Inside MLS Message Interop

dispatch/artarea meeting, IETF113

21-Mar-2022

Rohan Mahy, Wire
rohan@wire.com
Wire: rohan_wire

What is MLS?

- Messaging Layer Security protocol defined in MLS WG (sec area).
- It is an efficient group-keying protocol which encrypts application data to the group members at that moment.
- Strongly motivated by group chat/IM applications wanting efficient group security with security properties similar to Double Ratchet (used in Signal, Telegram, WhatsApp, Wire, etc).
- Supports groups with members from different federated systems.

- Status:
  - Multiple independent implementations, ex: Cisco and OpenMLS
  - Preparing for Working Group Last Call now

Why do we need interoperability inside MLS application data?

- IM Customers already asking for federation now (communication among domains). Interop of “application data” inside MLS cannot be far away.
- To have interoperability of e2e encrypted application data, we need a **common format** and a **way to negotiate it**.

- The good news: We already have the Common Presence and Instant Messaging format (CPIM) that partially addresses this problem.
- The bad news: The common IM features have changed since CPIM, and the assumption about what end-to-end security means (MLS vs. S/MIME or PGP).

draft-mahy-dispatch-immi-mls-mime: Negotiate

- The base MLS protocol does not contain any way to specify the format of its “application data”
- We probably need three things to do basic negotiation in MLS:
  - A way to communicate the format of an application message
    - (proposed in https://github.com/mlswg/mls-protocol/pull/605)
  - A way for a client to specify all the MIME types they support (MLS KeyPackage extension)
  - A way for a group to specify MIME types which must be understood to participate in that group (MLS GroupInfo extension)

- Are these useful?
- If so, where should they live?

draft-mahy-dispatch-immi-content: Convey

- A profile which can represent common IM features, mostly using specs/semantics we (IETF) already had lying around.
- Features:
  - plaintext and rich text messages
  - replies, reactions, mentions, and knock
  - edit / delete previously sent messages
  - expiring messages
  - read/delivery receipts
  - files/attachments
- Easy to use this as a common format and include a fancier or proprietary format in the same group.

- Is there interest in defining a way to solve this problem?
- If so, where should they live?
- Please feel free to comment on specifics in my draft

Thank you!

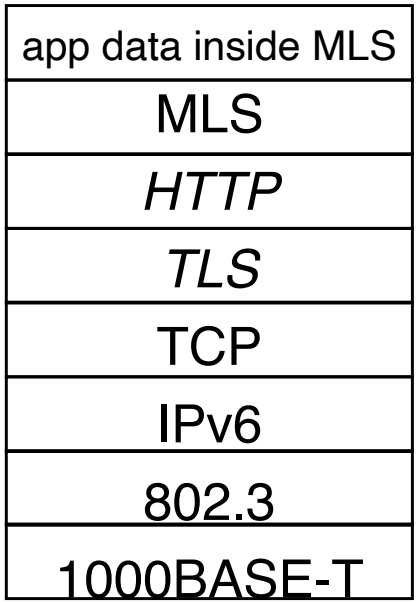
Questions?
Comments?

Sending common and proprietary formats simultaneously

```
Content-type: multipart/alternative; boundary=XcrSXMwuRwk9

--XcrSXMwuRwk9
Content-type: message/cpim

From: <im:alice-smith@example.com>
DateTime: 2022-02-08T22:13:45-00:00
Message-ID: <28fd19857ad7@example.com>

Content-Type: text/plain; charset=utf-8

Test Message
--XcrSXMwuRwk9
Content-type: application/vnd.examplevendor-fancy-im-message

<content of example vendor's fancy format>
```

Proposal:
Blue is not sent in the MLS application data. Green is sent as an optional MLS field before MLS application data.

Example network stack



Communication between MLS client and "Distribution Service" is not defined, but HTTPS is a reasonable implementation choice.