Workgroup: Internet Engineering Task Force
Internet-Draft: draft-bertola-mimi-discovery-00
Published: 25 August 2023
Intended Status: Experimental
Expires: 26 February 2024
Authors: V. Bertola
Open-Xchange

Discovery of MIMI Service-Specific Identifiers via DNS

Abstract

This document introduces a possible solution for the discovery problem in MIMI (More Instant Messaging Interoperability). The problem is defined in a narrow sense, only including the conversion of a non-service-specific user identifier (email address, telephone number or a new MIMI-specific format) into one or more service-specific user identifiers, then retrieving the necessary information to establish a connection with their provider. The solution is based on the Domain Name System, so that it could be easily and quickly deployed on existing, well-known and broadly available infrastructure.

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 26 February 2024.

Table of Contents

1. Introduction

The More Instant Messaging Interoperability (MIMI) working group was established to develop the missing technical elements for a global, federated instant messaging system, which would enable interoperation across existing and new messaging services.

One of the issues that need to be addressed is how to allow a user (A) to enable another user (B) to start a communication with them, when they never communicated before. In a closed service supplied by a single provider, sharing user A's personal identifier would be sufficient, as all other elements are fixed. However, to do so in a federated environment, user B's client needs to know which provider is serving user A, which protocol endpoint should be contacted at user A's provider, and how to identify user A within that provider's own infrastructure.

Providers may, and possibly will, supply MIMI identifiers to their users; however, as different providers may use different user naming conventions and formats, as endpoints may change over time, and as users may switch from one provider to another, it is preferable to introduce a uniform, shorter, service-independent identifier that user A can give user B to allow them to initiate a conversation.

This document describes formats for these service-independent identifiers and exposes a mechanism for converting them into service-dependent information that would allow the initiation of a MIMI communication. The conversion process uses the existing DNS infrastructure, through the establishment of new record types.

The solution has been conceived for maximum flexibility, allowing both user-run and provider-run identifiers and both the introduction of new MIMI-specific identifiers and the reuse of existing ones (email, telephone number). Over time, not all these mechanisms may become broadly adopted, but the discovery solution should not preempt or limit the choices.

This document focuses on the narrow discovery phase only. The way through which user B gives user A their service-independent identifier, possibly through other means of digital, physical or in-person communication, is out of scope. Additionally, the steps to establish the communication channel after user B's client discovers user A's service information, such as the exchange of cryptographic information, are also out of scope.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Definitions

* MIMI User Identifier (MUI): A globally unique, service-independent identifier that refers to a specific MIMI user. MUIs follow a new specific format defined below. Nothing prevents users who want to segment their communications to own more than one MUI.

* External User Identifier (XUI): A globally unique, service-independent identifier that refers to a specific user in another type of communication service, such as an email address or a telephone number. Users may want to associate their existing XUIs to one or more MIMI service accounts, so to use them as identifiers in MIMI with or in place of an MUI.

*MIMI Service Endpoint (MSE): A technical identifier that points to a server providing MIMI service to a user, containing all the necessary information for a MIMI client to connect to it. The specific format and content of this identifier depend on other parts of the MIMI protocol set and are out of scope for this document.

*MIMI Service (also just "service"): A set of MSEs run by the same organization.

*MIMI Provider (also just "provider"): The organization running a MIMI service.

*MIMI Service-Specific Account Label (MSSA): An identifier that refers to a specific user account within a MIMI service, and is unique within that service and only within it. Different users may own the same MSSA in different services. Different services may have significantly different formats for their MSSAs.

*MIMI Service-Specific Identifier (MSSI): The coupling of an MSSA and an MSE; as a consequence of the previous definitions, it is globally unique and it includes all the necessary information to initiate a conversation with the user that owns that account on that service.

3. Use Cases and Requirements

The document aims to support the three following use cases:

1. User A wants user B to start a conversation with them via MIMI. User A gives user B their MUI in some non-MIMI way. This implies the need for the MUI to be as concise as possible, easily understandable both in spoken and written form and transmittable via the most common communication systems.

2. User A wants user B to start a conversation with them via MIMI. After setting up an association between one of their XUIs and their MIMI service, user A gives user B that XUI in some non-MIMI way.

3. User B already knows user A's XUI from another non-MIMI service through which the two users are already connected. As long as user A has accepted to make themself reachable via MIMI through that XUI, user B wants to use that XUI to start a MIMI conversation.

The solutions for these use cases should meet the following requirements:

- It should be possible for users to own their MUIs, without the need for authorization or delegation by service providers. It should also be possible for service providers, or for third parties, to provide MUIs and MUI management tools to users who do not want to acquire them on their own.

- An MUI or XUI could be associated to a single MSSI or to more than one at the same time, as per the user's wish. In the latter case, user B's client will decide on its own what to do and specifically which one to use, though user A may express preferences.

- XUIs should be associated to MIMI service(s) only if the user so desires.

- Users should be able to keep their MUI when moving from one service to another, as long as they use an MUI that is not owned by their current provider.

- Users should be able to add or remove any of the services associated to their MUI in the simplest possible way.

- Any change in the provider's MSEs or naming scheme for MSSAs should not require a change of MUI.

- It should be possible to construct MUIs that cannot be easily guessed, though users that value simplicity may also choose to use straightforward ones.

- The architecture should be as decentralised as possible, to prevent single points of surveillance for MIMI discovery activities.

- It should not be easily possible for third parties (parties who are not providing service to user A) to discover user A's MUI or MSSI or to discover MUIs or MSSIs in batches.

- It should be possible for any party, including individual end-users, to self-host their MIMI services, including the discovery part. Any unavoidable barrier in terms of cost and technical capabilities should be kept as low as possible.

- It should be possible to introduce new types of XUIs, not foreseen at the moment, for use with MIMI in the future, though this may require further specification.

It is assumed that the same MSE that can be contacted to initiate a conversation can also be contacted for any other standard MIMI action

that user B may want to perform on user A, such as retrieving further information (if allowed) or checking for presence. Thus, discovering user B's MSSI is sufficient for any desired MIMI operation.

In addition to the functional requirements, any solutions should be based on open standards and have a clear, low-barrier path to widespread implementation and deployment, which should also take into consideration economic sustainability and any likely security, privacy and legal issues.

4. Format and Resolution of MUIs

4.1. MUI Format

An MUI takes the form of a DNS Fully-Qualified Domain Name (FQDN), as defined in Section 2 of [RFC8499]; for example, mymimi.example.com or user.example.net. As in these examples, it is normally written relative to the root zone (without a trailing dot).

Any string valid as a FQDN can be used, including the internationalized domain names introduced in [RFC5890]. However, for the MUI to work, the user will need to control or at least have access to the zone of the highest level subdomain in the MUI.

The MUI format has been chosen so that it is immediately and visually distinguishable from the other type of user identifier most commonly used in messaging, the email address. Optionally, to clarify the context in which the FQDN is meant to be used, an MUI can be turned into a URI [RFC3986] by prefixing it with a MIMI-specific URI scheme (to be defined) and a colon.

4.2. Association of an MUI to MSSIs

An MUI is associated to an MSSI by adding into the DNS a resource record of the new type MIMI, having as value an MSSI definition string in the format specified in the following paragraph.

An MUI can be associated to several MSSIs by adding multiple MIMI records for the same MUI. See below in this document for examples.

4.3. MSSI Definition String

The MSSI definition string includes all the information necessary to discover the target MSSI and establish a connection. It makes use of name-value pairs, following the format used by the DMARC [RFC7489] and DKIM [RFC6376] specifications.

The following tags are defined for use as names within this string:

*v (plain-text; REQUIRED): version of the string format in use; for this document, "MIMI1". Any value which is not defined in a non-obsolete specification of this discovery mechanism MUST lead the client to ignore the entire record.

*a (plain-text; REQUIRED): the MSSA of the user in the target instant messaging service.

*e (plain-text; REQUIRED): the MSE of the target instant messaging service.

*pv (plain-text; OPTIONAL): the highest protocol version of MIMI supported by the target service at the endpoint. Acceptable values for this parameter are defined in other MIMI specifications.

*p (integer; OPTIONAL): a priority number of this service in respect to others, decided by the discovered user. This parameter allows the user to specify on which service(s) they would like to be contacted first if possible. Lower priority numbers imply higher priority.

*s (plain-text; OPTIONAL): an identifier of the instant messaging service type or brand, drawn from a specific IANA registry (see the IANA Considerations section). As MIMI only provides a basic interoperable layer, but specific instant messaging services may offer additional features on top, it may be useful for the user to specify more precisely which application or service is provided at the target endpoint.

4.4. Resolution of MUIs via DNS

User B's client performs a DNS query for user A's MUI asking for all existing MIMI records, using whatever DNS resolver service they want to use.

If the query succeeds, the client learns all the necessary elements to initiate a communication; if more than one record is retrieved, the client will decide which service to connect to, taking into account both user A's indication and user B's preferences, capabilities and requirements.

If the query fails, then the MUI cannot be resolved and the client returns failure.

4.5. MIMI DNS Record Examples

```
mymimi.example.com MIMI "v=MIMI1; a=myuser; e=mimiserver.example.com"
```

This record points to a generic MIMI service at mimiserver.example.com, where the user can be identified with the account name myuser.

```
mymimi.example.com MIMI
    "v=MIMI1; p=2; a=+15551234567; e=mimi.whatsapp.com; s=whatsapp"
mymimi.example.com MIMI
    "v=MIMI1; p=1; a=myname99; e=im.telegram.org; s=telegram"
```

This record discloses two different MIMI services. The first one is a WhatsApp service at mimi.whatsapp.com, identifying the user with a telephone-number-like string; the second one is a Telegram service where the user is known as myname99. If possible, the user would prefer Telegram over WhatsApp.

5. Format and Resolution of XUIs

5.1. Email Addresses

This specification allows for the usage as MIMI XUIs of email addresses, as defined in Section 3.4.1 of [RFC5322].

Two solutions are envisaged for the conversion of email addresses to MSSIs. They depend on who supplies the resolution service; in the first mechanism, the resolution is supplied by the email provider who controls the XUI namespace, while in the second mechanism, which does not require active cooperation by the email provider, it is necessary to introduce a form of oracle.

5.1.1. Resolution of Email Addresses via DNS

The resolution of E-mail addresses via DNS happens in two steps. In a first DNS query, the client learns which domain to use for the resolution of email addresses from that specific email provider for MIMI purposes; in the second query, the actual MSSI definition strings are retrieved. This allows the email provider to host MIMI resolution on a specialized domain and server, independent from the email infrastructure, or even to delegate it to third parties.

The first step is performed through the new DNS record MIMIX, which associates an email domain - the part of email addresses after the '@' sign - with a MIMI resolution domain. For example, assuming that the email address is mailbox@example.com, the DNS record would look as follows:

```
example.com MIMIX mimi.example.net
```

In the second step, the client appends the local part of the email address - the part before the '@' sign - in front of the MIMI resolution domain, obtaining an MUI. For the above example, the

resulting MUI would be mailbox.mimi.example.net. The MUI is then resolved as per Section 4.

It is expected that email providers interested in supplying this service to their users would enable them to manage their MIMI associations directly through some form of user interface, which would in turn prompt the creation or modification of the necessary DNS records. Alternatively, small and self-hosted email system administrators could manually create the records.

5.1.2. Resolution of Email Addresses via Oracles

If the email provider does not supply a MIMI resolution mechanism by entering the required records into the DNS, it is then necessary to introduce some form of oracle: a database run by a third party that allows the user, or a MIMI provider on their behalf, to enter the appropriate associations so that any client can then retrieve them.

Oracles of this kind present significant issues: they do not have an immediate business model; they need to validate on their own the existence and ownership of the XUI; they need to allow queries from any client from anywhere, while being able to avoid data harvesting and protect privacy.

Moreover, it is necessary for clients to know where these oracles are and determine whether they can be trusted. This can be addressed either by only having one system-wide oracle or network of oracles, with potential regulatory, privacy, competition and cost issues, or by having each user or client discover and maintain a list of trusted oracles on their own, which may be hard and multiply the queries that need to be made before finding a result.

For these reasons, oracles are considered to be a least good solution in comparison to decentralised, distributed resolution via DNS. However, especially in the initial phase of MIMI deployment, such a service may be necessary; for this reason, it can be reasonably expected that new entrants in the instant messaging space may be willing to pool resources, establish one or more oracles and advertise their existence. Participation in these oracles may actually become a regulatory requirement in cases where the use of MIMI will be mandated by law.

Clients SHOULD only query any oracle they know if direct resolution via DNS fails, i.e. if no valid MIMIX record is found for the email domain of the email address.

The actual details of building and querying an oracle are left to other developments; [I-D.rosenberg-mimi-glados] could be the starting point for that discussion.

5.2. Telephone Numbers

Similarly to email addresses, two solutions are envisaged for the association of telephone numbers to MSSIs. They depend on who provides the resolution service; in the first mechanism, the resolution is provided by the user's telephone operator via the use of ENUM [RFC6116]. In the second mechanism, an oracle is consulted.

5.2.1. Resolution of Telephone Numbers via DNS

The resolution of telephone numbers happens with a single query. First, the telephone number is converted into a fully-qualified domain name using the ENUM specification; then, the DNS is queried to retrieve one or more MIMI records for that name, using it as an MUI and proceeding as per Section 4.

5.2.2. Resolution of Telephone Numbers via Oracles

The same oracles that provide resolution for email addresses, as described in Section 5.1.2, could also offer resolution for telephone numbers, using the same mechanism. The same considerations apply.

6. IANA Considerations

IANA should create a new registry, possibly in a grouping devoted to MIMI, for the values of the 's' parameter in MSSI description strings.

Values should be unique and should be assigned on a first-come, first-serve basis, to instant messaging services that can demonstrate the existence of at least one deployed server.

Different values should be assigned for different server software and customised protocols, but not for different deployments of the same server software and/or protocol, though different values could be assigned to new major versions if they introduce significant incompatibilities.

7. Security Considerations

By reusing the DNS infrastructure, all DNS security mechanisms and practices would apply to MIMI resolution as well.

Separate security considerations for oracles will be provided once the oracle architecture has been defined.

8. Privacy Considerations

Instant messaging is a very sensitive application for privacy. Users should be protected from receiving undesired messages; the connection

between two users, which is exposed by the discovery procedure, should be kept confidential by default. These considerations have informed some of the requirements in Section 3.

A potential loss of privacy can derive from the use of personally identifiable information as part of the user's MUI or MSSIs. The flexibility given by this specification allows users to manage this risk for MUIs, by choosing the domain name and the hostname included in the MUI carefully. Some users may prefer getting the MUI within the domain of a big service provider, rather than in their own personal domain name. The risk of disclosing personal information within MSSIs is harder to manage for users, as it may be the service provider to force them to use personal information as part of their account name. However, this is a problem that already exists today and is not made worse by this specification.

By reusing the DNS infrastructure, all DNS privacy mechanisms and practices would apply to MIMI resolution as well. In particular, the user can choose which DNS resolver to use and how to connect to it; through these choices, they can ensure that they trust the resolver operator that will directly receive their MUI resolution query, and that their communication with that resolver will be protected, for example through encryption and proxying. In this way, it can become hard or impossible for the operator of the target user's MUI zone to learn who the resolution request is coming from.

Operators that manage the authoritative DNS zone for a big number of MUIs may still learn information, such as any personal information that is disclosed in the MUIs and MSSIs. Again, this risk can be reduced or eliminated by a careful choice of identifiers and operators by the user.

The use of XUIs such as email addresses and telephone numbers introduces another risk for the user - the fact that people that know that information from other services now get to reach them over MIMI as well. Indeed, the user may have given their email address to someone a long time ago without any intent to be reachable via instant messaging, which is a more direct and potentially intrusive communication system; also, their email address or telephone number could very likely have been collected into spamming lists. This is why the use of XUIs in MIMI should always be optional, and it should be possible for users to only use native MUIs.

Separate privacy considerations for oracles will be provided once the oracle architecture has been defined. In particular, oracles potentially constitute a centralized point of surveillance of user introductions and connections, and this makes them undesirable if they can be avoided.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8499] Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS Terminology", BCP 219, RFC 8499, DOI 10.17487/RFC8499, January 2019, <https://www.rfc-editor.org/info/rfc8499>.

9.2. Informative References

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

[RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, DOI 10.17487/RFC5322, October 2008, <https://www.rfc-editor.org/info/rfc5322>.

[RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, DOI 10.17487/RFC5890, August 2010, <https://www.rfc-editor.org/info/rfc5890>.

[RFC6116] Bradner, S., Conroy, L., and K. Fujiwara, "The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)", RFC 6116, DOI 10.17487/RFC6116, March 2011, <https://www.rfc-editor.org/info/rfc6116>.

[RFC6376] Crocker, D., Ed., Hansen, T., Ed., and M. Kucherawy, Ed., "DomainKeys Identified Mail (DKIM) Signatures", STD 76, RFC 6376, DOI 10.17487/RFC6376, September 2011, <https://www.rfc-editor.org/info/rfc6376>.

[RFC7489] Kucherawy, M., Ed. and E. Zwicky, Ed., "Domain-based Message Authentication, Reporting, and Conformance

(DMARC)", RFC 7489, DOI 10.17487/RFC7489, March 2015, <https://www.rfc-editor.org/info/rfc7489>.

[I-D.rosenberg-mimi-glados]
Rosenberg, J., "Global Lookup and Discovery of Services (GLADOS)", Work in Progress, Internet-Draft, draft-rosenberg-mimi-glados-01, 24 July 2023, <https://datatracker.ietf.org/doc/html/draft-rosenberg-mimi-glados-01>.

Author's Address

Vittorio Bertola
Open-Xchange
Via Livorno 12
10145 Torino TO
Italy

Email: vittorio.bertola@open-xchange.com