

Workgroup:
More Instant Messaging Interoperability
Internet-Draft:
draft-ralston-mimi-matrix-transport-01
Published: 9 May 2023
Intended Status: Informational
Expires: 10 November 2023
Authors: T. Ralston
The Matrix.org Foundation C.I.C.
M. Hodgson
The Matrix.org Foundation C.I.C.
Matrix Message Transport

Abstract

This document describes Linearized Matrix's transport considerations.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at <https://turt2live.github.io/ietf-mimi-matrix-transport/draft-ralston-mimi-matrix-transport.html>. Status information for this document may be found at <https://datatracker.ietf.org/doc/draft-ralston-mimi-matrix-transport/>.

Discussion of this document takes place on the More Instant Messaging Interoperability Working Group mailing list (<mailto:mimi@ietf.org>), which is archived at <https://mailarchive.ietf.org/arch/browse/mimi/>. Subscribe at <https://www.ietf.org/mailman/listinfo/mimi/>.

Source for this draft and an issue tracker can be found at <https://github.com/turt2live/ietf-mimi-matrix-transport>.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents

at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 10 November 2023.

Copyright Notice

Copyright (c) 2023 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

- [1. Introduction](#)
- [2. Transport Requirements](#)
- [3. Interoperability](#)
- [4. Encryption](#)
- [5. Security Considerations](#)
- [6. Informative References](#)
- [Acknowledgments](#)
- [Authors' Addresses](#)

1. Introduction

Linearized Matrix [[I-D.ralston-mimi-linearized-matrix](#)] describes an API surface for accessing Matrix rooms in a simple, easy to implement, fashion. Normally a Matrix room would be a Directed Acyclic Graph, requiring servers to implement transport functions to fill gaps, however with Linearized Matrix's API surface the transport considerations are reduced to simply passing events (messages) between servers.

2. Transport Requirements

TODO: This section needs a lot more words.

*HTTPS+JSON for ease of use (or similar)

*Should be reliable (not require constant connection, resilient to network faults where possible)

*Should be simple to understand (RESTful, for example)

*Exact details TBD.

3. Interoperability

Matrix's split of Federation and Client-Server APIs allow homeservers to implement the API surface which is most relevant for its application. For interoperability, only the Federation API is relevant. The APIs have been designed to intrinsically support load balancing and active/active horizontal scaling - for instance, it's valid for different parts of a server to race together when sending a message in a room, avoiding the need for global locks within the server.

TODO: Is this still true with Linearized Matrix? Can we really avoid locks, or are we going to need both active servers to behave the same?

The steps needed for an existing system to be interoperable with another over Matrix would mean implementing portions of the Federation API.

TODO: Define that minimized API surface (much of the Federation API is actually for DAGs, which we don't care about here).

4. Encryption

End-to-end Encryption is deliberately layered on top of the Matrix transport (Client-Server or Federation APIs). Currently a combination of Double Ratchet (Olm) encryption and group ratchet encryption (Megolm) is specified in the End-to-End Encryption section of the Client-Server API [[CSEncryptionApi](#)], but Matrix over MLS [[I-D.ietf-mls-protocol](#)] (with minor bookkeeping to compensate for the lack of a centralised sequencing function in Matrix) is being specified as DMLS. [[DMLS](#)]

5. Security Considerations

TODO Security. Matrix has its own threat model that needs to be described here to protect against malicious actors (at the transport level).

6. Informative References

[[CSEncryptionApi](#)] The Matrix.org Foundation C.I.C., "End-to-End Encryption | Client-Server API", 2022, <<https://spec.matrix.org/v1.4/client-server-api/#end-to-end-encryption>>.

[DMLS]

Chathi, H., "Decentralised MLS", Web <https://gitlab.matrix.org/matrix-org/mls-ts/-/blob/dd57bc25f6145ddedfb6d193f6baebf5133db7ed/decentralised.org>, 2021, <<https://gitlab.matrix.org/matrix-org/mls-ts/-/blob/dd57bc25f6145ddedfb6d193f6baebf5133db7ed/decentralised.org>>.

[I-D.ietf-mls-protocol]

Barnes, R., Beurdouche, B., Robert, R., Millican, J., Omara, E., and K. Cohn-Gordon, "The Messaging Layer Security (MLS) Protocol", Work in Progress, Internet-Draft, draft-ietf-mls-protocol-20, 27 March 2023, <<https://datatracker.ietf.org/doc/html/draft-ietf-mls-protocol-20>>.

[I-D.ralston-mimi-linearized-matrix] Ralston, T. and M. Hodgson, "Linearized Matrix API", Work in Progress, Internet-Draft, draft-ralston-mimi-linearized-matrix-00, 25 March 2023, <<https://datatracker.ietf.org/doc/html/draft-ralston-mimi-linearized-matrix-00>>.

Acknowledgments

TODO acknowledge.

Authors' Addresses

Travis Ralston
The Matrix.org Foundation C.I.C.

Email: travisr@matrix.org

Matthew Hodgson
The Matrix.org Foundation C.I.C.

Email: matthew@matrix.org