Google

Mimi Discovery Requirements

Giles Hogben, Femi Olumofin

Functional Requirements

For a given messaging service identity handle (phone number or alphanumeric UserID):

1. **Endpoint discovery**: discover receiver service IDs to retrieve public key material and send message payload e.g. Platform1.org/send/, Platform1.org/kds
2. **Default service discovery**: Discover optional default receiver service ID user preference for a given PN/UserID (e.g. `default:Platform1.org`)
3. **Global uniqueness**: Fully-qualified service identifiers should be globally unique
4. **(P1) Key verification**: Provide an independently trusted party to assert and verify the association between a public key and a UserID

Privacy Requirements

1. **Social graph**: Resolver or discovery services should not learn the PN/UserID a client is querying for (i.e. who is sending a message to who)
2. **Querying user identity:** A resolver service should not default to sharing the querying user identity with other resolver services when it requires their help for discovery
3. **Metadata**: Resolver service should not learn the exact timing of when a message is sent

Non-requirements

1. **Hiding service reachability:** (the link between a UserID and a service). E.g. +16501234567, reachable on Messages
    - All major E2EE messaging services already publish unACL'd reachability information without opt-out, Whatsapp, Telegram (not including name or any other info)
2. **Hiding the value of UserIDs or public keys:** e.g. the existence of the PN, +16501234567
3. **Hiding the association between a public key and a UserID:** e.g. PN +16501234567 has pubkey x
4. **Contact lookup by name** (or anything except username)

Other non-functional requirements

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