Mitigating Abuse in Online Private Communication

Nirvan Tyagi and Helen Nissenbaum

The ability to communicate privately over the internet is now enjoyed by billions of people, due primarily to the adoption of end-to-end encryption by popular platforms like WhatsApp and iMessage. End-to-end encryption provides strong confidentiality and integrity guarantees, meaning the platform cannot read nor interfere with conversations between its users. Unfortunately these same properties hamper the ability of the platform to perform critical safety services on behalf of its users. We have seen in WhatsApp the negative effects of safety systems being unable to see message content directly: hate speech inciting lynchings, political disinformation campaigns, and child pornography rings are a few examples of abuse evading moderation found on WhatsApp. Such failings have led to calls for the end of private online communication.

In this work, we advocate for an alternate approach tailored to the types of abuse a platform wishes to protect against. Instead of removing the privacy guards on online communication to enable safety systems, we will examine the safety systems, determine what information is needed for effective abuse mitigation, and build “abuse reporting” mechanisms that reveal to the safety systems only this specified information in the appropriate contexts (enforced by policy and/or cryptographic methods). Ultimately, this approach still represents a weakening of the original privacy properties of the communication system -- a necessity to achieve user accountability -- but carefully designed safety systems combined with abuse reporting mechanisms can produce an intermediate solution space that showcases less stark tradeoffs to users and policymakers.

In line with this approach, Facebook has recently announced a new safety system for detecting financial scams and instances of child grooming that is run with only communication metadata as input, i.e., does not need to see the message content [1]. The widely-deployed private communication platforms we are considering (WhatsApp, iMessage) hide message content, but do not attempt to hide metadata, or who is communicating with whom, at what time, and how frequently. However, privacy researchers have long warned about the privacy threat of such rich communication metadata where the information revealed through inference is just as sensitive as the underlying message content. In this light, perhaps, it is not surprising that safety systems can function effectively given this trove of user communication metadata.

To arrive at a meaningful intermediate solution, our work advocates for safety systems tailored to different types of abuse that further minimize the amount of information needed to function effectively. Consider the aforementioned example of child grooming. The system may look for messages between adults and teenagers. Such a system would not need access to all the communication metadata, but may function effectively given only broad tags of users based on age, or even more specifically given instances only of adult accounts that are active in an abnormally large number of conversations with teenagers. It would be the job of the abuse reporting mechanism to procure these instances from the private communication protocol.

We will illustrate the need for a variety of mitigation techniques by studying a few examples. First, consider the example of online harassment. Here, as the abused party is a part of the conversation, they can report the abuse to the platform, who can then take appropriate action, e.g. by banning the abuser. This type of user-driven reporting can be thought of as leading to “reactive” actions by the platform. Such...
a strategy would not work for other types of abuse, such as distribution of child sexual abuse material (CSAM) or copyrighted materials. Here, the conversation participants are complicit in the abuse and will not self-report, requiring other strategies such as social graph analysis or client-side filtering, which we call “proactive” actions by the platform. Lastly, there are types of abuse that lie in between. For example, in the spread of misinformation, users that are swept up in the virality and forwarding will not self-report, but eventually the misinformation will reach parties able to identify the abuse. Or in child grooming, the victim may not realize the abuse until weeks or months down the line. Here, reactive actions may be paired with proactive actions to effectively protect users.

User-driven reporting: Strengths and limitations of reactive abuse mitigations. One approach for addressing abuse on a platform is to wait for users to identify and report abuse themselves. This approach has the obvious benefit that the platform need not learn anything about user communications until after an abuse report is made. In fact, end-to-end encrypted messaging protocols have been extended to cryptographically support user-driven reporting. “Message franking”, a technique pioneered by Facebook [2] and improved on in subsequent work [3-5], targets online harassment and allows users to securely report abusive messages attributing them to a sender. At a high level, message franking operates much like its non-digital counterpart of postal franking: the platform cryptographically marks encrypted messages as it delivers them to recipients. A recipient can later report a marked message along with a key to allow the platform to learn the abusive content. The cryptographic mark allows the platform to validate the content and sender, preventing erroneous reports from framing a sender.

This approach enables harassment moderation systems by revealing the minimal amount of information needed for them to function, namely the offending sender and the abusive content. However, other types of safety systems require different information to function effectively.

Consider the task of moderating viral misinformation, another type of abuse commonly found on online communication platforms. Misinformation can be spread rapidly through a network by users forwarding a received message to their contacts. It has been found that misinformation is often injected into a network as part of coordinated efforts by a small set of accounts in what is known as a misinformation campaign [6]. Protecting an online community against misinformation campaigns is not an easy task, and developing effective countermeasures is an active area of research. Some actions platforms have chosen to take are (1) to identify the source accounts injecting the misinformation and ban them, and (2) to identify users that share misinformation and inform them.

The user-driven reporting mechanism designed for online harassment falls short of enabling these types of safety systems for misinformation. A report reveals to the platform the misinformative content and the last sender of the message; it doesn’t help the platform identify the larger landscape (of possibly millions) of users affected or identify the source of the message. In recent work, an alternate user-driven reporting mechanism is proposed for use with misinformation safety systems. Message tracing [7] allows for users to report a “tracing key” for a received message that enables the platform to learn the forwarding tree of a message, i.e., the original sender of the message along with where the message has been shared.

Even with the message tracing mechanism designed specifically for misinformation, we start to see some of the inherent limitations of user-driven reporting: at the point a piece of misinformation is reported,
much of the damage of the misinformation campaign has already been dealt. Informing users that they have previously shared misinformation may slowly lead to users making more informed decisions around sharing, however in the meantime, millions of people will continue to fall victim to misinformation causing real negative consequences.

The reactive nature of the countermeasures taken from user-driven reporting are a good fit for addressing online harassment, but are less effective in other contexts. For unknowing victims of misinformation or child grooming, user-driven reporting cannot be relied upon to surface abuse in a timely manner. In contexts where the victim is not a communication participant, as in the distribution of CSAM, user-driven reporting will never surface the abuse. Different techniques will be needed.

**Challenges of proactive abuse mitigations.** Whereas user-driven reporting leads to reactive countermeasures *after* an abuse has been reported, a proactive abuse mechanism would surface possible instances of abuse *before* being explicitly identified by a user. An often discussed proposal of a proactive abuse mechanism is client-side scanning. Client-side scanning refers to the practice of scanning unencrypted content on the user device, taking action if restricted content is detected, e.g. by notifying the user, blocking content from being sent, or reporting content to the platform. For misinformation, warning notifications can be shown to users to deter them from sharing identified misinformation. For preventing the spread of CSAM, actions like blocking content and/or reporting to the platform can be made. Client-side scanning has been a particularly divisive topic due to its direct reliance on scanning unencrypted content (more privacy-preserving scanning of encrypted content is generally computationally expensive).

An alternative proactive approach was recently deployed by Facebook [1]. Instead of proactively scanning unencrypted client-side content, the system proactively scans communication metadata. Communication metadata is sensitive as well, but perhaps may be a more tractable target than content for the application of privacy-preserving mechanisms, e.g., when a forwarding tree gets to a certain size (virality) it is revealed, or networks of users in contact with suspected CSAM distributors are revealed.

An overarching concern about the deployment of proactive abuse mitigations is how users can be confident that the patterns being revealed are limited to the ones agreed upon by society (e.g., through public policy or wide announcement) and do not include “backdoor” patterns. In some cases, we can rely on cryptographic constructions to enforce what can be revealed, but as mentioned before, not all patterns can be matched efficiently. For these situations, proactive abuse mitigations may be paired with transparency systems, so if a platform misbehaves to reveal information inappropriately, such misbehavior may be detected and caught after the fact. Cryptography and other security approaches like trusted hardware may play a role in building these transparency systems, but it may also be enforced by public policy.

**Conclusion.** Different types of online abuse require different countermeasures for effective mitigation. We advocate for revealing only the minimum amount of user data needed to effectively mitigate against abuses by using a combination of reactive and proactive techniques enforced by both cryptography and policy. Continuing to address these concerns is critical for the health of online communication.