Poster: No safety in numbers: traffic analysis of sealed-sender groups in Signal

Eric Brigham
Department of Computer Science
University of Minnesota
Minneapolis, United States
brigh169@umn.edu

Nicholas Hopper
Department of Computer Science
University of Minnesota
Minneapolis, United States
hoppernj@umn.edu

***Index Terms*—Signal, end-to-end encrypted messaging applications, statistical disclosure attacks**

I. Introduction

Secure messaging applications often offer privacy to users by protecting their messages from would be observers through end-to-end encryption techniques [8] [9] [11]. However, the *metadata* of who communicates with whom cannot be concealed by encryption alone. Signal's Sealed Sender mechanism attempts to enhance its protection of this data by obfuscating the sender of any message sent with the protocol [5]. However, it was shown by Martiny *et al.* [6] that due to the message delivery protocols in Signal, the record of who *receives* messages can be enough to recover this metadata. In this work we extend the attack presented in [6] from deanonymizing communicating pairs to deanonymizing entire group conversations.

II. Background

Statistical disclosure attacks, or SDA, are a known attack vector which can be used to link senders and receivers of messages in anonymous mix networks [3] [4] [7] [10]. The traditional attack methods rely on two key components that seemingly make them ineffective in an environment like Signal: identities of senders are known and there is a mix entity present [1]. Under Signal's Sealed Sender mechanism, the identity of a message's sender would be hidden, and there is no mix present. The authors of [6] argue that in the presence of immediate responses, both of these apparent shortcomings can be overcome and a modified SDA attack can be developed and leveled against the users by the server in order to deanonymize communicating pairs. Crucially, they observe that immediate responses are guaranteed in Signal due to the fact that delivered receipts cannot be disabled by the user. The authors show both convincing theoretical analysis as well as simulations that the attack does achieve its goal. In this work we extend their proposed attack from the setting of communicating pairs to that of groups, where there are three or more users communicating through a single channel.

Supported by NSF grant 1814753

III. The Attack

We consider an attack setting where a single user Bob is being targeted in the hope of discovering his group G of k associates who are communicating together through a single channel. In Signal, when Bob sends a message through a group channel, within a short period of time which we refer to as the *epoch*, all members of the group can be seen to have received a message, and shortly thereafter, Bob will receive delivered receipts from all members; we refer to this sequence of delivered receipts as the *flurry*. Contrasting this with a random epoch of the same length where Bob has not sent a group message, the group members may or may not be receiving messages, but Bob will *not* receive a quick succession of messages. We observe then that by monitoring who receives messages most often before a flurry, we can identify the likely members of the group.

This behavior was confirmed by examining output logs generated by group messaging with the Android emulator Genymotion similarly to [2]. While the proposed attack generalizes to a group of any size, we use a group of three members for simplicity of exposition. We assume that all members of the group are online at the time of the attack, otherwise delivered receipts cannot be sent. Similarly to [6] we define random and target epochs, but the foundation of our attack is the *flurry*. The following definition/terms will be used throughout:

- **Flurry** a string of 'To Bob' messages which appear to the server as being sent in succession, or very close to it. This occurs after Bob sent a message through the group channel, after which we would necessarily observe delivered receipts to Bob from all members of the group. Please refer to Fig. 1 for a visual representation.
- **Attack Window** : the time-frame in which the attack takes place, necessarily spanning multiple messages sent to and from Bob
- **Target Epoch**: an epoch during the attack window preceding a flurry of "To Bob" delivered receipts. This contains some number of messages sent by Bob using sealed sender with all recipients(group members) observable.
- **Random Epoch**: an epoch taken from the total set of messages, chosen uniformly at random independent from Bob's activities.

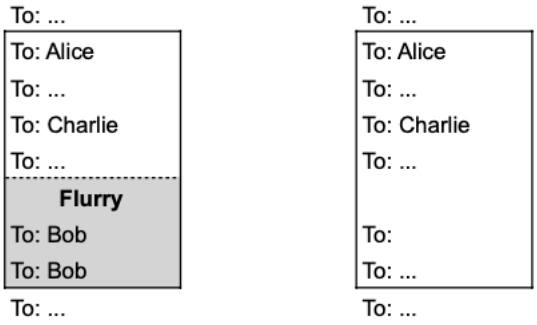

Fig. 1. The left box corresponds to a target epoch due to the existence of the flurry, while the right box could correspond to some random epoch

We define our attack on the assumption that a flurry exists. The attack follows the basic functioning presented in [6], but is modified for a group setting and predicated on the existence of a flurry. It is as follows:

1) Create an empty table of counts, initializing all values to zero.
2) Sample a target epoch. For each user that received a message during the target epoch, increment their count in the table.
3) Sample a random epoch. For each user that received a message during the random epoch, decrement their count in the table.
4) Repeat steps two and three for some number n target and random epochs (same amount each).
5) The users in the table with the highest counts are most likely to be in a group together. Note that because a flurry is always produced when Bob sends to the group, but is unlikely to occur in pairwise communication patterns, the process will produce the highest scores for group members communicating with Bob.

Note it possible to define an attack on the absence of a flurry, for instance if Bob were a passive observer of the group, but this is left to a full version of the paper.

IV. Theoretical Analysis

The subsequent analysis will rely on the following assumptions, of which one, two, and four are identical to those found in [6], the third and fifth are unique to this setting:

1) The probability of receiving a message during a particular epoch is independent of receiving a message during any other epoch
2) Each user has a fixed probability of receiving a message during a random epoch, call it r_u
3) Any group member u has a fixed probability t_u of receiving a message during a target epoch, with $t_u > r_u$
4) All users not in the group have the same probability of receiving a message during a target or random epoch, $t_u = r_u$
5) Bob is in **one** group (Note that in this case, assuming *no* false positive flurries, we have $t_u = 1$ for all $u \in G$.)

In this section we present theoretical analysis of attack success. It follows much in the same way as the analysis presented in [6], but for a group setting.

Theorem: Given m total users in a messaging system. Let each group member $u \in G$ have probabilities r_u, t_u of appearing in random or target epochs respectively. Then under the stated probability assumptions, the probability that all k group members are ranked higher than all non-associates after n random and target epochs is at least:

$$1 - \frac{m|G|}{C^n}$$

Where the parameter $C = \min_{u \in G} exp((t_u - r_u)^2/4) > 1$, depend solely on group member probabilities r_u, t_u. The proof of this theorem is left to a full version of the paper.

The consequences of this are similar to those found in [6], chief among them being that the number of epochs needed to de-anonymize group members with high probability scales logarithmically with the total number of users – meaning that the attack is efficient.

V. Future work

While all that has been included here are theoretical results, simulations are in progress and will be included for a full paper. We also hope to explore the dynamics of the target user being in multiple groups simultaneously. Finally, we hope to develop and test defense mechanisms that do not require Signal to modify their own implementation.

References

[1] David L Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–90, 1981.
[2] Cas Cremers, Charlie Jacomme, and Aurora Naska. Formal analysis of session-handling in secure messaging: Lifting security from sessions to conversations. In *Usenix Security*, 2023.
[3] George Danezis. Statistical disclosure attacks: Traffic confirmation in open environments. In *Security and Privacy in the Age of Uncertainty: IFIP TC11 18 th International Conference on Information Security (SEC2003) May 26–28, 2003, Athens, Greece 18*, pages 421–426. Springer, 2003.
[4] George Danezis, Claudia Diaz, and Carmela Troncoso. Two-sided statistical disclosure attack. In *Privacy Enhancing Technologies: 7th International Symposium, PET 2007 Ottawa, Canada, June 20-22, 2007 Revised Selected Papers 7*, pages 30–44. Springer, 2007.
[5] J. Lund. Technology preview: Sealed sender for signal. *Signal Blog: https://signal.org/blog/sealed-sender/https://signal.org/blog/sealed-sender/*, 2018.
[6] Ian Martiny, Gabriel Kaptchuk, Adam J Aviv, Daniel S Roche, and Eric Wustrow. Improving signal's sealed sender. In *NDSS*, 2021.
[7] Nick Mathewson and Roger Dingledine. Practical traffic analysis: Extending and resisting statistical disclosure. In *Privacy Enhancing Technologies*, volume 3424, pages 17–34. Springer, 2004.
[8] Trevor Perrin Moxie Marlinspike. The x3dh key agreement protocol. *Signal Blog: https://signal.org/docs/specifications/x3dh/*, 2016.
[9] Trevor Perrin Moxie Marlinspike. The sesame algorithm: Session management for asynchronous message encryption. *Signal Blog: https://signal.org/docs/specifications/sesame/*, 2017.
[10] Carmela Troncoso, Benedikt Gierlichs, Bart Preneel, and Ingrid Verbauwhede. Perfect matching disclosure attacks. In *Privacy Enhancing Technologies: 8th International Symposium, PETS 2008 Leuven, Belgium, July 23-25, 2008 Proceedings 8*, pages 2–23. Springer, 2008.
[11] Nik Unger, Sergej Dechand, Joseph Bonneau, Sascha Fahl, Henning Perl, Ian Goldberg, and Matthew Smith. Sok: secure messaging. In *2015 IEEE Symposium on Security and Privacy*, pages 232–249. IEEE, 2015.