MLS Federation

IETF 104

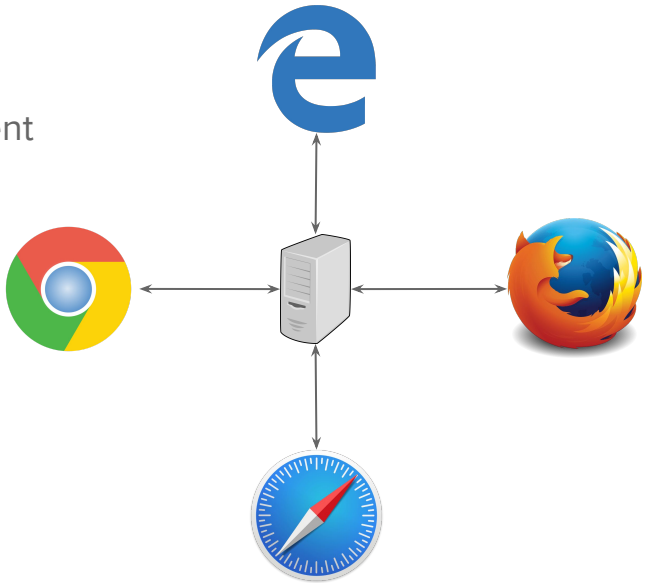
Emad Omara

Goals

- Standardize the minimum information needed to allow different MLS clients to encrypt/decrypt messages to each other
- The system should support single (shared) or multiple delivery services

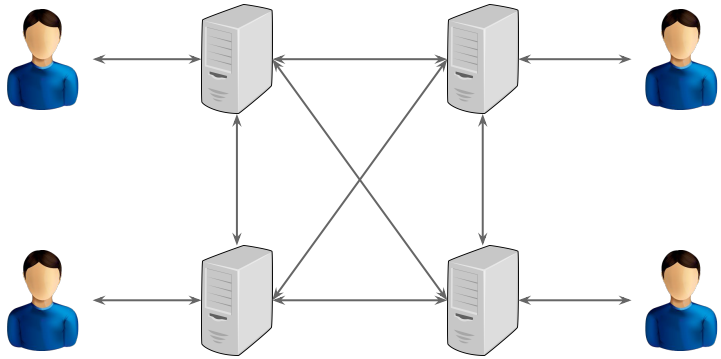
Use cases

- Different clients and Single (shared) delivery service
  - E2EE Web applications
    - Web developers write applications that support E2EE messages between peers
    - MLS implementation should be in the browser
    - Apps interact with MLS through JS APIs
  - E2EE Conference call
    - MLS in the browser can be used as a key management for E2EE multi party conference call (MLS-SRTP)



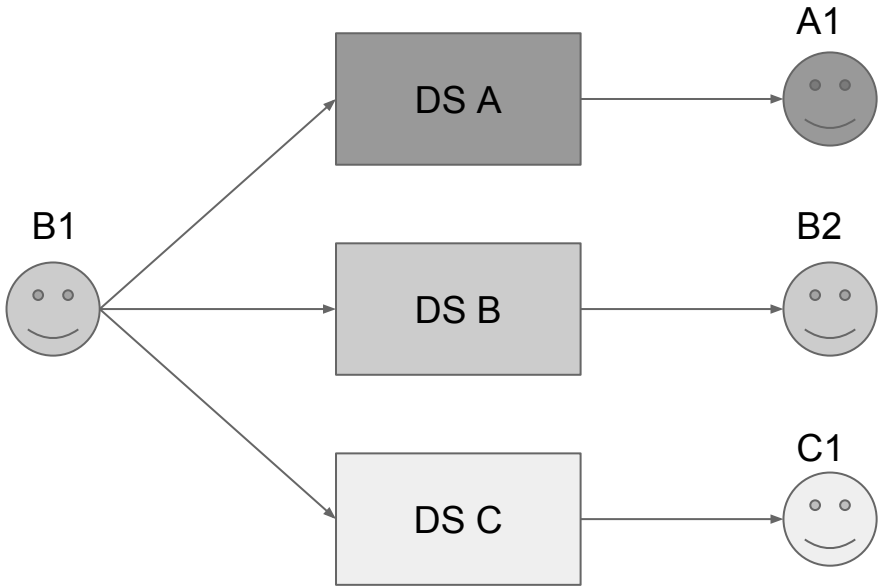
Use cases

- Different clients and different delivery services
  - E2EE messaging across different systems
    - Each messaging provider has its own server and clients
    - Clients from different providers can encrypt/decrypt to each other
  - Can be done in two ways
    - Client-side fanout
    - Server-side fanout



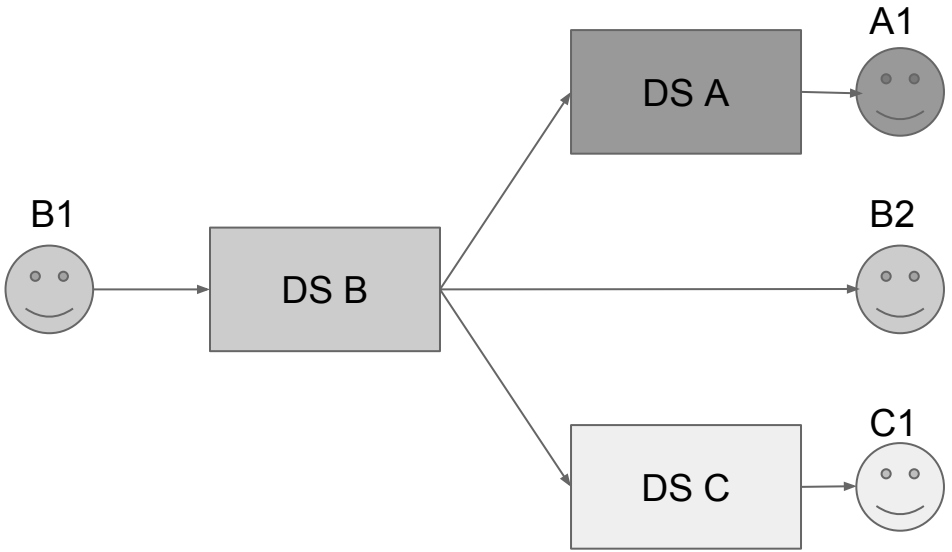
Client fan-out

- Client B1 establishes multiple connections with different servers to retrieve keys and deliver messages
- May not scale!



Server fan-out

- Client B1 establishes one connection with its server
- Operator B server will proxy all key requests to other servers and fan out the messages



Scope

- Only focus at the crypto layer

- Mapping between userID to DS is up to the application

- How different DS communicate is up to the application *unless it becomes necessary!*

Requirements

- Define the format of the Identity/Init key request & response
  - Request should include
    - Remote user id
    - Local supported ciphersuites
  - Response should include
    - List of InitKey bundles (one for each device)
    - Each init key bundle will also have the long term identity key

Challenges

- Version/Ciphersuites negotiation
  - Should MLS do explicit version/ciphersuites negotiation in addition to the userInitKey negotiation?
- Handshake message ordering
  - Ordering is critical for TreeKem, how it can be enforced with multiple DS?
- Metadata sharing
  - In case of multiple deliver services, how the group state and other metadata are synchronized across all of them?
- Multi-device
  - How clients get notified when other clients controlled by other DS add/remove devices ?

Adopt or not to adopt?