```
Network Working Group                                       J. Rosenberg
Internet-Draft                                                     Five9
Intended status: Informational                               J. Peterson
Expires: 5 September 2024                                     TrasnUnion
                                                            4 March 2024
```

MIMI Discovery Requirements and Considerations
draft-rosenberg-mimi-discovery-reqs-01

Abstract

This document defines requirements and use cases for the discovery problem in the More Instant Messaging Interoperability (MIMI) working group. The discovery problem refers to the process by which a message sender can identify the provider(s) associated with a desired messaging recipient, who is normally identified by an email address or phone number.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction 2
2. Definitions 3
3. Prior Efforts 5
4. Reference Architecture 6
5. App Provider Variations 7
5.1. Consumer OTT 7
5.2. Consumer Operator Aligned 8
5.3. Enterprise Cloud 8
5.4. Enterprise On-Prem 9
5.5. Consumer On-Prem 9
6. Core Requirements 10
7. Identifier Types 10
8. Provider Cardinalities 10
9. Caching 11
10. Number Portability 11
11. SII Release 12
12. SII Claim 13
13. Organizational Requirements 13
14. Blackhole Prevention 15
15. Spam Prevention 16
16. DP Social Graph Privacy 17
17. Encryption 17
18. AuthN 17
19. Hard Problems 18
20. Informative References 18
Authors' Addresses 20

1. Introduction

The More Instant Messaging Interoperability (MIMI) working group is chartered to enable federated messaging, voice, and video service between application providers, such as WhatsApp, Facebook Messenger, and other vendors. The MIMI protocols cover the exchange of encrypted content [I-D.ietf-mimi-content] through transfer protocols [I-D.ralston-mimi-linearized-matrix]. These protocols allow a user in one provider to initiate 1-1 and group messaging with a user in a second provider. The protocol requires that the originator of the communication know two things about the target user - their messaging provider, and a unique identifier for that user within that provider. The specifications recognize that the originator will not always know the provider for the target user, or the service-specific identifier for that user on that provider. The problem is further complicated by the fact that a users often make use of multiple messaging applications, in which case the preferences of the target user need to be taken into account as well. These preferences are even less likely to be known by the originator of communications.

Rather, in many cases one user will have an email address or phone number for the target user, obtained from their address book on their mobile device. Neither the phone number or email address identify the messaging provider that the target user is using. Unlike email service, the domain portion of a user's email address has no bearing on what messaging provider they use. A user joe@gmail.com might be using WhatsApp or iMessage, neither of which are Gmail. Thus - the core problem is - how to take one of these service independent identifiers and learn the messaging service that user is using, and how to send messages to them on that messaging service.

The MIMI framework hypothesizes the existence of a discovery or directory service to solve this problem. The discovery service would allow the originator to take a servide independent identifier for a target - such as a mobile phone number or email address - and perform a lookup to determine the preferred service(s) of the target user, along with enough information to reach them on that service.

This document describes requirements and use cases for solutions to the discovery problem.

2. Definitions

* Service Independent Identifier (SII): A type of identifier for a user that is unique (such that an SII is associated to only a single user), and independent of any specific communications service. There are two specific identifiers in this case - a phone number (landline or mobile), or an email address.

* Service Specific Identifier (SSI): A type of identifier for a user that is unique (such that an SSI is associated to only a single user), and achieves its uniqueness by being composed of two parts - a user part, scoped to a provider of communication services, and a unique identifier for the communication service provider. In some services, the user part is not globally unique across services. Examples of this case are Wire, Twitter and Skype, where user handles are flat - @jdrosen2 on Twitter, for example. In other services, the user part is globally unique, and

corresponds to the email address or mobile phone number (SII) for the recipient. Examples of this case are WhatsApp, iMessage, and Facetime.

- Personally Identifying Information (PII): Information about a target user that is not unique, but can be used to facilitate a search for the target user. Typically this would be the first name and/or last name of the recipient. The search would provide a list of possible matches, along with additional information, such as display names and avatars, which help the initiator find the specific person to which communications is desired.

- Application Provider (AP): A provider of messaging, voice, video and communications services to end users. An application provider is the entity that would implement the MIMI protocols. Examples of application providers are WhatsApp, Facebook Messenger, iMessage, Wire, Matrix, and so on.

- Discovery Provider (DP): A provider of discovery services, capable of mapping an SII to an SSI. This entity does not yet exist, and this document defines requirements for the protocols and processes behind it.

- Telephone Number Service Provider (TNSP): An entity which has authoritative ownership of the phone number used by a user. In the case of a mobile phone number, this would be their mobile operator (e.g., Verizon or AT&T in the United States). For a landline number, in would be their landline voice provider, which can include incumbent landline providers, but may also include non-traditional providers of voice and SMS services, like CPaaS (Communications Platform as a Service), such as Twilio or Nexmo, CCaaS (Contact Center as a Service), such as Five9 and NICE/InContact, and UCaaS (Unified Communications as a Service) providers, such as RingCentral, Webex and Zoom. We use Number Provider (TNSP) and not "operator" to keep this general purpose and to emphasize the fact that the key consideration for the discovery service is the assignment of the number to the user, not the provision of communications services against that number.

- Email Provider (EP): An entity which has authoritative ownership of the domain name portion of the email address used by a user. This would be Google for gmail.com, or Verion/AOL for aol.com as two examples. The EP for an email address can also be an enterprise, such as Cisco for cisco.com email addresses. As with telephone numbers, the EP is simply the provider of the address, and may not also be the provider of all communications services against that address.

* Cloud Provider (CP): An entity providing services to enterprises for voice, video and messaging services, acting as the Application Provider for employees of its enterprise customers. The CP is the TNSP for some numbers used by the enterprise, but not always.

3. Prior Efforts

Discovery services are far from new on the Internet.

The whois protocol, originally specified in [RFC0954] and later revised by [RFC3912], was largely focused on the mapping of domain names, to services associated with those domain names, and was one of the first discovery services deployed on the Internet. The DNS SRV record was specified in [RFC2782] and allows a similar discovery process - given a domain name, allows a querier to learn the set of services, such as VOIP based on the Session Initiation Protocol (SIP) [RFC3261] [RFC3263]. The SRV record was adapted to messaging in particular [RFC3861]. Whois and DNS SRV records both assumed that the lookup was keyed by a domain name, and thus they were not that useful for looking up an identifier that is not domain scoped, such as a mobile phone number.

This was first addressed through the specification of ENUM [RFC3761] in 2004. ENUM defined the usage of DNS to lookup phone numbers, by convering a phone number to a DNS name by reversing the digits and adding the suffix "e164.arpa". This allowed portions of the namespace to be delegated to telco providers that owned the number prefix in question. Though technically simple to define, its public deployment was hampered by the challenges of establishing authority for the prefixes. Private ENUM [RFC6116] services however have become relatively common, facilitating routing for many functions, including MMS routing in the messaging space.

Another attempt was made with ViPR (Verification Involving PSTN Reahability) [I-D.rosenberg-dispatch-vipr-overview] [I-D.petithuguenin-vipr-pvp]. VIPR made used of a peer-to-peer network based on RELOAD (Resource Location and Discovery) [RFC6940], running between enterprises. It solved the problem of authority problem by authorizing records based on proof of forward routability. However, it had the same network effects problem as ENUM. It also addressed the incentive problem, by focusing on enterprises for which bypassing the phone network would provide cost savings. However, the network effects problem proved insurmountable (amongst other challenges unrelated to the protocol), and it was never widely deployed.

Discovery and lookup services are now common place on the Internet but are scoped entirely within large providers, such as Facebook, Twitter, WhatsApp and other providers.

The MIMI discovery service requires a solution that spans across providers.

4. Reference Architecture

The reference architecture is shown below.

```
           +------+        +------+
           | Disc |        | Disc |
           | Prov |--------| Prov |
           |  1   |        |  2   |
           +------+        +------+
               |               |
               |               |
         +-----+----+          |
         |          |          |
    +--------+ +--------+ +--------+
    |  App   | |  App   | |  App   |
    |Provider| |Provider| |Provider|
    |   1    | |   2    | |   3    |
    +--------+ +--------+ +--------+
     |      |        |     |      |
     |      |        |     |      |
   +----+ +----+  +----+  +----+ +----+
   |User| |User|  |User|  |User| |User|
   |  1 | | 2  |  | 3  |  | 4  | | 5  |
   +----+ +----+  +----+  +----+ +----+
```

Figure 1: Discovery Architecture

There are many users in the system, with each user making use of zero or more communication applications, each provided by an Application Protocol (AP). Those application providers, in turn, connect to a Discovery Provider (DP) which is capable of mapping the SII to an SSI. In some cases, APs may themselves act as DPs. As shown in the diagram, one of the requirements is that there can be more than one DP, in which case there will be a need for some kind of inter-DP communication or federation.

5. App Provider Variations

There are many variations on who the app provider is, and what their relationship is with the user of the messaging service and the number and email providers for the identities. We can invision the following variants, all of which should be supported by both MIMI and the discovery service.

The variations are discussed below in order of decreasing commonality. For each one, we also discuss their cardinality (how many of them there are), how large of a population of users they serve, and how likely they are to participate in MIMI and the discovery service.

5.1. Consumer OTT

In this case, the App Provider offers services to consumers. The consumer has an email address and/or phone number, but the EP and TNSP for those identifiers have no relationship whatsoever with the AP. Examples include Apple’s iMessage, Facebook Messenger, Wire and so on. However it does not include Google Messaging (RCS) which is the next category.

A very small number of these providers dominate messaging today. Many, but not all, are gatekeepers. Most are extremely large, supporting millions or more consumers. It is reasonable to expect the smaller, non-gatekeeper ones, to directly request interop with the gatekeepers as it is core to their business.

The smaller consumer OTT providers will be highly incented to participate in MIMI. They may, or may not, be incented to participate in the discovery service. Some of them do not make use of SII’s in their services. For those providers, they would require that their users be reached because users in other APs know the SSI instead. Similarly, for their own users to communicate with other consumer OTT providers, they may require their users to know their SSIs.

If we were to only consider the consumer OTTs, we might conclude that SII to SSI mapping (discovery) is not needed - users can just use a drop-down menu of providers when reaching out to another user (this is sometimes called a nascar menu). However, it becomes untenable when you consider the additional use cases below.

5.2. Consumer Operator Aligned

In this case, the app provider offering services to its consumers is affiliated to the Telephone Number Service Provider (TNSP) for its users, and therefore has authoritative knowledge of the ownership of phone numbers for its users. The primary use case here is Google Messaging, provided through the Rich Communications Service (RCS) providers, which are the mobile operators, or Google who can operate it on their behalf. It may also include residential triple-play providers (MSOs and so on) that enable messaging for landline numbers.

There are many operators globally, numbering in the hundreds. Today, the only ones offering consumer messaging are the mobile operators.

5.3. Enterprise Cloud

In this case, another entity - the Cloud Provider (CP) is involved - which is a CCaaS, UCaaS or CPaaS provider offering communications services. The enterprise contracts with the Cloud Provider, which acts as the Application Provider (AP). The Cloud Provider is often also the Number Provider for the enterprise numbers used by enterprise employees. However, the Cloud Provider will often instead themselves contract with TNSPs to obtain numbers for its enterprise customers, which can then route to it over private SIP trunks for voice, and usually some non-SIP APIs for messaging. Examples of enterprise cloud APs are Five9, Cisco Webex, RingCentral, Microsoft Teams, Zoom, and so on.

The enterprise use case brings an additional consideration as well - in that many numbers (and email addresses) represent a service rather than a user. THink of the 1-800 number for a business, or an email address for customer support, or a phone number for an enterprise helpdesk. These are all services, behind which one or more users may reside.

There are a relatively small number of larger enterprise cloud players, perhaps numbering the few dozen. They tend to each have a smaller number of users than the consumer OTT providers (typically in the hundreds of thousands to millions of users). They also have economic incentive to request interop with the gatekeepers, since it reduces their direct costs for routing messages, voice and video calls. It would also likely increase the appeal of their products, which could offer consumer interconnection as part of their offerings, along with b2b federation between cloud providers.

For enterprise clouds to participate in MIMI, the discovery solution is much more important. This is because these companies are often not brands that are not consumer recognizable, there are too many of them to fit in a selector UI, and it is often impossible for a sender of a message to figure out what provider the recipient is on. This is especially problematic for the case where the SII represents a service and not a user.

5.4. Enterprise On-Prem

In this case, the app provider offering messaging services is an enterprise, who is doing so through on-premise messaging software they deploy and operate. The enterprise will always be the Email Provider (EP), but they are not the Telephone Number Service Provider (TNSP). That said, the enterprise connects to the TNSP via SIP trunks to enable calls to/from those numbers to reach it. One can think of this as the case where the enterprise is its own cloud provider. These cases are less common these days, but still exist. Examples are any enterprises running Cisco Jabber on-prem or Microsoft OCS or LCS on prem.

There are of course a large number of enterprises in the world which have historically had some kind of on-prem software, numbering in perhaps the hundreds of thousands. The ones which still do so is much smaller, but still a much larger number than the number of enterprise cloud providers. These enterprises are less likely to request interop with gatekeepers, just because they each serve a much smaller number of users and their incentives for doing so are less.

For enterprise on-prem use cases, the discovery service is absolutely required for their users to be reached for inbound communications. There is simply no way that other users will be able to select from a dropdown list of company names.

5.5. Consumer On-Prem

In this last case, the app provider offering messaging services is the consumer themselves, who is running some software in their home network or in a public cloud compute environment, which they deploy and operate. The consumer is neither a TNSP or an EP. This is a relatively uncommon case these days. It was not uncommon for people to run their own mail services for their home, but since messaging has predominantly been cloud based it is not as common there. That said, it is certainly possible for a consumer to run (for example) their own Matrix server in their home for their family.

It is extremely unlikely a consumer on-prem user would ever request interop with a gatekeeper. And, discovery is absolutely needed for the user to be reached for inbound communications.

6. Core Requirements

There are four key requirements:

1. Mapping: The service must provide a way to map from a SII to one or more application providers, and where necessary, to SSIs valid for those applications.

2. Validity: The mappings provided by the service must represent the wishes of the user associated with the SII, mapping to an application they are a user of, and the mapped SSI must be the one associated with this user. The core issue is one of trust, and how to determine that the mappings provided by the service are accurate.

3. Critical Mass: The network effects problem is perhaps the hardest to solve. But, to be viable, any solution must be able to reach a critical mass of mappings so that it becomes useful to consume, and thus useful to further populate.

4. Incentive Alignment: There must be an incentive structure which motivates the population of mappings into the service, and for the consumption of those mappings.

7. Identifier Types

1. Mobile SIIs: SIIs must include mobile phone numbers.

2. Landline SIIs: SIIs must include landline phone numbers.

3. Email addresses: SIIs must include email addresses

8. Provider Cardinalities

1. Zero APs: The system should work when a user - and their SIIs - are not associated with any discoverable APs. In this case, the discovery operation should indicate a no-match. This would enable an originating user to learn that they cannot reach that SII (short of sending an email or SMS, say).

2. One AP: The system should work in the simple case when a user as a single AP.

3. Multiple APs with Default: The system should enable a user to have multiple APs. The discovery service should enable user preference to be considered, so that a user can choose a default AP to use.

4. Business vs. Consumer AP: It should also be possible for a user to indicate that different APs are used for business purposes vs. consumer purposes. As an example of this case, user Alice might use WhatsApp for friends and family, but use Microsoft Teams at work. Her mobile number is used as an SII in both providers. When a user Bob on Webex Teams searches for that number, Bob would only get the Microsoft Teams SSI because their Webex Teams administrator has specified that messaging is between business APs by default. In another use case, Bob would get both of these back and would have the ability to choose whether to use the business or personal AP.

5. Circle Based APs: It should be possible for a user to specify that different APs are to be used for different contacts. For example, user Alice might use WhatsApp when talking to friends, but use iMessage when talking to family. When Bob, Alice’s friend enters her number into his messaging app, the result depends on whether Alice has specified that he is a friend vs. family member [NOTE: I think this is probably more than we need and it adds a lot of complexity. I include it here for completeness to explore how deep this rabbit hole goes].

9. Caching

Given the significant volume of inquiries which might be sent, caching is a useful feature of the discovery service.

1. Cacheability of Results: The discovery service should allow for mappings to be cached by the AP. The DP must be able to tell the AP the duration over which the mapping can be cached.

2. Cache Invalidation: To handle changes in preferences or SII releases, it must be possible for the DP to inform the AP when a mapping is no longer valid ahead of its cache expiration.

10. Number Portability

When the SII is a phone number, porting comes into consideration.

The requirements depend on whether the user’s operator - basically their number provider (TNSP) - is also the Application Provider (AP). When these are intertwined, porting a number also changes providers. Consequently, we can break this down into four distinct use cases and requirements.

1. Donating Operator is not the AP, and neither is the recipient. The number port should change nothing, the discovery service should continue to resolve to the AP.

2. Donating Operator is not the AP, but the recipient operator is an AP. The user now effectively has two APs - the OTT one before the port, and now a second one because their new operator is an AP, in essence enrolling them in the service by virtue of being an AP. In this case, it should be possible for a user to express a preference about where to receive incoming messages.

3. Donating Operator is the AP, but the recipient operator is not an AP. In this case, by porting away from their prior operator who was also an AP, the user has terminating their relationship with the messaging provider, and now has no provider at all, since their new operator is not also an AP. As it relates to the discovery service, once the port is complete, the user should be shown as no longer discoverable, and their prior mapping is deleted.

4. Both Operators are APs: In this case, the user has basically moved providers from one to another. As it relates to the discovery service, once the port is complete, the discovery service should indicate that their SSI is now on the new operator/AP.

11. SII Release

If a user is associated with a phone number by virtue of being a customer of a TNSP that is providing them that number, their association with that number will end once the user terminates their relationship with their TNSP. It is typical in telephony systems for that number to go into a waiting pool for several months before it can be reassigned to a different user.

For email addresses, it is also possible for a user to lose their association with an email address when they end service with that provider. Although reclamation of email addresses is possible, it is less common. Nontheless, it is technically possible.

This release process adds requirements for the discovery service.

1. SII Release Timeliness: If a user terminates service with a TNSP or EP, and thus loses their association with a number or email address from that provider, any mappings in the discovery service keyed by that SII should be removed within a month. Note that, this is an extremely difficult requirement to meet. It is certainly not met today by most messaging systems internally that use numbers as identifiers. For any OTT AP, the only way this requirement can be met is periodically reverifying ownership of the number through an SMS or phone call. This is burdensome to the user, and consequently, generally not done. Meeting this requirement without disruptive re-verifications requires the discovery providers (DPs) to have feeds into global number databases. For email addresses, this is even more untenable.

12. SII Claim

When a user starts their association to a number or email address, we can think of this as a "claim". Their claim is rooted in the start of services from the Number Provider (TNSP), Cloud Provider (CP) or Email Provider (EP) towards the user. This introduces a timeliness requirement.

1. SII Claim Timelines: Once a user is associated with an SII by virtue of obtaining service from a TNSP, EP, or CP that owns the given SII, it must be possible for the user to utilize that number with an AP and become discoverable immediately upon provision of service. This reflects a real, common use case. A user gets a new mobile phone with a new mobile phone number, and before even leaving the store, installs WhasApp or uses Google Messaging on their Android (which is RCS based) and expects it to work. Furthermore, they will contact their friends and family right away, giving them the new number, and expect to be reachable. The same applies to email addresses, though those change less frequently in the consumer space. In the enterprise space however, email addresses are frequently assigned and similarly, we want the user to be immediately discoverable.

2. When a user associates their SII with an Application Provider, there must be some way for the app to validate that the user controls the SII. Moreover, the app must have some way to prove that this validation was performed to third parties, such as other app providers, in order to prevent blackholes and similar attacks.

13. Organizational Requirements

A key consideration is - who runs, or can run, the discovery service?

1. Multiple Providers are Possible: One can imagine a design for the discovery service in which there is a single, worldwide global provider of the discovery service. This would certainly simplify the protocol and its security properties. There are some precedents for a singleton provider of service in the Internet - see ICANN and IANA. However, neither of these run operational services. Even the Internet’s primary global service - the DNS - is in practice distributed amongst many different entities that run and operate the top level domain name servers. As a result, the discovery service should follow a similar pattern and allow for multiple providers of the discovery service.

2. Organizational Principles deliver trust: Once we accept that there can be many such providers of discovery services, how would an application provider (AP) know whether to trust the mappings that it provides? One answer is - this is just left to the market to decide, and the IETF has nothing to say on the matter. The alternative is that - the IETF defines the solution in such a way that there are ways for trust to be established. As one such example, the solution could be specified such that the solution for phone numbers makes use of existing number ownership structures that support STIR/SHAKEN [RFC8224] [RFC8225]. Or, it could define the solution in such a way that entities which already hold this routing information for messaging apps (i.e. using the Pathfinder service from Neustar which provides this mapping today for the GSMA) expose APIs for it.

3. PII Residency within Geopolitical Boundaries: There are increasingly regulations being passed, like GDPR, which require that personal data remain within certain geopolitical boundaries. Since the discovery service may contain such information, it must be possible for the DPs to sit within a geopolitical boundary and hold data for users within those geopolitical boundaries.

4. Invisible to Consumers: There are a class of solutions wherein a DP is directly visible to consumers, who would sign up, verify their number with it, and configure their preferences with it. However, this is unlikely to work in practice. It suffers from a significant network effects problem, such that signing up for the service would provide no value to its users until critical mass is reached. This would disincent users from signing up in the first place. As a result, the only solutions which can really work are those which are invisible to users, where the App Provides themselves send request to - or act as - DPs. That does however raise the question of how user preferences are expressed in the system.

5. Numerous App Providers: This is as much a requirement in MIMI, as it is for the discovery protocol. But, the goal is that we want a system wherein there can be a lot of app providers, many of which are smaller in size. This becomes even more obvious when we consider enterprise use cases, where a business might be its own provider for its own employees, and want them to be able to message consumers as well as other businesses using business numbers or business email addresses. In such a scenario, the number of APs can be in the thousands or more.

6. DP Federation: Because there are multiple DPs, run by different entities, it must be possible for some kind of federation so that an AP can request a mapping from one DP, and the mapping can be provided even if it resides within a different DP. Note that - this requirement could be contested. There is an alternative world view, wherein each AP needs to connect to every DP, with each DP holding a subset of the mapings. The drawback of such a system is, if we think DPs are aligned against geopolitical or organizational boundaries, it may be impossible or impractical for such a full-mesh configuration.

7. DP Federation Policy: Due to geopolitical considerations, it must be possible for a DP to decide to federate, or not federate, with other DPs. Such policies are outside the scope of this work, but this fact may result in some SIIs not being discoverable in certain geographical or political regions.

14. Blackhole Prevention

If we accept the requirement above that there can be a large number of app providers, including enterprises themselves, there is a large risk that one of them is malicious. The main attack we wish to prevent, is for an AP to claim it has a user associated with a given SII, when it in fact does not. Though MLS would (to the degree e2e identity works against that SII) prevent the recipient from reading messages sent to that SII, it is certainly possible that they can "blackhole" them. This is an attack wherein the malicious AP causes the SII to map to its own SSI, rather than the legitimate SSI for the user. This would deny receipt of messages at the legimiate SSI, and thus is a form of denial of service.

The concern over blackhole attacks introduces several key requirements.

1. Malicious AP cannot blackhole against a legitimate AP: A critical security requirement for the discovery service, is that is not possible for a malicious AP to create a blackhole.

2. Malciious AP cannot make a user appear discoverable even though they are not: In this case, a user Bob is not a user of any AP. In a functioning system, they would show as not-discoverable to users searching for them based on their SII. In this attack, a malicious AP tries to convince the discovery service that they are in fact a user of the malicious AP. Even though the malicious AP cannot decrypt the incoming messages, they will cause other users to now view user Bob as discoverable. This is a less severe version of the above attack, but is still an attack. It would potentially fool senders into thinking they have reached a target that is ignoring them, which can cause unintended consequences.

3. Ultimately, the DP must have a direct assurance that a particular SII has been authentically associated with an Application Service before allowing that app to be discovered as a mapping for the SII.

15. Spam Prevention

Spam is a significant concern in the system, and its risk grows exponentially with the number of APs connected to the system. As noted above, many use cases have a large number of APs, which can pose a serious risk. Spam prevention needs to be considered at both the MIMI layer (using techniques like connection requests and reputation safeguards), but can also be addressed at the discovery service.

Note that SIIs act as "front doors" for end users today, and there is an inherent risk in having one - especially telephone numbers, as the numbering space can be relatively easily enumerated. Making an SII discoverable necessarily opens the door to receiving unwanted or unsolicited communications, much of the mitigation of which will be the responsibility of apps and of user applications.

1. No Enumeration: The system must protect against an enumeration attack. An enumeration attack is one wherein a malicious AP attempts to look up a large number of SIIs - especially phone numbers which can easily be enumerated as they are finite - in order to learn the SSI assocaited with each. Once an SSI is known, the malicious AP has an address it can add to its spam list. Today, many people avoid listing their email addresses or phone numbers on public websites to prevent spam sites from scraping those identifiers to add them to target lists. We don’t want the discovery service to be a nice, convenient and easily farmable source of identifiers for sending spam.

2. Rate Limits: The system must provide rate limit capabilities to restrict an AP from sending too many discovery requests. There must be a way for the Discovery Provider (DP) to assess what a reasonable rate limit might be for that AP.

16. DP Social Graph Privacy

The Discovery Provider (DP) will receive requests from APs to map a given SII to a provider and/or SSI. These requests themselves create a form of social graph, indicating what SIIs are often requested, and which are not. This leaks information to the DP. The following requirement tries to limit exposure of the DP to this information.

1. DP Unaware of Requested Number: A DP must protect at least one end of the social graph during a request: the DP must be kept ignorant of either the querier's identity (including IP address) or the SII of interest in requests. For exampple, IP blinding could conceal the querier's identity, or techniques such as Private Information Retrieval (PIR) could conceal the SII from the DP.

2. DP Minimal Federation: The federation techniques should avoid propagating mappings from one DP to another DP unless there is a legitimate need for that DP to know of a mapping - for example in order to satisfy a query. While the business relationships that may underlie DP federation are outside the scope of these requirements, federations may institute their own policies to protect consumers and private business data.

3. DP User hiding: A DP should not share the querying user identity with other DPs when it requires their help for discovery.

17. Encryption

At the risk of stating the obvious, but:

1. Encrypted Transport: Exchange of information between DPs, or between DPs and APs, should always be encrypted in transit.

18. AuthN

Also obvious, but:

1. Authentication: It must be possible for two DPs federating to identify each other, and it must be possible for a DP and AP communicating with each other, to identify the other party.

19. Hard Problems

From these requirements, a few areas have emerged that warrant particular attention in potential solutions:

1. Multiple mapppings. If there are multiple candidate app mapppings discovered for a given SII, what do we expect the behavior will be at a protocol level? Will a message be sent to each app? Will a nascar menu be presented to the user? Or will just one be selected through some sort of preferences mechanism? In the last case, especially when apps themselves act as DPs, is it legitimate for apps to prefer to route an SII to its own service rather than to competitors?

2. Preferences and capability negotiation. If there are multiple potential mappings for an SII, how much should the preferences of the sender and recipient of communications be weighed, and how should those preferences be expressed? Because users may tacitly or explicitly establish contexts for their messaging contacts (business on one app, personal on another, say), how rich would the expression of such preferences need to be?

3. Authentication and expiration of mappings. How rigorous does the process need to be for validating mappings in order to prevent blackholes and similar threats? How do the mappings created for discovery relate to the identities asserted at the protocol level, e.g. [I-D.mahy-mimi-identity]? Once an SII has been claimed by a user and enrolled at one or more messaging apps, how long should that mapping persist before expiring, as some SIIs change ownership over time?

4. Protecting user privacy. How much information can we shield from the DP, or indeed the appp itself, while still enabling a messaging system? How do we prevent enumeration attacks if we want these mappings to be basically publicly available? How do we balance user privacy with spam protection? What is the threat landscape for pervasive monitoring of social graphs associated with messaging?

20. Informative References

[I-D.ietf-mimi-content]
Mahy, R., "More Instant Messaging Interoperability (MIMI) message content", Work in Progress, Internet-Draft, draft-ietf-mimi-content-01, 23 October 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-mimi-content-01>.

[I-D.mahy-mimi-identity]
Mahy, R., "More Instant Messaging Interoperability (MIMI) Identity Concepts", Work in Progress, Internet-Draft, draft-mahy-mimi-identity-02, 10 July 2023, <https://datatracker.ietf.org/doc/html/draft-mahy-mimi-identity-02>.

[I-D.petithuguenin-vipr-pvp]
Petit-Huguenin, M., Rosenberg, J., and C. F. Jennings, "The Public Switched Telephone Network (PSTN) Validation Protocol (PVP)", Work in Progress, Internet-Draft, draft-petithuguenin-vipr-pvp-04, 12 March 2012, <https://datatracker.ietf.org/doc/html/draft-petithuguenin-vipr-pvp-04>.

[I-D.ralston-mimi-linearized-matrix]
Ralston, T. and M. Hodgson, "Linearized Matrix", Work in Progress, Internet-Draft, draft-ralston-mimi-linearized-matrix-04, 10 January 2024, <https://datatracker.ietf.org/doc/html/draft-ralston-mimi-linearized-matrix-04>.

[I-D.rosenberg-dispatch-vipr-overview]
Rosenberg, J., Jennings, C. F., and M. Petit-Huguenin, "Verification Involving PSTN Reachability: Requirements and Architecture Overview", Work in Progress, Internet-Draft, draft-rosenberg-dispatch-vipr-overview-04, 25 October 2010, <https://datatracker.ietf.org/doc/html/draft-rosenberg-dispatch-vipr-overview-04>.

[RFC0954] Harrenstien, K., Stahl, M., and E. Feinler, "NICNAME/WHOIS", RFC 954, DOI 10.17487/RFC0954, October 1985, <https://www.rfc-editor.org/info/rfc954>.

[RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, DOI 10.17487/RFC2782, February 2000, <https://www.rfc-editor.org/info/rfc2782>.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <https://www.rfc-editor.org/info/rfc3261>.

[RFC3263] Rosenberg, J. and H. Schulzrinne, "Session Initiation Protocol (SIP): Locating SIP Servers", RFC 3263, DOI 10.17487/RFC3263, June 2002, <https://www.rfc-editor.org/info/rfc3263>.

[RFC3761] Faltstrom, P. and M. Mealling, "The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)", RFC 3761, DOI 10.17487/RFC3761, April 2004, <https://www.rfc-editor.org/info/rfc3761>.

[RFC3861] Peterson, J., "Address Resolution for Instant Messaging and Presence", RFC 3861, DOI 10.17487/RFC3861, August 2004, <https://www.rfc-editor.org/info/rfc3861>.

[RFC3912] Daigle, L., "WHOIS Protocol Specification", RFC 3912, DOI 10.17487/RFC3912, September 2004, <https://www.rfc-editor.org/info/rfc3912>.

[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>.

[RFC6940] Jennings, C., Lowekamp, B., Ed., Rescorla, E., Baset, S., and H. Schulzrinne, "REsource LOcation And Discovery (RELOAD) Base Protocol", RFC 6940, DOI 10.17487/RFC6940, January 2014, <https://www.rfc-editor.org/info/rfc6940>.

[RFC8224] Peterson, J., Jennings, C., Rescorla, E., and C. Wendt, "Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 8224, DOI 10.17487/RFC8224, February 2018, <https://www.rfc-editor.org/info/rfc8224>.

[RFC8225] Wendt, C. and J. Peterson, "PASSporT: Personal Assertion Token", RFC 8225, DOI 10.17487/RFC8225, February 2018, <https://www.rfc-editor.org/info/rfc8225>.

Authors’ Addresses

Jonathan Rosenberg
Five9
Email: jdrosen@jdrosen.net

Jon Peterson
TrasnUnion
Email: jon.peterson@transunion.com