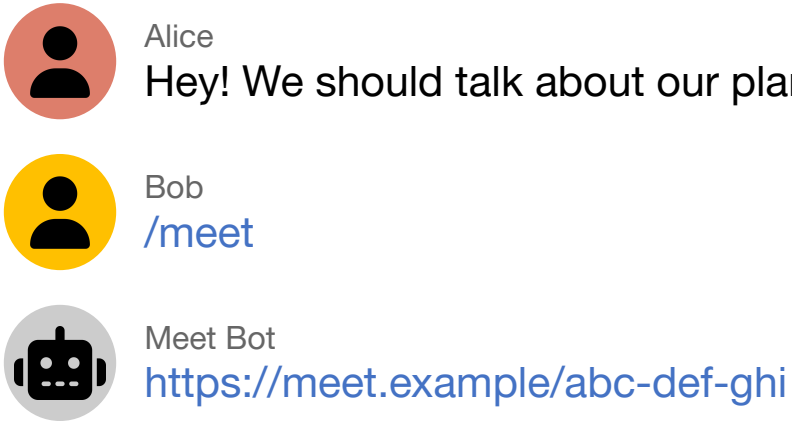
Bots can Snoop

Uncovering and Mitigating Privacy Risks of Bots in Group Chats

Kai-Hsiang Chou*, Yi-Min Lin*, Yi-An Wang, Jonathan Weiping Li,
Tiffany Hyun-Jin Kim, Hsu-Chun Hsiao

國立臺灣大學
National Taiwan University

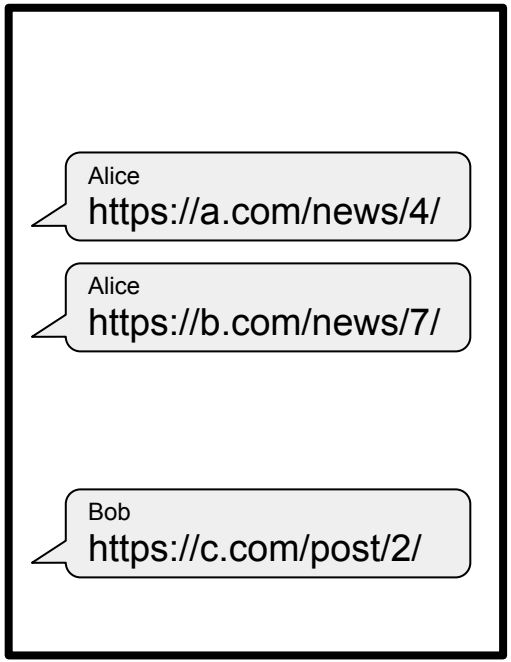
Group Chatbots



- On Discord, about **430,000 bots** were active across roughly **30% of groups**.
- Telegram hosts over **10 million bots**, one of the largest bot ecosystems.

Chatbots Read Irrelevant Messages

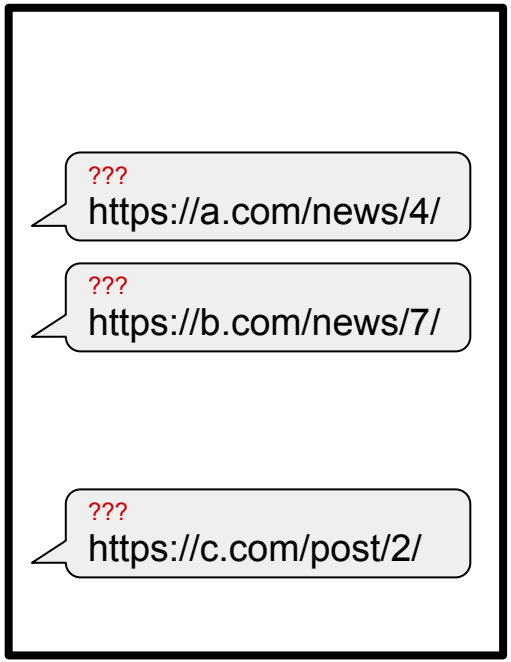
Case study:
A Discord chatbot accesses 400× more messages than necessary.



Selective Message Access

Chatbots Learn Sender Information

Case study:
3.6% of users encounter the same chatbot in multiple groups.



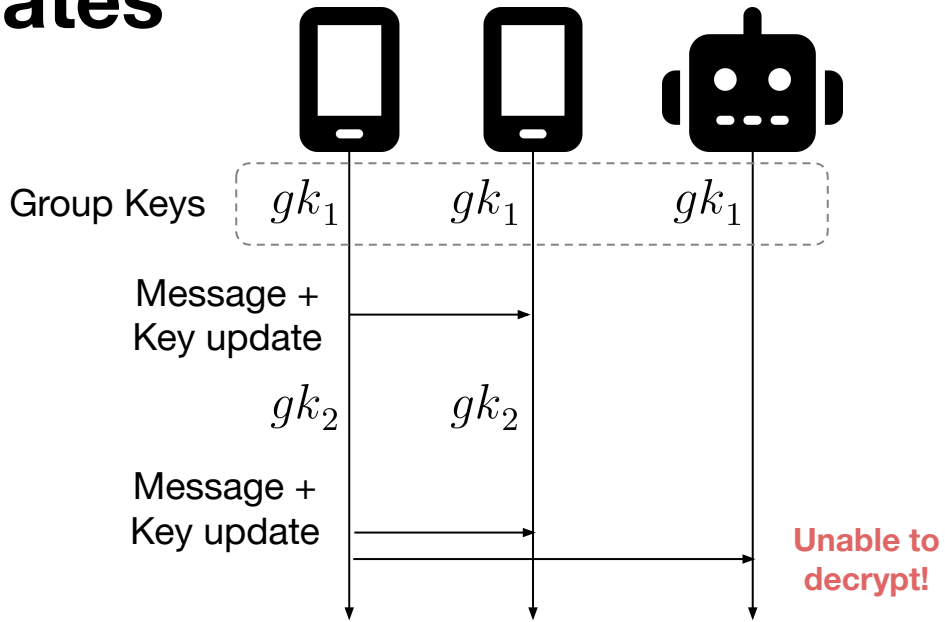
Sender Anonymity

Evaluation of Existing Platforms

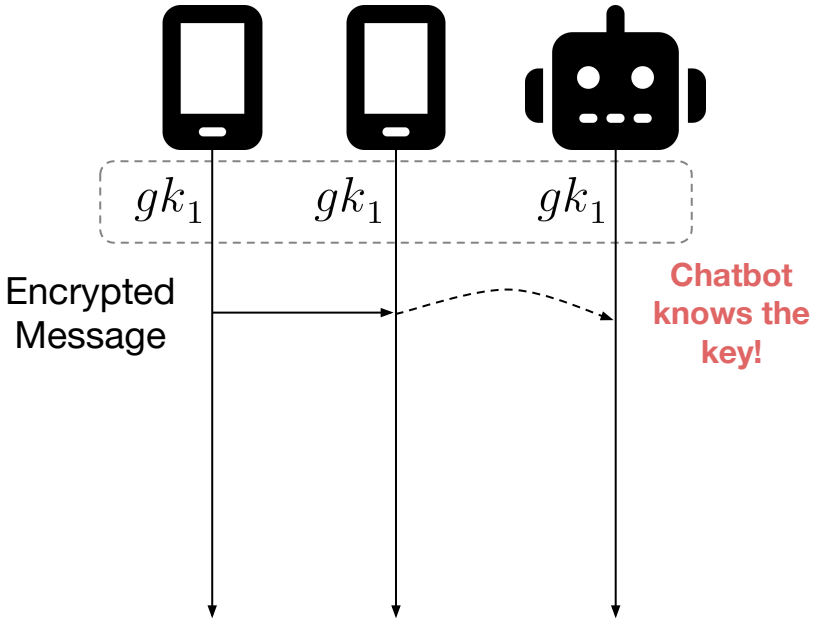
No platform supports **all** desired properties.

	Userbots on WhatsApp, Viber, and Signal	LINE	Telegram	Discord	Slack	Keybase
E2EE	✓	▲				✓
Selective Message Access			✓	✓	✓	✓
Sender Anonymity					✓	

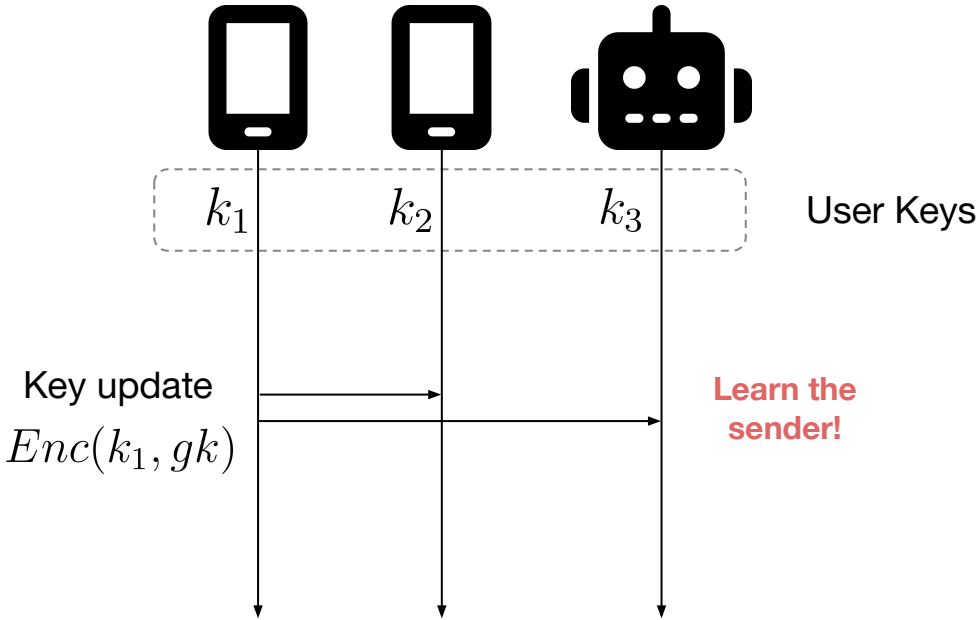
Challenge: Chatbots May Miss Key Updates



Challenge: Chatbots Know the Keys

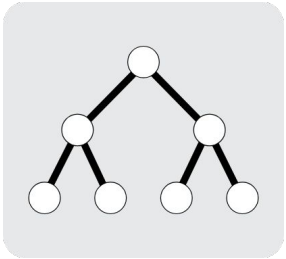


Challenge: Key Updates Leak Senders



Our Solution: SnoopGuard

Built on tree-based continuous group key agreement (CGKA) schemes.



✔ Selective message access
✔ Sender anonymity
✔ End-to-end encryption

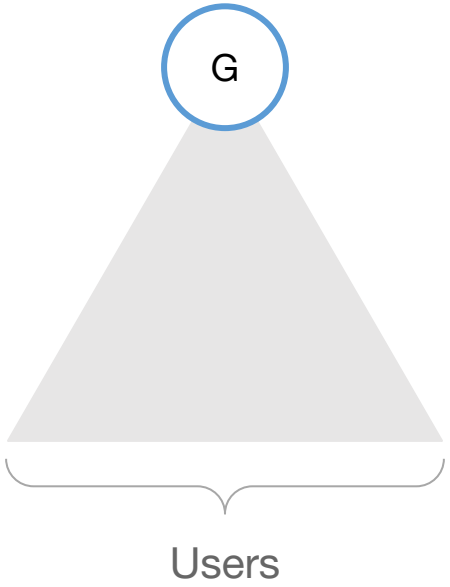
Easy integration with existing protocols.

MLS

Signal

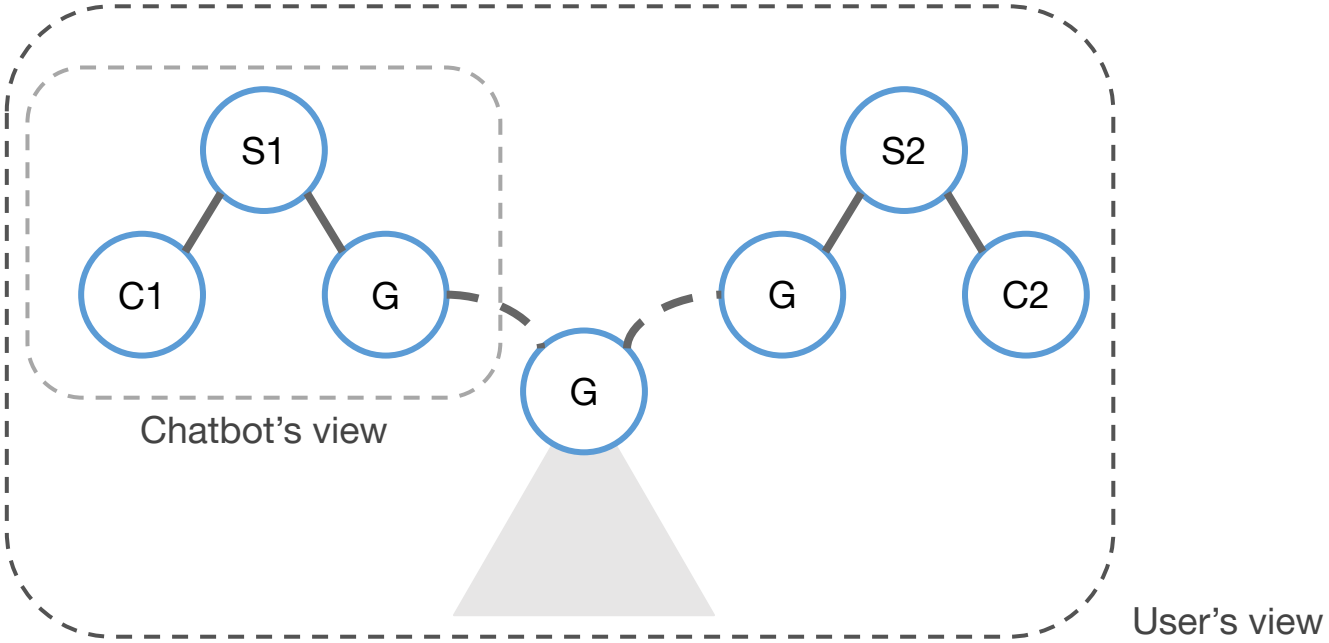
Compressed Multi-Roots Tree (CMRT)

- CMRT extends an existing CGKA scheme.



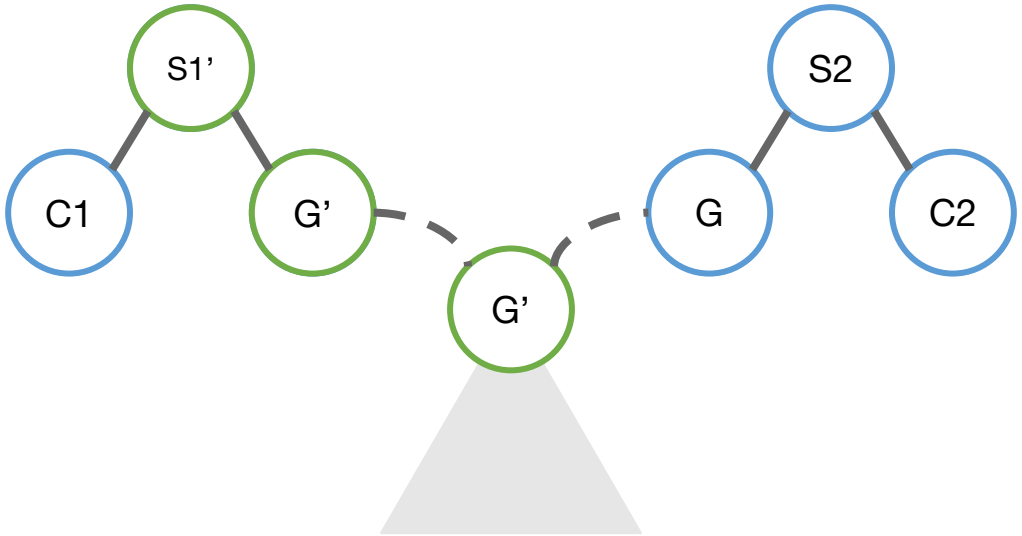
Compressed Multi-Roots Tree (CMRT)

- Chatbot addition



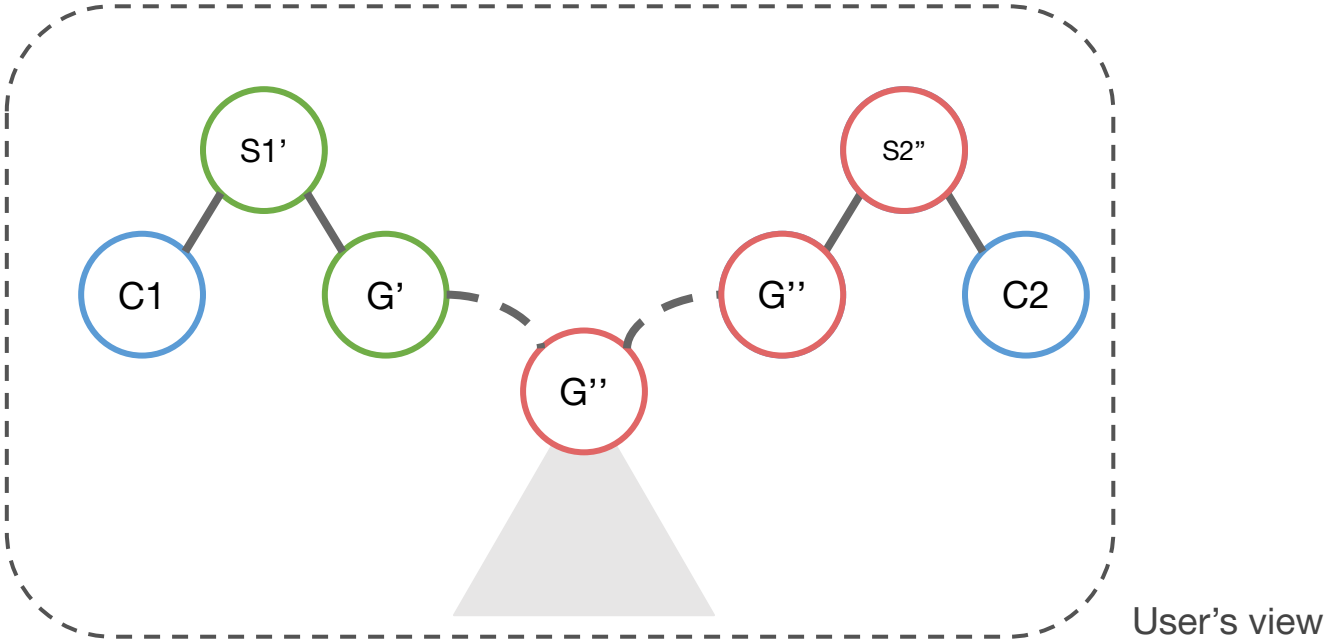
Compressed Multi-Roots Tree (CMRT)

- A user shares a message with chatbot 1.



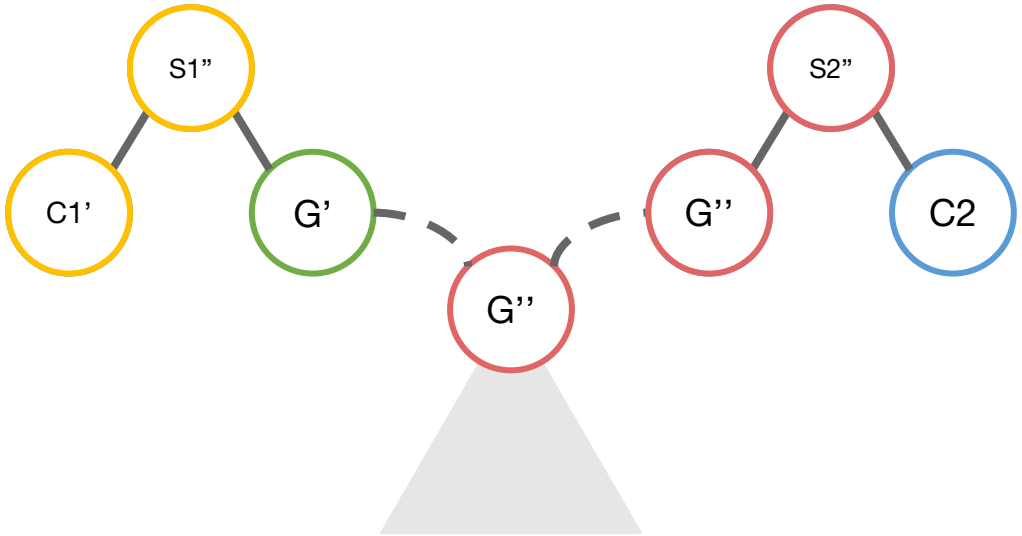
Compressed Multi-Roots Tree (CMRT)

- A user shares a message with chatbot 2.



Compressed Multi-Roots Tree (CMRT)

- Chatbot 1 sends a message.



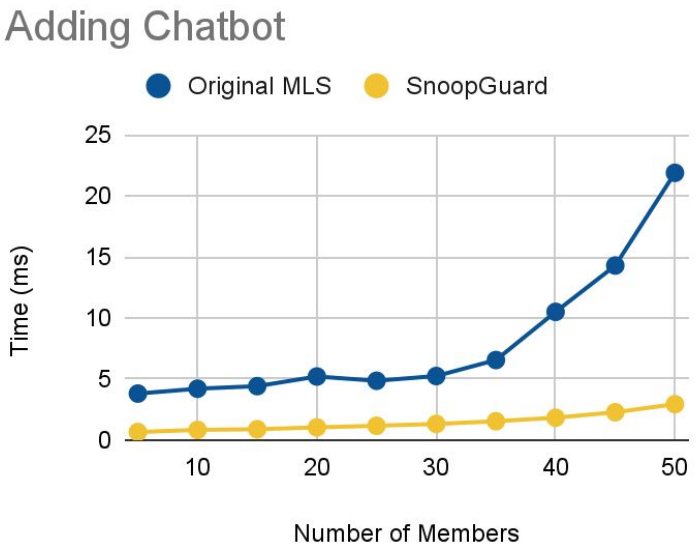
Theoretical Performance Analysis

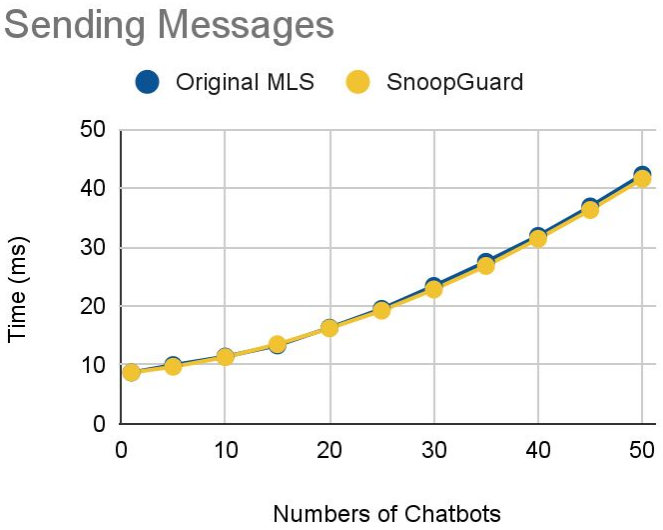
For a group with n users and m chatbots:

	Chatbot Addition	Sending a Message	Receiving a Message
SnoopGuard + MLS	***O(n+m)***	***O(log(n)+m)***	***O(log(n))***
MLS	*O(n+m)*	*O(log(n+m))*	*O(log(n+m))*
Sender Keys Protocol	*O(n+m)*	*O(1)*	*O(1)*

Empirical Performance Analysis

SnoopGuard introduces minimal overhead:



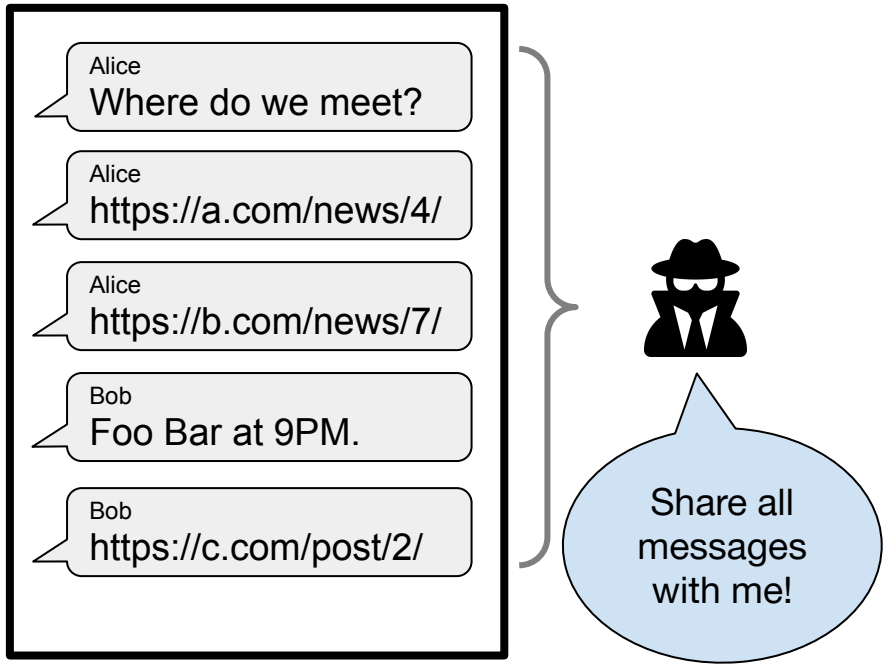


Limitation: Trigger Function Dependence

- Effectiveness of SnoopGuard relies on trigger function accuracy.
- Poorly designed or malicious functions could mark all messages as relevant.

Potential Mitigation:

- Vetting process to detect misuse.



Roadmap Toward a Privacy-Preserving Solution for Chatbots

1. Enhancing Usability
   a. Design intuitive interfaces for chatbot presence and permissions
   b. Provide clear workflows to grant/restrict access

2. Designing Permission Models
   a. Need dynamic, adaptive frameworks for evolving group dynamics

3. Understanding User Perceptions
   a. Study user trust, acceptance, and privacy concerns in group chats

Summary

- We showed that chatbots can learn irrelevant messages and identify senders.
- We highlighted the challenge of limiting chatbot access without breaking end-to-end encryption.
- We proposed *SnoopGuard*, a secure group messaging protocol that enables selective message access and sender anonymity while preserving end-to-end encryption.

Thank You