Google

Interoperable Private Identity Discovery for E2EE Messaging

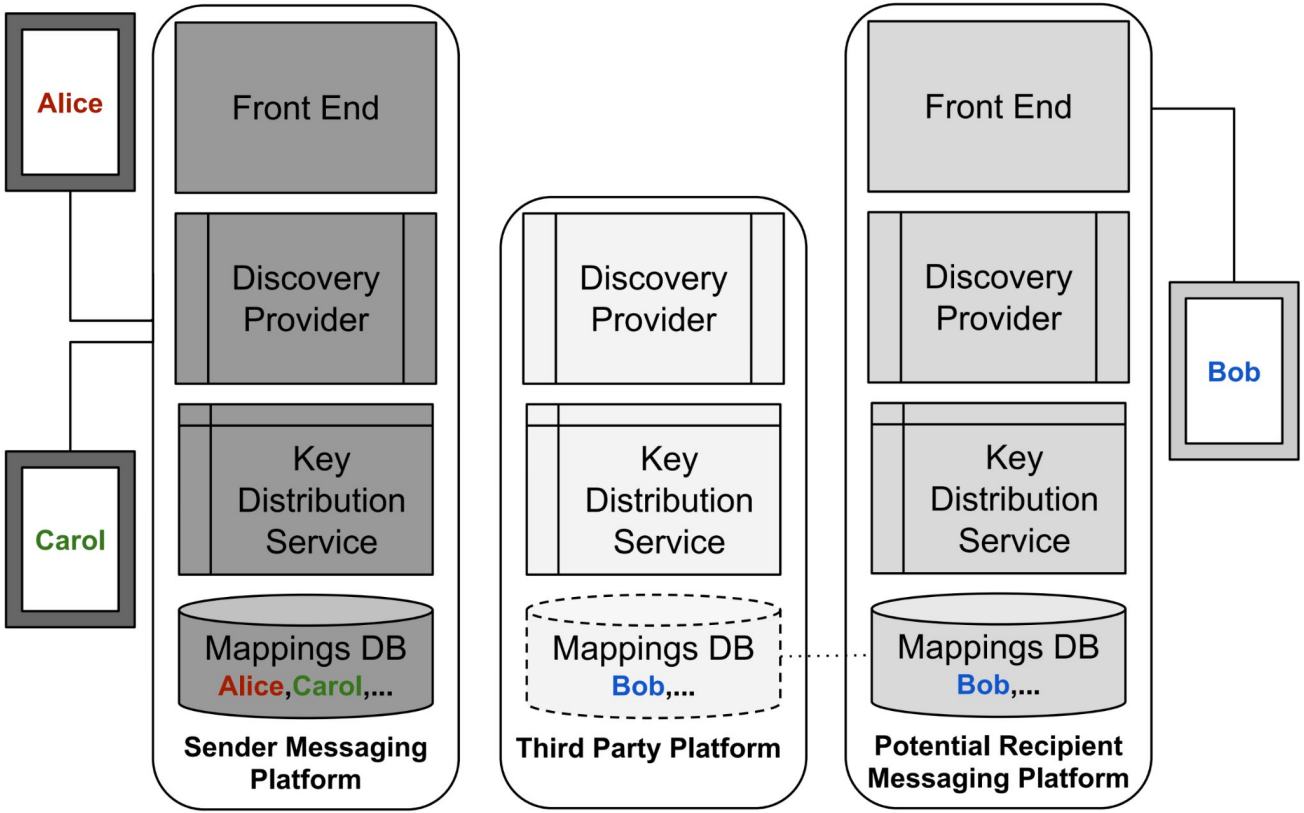
Draft proposal for Mimi

Giles Hogben, Femi Olumofin

Discovery problem statement

- Resolving the SSI that a user's SII maps to, while preserving privacy
- Alice performs discovery of Bob's SSI, using Bob's SII
- Challenge: minimize leakage of possible connection/relationship between Alice and Bob

Threat actors



- Alice & Carol register with SMP
- Bob registers with PRMP
- PRMP is in contract with TPP
- … indicates replication or online query

Mapping data access options for DP:

1. Acquiring mapping copies from partner DPs
2. Querying partner DPs on an as-needed basis
3. Solely relying on its own database with potential limitations in resolving scope

Privacy requirements

1. **Social graph**: Discovery service providers should not learn the SII a user is querying for unless they are sending or receiving a message to that user
2. **Querying user identity**: A discovery service provider should not share the querying user identity with other discovery services when it requires their help for discovery
3. **Metadata**: Discovery service should not learn the exact timing of when a message is sent (after discovery) unless they are sending or receiving the message

Requirements by threat actors

Service	Minimum privacy requirements	Optimal privacy requirements
Sender Platform	Do not hide SII	Hide SII 🔺
Recipient Platform	Do not hide SII	Do not hide SII
Non-recipient Platform with SSI	Hide SII	Hide SII
Non-recipient Platform without SSI	Hide SII	Hide SII
Third party service	Hide SII	Hide SII

- 🔺 **Issue**: Clients and sender platforms will perform discovery for contacts they never message. Disclosing the discovered SII to the sender's platform during discovery is premature, hurts privacy
- **Optimal privacy**: Hide SIIs during discovery until they are used for E2EE messaging

Requirements by threat actors

- Without hiding SSI, discovery enables threat actors to aggregate users' social graph fragments across different services
- Minimum requirement
  - Hide the queried SII from all actors except the Sender & Recipient platforms
- Optimal requirements
  - Hide SII from all except the Recipient platform
- Rationale
  - Spam prevention requirements only apply to sent messages (not discovery)
  - Standard IP based techniques will be effective DDoS mitigation for discovery services
  - Client costs for SSI hiding mechanisms scale well with database size + number of services

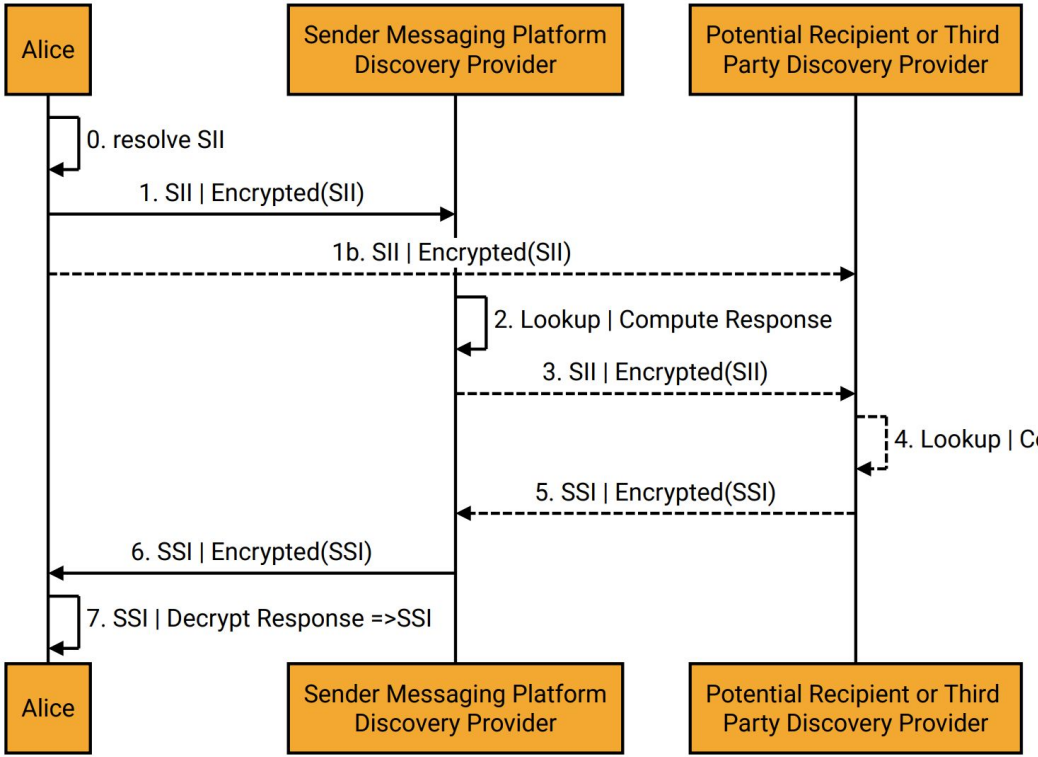
Privacy non-requirements

1. **Hiding SII <> service mapping**: Hiding service reachability or the existence of a mapping between an SII and SSI for a service provider is an explicit non-goal. All major E2EE messaging services already publish unACL'd reachability information without opt-out i.e. +16501234567, reachable on Messages, Whatsapp, Telegram (not including name or any other info)
    - Should not be a privacy goal (and would not be feasible to implement)
    - **However it may be a business goal to prevent scraping of the full list of account-holders**
2. **Contact/user lookup by name** or anything except an SII

Other non-functional requirements

1. No single entity should be financially responsible for resolving all discovery queries (e.g. even within a geographical region)
2. Costs for each participating entity of storing and resolving SII should be proportional to their number of participating users
3. Performance should support each client device resolving users' contact SIIs at least once every 24 hours

SSI discovery



Notes:

- Alice's identity as the request originator should remain undisclosed to DPs
- Alice is not required to hide discovery requests when the processor DP is within the Sender Messaging Platform
- Alice's client may, but is not required to hide discovery requests from Potential Recipient DPs (optimal vs. minimal reqs)

- With privacy
  - Private Information Retrieval (PIR)
  - Private Set Membership (PSM)

Private Information Retrieval (PIR) - discovery option 1

- PIR allows Alice to privately discover the SSI associated with an SII
  - The DP cannot any additional information about which mapping
- Lattice-based PIR framework applicable with standard schemes, including open source
- DP holds 10BN mappings, 1.28TB size, 10k shards -> 1M mappings each

Parameter/Metric	**Cost estimate**
Server Storage Per Device	14 MB
Client Device Storage (for 10 billion records)	5 MB
Upload Bandwidth Per Query	14 KB
Download Bandwidth Per Query	21 KB
Client Time Per Query	0.1s
Server Time Per Query (Single Thread)	0.8-1s

Private Set Membership (PSM) - discovery option 2

- PSM allows a client to obtain the associated SSI from a DP without revealing the SII or whether a match was found
- An open source PSM implementation is similarly available
- DP holds 10BN mappings, 1.28TB size, 1k shards -> 10M mappings each

Parameter/Metric	**Cost estimate**
Client Device Storage (for 10 billion records)	1 MB
Communication	2.8 MB
Client Time Per Query	0.1s
Server Time Per Query (Single Thread)	1-2s

PIR vs PSM for discovery privacy

	PIR	PSM
Computational problem basis for privacy guarantee	Ring Learning with Errors (RLWE)	Ring Learning with Errors (RLWE)
Application	Suitable for retrieving records of any number of bytes	Suitable for retrieving identifiers or SSI of 256, 128, or fewer bytes
Client learning bounds	Clients can learn about l SSIs by making only k requests, where k is less than l Symmetric PIR schemes limit client learning, but at a higher cost	Guarantees clients do not learn any information about other mappings in the DB
Rate limiting and DDoS prevention	Logged in usernames and IP addresses may be used	Can leverage cryptographic hash functions expensive to compute (e.g., Argon2, Script) by clients

Trusted authorities for mapping SIIs to SSIs

Which actors should be trusted authorities for mapping SIIs to SSIs?

- MSPs are trusted authorities for creating mappings
  - Mapping creation should be considered out of scope for this proposal
  - MSPs should verify ownership of SIIs (OTP code to phone via text or call, or email)
- MSPs may share mappings with 3P discovery providers
  - Delegate discovery providers should be lookup providers only
- A 3P DP may also authenticate mappings or act as a pass-through for signed mappings for an MSP or another identity provider
- TLS is sufficient to authenticate the mapping assertion

Discovery scaling

Does discovery need to scale to accommodate 10s, 100s, or 1000s of service?

- Discovery requests should be sent to specific MSP DP attached to a messaging client or a 3P DP
- Providers decide how to process requests; fan-out or use own mappings
- Uncontrolled fan-out can be costly and may lead to DDoS patterns (series of recursive requests with possible loops across multiple DPs)
  - Mitigate by accompanying each request with a fan-out depth limit and UUID
  - DPs will be motivated to restrict fan-outs for discovery with encryption given the attendant compute costs for response processing
- Nonetheless, the protocols should be feasible (in terms of computation and communication cost) for 1000s of services

Acceptable leakage for discovery

What is it acceptable for queries to reveal about the social graph, and to whom?

- A query **should not reveal** the SII in a user's query to discovery providers unless the discovery provider is also within the Sender's platform or the Recipient's platform with the SSI mapping
- An encrypted query doesn't leak any information about the SSI. However, a small amount of leakage w.r.t the shard or mapping subset is acceptable to achieve high performance in billions-scale mappings DB
    - We take 1 out of a random million as a sufficient minimum level of privacy for indistinguishability of the SSI
- Returning an SSI set of different cardinalities leaks information to a discovery provider about the likely sets of SSIs that are of interest for a query
- A one-to-one mapping of SII to SSI does not leak such information
- A discovery provider cannot tell when a privacy-preserving discovery returns an empty result or a single SII. However, it will be able to tell when a large number of SSIs are returned

Rate limiting

Is rate limiting useful to prevent scraping?

- Discovery providers should consider rate-limiting to mitigate leakage of their mappings DB, and computational costs for processing requests
- Users should be able to look up at least 50 SII per discovery provider per messaging provider in a 24-hour period
- Third-party discovery providers are exempt from the minimum discovery load per user requirement unless required by their contract with MSPs

Multiple SIIs mapping and query routing

- *An SII may map to multiple SSIs. Should the requestor learn all of them, and if so, how?*
  - *One service that returns all SSIs for an SII?*
  - *Query each service provider independently?*
  - *User figures out out-of-band what service provider to query?*
- An SII may map to multiple SSIs within a single MSP, but is not recommended:
  - **MSPs**: Provider's choice to allow within a single MSP
  - **Privacy**: Multiple SSIs makes privacy challenging (response size fingerprints possible interest)
  - **Users**: May not want to group multiple SSIs together for privacy reasons
  - **Indexing**: A scheme could be devised where an SII is suffixed with an index during registration and discovery
    - Example: +1234567890, a user may map +12345678900 to the first Whatsapp SSI, and +1234567891 to the second Whatsapp SSI and so on
- Users Should figure out out-of-band what DP should process a query
- DPs should not be required to fork out discovery requests to other providers (optional)

Cross service identity spoofing

- Messaging services currently use various identifiers like email addresses, phone numbers, or service-specific usernames
- Cross service identity spoofing and impersonation arises with interoperability, as user identities may not be unique across different platforms
  - Alice messages Bob at bob@Threema
  - Eve messages Alice impersonating Bob using bob@FooService
  - Alice needs some indicator or UI to know that bob@Threema isn't bob@FooSercice and that when bob@FooService messages, it should not be assumed that bob@FooService is bob@Threema
- Options for solving
  - SII must be globally unique or fully qualified (cannot be a per-service username)
  - SII registration should include a step to store supported services for each contacts (in Contacts/Address book
  - Treat messages from unknown senders as spam or untrusted

Questions ?

https://datatracker.ietf.org/doc/draft-party-mimi-user-private-discovery/