



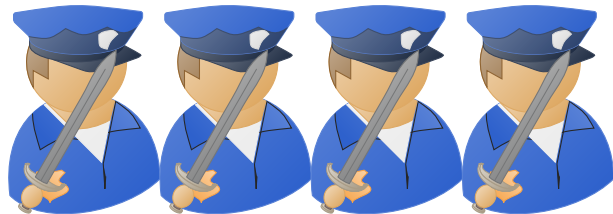
The Ins and Outs of Anonymous Group Communication

Christoph Coijanovic | April 16, 2026

Why do we need anonymous group communication?



versus

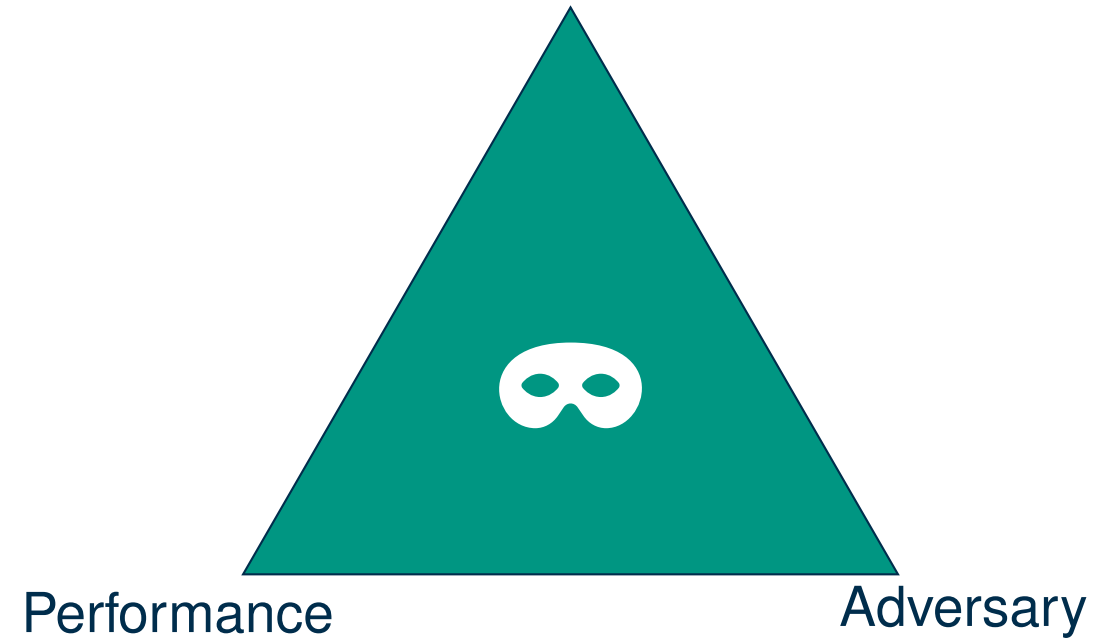


Political activists need a group chat to organize their protest

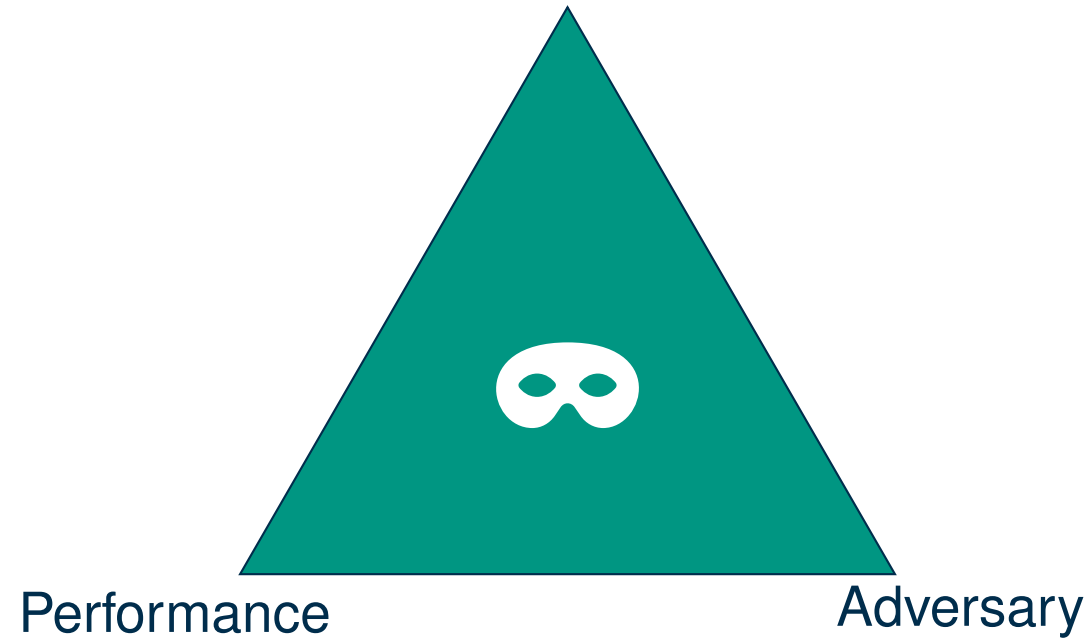
Standard encrypted messengers are not sufficient, as they allow the activists to be linked based on metadata

Let's build an anonymous group messenger to help them 🙌

Privacy Goal



Privacy Goal



Privacy goals for anonymous communication

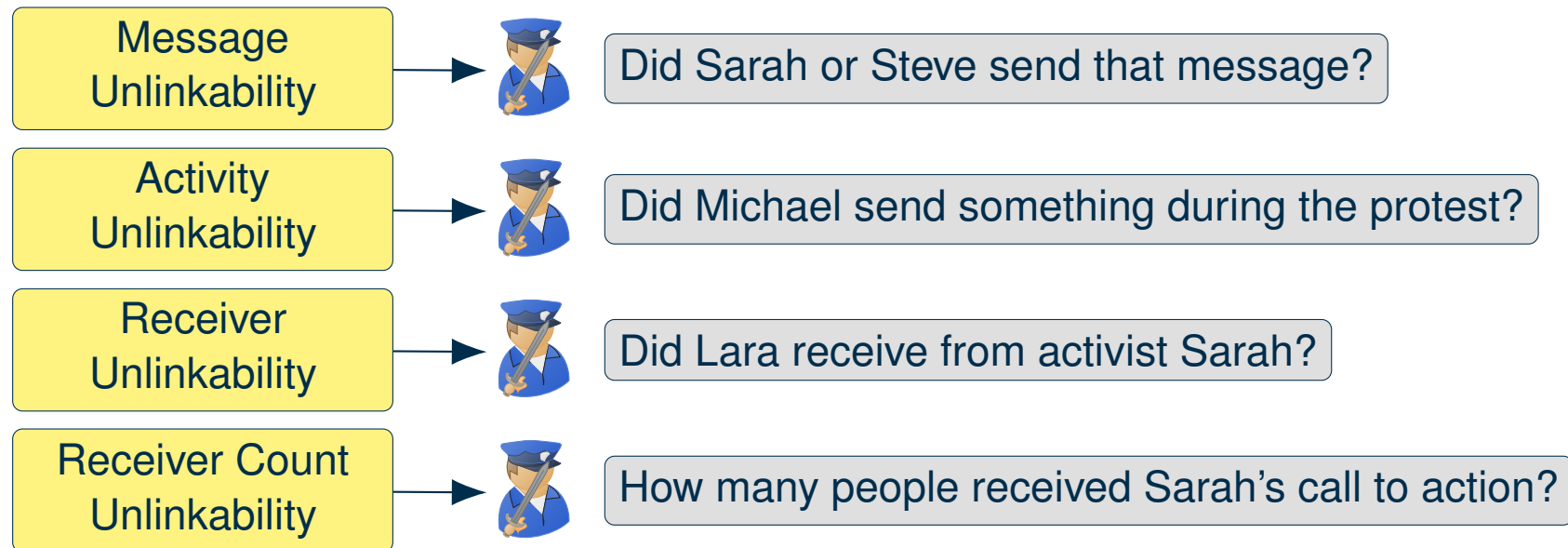
“Anonymity” is not unambiguously defined: Protocols can aim to protect different categories of metadata.



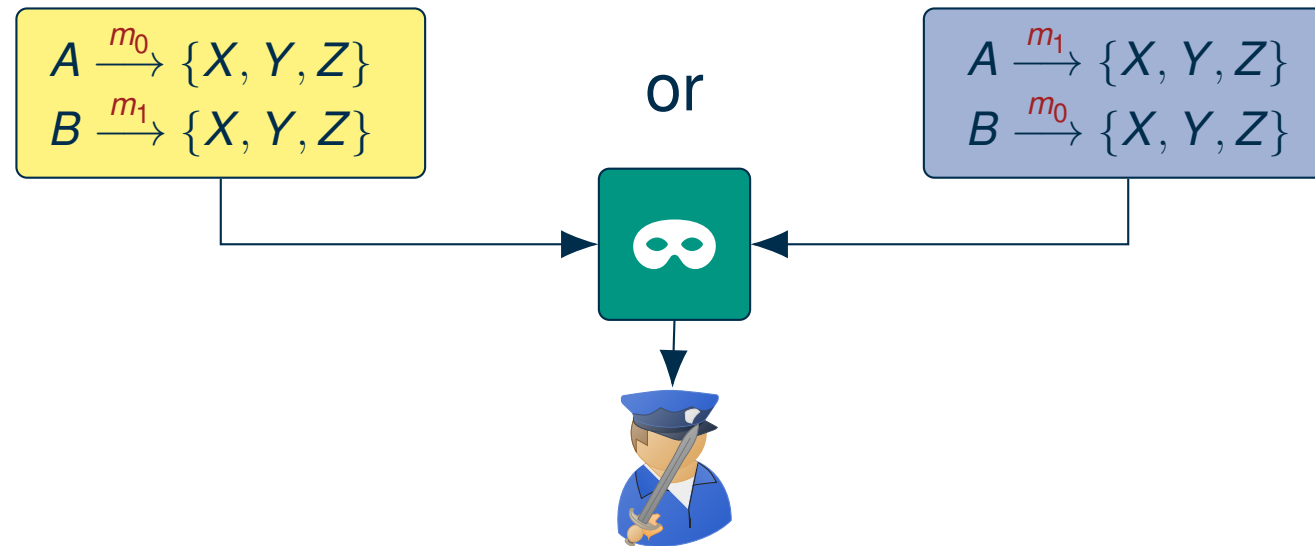
- 💡 We need generalized goals that are applicable to many systems
- 💡 We need formalized goals that enable provable privacy

Generalized privacy goals for anonymous group communication

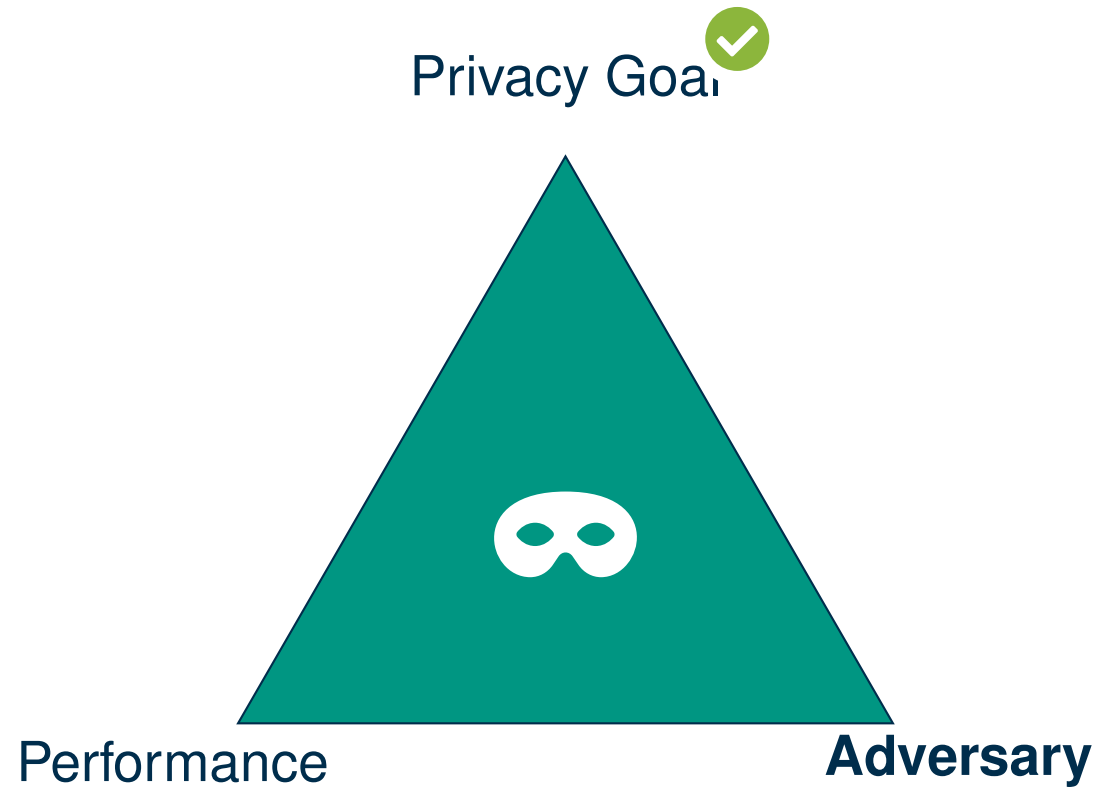
Most goals in literature can be approximated by the following four:



Formal privacy goals through an indistinguishability game



- **Goal:** Distinguish between two communication patterns (■ and ■) based on protocol output
- Patterns may only differ in the metadata that the goal aims to protect.
- Here: Who sends which message (Message Unlinkability)



Adversaries

- Depending on their abilities, adversaries can observe different metadata.
- Today, we focus on passive adversaries
- We distinguish between three classes

Global network adversary

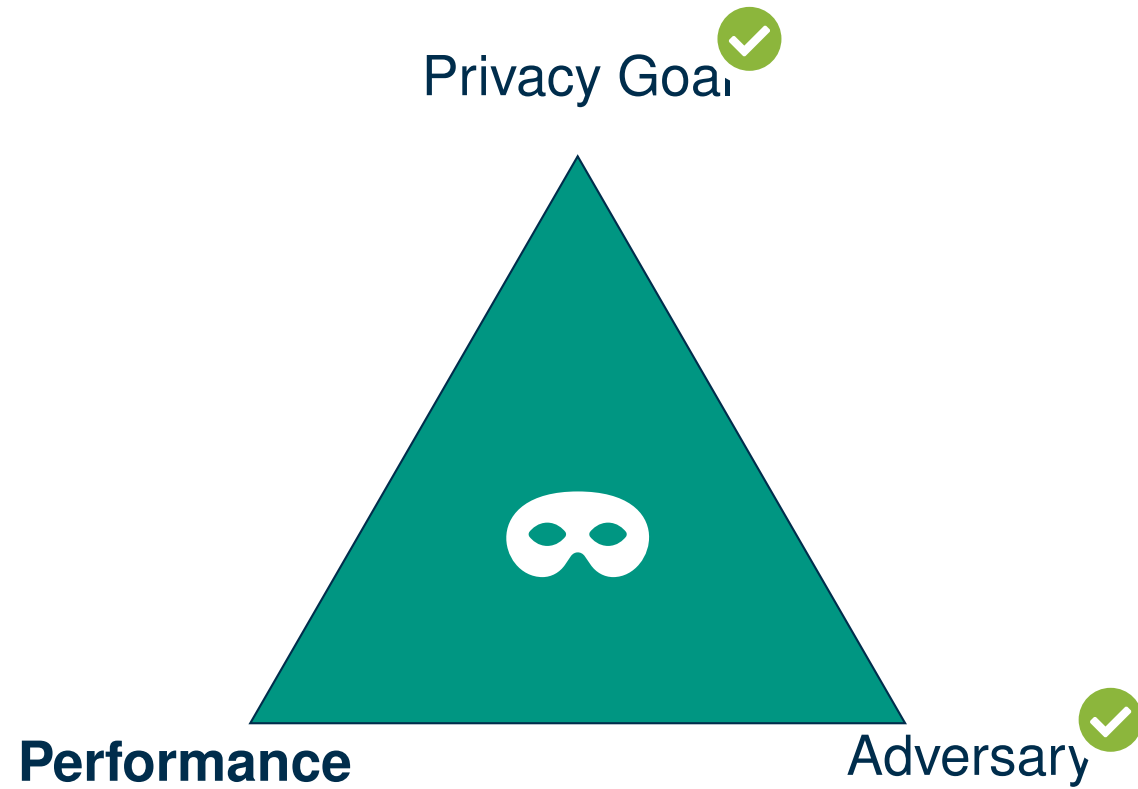


Malicious servers



Malicious group members





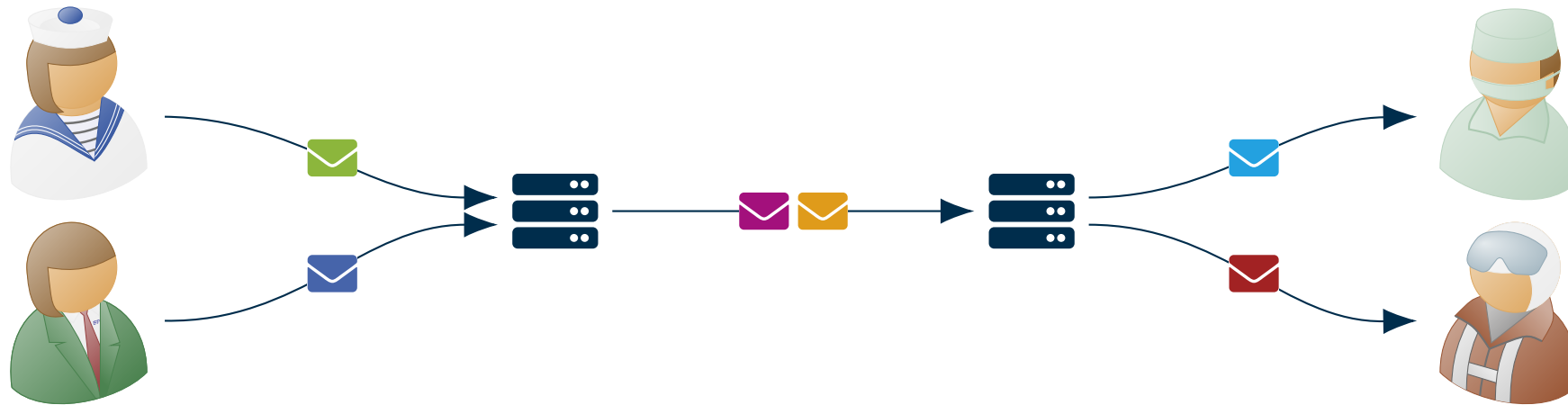
Performance

- Anonymous group communication networks hide metadata through cryptographic means
- Hiding metadata requires computational and/or network overhead, which impacts performance
- Concrete system performance is mainly determined by the underlying privacy primitive:

Mix networks

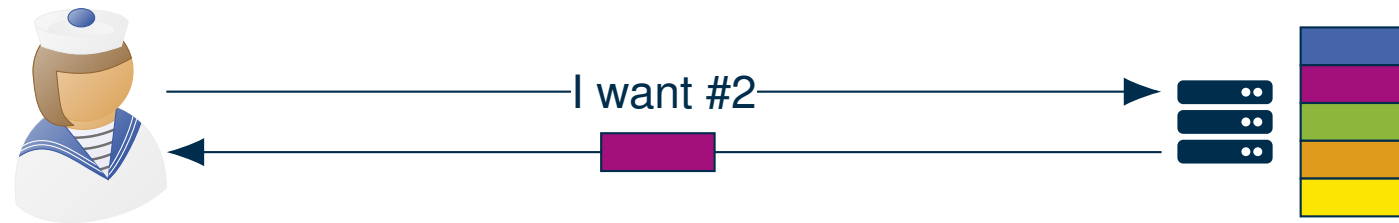
Private information retrieval

Primitive: Mix Networks

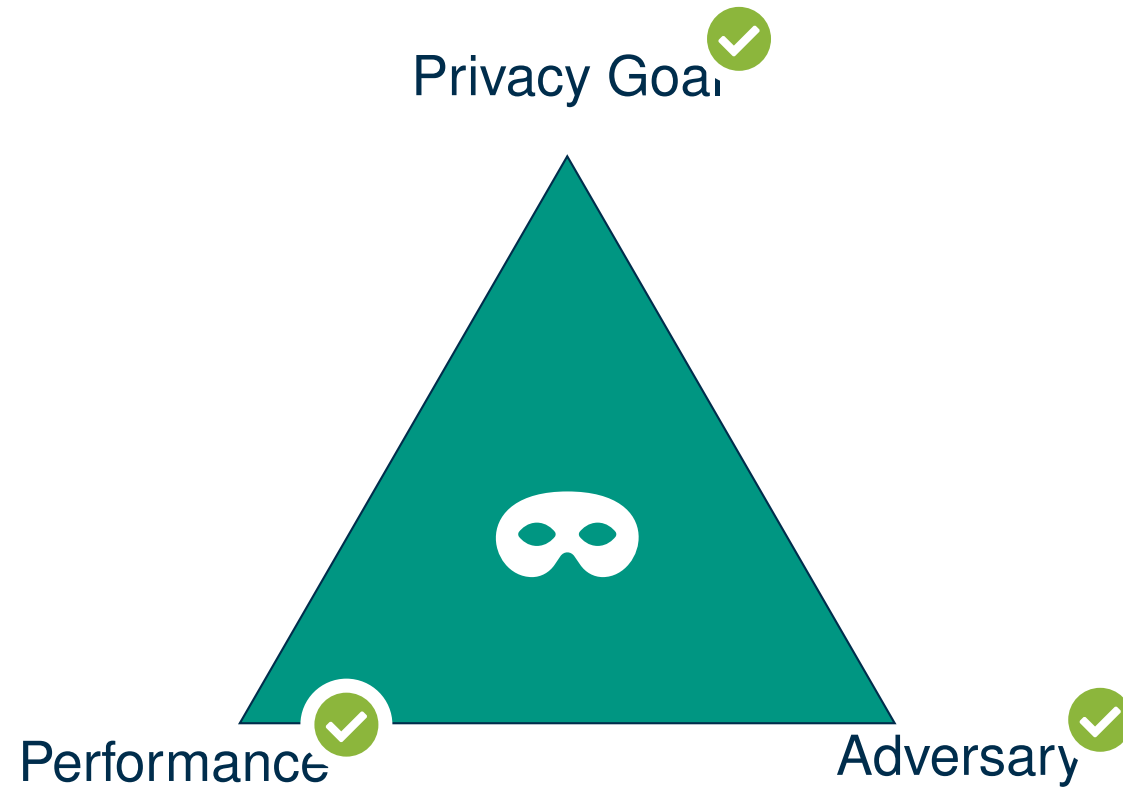


- 💡 Messages from multiple senders pass through a chain of servers
- 💡 Each server unlinks incoming from outgoing messages


Primitive: Private Information Retrieval



- 💡 Clients request items by index from server with remote database
- 💡 Server does not learn which item the client wants




Toy protocol Mix

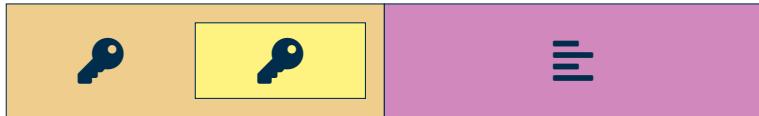
 Mix: Anonymous group communication based on mix networks.

- ⚠ Servers need to unlink incoming from outgoing messages
- ⚠ Message replication should occur within the network

Toy protocol Mix

 Servers need to unlink incoming from outgoing messages

 Prevent linking based on content




- Incoming = header + payload
- header is encrypted for current server
- header = key material + next header
- Outgoing = next header + encrypted payload

 Prevent linking based on order



Shuffle(, , )

- Collect incoming messages until threshold is reached
- Apply random permutation
- Release all at once

Toy protocol Mix

 Message replication should occur within the network (without losing unlinkability!)





- Incoming **header** contains multiple sets of key material and next headers
- Outgoing A = **next header** + **payload** encrypted under 
- Outgoing B = **next header** + **payload** encrypted under 

Toy protocol PIR

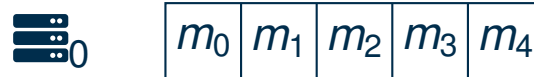
 PIR: Anonymous group communication based on private information retrieval.



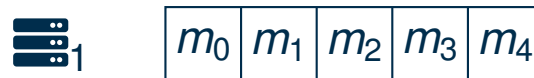
-  Prevent the server from learning information about the retrieved index
-  Clients need to know what to retrieve

Toy protocol PIR

 Prevent the server from learning information about the retrieved index



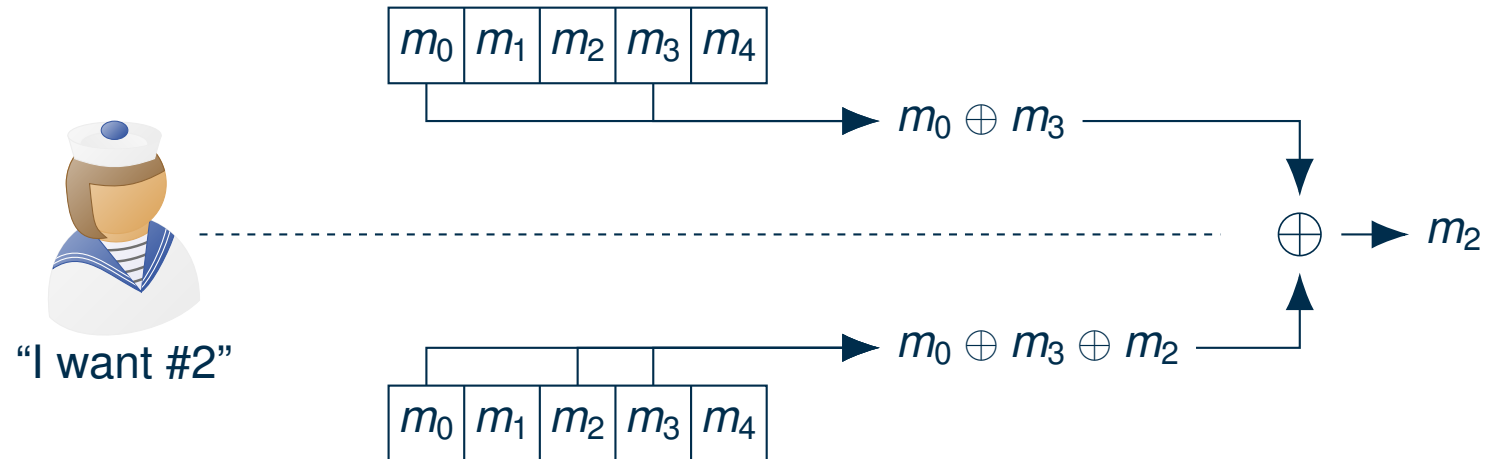
Duplicate database between two non-colluding servers



Toy protocol PIR

⚠ Prevent the server from learning information about the retrieved index

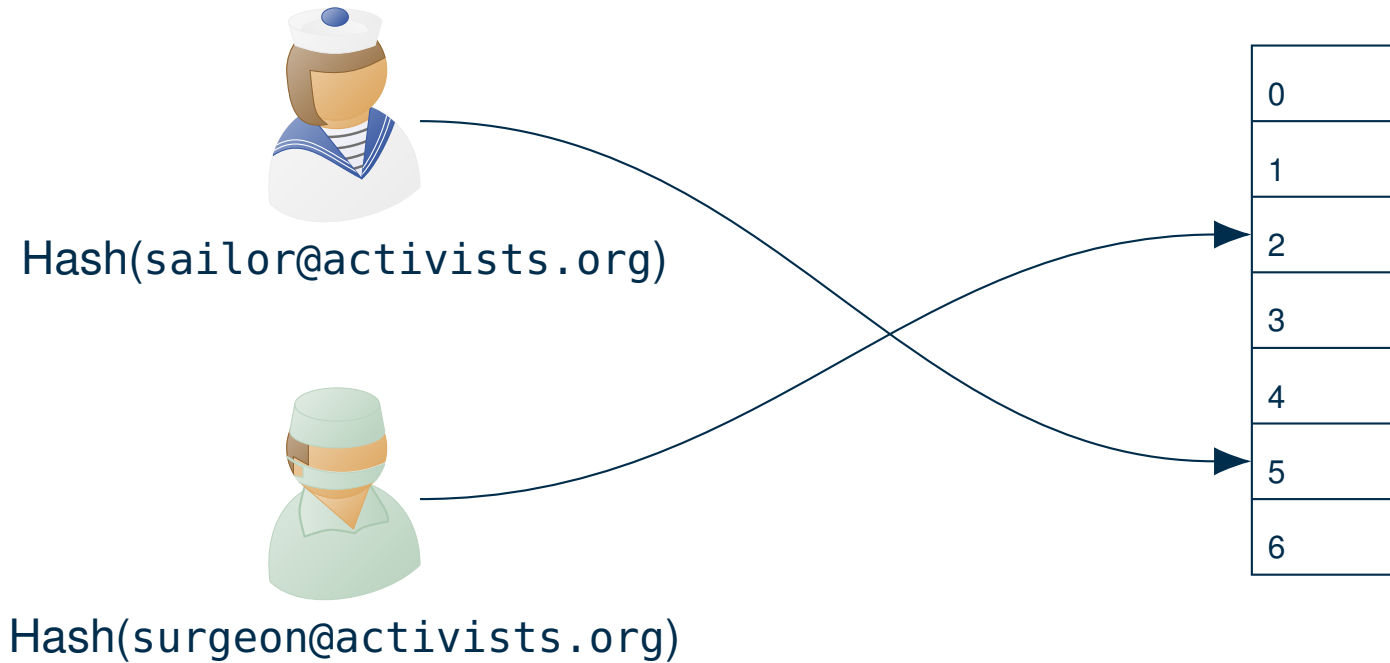
Request random set of indices from first server.



Request same random set with real index flipped from second server.

Toy protocol PIR

 Clients need to know what to retrieve



Mix versus PIR – Privacy

Mix

Message Unlinkability

- ▶ Network adversary ✓
- ▶ $S - 1$ servers ✓
- ▶ $G - 2$ group members ✓

PIR

Message Unlinkability

- ▶ Network adversary (w/ TLS) ✓
- ▶ 1 server (w/o E2EE) ✗
- ▶ S servers (w/ E2EE) ✓
- ▶ 1 group member ✗

Activity Unlinkability is achieved against the same adversaries as Message Unlinkability if cover traffic is used.

Receiver (Count) Unlinkability

- ▶ Network adversary ✓
- ▶ $S - 1$ servers ✓
- ▶ $G - 1$ group members ✓

Receiver (Count) Unlinkability

- ▶ Network adversary (w/ TLS) ✓
- ▶ $S - 1$ servers (w/o E2EE) ✓
- ▶ $G - 1$ group members ✓

Mix versus PIR – Performance

Mix

- + Symmetric crypto → computationally cheap
- + Adding additional servers can increase capacity
- × Shuffling incurs high end-to-end latency

PIR

- × Oblivious processing → computationally expensive
- × Adding additional servers only distributes trust
- + Low latency if sufficient computation
- + There is also PIR w/o server trust (more computationally expensive)

Conclusion

Anonymous group communication is cool!

Both  Mix and  PIR can protect our activists 

There are many (more) approaches, each with different tradeoffs.