in SMS is quantified by the set of threats that it faces. It is usually difficult for a single attacker to gain access to the contents of a message sent via SMS or to follow the SMS message through the network. This is because most attackers do not have access to the underlying transport framework used by SMS to deliver messages. Without gaining some sort of access to the mobile device sending the message, the only option left to the attacker is to intercept and decrypt the message by being present on the endpoints of the system where cell towers communicate with end nodes. For this reason, SMS is not often considered vulnerable. This assumption does not take into account situations where an observer has access to the entire underlying cellular network or at the very least a sizable portion of it. Such is the situation when considering cellular telecommunications companies who, for dubious reasons, decide to observe and track customers, or in the case of oppressive governments who use control of electronic media to stifle speech and quash protest. In the case of the former, the entity can track a message up until the point in which it leaves its network (if this does occur). In the latter case, it is entirely possible that, as long as all communicants were using communication providers controlled or influenced by a governing organization, the organization could follow all messages from sender to receiver and even use that information to locate the physical whereabouts of that person or persons. Regardless of which case one considers, as long as an observer has access to the network the sender is connected to, enough information is usually present in a message to associate two phone numbers and observe communications between them.

The proposed communication framework would address the need for privacy and anonymity in communication amongst small groups of individuals. Application of Chaum Mixing and encryption techniques designed for SMS would allow users to remain hidden and their conversations private from any observer who could watch the underlying network. In the past, the price of an SMS message was a limiting factor in establishing this kind of network for multiple; users however, the increasing popularity of unlimited SMS plans is removing this barrier for most. SMS also faces restrictions that Chaum Mixing does not take into account, which requires a modified form of mixing that uses current research into privacy protection and information-theoretic metrics for anonymity. An outline of the rest of the work is as follows: In Section 2, an overview of current anonymous communication methods as well as a detailed outline of Chaum Mixing is given. Section 3 details how key properties of Chaum Mixes can be applied to SMS. The problems faced by Chaum Mixing and the application of mixes to SMS are described in Section 4. Section 5 proposes solutions to those problems mentioned in Section 4 and Section 6 addresses future work.

2. Current Anonymous Communication and Chaum Mixing

Anonymity networks are used in a variety of technology domains. Onion Routing [5] is a set of protocols that allow its users to communicate anonymously via TCP/IP. The onion router makes use of a global network of volunteer nodes, with each node obfuscating the traffic that passes through it. A user using Onion Routing would direct its traffic through several nodes in sequence. All data to be sent is encrypted in a layered structure with each layer containing routing information specific to a node in the sequence. When a node receives data it removes the outer layer, determines where data should be sent, and then forwards that data to the next node. The final data is sent as plaintext from the last node to the intended recipient. An example of Onion Routing with letters is shown in **Figure 1**.

Freenet [6] is a distributed system for data storage and retrieval designed to allow anonymous file sharing and information publishing among its users. Freenet is composed of volunteer users who allow those on the net to make use of unused storage on their machines. Files in Freenet can be identified through the use of a key, existing in two parts, derived through the use of a hash of the file. The first part of the key is public and used to identify the file while the second half is used to sign the file, allowing for integrity checking. The public key and file can then be posted to the network where the file will be distributed through the system with similar files being located within close proximity.

Mixminion [7] is a system that provides the ability to make email anonymous. Also known as a type III remailer, it is the successor to the type I Cypherpunk and type II Mixmaster remailers. Mixminion routes messages through a series of email servers to hide the actual location of an originating email. By encrypting the email in layers and removing a layer of encryption at each server, Mixminion assures that any message entering a server cannot be correlated to a message leaving that server. Mixminion improves upon the Type I and II

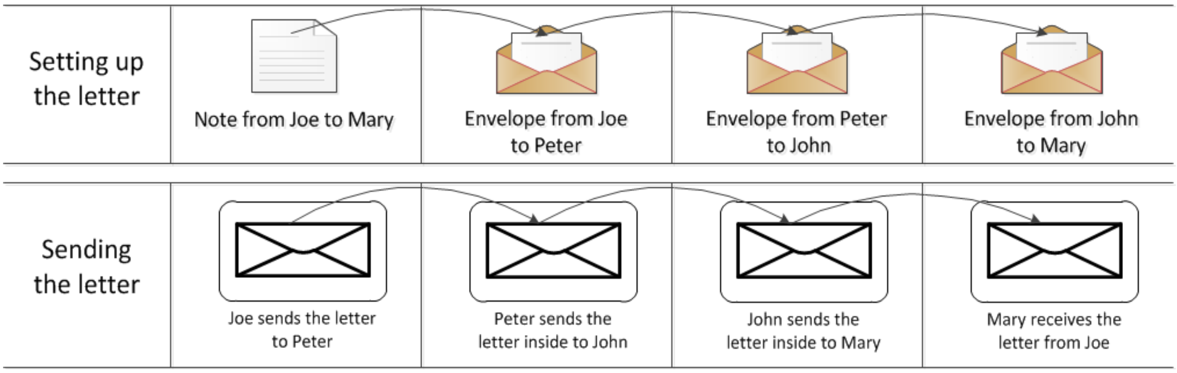
Setting up the letter
Note from Joe to Mary
Envelope from Joe to Peter
Envelope from Peter to John
Envelope from John to Mary
Sending the letter
Joe sends the letter to Peter
Peter sends the letter inside to John
John sends the letter inside to Mary
Mary receives the letter from Joe

Figure 1. Onion routing as explained with envelopes. Each envelope is a layer of encryption and each person is a router.

remailers by making it simple for clients to obtain information about the available remailing servers as well as covering security flaws exposed in Cypherpunk and Mixmaster.

Each of the above systems draws inspiration from Chaum's initial work on mix networks. The mix network is an algorithm that can be broken down into several steps that, when used together, provide anonymity to a user wishing to communicate without revealing the user's identity. Chaum defines a system of users and mixes, where a mix is a node that processes an item of mail to be delivered. Mail to be sent through the system by a user S to a recipient X is first padded with random bits and encrypted using a public key K belonging to X. Let M be the message to be sent, R be a set of random bits added to prevent any sending of the same message text from encrypting to the same value, and C(x,y) be the encryption function where x is the key and y is the plaintext. The encrypted message EM is then EM = C(k,{M,R}). Encryption prevents any intermediate party from reading the data within, while random padding prevents an observer from reverse engineering the message data through repeated encryptions with the recipient public key.

At this point, the sender decides the number of mixes that the message should be sent through and the route that message should take. It is assumed that any user of the system can access a list of available mixes with corresponding public keys. Once the route is decided, the sender will then encrypt the message in layers using keys associated with each selected mix. Each layer contains all the encrypted layers below it, an address for where the message should be sent next, and some random bits. Layering is in reverse order beginning with the last mix in the route and ending with the first. Using the same terminology as before and assuming 3 mixes are used in numerical order, let Ki be the key of mix i, Ri be a set of random bits for a mix, and Ai being the address of mix I. The payload sent by S, called P is then constructed as:

$$P_1 = C(K_3,\{E_M,R_3,A_X\}) \tag{1}$$

$$P_2 = C(K_2,\{P_1,R_2,A_3\}) \tag{2}$$

$$P_{Final} = C(K_1,\{P_2,R_1,A_2\}) \tag{3}$$

When the message is sent through the system each mix decrypts its layer, retrieves the address data, and forwards the message to the next mix. The final mix sends its data to the intended recipient who can decrypt the message. The decryption process D_i where mix i decrypts is as such:

$$D_1(P_{Fin}) = D_1(C(K_1,\{P_2,R_1,A_2\})) = P_2,R_1,A_2 \tag{4}$$

$$D_2(P_2) = D_2(C(K_2,\{P_1,R_2,A_3\})) = P_1,R_2,A_3 \tag{5}$$

$$D_3(P_1) = D_3(C(K_3,\{E_M,R_3,A_X\})) = E_M,R_3,A_X \tag{6}$$

Aside from the encryption scheme, Chaum also emphasized the importance of how and when data was sent from a mix. In order to prevent correlation between the arrival of data and the departure of a message, mixes must send received data using fixed size, lexically organized batches. Each batch would be sent to at least any user or mix that would be receiving data within that batch. It is left up to the next recipient to select appropriate messages from the batch and discard the rest. In order to prevent correlations based off of the number of input and output messages, each mix should include randomly addressed dummy messages in order to achieve some minimum batch size. It is also the responsibility of a mix to prevent duplicate messages or repeats. A repeat would allow an attacker to correlate the repeated input with the repeated output and thus violate anonymity for that message in the mix.

3. Applying Chaum Mixing to SMS

The proposed system achieves anonymity by identifying key elements of Chaum's algorithm and using them to create a protocol appropriate to SMS and the messages that it generally carries. In this case, each SMS device acts as a Mix, and the underlying cellular network plays the role of computer messaging networks. Software running on each device enforces rules on message transfer. The proposed system is described in its most secure form below, though several parameters (adjustable by group administrators) that improve efficiency but reduce security are mentioned in Section 5.

Encryption is needed to obfuscate messages as they pass through mix networks and hide the contents of the message from any potential observers. As in Chaum's work, encryption would be based on a set of keys. These keys would be privately available to members of the group, with each key corresponding to a single SMS capable mix device. The list of keys and associated devices would map to each member of the small group using the network. As SMS payloads are limited to 1120 bits, Eliptic Curve Cryptography is used for encryption as it achieves comprable strength to algorithms such as RSA while requiring much smaller keys (and thus smaller block sizes). Encrypting in layers, as is done in the Chaum network, is not appropriate to SMS. Message payloads in SMS are limited to 1120 bits, and each layer of encryption requires some of the available data to be sacrificed for use in random bits. As an alternative, the proposed system would use a two-step approach. First, the message is encrypted with ECC for the final recipient. Once the message is encrypted and a route selected (as will be discussed later) the sender adds a digital signature and a two-tuple consisting of some random bits along with a route number. Finally, the sender encrypts this with the public key of the next mix, and sends the message through the network. In the case of a single mix, once the message is received, the mix will decrypt the message, remove the route number and random bits, attach a new tuple, and then send the message to the intended recipient.

When a cascade of mixes is selected, the process is similar. The initial message is encrypted in the same way as described above, appended with the mentioned tuple, and then encrypted for the first mix. Once the message is received, the mix will decrypt the message, remove the route number and random bits, attach a new tuple, and then send the message to the next mix. As in Chaum, the mixing process repeats itself until the message reaches the end of the mix cascade, at which point the final mix will send the message to the intended recipient. In **Figure 2** we see an example of encrypting in a cascade. One will note that the sending protocol is the same for a single mix and a cascade of mixes and that the available bits in a message remain the same no matter how many mixes the message is routed through. This is because, prior to sending any messages, the sender negotiates a route with the system, reducing the amount of data necessary for encoding address information.

Route negotiation is similar to that used by other anonymity networks such as TOR [2] and is shown in **Figure 3**. When one user of the network wishes to send a message to another member, a number of mixes are selected randomly by the sender as well as an ordering for how messages should be sent.

Communication relating to route negotiation is handled in the same way as normal anonymous communication between two users. The sender, labeled S, will create a RFP (Request for Participation) message asking a mix, labeled X, to participate. The RFP will be encrypted appropriately (including a tuple), and then sent to X. A portion of the RFP contents includes a route number. If X agrees to participate, it will associate the phone number the message came from with the route number it received and acknowledge that it has done so. This acknowledgement will include a digital signature that only X can create. From this point on, any message with that combination of phone number and route number will be treated as a member of that path or route. This identification is bidirectional; S and X will use this route number when the stream of messages is flowing from S to X and vice versa.

After receiving the first acknowledgement, S will send a second message, a Request to Extend (RTE), to X which provides an address of a mix, labeled Y, to add to the end of the route. X will then select a route number

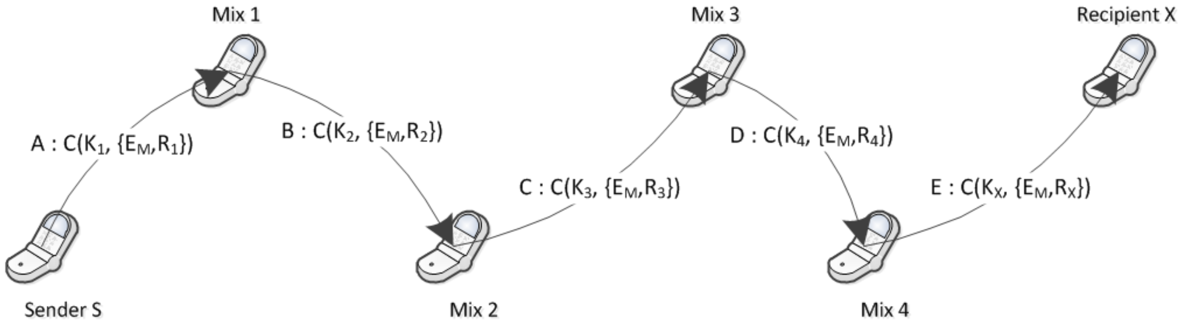

Figure 2. Sending an encrypted message.

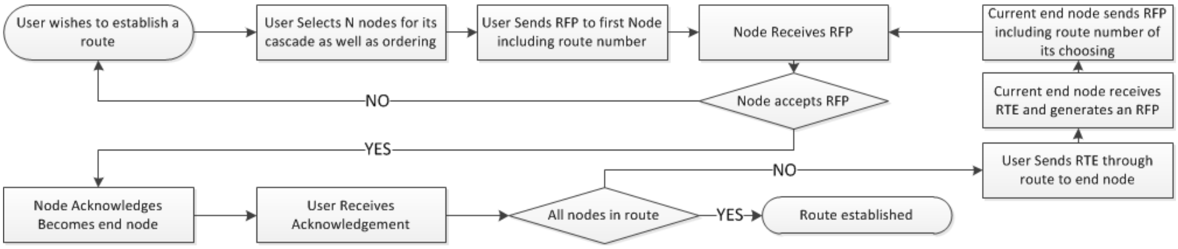
User wishes to establish a route
User Selects N nodes for its cascade as well as ordering
User Sends RFP to first Node including route number
Node Receives RFP
Current end node sends RFP including route number of its choosing
Current end node receives RTE and generates an RFP
User Sends RTE through route to end node
NO
Node accepts RFP
YES
NO
Node Acknowledges Becomes end node
User Receives Acknowledgement
All nodes in route
YES
Route established

Figure 3. Establishing a route.

to be used between X and Y, and then send an RFP to Y thus beginning the process again. The only difference is that an acknowledgement from Y or any subsequent messages will travel through the route back to S. This cycle would continue until the recipient, labeled R, has been reached. At this point, a RFP is sent to R but not an RTE. If any node is participating in a route but does not connect to any further nodes, it is assumed to be the intended recipient. From this point on, S can then send messages to R. Because each mix stores a 4-tuple containing two phone numbers and two route numbers it is not necessary for S to attach each address as was originally proposed by Chaum. Instead, S needs only append a route number for the first mix to which it sends. Each subsequent mix will remove the route number that was received and add the appropriate number for the next recipient. Since negotiation uses the same set of messages, it is impossible for a mix to know its location in relation to the rest of the route.

In the proposed system, digital signatures are created using the Elliptic Curve Digital Signature Algorithm (ECSDA). Grillo *et al*. proposed using ECC (elliptic curve cryptography) [8] as a method of signing SMS messages because ECC permits shorter keys (a 160-bit ECC key is equivalent in strength to a 1024-bit RSA key), and is comparably faster in operations that involve private keys. In their work, they proposed using ECDSA based off of the curve P-192 resulting in a 48-byte key that left 92 bytes for a message data payload. One prime concern in any mobile system is power consumption. ECC is well suited as it has been shown to consume less power than other methods of encryption [9]. In the proposed system, each mix acknowledgement is signed with a digital signature that prevents a compromised mix from disregarding the address specified in the RTE. Messages sent back and forth between the users are also signed to guarantee the validity of the sender.

Sending messages in a batch format as proposed by Chaum is also something that is not possible with SMS. Each message can only be sent one at a time. To overcome this constraint in the proposed SMS Chaum Mixnet, the system sends messages on a time quantum. All received messages must wait until the end of the current time slot to be sent. When a slot does expire, a number of messages marked to be sent are transmitted in rapid succession, and in some random order. In the case where the number of messages waiting to be sent is below a certain threshold, the system will create dummy messages and forward them to random phone numbers within the group. These messages would be assigned a dummy route number. To an outside observer, this message looks no different from all the others. Devices in the network will discard any message with the dummy route to save buffer space. Each device maintaining a steady rate of sending, be it real or dummy messages, makes it difficult for an observer to gather statistical data of sent and received messages.

As with the original Chaum Mix, nodes must also prevent duplicate messages or repeats from being processed.

This can be accomplished by taking a hash of received messages and then comparing new messages to that hash.

4. Problems Faced by Chaum Mixing and an SMS Chaum Mix

Research conducted following Chaum's initial work has noted several security weaknesses that appear in anonymity networks. Edman and Yener [7] have classified these networks based on the latency requirements of the users they serve, specifically low- and high-latency. High-latency networks refer to systems that can tolerate delays of up to several hours or more such as email, while low-latency networks encompass the domains of interactive applications like web browsing. High-latency systems are also referred to as message based systems while low-latency systems are known as connection based. Each kind of system is subject to different attacks each with the goal of compromising the users of the network. The proposed system is categorized in the middle of the two categories. While high delays are tolerable and even unavoidable at times, SMS communicates small amounts of information that often take the form of a conversation, granting some connection like properties. Therefore, it is important to examine the threats that both low- and high-latency systems encounter, and determine which threats need to be mitigated.

Edman and Yener [7] as well as Marques and Zuquete [10] have outlined several methods of attack in their works including the blending attack, the predecessor attack, the intersection attack, the timing attack, and the Sybil attack. Each attack is discussed below.

Blending attacks as referred to by [11-13] consist of two types of attack, the (n-1) attack and the trickle attack, which are directed against low-latency systems. The attacks build on the knowledge that buffer sizes in these systems are finite and that some form of flushing algorithm is used to send messages to make more room and to facilitate delivery.

The (n-1) attack is specific to the classic mix proposed by Chaum where the mix will send messages when some threshold for a number of messages has been met. In this attack, an adversary will find and delay a legitimate message M that it wishes to track. Then, the attacker will flood the next mix that M should be sent to with fake messages until the mix flushes. Once the flush occurs the attacker will send M along with more fake messages until the mix flushes a second time. The attacker then intercepts this second batch and eliminates or removes the dummy messages it provided. What is left is a small set of messages that could be M, or possibly only M, thus reducing or eliminating the effectiveness of the mix.

The trickle attack works in a similar way to the (n-1) attack but is directed at mixes that batch all messages received within a certain time frame. In this scenario, the attacker waits for a batch to be sent and then allows only a single message M into the mix. On the next flush, the attacker is guaranteed to see the mix output the decrypted message M.

In the predecessor attack, first suggested by Reiter *et al*. [14] and then formalized by Wright *et al*. [15], a group of attackers joins an anonymous network and attempts to identify the sender in a connection. Because communication paths are often remade on set time periods it becomes increasingly likely that, as time passes, the sender will select an attacker as the node that directly follows it. If the attackers can identify a stream or connections after a path is remade, they can eventually identify the sender. A predecessor attack can also be conducted by having an adversary as the first node into the network and the last node out of the network. By gathering timing information from the endpoints of the network, a group of attackers can discover which nodes are communicating with each other.

The intersection attack is a passive attack that requires an attacker to be able to monitor which users are using an anonymity network at a given time, as well as the connections between devices in the network. An attacker first selects a user X to track and then the first mix to which a user sends. By carefully watching for batch outputs that occur after X has sent a message, an attacker can determine which communications belong to X and which do not. By comparing batches that include a message from X but that only contain one recipient in common, an attacker can determine where those messages from X are being sent. Through repetitions of this process, an attacker can determine all communication pairs through any number of mixes.

The timing attack is a more generalized version of the methodology used by the predecessor attack. In the timing attack, users with access to the network passively monitor communication flows based on message timing. If multiple users are in a network, they can determine if they are on the same line of communication. Timing attacks need not be only passive. Zhu *et al*. [16] proposed that an active adversary could inject traffic that followed a specified pattern or in some other way to induce a timing delay along a route. This delay could then be tracked through any nodes an observer could view, even back to the initial sender.

The Sybil attack, when applied to a distributed environment as described in [17], allows an adversary to gain disproportionate influence over the network. The basics of a Sybil attack involve an attacker creating multiple identities that serve their purpose. To execute this attack an adversary creates multiple nodes and then joins each node to a peer-to-peer anonymity network. Though each node acts as a separate unconnected user, they are all secretly controlled by the adversary. The ability for an adversary to act as multiple identities or personalities is where the Sybil attack gains its name. These nodes can then collude, sharing data with the attacker and performing other kinds of attacks in sequence. The more nodes that can be attributed to a single entity, the more effective the attack is.

5. Addressing Security Concerns

After defining the security concerns that may be faced by the Chaum SMS Mix it is imperative that each issue be addressed to the fullest possible extent in order to prevent the system from being compromised. In order to do this, solutions for how to mitigate or eliminate each attack described above are discussed below.

N-1 attacks as directed against traditional mixes rely on two key assumptions. An attacker must be capable of sending a large number of messages to flood a system and, an attacker must be able to pick the message in question out of the remaining batch. In order to make flooding difficult, our system places two requirements on sending messages. First, in order for any messages to be sent and not immediately discarded they must be on a pre-negotiated route. Second, in order to be allowed to create a route in the first place a node must be in the privately available list of users and keys. Since each group uses a small list which is maintained by the group itself, an adversary must compromise the list or infiltrate the group to set up a route. If an adversary cannot accomplish either of those things, then a route cannot be established and a message flood cannot be initiated. In order make message identification difficult in case of a flood, the proposed system generates dummy messages.

A dummy message is a message containing random text whose route number is set to a reserved unused route. Every batch sent out by a mix will contain some amount of dummy traffic with the possibility of a batch containing only dummy traffic. This traffic works to maintain a constant flow of messages through the system. As the SMS Chaum Mix is designed for use by small groups, it is impossible to guarantee that significant numbers of messages are present in the system. Without these numbers a mix is forced to send out very small batches or hold a message for long periods of time, and becomes increasingly susceptible to the n-1 attack. In the proposed system, each real message received is placed into a queue. When a group of messages are to be sent, the system will randomly select a number of slots in the batch for dummy messages addressed to random users and then fill in the rest of the batch with real traffic. Real and dummy traffic is then sent from the mix into the system. Recipients of dummy messages will identify the reserved route number and immediately discard the message uppon decryption. Since dummy messages originate from within a mix, if a user does face an n-1 attack then an adversary, even one that could restrict incoming messages to a mix, would not be able to pick the target message out of a batch.

To combat the trickle attack the system makes use of the same methodology as above with a single addendum. If there are not enough messages to make a full batch then extra dummy messages are added until the batch is of the decided static size and if there are no messages then a batch of dummy messages is created. In this way, even if an attacker only allows a single message through, the output rate and batch size of a mix will stay the same and the attacker will still have to sort through just as many messages. Furthermore, the system will randomly select some batches with a certain frequency to contain only dummy messages to further hide real traffic.

In order to combat the predecessor attack, the SMS Chaum Mix builds off of the ideas of Wright *et al*. [18-20] that suggests the idea of an entry guard as a countermeasure to the predecessor attack. It was noted that in the case where the first node in a network was trusted, the scenario in which the user was the predecessor to the attacker was not possible. In every case all an attacker concludes is that the initiator of the communication is one of the entry guards. The proposed system would select a small number of devices for this role and then allow all users to begin their connections through these devices. In a real life scenario, a simple way to do this is would be to purchase several cash SMS enabled phones, install the anonymity software and connect them to the network, and then lock them in different rooms where they were plugged onto some sort of power supply. In order to keep these guards from being a single point of failure in the system, users will not have to connect to them, but will try to do so first. Though the devices are designated entry guards, they will still be available for use as normal mixes in the network. That way a node cannot use the knowledge of which devices are entry guards to provide it with more information as to its location in a route.

Intersection attacks are very difficult to combat. In their survey of anonymous communication systems, Edman *et al*. [7] were able to find no efficient methods that could prevent the success of such an attack with absolute certainty. Berthod and Langos [21] proposed that intersection attacks can be mitigated through the use of dummy messages that create a constant flow of data throughout the network. This proposal is based off of the idea that intersection attacks are possible if only a subset of users is making use of the network at the same time because a user can only be communicating with that subset. If all users are communicating, then the intersection attack becomes more difficult. In order to combat this attack, the proposed system sends dummy messages from each user on a time quantum as mentioned. The probability for any user to receive a number of dummy messages follows the binomial distribution:

$$F(k;n,p)=\binom{n}{k}p^k(1-p)^{(n-k)} \tag{7}$$

where n is the total number of dummy messages sent, k is the number of messages received and p is the probability of receiving a dummy message in a time quanta. This the probability can further be decomposed to:

$$F(k;n,p)=\binom{x*s}{k}\frac{1}{x-1}^k\left(1-\frac{1}{x-1}\right)^{(x*s)-k} \tag{8}$$

where x is the number of participants, s is the number of messages each user sends. The probability p becomes $1/x-1$ because each user selects the recipient of a dummy message from all other users. The probability of receiving at least 1 dummy message is then the probability of recieving no dummy messages subtracted from 1:

$$1-P(NoMsgs)=1-\binom{x*s}{0}\frac{1}{x-1}^k\left(1-\frac{1}{x-1}\right)^{(x*s)-k} \tag{9}$$

Based off of the number of participants the users could adjust the number of dummy messages sent by the proposed system until the probability of any single user receiving a number of dummy messages was significantly high. The number of dummy messages sent would either directly impact the number of messages sent in a batch (as batch size would grow or shrink in order to keep the defined ratio of dummy to real messages) or would directly impact the ratio of messages to be sent.

Along with sending dummy messages, the proposed system also allows routes to expire after a given amount of time. When a route expires the user has to renegotiate a route to continue to send messages through the network. Since intersection attacks rely on the idea of gathering sets of messages that only intersect once, and using that to identify either a communication partner or the next mix towards that partner, renegotiating routes would force an attacker to begin an unfinished intersection attack again.

Timing attacks are more applicable to systems with very low latency than to SMS Chaum Mixing. Passive timing attacks, that view a conversation between two users, are difficult to execute because of the delays that can be inserted by SMS itself, as well as the delay that comes from the human element of SMS messaging (that a person may not immediately reply to a message). Active forms of timing attacks, specifically ones that involve traffic shaping, are difficult to execute due to several factors. As discussed earlier, traffic can only be sent on pre-negotiated routes, and that is impossible for someone who has not been physically added to the network by its users. Second, unlike low latency systems where traffic can be described as a stream, messages in SMS Chaum Mixing are individual messages. Finally, traffic shaping is difficult because all message sending is done on a time quantum. A user can put a number of messages into the system, but those messages will only travel at a certain rate regardless of how quickly they are inserted.

The Sybil attack is difficult in SMS Chaum Mixing because the proposed system is designed for use by small groups who can personally add devices to the network and have some oversight into this process. It is even possible that some form of social networking as described in [10] might be used when reviewing new additions to the network in order to prevent an attacker from joining. Even then, a user who had infiltrated the group would still have to insert several nodes in order to engage in this kind of attack.

Another form of attack not explicitly mentioned above is an attack by traffic analysis. Similar to the timing attack, traffic analysis involves an adversary passively observing a network of mixes. An adversary will observe the inputs and outputs of a mix in an attempt to create source-destination pairs for these messages. Because mixes do not have an unlimited buffer size, a knowledge of what senders have messages in the buffer as well as the mixing or flushing strategy can allow an adversary to draw conclusions about sender-receiver pairs. In [22], Venkitasubramaniam proposes a mixing strategy that demonstrates asymptotic optimality. In this strategy the first third of the packets that arrive are set aside. The remaining packets are divided into two groups (1 and 2) such that all groups are of equal size. The distribution of packets from a mix-route pair in each of those groups, assuming unequal arrival rates, is proportional to the multinomial distribution:

$$Pr\{m_1,\dots,m_k\} \propto \begin{pmatrix} \frac{B}{3} \\ m_1 \dots m_k \end{pmatrix} \lambda_1^{m_1} \dots \lambda_k^{m_k} \tag{10}$$

where (m_1 ... m_k) is the set of possible compositions, B is the total buffer size, and λ_i is the arrival rate from a mix-route pair i. Once the buffer is filled, the first third of the packets that were set aside as well as group 1 are combined and marked for sending. As each new message arrives one of the messages from the marked group are sent based on some random ordering. After all messages marked for sending have been sent, the messages that arrived are again divided into two groups and the process repeats. The proposed system uses this strategy for sending messages, allowing the user to set the buffer size to some constant value across the system. A summary of the mentioned attacks is shown in **Table 1**.

The above system is designed specifically to be as secure as possible. However, it is recognized that the addition of security comes at the cost of throughput. In order to make the system flexible several security paramaters can be adjusted that reduce security but inprove overall operation. Group administrations would be able to do the following: enable or disable the sending of dummy messages, adjust the ratio of real to dummy traffic, adjust the minimum batch size, set a maximum and minimum route length, and finally limit the traffic a single user can generate.

6. Conclusions & Future Work

In this work, a Chaum mix inspired, smartphone based network that uses the Short Message Service (SMS) is proposed. It is designed to provide anonymity for small groups of individuals numbering 30 to 40 members that wish to exchange short messages. The group maintains a small private list of phone numbers, which is maintained

Table 1. Condensed list of attacks.

Attack Name	Attack Summary	Attack Description
N-1 Attack & Trickle Attack	The attacker forwards fake traffic with a single legitimate message or the attacker delays all but a single legitimate message from entering a mix.	• Limit number of users to a small group whose participants approve members • Require membership in order to send messages • Use dummy traffic ratios to prevent one user's messages from dominating a batch
Predecessor	Given enough path reformations, an adversary eventually becomes the predecessor to the sending node, allowing an attacker to gather data used to identify a sender.	• Select a small number of trusted nodes as entry guards. All communication routes begin through these nodes
Intersection	Given enough path reformations, an adversary eventually becomes the predecessor to the sending node, allowing an attacker to gather data used to identify a sender.	• All users communicate via real or dummy messages at each time quanta, thus preventing an attacker from determining which users are using the network at which times • Reform routes at set intervals
Timing	A single user controls multiple nodes in the system, giving more influence then should be allowed and allowing for information collusion between members of a route.	• High traffic latency • Fixed sending rate
Sybil	An attacker shapes traffic to form observable timing metrics through the system.	• Limit number of users to a small group whose participants decide who can be in the group • Require membership in order to send messages
Traffic Analysis	An attacker observes all messages entering and leaving a node and uses this information to correlate senders and receivers	• Dummy Traffic • Venkitasubramaniam's asymptotically optimal mixing strategy

by the group itself. Each smartphone in the small group acts like a mix by accepting and retransmitting messages to obscure which users are communicating. Not all nodes must be attached to a user with the possibility of dedicated and disposable smartphones, scattered over an area, mixing messages for added security and increased. ECC public key cryptography is used so that encrypted messages fit into the SMS payload.

It is understood that battery life is a important resource in the mobile domain and as such future work would involve empirical testing in order to quantify and mitigate power consumption. In implementation, this could be accomplished in part by throttling either through imposing sending limitations, limiting the number of routes that could pass through a phone, optimizing dummy traffic, or attempting to assign high traffic routes to unused nodes upon route reformation. Future work would also consist of writing and testing the application on a varying set of mobile devices. This would involve determining delay metrics for SMS messages sent with and without mixing as well as fine tuning values like buffer size and the maximum dummy traffic ratio for the system. Though a method of buffer flushing was proposed, future work might also consist of trying other flushing methods for systems of differing topologies.

Funding

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

References

[1] Chaum, D.L. (1981) Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. *Communications of the ACM*, **24**, 84-90. http://dx.doi.org/10.1145/358549.358563

[2] Dingledine, R., Mathewson, N. and Syverson, P. (2004) Tor: The Second-Generation Onion Router. Technical Report, DTIC Document.

[3] Jerichow, A., Muller, J., Pfitzmann, A., Pfitzmann, B. and Waidner, M. (1998) Real-Time Mixes: A Bandwidth-Efficient Anonymity Protocol. *IEEE Journal on Selected Areas in Communications*, **16**, 495-509.

[4] International Telecommunications Union (2013) The World in 2013: ICT Facts and Figures. http://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2013.pdf

[5] Goldschlag, D.M., Reed, M.G. and Syverson, P.F. (1996) Hiding Routing Information. *Information Hiding*, 137-150. http://dx.doi.org/10.1145/1592451.1592456

[6] Clarke, I. and Sandberg, O., Wiley, B. and Hong, T.W. (2001) Freenet: A Distributed Anonymous Information Storage and Retrieval System. *Designing Privacy Enhancing Technologies*, 46-66. http://dx.doi.org/10.1007/3-540-44702-4_4

[7] Edman, M. and Yener, B. (2009) On Anonymity in an Electronic Society: A Survey of Anonymous Communication Systems. *ACM Computing Surveys* (*CSUR*), **42**, 5.

[8] Grillo, A., Lentini, A., Me, G. and Italiano, G.F. (2008) Transaction Oriented Text Messaging with Trusted-SMS. *Annual IEEE Conference on Computer Security Applications*, *ACSAC* 2008, 485-494.

[9] Potlapally, N.R., Ravi, S., Raghunathan, A. and Jha, N.K. (2006) A Study of the Energy Consumption Characteristics of Cryptographic Algorithms and Security Protocols. *IEEE Transactions on Mobile Computing*, **5**, 128-143.

[10] Marques, R. and Zuquete, A. (2011) Social Networking for Anonymous Communication Systems: A Survey. 2011 *International Conference on Computational Aspects of Social Networks* (*CASoN*), 249-254.

[11] Diaz, C. and Preneel, B. (2004) Taxonomy of Mixes and Dummy Traffic. *Information Security Management*, *Education and Privacy*, 217-232.

[12] O'Connor, L. (2005) On Blending Attacks for Mixes with Memory. *Information Hiding*, 39-52.

[13] Serjantov, A., Dingledine, R. and Syverson, P. (2003) From a Trickle to a Flood: Active Attacks on Several Mix Types. *Information Hiding*, 36-52.

[14] Reiter, M.K. and Rubin, A.D. (1998) Crowds: Anonymity for Web Transactions. *ACM Transactions on Information and System Security* (*TISSEC*), **1**, 66-92. http://dx.doi.org/10.1145/290163.290168

[15] Wright, M.K., Adler, M., Levine, B.N. and Shields, C. (2004) The Predecessor Attack: An Analysis of a Threat to Anonymous Communications Systems. *ACM Transactions on Information and System Security* (*TISSEC*), 7, 489-522.

[16] Zhu, Y., Fu, X.W., Graham, B., Bettati, R. and Zhao, W. (2005) On Flow Correlation Attacks and Countermeasures in

Mix Networks. *Privacy Enhancing Technologies*, 207-225. http://dx.doi.org/10.1007/11423409_13

[17] Douceur, J.R. (2002) The Sybil Attack. *Peer-to-Peer Systems*, 251-260. http://dx.doi.org/10.1007/3-540-45748-8_24

[18] Overlier, L. and Syverson, P. (2006) Locating Hidden Servers. 2006 *IEEE Symposium on Security and Privacy*, 15 p.

[19] Syverson, P., Tsudik, G., Reed, M. and Landwehr, C. (2001) Towards an Analysis of Onion Routing Security. *Designing Privacy Enhancing Technologies*, 96-114. http://dx.doi.org/10.1007/3-540-44702-4_6

[20] Wright, M., Adler, M., Levine, B.N. and Shields, C. (2003) Defending Anonymous Communications against Passive Logging Attacks. *Proceedings of the* 2003 *Symposium on Security and Privacy*, 28-41.

[21] Berthold, O. and Langos, H. (2003) Dummy Traffic against Long Term Intersection Attacks. *Privacy Enhancing Technologies*, 110-128.

[22] Venkitasubramaniam, P. (2010) Anonymous Networking under Memory Constraints. 2010 *IEEE International Conference on Communications* (*ICC*), 1-5.