Content-Oblivious Trust & Safety Techniques: Results from a Survey of Online Service Providers

Riana Pfefferkorn
USENIX Enigma
Feb. 1, 2022
Background

● Motivation
  ○ Hot topic: How to do trust & safety (T&S) in an increasingly E2EE world?
  ○ Discussions often presume that doing T&S requires access to content
    ■ 2 “solutions” I’ll discuss: “break E2EE” & “scan all the things”

● Goal: learn more about “content-oblivious” T&S techniques currently in use

● Terminology
  ○ “Content-dependent”: e.g. automated scanning, review by human moderators
  ○ “Content-oblivious”: e.g. metadata, user reports, limits on group size
About the survey

● Survey administration
  ○ April-June 2021
  ○ Distributed to individuals at a variety of online services
  ○ Aimed at providers of apps & services for online communications and/or data storage
  ○ Analysis includes responses from 13 of 58 survey recipients (22.4%)

● About the participants
  ○ Several services are E2EE messaging apps, but most aren’t
  ○ FB Messenger, IG Messaging, WhatsApp, Wikidata, MetaFilter, Lobste.rs, Y! Groups
  ○ From ~2K MAUs to ~2B MAUs
  ○ Median anonymous participant has >200M MAUs
Categories of abuse covered in survey

- Intellectual property infringement
- Spam
- Phishing or malware
- Child sexual abuse imagery (CSAI)
- Child sexual exploitation (e.g. grooming, enticement) (CSE)
- Terrorism or violent extremism
- Pornography, sexual content, or obscenity (non-child)
- Dis-/misinformation
- Harassment, threats, (s)extortion, or intimidation
- Hate speech
- Self-harm
- Bots or inauthentic behavior
- Other
Summary of key findings

- Everyone employs both content-oblivious and content-dependent techniques
- Most popular technique: user reports
  - All 13 offer some kind of abuse reporting (in-app and/or “off-app”)
  - Less prevalent: metadata, automated content scanning, etc.
- Most useful technique: also user reports
  - Considered the most useful for detecting 9 of 12 abuse categories
  - Yet user-reporting tools don’t consistently enable reporting of all 12
- The outlier: CSAI
  - Strong consensus re: automated content scanning

Let’s walk through some charts...
Use of various techniques to detect, prevent, & mitigate abuse

- In-app reports
- Human review
- Outside reports
- Metadata
- Automated scans
- Share/forward limits
- Group size limits
- Other
- None of the above
- N/A
- Don't know
- Refuse to answer
Use of content-oblivious techniques to detect various abuse types
Technique deemed most useful for detecting each abuse type

- IP infringement
- Spam
- Phishing / malware
- CSAI
- CSE
- Terror or VE
- Porn
- Mis- or disinfo
- Harassment
- Hate speech
- Self-harm
- Bots
- Other

Legend:
- Automated scans
- Metadata
- User reporting
What Does This Imply about E2EE’s Impact on Abuse Detection?

● E2EE (which hinders access to content) impedes automated scanning (because it’s content-dependent), but not user reporting or metadata (because they aren’t).
● I find that user reporting ≥ automated scanning for all abuse types except CSAI.
● Implication: E2EE doesn’t impact abuse detection efforts uniformly.
What Does This Imply about E2EE’s Impact on Abuse Detection?

● Rather, E2EE’s impact on abuse detection probably varies by abuse type.
  ○ Least impact: content-oblivious tools >>> automated scanning
    ■ Examples: harassment, hate speech, self-harm
  ○ Some impact: content-oblivious tools ≈ automated scanning
    ■ Examples: CSE, spam
  ○ Greatest impact: content-oblivious tools <<< automated scanning
    ■ Sole example: CSAI

● If so, calls to break E2EE or compel automated scanning are mostly a *non sequitur*. 
  ○ Why break E2EE if it doesn’t impede the most useful tool?
  ○ Why mandate something that is mostly *not* very useful?
Abuse Isn’t a Uniform Problem Requiring a Uniform Response

- Policy debates often focus on CSAI, as though it’s representative of all abuse.
- But CSAI is not like other types of abuse.
  - What works best against CSAI doesn’t work best for other abuse types & vice versa
  - Even child safety isn’t a uniform problem: CSE ≠ CSAI
    - CSE lacks CSAI’s strong consensus re: automated scanning
      - E2EE may affect CSE detection < CSAI detection
      - If so, E2EE’s overall effect on child safety is less than CSAI alone would suggest
- Can’t accurately forecast E2EE’s impact based on CSAI alone.
- Optimize for CSAI → shortchange other abuse types.
Takeaways for Policy

● There is no silver bullet for online abuse.
  ○ Automated scanning isn’t a silver bullet
  ○ But neither are content-oblivious tools (and we shouldn’t pretend otherwise)
  ○ Content-oblivious tools also affect privacy etc. (e.g. metadata)

● No one-size-fits-all answers.
  ○ CSAI context is unique & can’t be the basis for T&S programs — or laws
  ○ Automated scanning mandates risk codifying a largely ineffective method
  ○ E2EE doesn’t break what is useful (user reports)
    → weakening E2EE = huge $\downarrow$ + little $\uparrow$

● Instead of broad mandates, providers need:
  ○ A suite of T&S tools for differing challenges
  ○ Legal flexibility to try new things, discard old ones, & evolve their strategies
    ■ (Abusive users are always evolving theirs)
Providers: Here’s What I Want from You

● Invest in better, more granular user reporting functionality.
  ○ Address the types of abuse your users are likely to encounter
  ○ Empower users while defraying E2EE’s impact
  ○ (But UX/UI issues are out of scope here)

● More transparency about your T&S programs & research.
  ○ “Before & after” data on abuse detection (T&S tooling changes, adding E2EE)
  ○ Research:
    ■ Why is automated scanning disfavored for most abuse types?
      ● Is the tech just immature?
      ● Inherently ill-suited to some abuse types?
    ■ Role of user education & guidance
      ● E.g. mis/disinfo labels, phishing warnings, child safety interventions
    ■ Poll users about desired anti-abuse features
Thank You

Read the Paper:
io.stanford.edu/COTS

Contact Me:
riana@stanford.edu
@Riana_Crypto
For each type of abuse, which do you find most useful for detection? (n=9)

This chart includes only the 9 services that use automated monitoring or scanning of content to fight abuse.
Do you do metadata-based abuse detection for any of these types of abuse? (n=9)

N.B. Respondents only shown this Q if checked “metadata” on the previous Q. Blank responses coded as “Refuse to answer.”
Do you enable in-app user reporting for any of these types of abuse? (n=12)

N.B. Respondents only shown this Q if checked “in-app reports” on the first Q. Blank responses to “Other” coded as “Refuse to answer.”
Respondents’ descriptions of metadata abuse detection tools

Then:
- From a former Yahoo! employee in the early aughts (pre-PhotoDNA), who worked on IDing CSAI on Groups (now defunct): “we used membership in confirmed [CSAI] groups to work through the members' group networks in order to recommend further groups for content moderator review. [...] [CSAI] groups also had distinctive usage patterns that differed from adult pornography groups.”

Now:
- Lobste.rs (a community link-aggregator site): “Account automatically prohibited from submitting links if heavily flagged by users”; new accounts (≦ 70 days old) are “unable to invite users, post links to unseen domains, or suggest story edits.”
- Anonymous: “We have trained models that run on various forms of metadata, including whether or not a human took action on similar data in the past to surface content for review.”
Respondents’ descriptions of user reporting functionality

Then:
- From the former Yahoo! employee: user-reporting tooling was fairly rudimentary at the time, “mostly reports to customer service agents that had to be reviewed manually.”

Now:*  
- MetaFilter (a community link-aggregator site): “Users can report issues and concerns about site content via an inline flagging mechanism on every comment and post, via a web-based contact form linked prominently on every page, via on-site mail to members of the moderation staff, via off-site email to individual or group company addresses for the moderation staff, and via an onsite posting queue.”
- Lobste.rs: “Users can flag UGC with a list of pre-selected reasons”; if problem doesn’t fit within those reasons, “the in-app private message feature is used to message moderators as a catch-all”
- Anonymous: Report feature was designed “primarily for spam/phishing,” but could be used for other forms of harmful content
- Anonymous: “In-app user reporting tools do not enable users to distinguish by type of abuse.”

* N.B. Some anonymous answers aren’t quoted in the paper, in order to protect participant anonymity.