MIMI Delivery Service

`draft-robert-mimi-delivery-service`

IETF116, Yokohama
Raphael Robert

Architecture

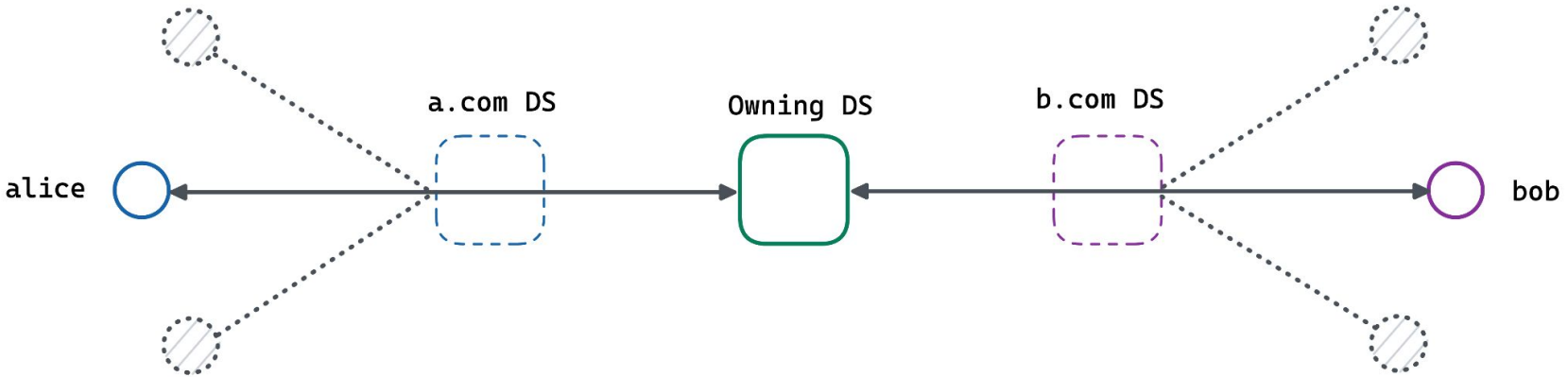
MLS has two dependencies:

- Authentication Service (AS)
- **Delivery Service (DS)**

Architecture

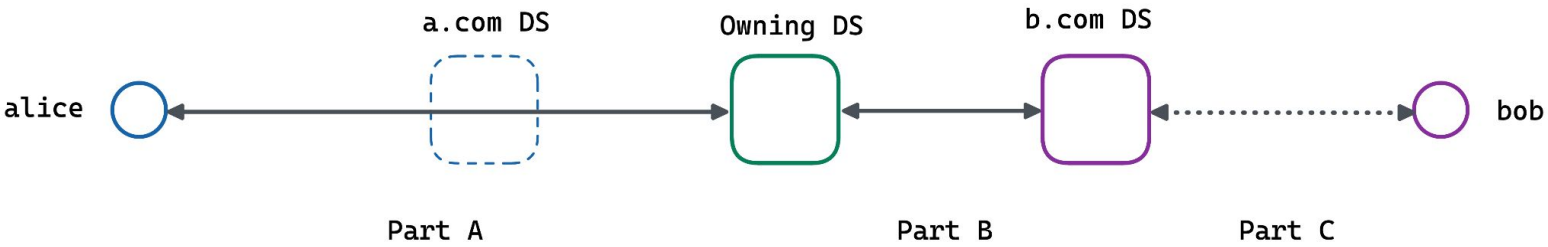
The Delivery Service comes in two flavors:

- **Consistent and Partition-tolerant, or Strongly Consistent**
- Available and Partition-tolerant, or Eventually Consistent



Network diagram of clients and a DS in a federated environment

Scope



- `draft-robert-mimi-delivery-service` is about the protocol between clients and the DS, and DS to DS (part A + B)
- It is independent of the underlying transport protocol (e.g. HTTP, XMPP, Matrix, etc.) as long as a request-response scheme is supported for Part A, uses TLS encoding

Protocol overview

Supports all operations that can be done with MLS:

- Send messages
- Inspect group membership
- Adding/removing members
- Updating key material
- Joining a group (via Welcome, External Commit, External Member Proposal or New Member Proposal)
- 1:1 connection requests
- Resync (when clients get corrupted)

Protocol overview

The protocol has the following capabilities:

- Multi-device support (more than one device per user)
- Can enforce ACLs server-side (e.g. admins vs regular members)
- Can do multi-level rate-limiting
- Distinguish between 1:1 connections and groups
- Consent-based connection request
- Spam protection

State on the Delivery Service

The DS holds the following information:

- A list of all groups it hosts
- For each group, it holds
  - The MLS ratchet tree
  - The GroupInfo (including a signature)
  - Information needed for per-client/per-user authentication and message fanout
- KeyPackages for
  - 1:1 connections
  - Groups

Privacy-preserving Delivery Service

- The client to DS protocol supports the same functions, only the syntax is slightly different
- The state on the DS cannot contain sensitive metadata:
  - The credentials/identities are replaced by pseudonyms
  - KeyPackages for groups are also pseudonymous
  - Part of the rate-limiting is enforced by using Privacy Pass
  - Identifying data, like e.g. push token is encrypted at rest
  - Optionally, encryption-at-rest can be used for all of the group state