Network Working Group
Internet-Draft
Intended status: Informational
Expires: September 6, 2018

J. Peterson
T. McGarry
NeuStar, Inc.

March 5, 2018

Modern Problem Statement, Use Cases, and Framework
draft-ietf-modern-problem-framework-04.txt

Abstract

The functions of the public switched telephone network (PSTN) are rapidly migrating to the Internet. This is generating new requirements for many traditional elements of the PSTN, including telephone numbers (TNs). TNs no longer serve simply as telephone routing addresses: they are now identifiers which may be used by Internet-based services for a variety of purposes including session establishment, identity verification, and service enablement. This problem statement examines how the existing tools for allocating and managing telephone numbers do not align with the use cases of the Internet environment, and proposes a framework for Internet-based services relying on TNs.

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 September 6, 2018.

Table of Contents

1. Problem Statement

The challenges of utilizing telephone numbers (TNs) on the Internet have been known for some time. Internet telephony provided the first use case for routing telephone numbers on the Internet in a manner similar to how calls are routed in the public switched telephone network (PSTN). As the Internet had no service for discovering the

endpoints associated with telephone numbers, ENUM [3] created a DNS-based mechanism for resolving TNs in an IP environment, by defining procedures for translating TNs into URIs for use by protocols such as SIP [2]. The resulting database was designed to function in a manner similar to the systems that route calls in the PSTN. Originally, it was envisioned that ENUM would be deployed as a global hierarchical service, though in practice, it has only been deployed piecemeal by various parties. Most notably, ENUM is used as an internal network function, and is rarely used between service provider networks. The original ENUM concept of a single root, e164.arpa, proved to be politically and practically challenging, and less centralized models have thus flourished. Subsequently, the DRINKS [4] framework showed ways that service providers might provision information about TNs at an ENUM service or similar Internet-based directory. These technologies have also generally tried to preserve the features and architecture familiar to the PSTN numbering environment.

Over time, Internet telephony has encompassed functions that differ substantially from traditional PSTN routing and management, especially as non-traditional providers have begun to utilize numbering resources. An increasing number of enterprises, over-the-top voice-over-IP (VoIP) providers, text messaging services, and related non-carrier services have become heavy users of telephone numbers. An enterprise, for example, can deploy an IP PBX that receives a block of telephone numbers from a carrier and then in turn distribute those numbers to new IP telephones when they associate with the PBX. Internet services offer users portals where they can allocate new telephone numbers on the fly, assign multiple "alias" telephone numbers to a single line service, implement various mobility or find-me-follow-me applications, and so on. Peer-to-peer telephone networks have encouraged experiments with distributed databases for telephone number routing and even allocation.

This dynamic control over telephone numbers has few precedents in the traditional PSTN outside of number portability. Number portability allows the capability of a user to choose and change their service provider while retaining their TN; it has been implemented in many countries; either for all telephony services or for subsets such as mobile. However, TN administration processes rooted in PSTN technology and policies dictate that this be an exception process fraught with problems and delays. Originally, processes were built to associate a specific TN to a specific service provider and never change it. With number portability, the industry had to build new infrastructure, new administrative functions and processes to change the association of the TN from one service provider to another. Thanks to the increasing sophistication of consumer mobile devices as Internet endpoints as well as telephones, users now associate TNs with many Internet applications other than telephony. This has

generated new interest in models similar to those in place for administering freephone (non-geographic toll free numbers) services in the United States, where a user purchases a number through a sort of number registrar and controls its administration (such as routing) on their own, typically using Internet services to directly make changes to the service associated with telephone numbers.

Most TNs today are assigned to specific geographies, at both an international level and within national numbering plans. Numbering practices today are tightly coupled with the manner that service providers interconnect, as well as how TNs are routed and administered: the PSTN was carefully designed to delegate switching intelligence geographically. In interexchange carrier routing in North America, for example, calls to a particular TN are often handed off to the terminating service provider close to the geography where that TN is assigned. But the overwhelming success of mobile telephones has increasingly eroded the connection between numbers and regions. Furthermore, the topology of IP networks is not anchored to geography in the same way that the telephone network is. In an Internet environment, establishing a network architecture for routing TNs could depend little on geography, relying instead on network topologies or other architectural features. Adapting TNs to the Internet requires more security, richer datasets and more complex query and response capabilities than previous efforts have provided.

This document attempts to create a common understanding of the problem statement related to allocating, managing, and resolving TNs in an IP environment, the focus of the IETF MODERN (Managing, Ordering, Distributing, Exposing, and Registering telephone Numbers) working group. It outlines a framework and lists motivating use cases for creating IP-based mechanisms for TNs. It is important to acknowledge at the outset that there are various evolving international and national policies and processes related to TNs, and any solutions need to be flexible enough to account for variations in policy and requirements.

2. Definitions

This section provides definitions for actors, data types and data management architectures as they are discussed in this document. Different numbering spaces may instantiate these roles and concepts differently: practices that apply to non-geographic freephone numbers, for example, may not apply to geographic numbers, and practices that exist under one Numbering Authority may not be permitted under another. The purpose of this framework is to identify the characteristics of protocol tools that will satisfy the diverse requirements for telephone number acquisition, management, and retrieval on the Internet.

2.1. Actors

The following roles of actors are defined in this document:

Numbering Authority: A regulatory body within a region that manages that regions TNs. The Numbering Authority decides national numbering policy for the nation, region, or other domain for which it has authority, including what TNs can be allocated, which are reserved, and which entities may obtain TNs.

Registry: An entity that administers the allocation of TNs based on a Numbering Authority's policies. Numbering authorities can act as the Registries themselves, or they can outsource the function to other entities. Traditional registries are single entities with sole authority and responsibility for specific numbering resources, though distributed registries (see Section 2.3) are also in the scope of this framework.

Credential Authority: An entity that distributes credentials, such as certificates that attest the authority of assignees (defined below) and delegates. This document assumes that one of more credential authorities may be trusted by actors in any given regulatory environment; policies for establishing such trust anchors are outside the scope of this document.

Registrar: An entity that distributes the telephone numbers administered by a Registry; typically, there are many Registrars that can distribute numbers from a single Registry, though Registrars may serve multiple Registries as well. A Registrar has business relationships with number assignees and collects administrative information from them.

Communication Service Provider (CSP): A provider of communications services, where those services can be identified by TNs. This includes both traditional telephone carriers or enterprises as well as service providers with no presence on the PSTN who use TNs. This framework does not assume that any single CSP provides all the communications service related to a particular TN.

Service Enabler: An entity that works with CSPs to enable communication service to a User; perhaps a vendor, a service bureau, or third-party integrator.

User: An individual reachable through a communications service; usually a customer of a communication service provider.

Government Entity: An entity that, due to legal powers deriving from national policy, has privileged access to information about number administration under certain conditions.

Note that an individual, organization, or other entity may act in one or more of the roles above; for example, a company may be a CSP and also a Registrar. Although Numbering Authorities are listed as actors, they are unlikely to actually participate in the protocol flows themselves, though in some situations a Numbering Authority and Registry may be the same administrative entity.

All actors that are recipients of numbering resources, be they a CSP, Service Enabler, or User, can also be said to have a relationship to a Registry of either an assignee or delegate:

Assignee: An actor that is assigned a TN directly by a Registrar; an assignee always has a direct relationship with a Registrar.

Delegate: An actor that is delegated a TN from an assignee or another delegate, who does not necessarily have a direct relationship with a Registrar. Delegates may delegate one or more of their TN assignment(s) to one or more further downstream subdelegates.

As an example, consider a case where a Numbering Authority also acts as a Registry, and it issues blocks of 10,000 TNs to CSPs, which in this case also act as Registrars. CSP/Registrars would then be responsible for distributing numbering resources to Users and other CSPs. In this case, an enterprise deploying IP PBXs also acts as a CSP, and it acquires number blocks for its enterprise seats in chunks of 100 from a CSP acting as a Registrar with whom the enterprise has a business relationship. The enterprise is in this case the assignee, as it receives numbering resources directly from a Registrar. As it doles out individual numbers to its Users, the enterprise delegates its own numbering resources to those Users and their communications endpoints. The overall ecosystem might look as follows.

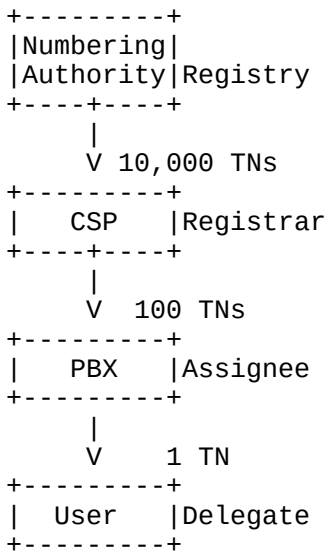

Figure 1: Chain of Number Assignment

2.2. Data Types

The following data types are defined in this document:

Administrative Data: assignment data related to the TN and the relevant actors; it includes TN status (assigned, unassigned, etc.), contact data for the assignee or delegate, and typically does not require real-time access as this data is not required for ordinary call or session establishment.

Service Data: data necessary to enable service for the TN; it includes addressing data and service features. Since this data is necessary to complete calls, it must be obtained in real time.

Administrative and service data can fit into three access categories:

Public: Anyone can access public data. Such data might include a list of which numbering resources (unallocated number ranges) are available for acquisition from the Registry.

Semi-restricted: Only a subset of actors can access semi-restricted data. For example CSPs may be able to access other CSP’s service data in some closed environment.

Restricted: Only a small subset of actors can access restricted data. For example a Government Entity may be able access contact information for a User.

While it might appear there are really only two categories, public and restricted based on requestor, the distinction between semi-restricted and restricted is helpful for the use cases below.

2.3. Data Management Architectures

This framework generally assumes that administrative and service data is maintained by CSPs, Registrars, and Registries. The terms "registrar" and "registry" are familiar from DNS operations, and indeed the DNS provides an obvious inspiration for the relationships between those entities described here. Protocols for transferring names between registries and registrars have been standardized in the DNS space for some time (see [14]). Similarly, the division between service data acquired by resolving names with the DNS protocol vs. administrative data about names acquired through WHOIS [15] is directly analogous to the distinction between service and administrative data described in Section 2.2. The major difference between the data management architecture of the DNS and this framework is that the distinction between the CSP and User, due to historical policies of the telephone network, will often not exactly correspond to the distinction between a name service and a registrant in the DNS world - a User in the telephone network is today at least rarely in a direct relationship with a Registrar comparable to that of a DNS registrant.

The role of a Registry described here is a "thin" one, where the Registry manages basic allocation information for the numbering space, such as information about whether or not the number is assigned, and if assigned, by which Registrar. It is the Registrar that in turn manages detailed administrative data about those assignments, such as contact or billing information for the assignee. In some models, CSPs and Registrars will be combined (the same administrative entity), and in others the Registry and Registrar may similarly be composed. Typically, service data resides largely at the CSP itself, though in some models a "thicker" Registry may itself contain a pointer to the servicing CSP for a number or number block. In addition to traditional centralized Registries, this framework also supports environments where the same data is being managed by multiple administrative entities, and stored in many locations. A distributed registry system is discussed further in [19]. To support those use cases, it is important to distinguish the following:

Data store: A Data Store is a service that stores and enables access to administrative and/or service data.

Reference Address: A Reference Address is a URL that dereferences to the location of the data store.

Distributed data stores: In a Distributed Data Store, administrative or service data can be stored with multiple actors. For example, CSPs could provision their service data to multiple other CSPs.

Distributed Registries: Multiple Registries can manage the same numbering resource. In these architectures, actors could interact with one or multiple Registries. The Registries would update each other when change occurs. The Registries have to ensure that data remains consistent, e.g. that the same TN is not assigned to two different actors.

3. Framework

The framework outlined in this document requires three Internet-based mechanisms for managing and resolving telephone numbers (TNs) in an IP environment. These mechanisms will likely reuse existing protocols for sharing structured data; it is unlikely that new protocol development work will be required, though new information models specific to the data itself will be a major focus of framework development. Likely candidates for reuse here include work done in DRINKS [4] and WEIRDS [12], as well as the TeRI [16] framework.

These protocol mechanisms are scoped in a way that makes them likely to apply to a broad range of future policies for number administration. It is not the purpose of this framework to dictate number policy, but instead to provide tools that will work with policies as they evolve going forward. These mechanisms therefore do not assume that number administration is centralized, nor that number allocations are restricted to any category of service providers, though these tools must and will work in environments with those properties.

The three mechanisms are:

Acquisition: a protocol mechanism for acquiring TNs, including an enrollment process.

Management: a protocol mechanism for associating data with TNs.

Retrieval: a protocol mechanism for retrieving data about TNs.

The acquisition mechanism will enable actors to acquire TNs for use with a communications service by requesting numbering resources from a service operated by a Registrar, CSP or similar actor. TNs may be requested either on a number-by-number basis, or as inventory blocks.

Any actor who grants numbering resources will retain metadata about the assignment, including the responsible organization or individual to whom numbers have been assigned.

The management mechanism will let actors provision data associated with TNs. For example, if a User has been assigned a TN, they may select a CSP to provide a particular service associated with the TN, or a CSP may assign a TN to a User upon service activation. In either case, a mechanism is needed to provision data associated with the TN at that CSP, and to extend those data sets as CSPs (and even Users) require.

The retrieval mechanism will enable actors to learn information about TNs. For real-time service data, this typically involves sending a request to a CSP; for other information, an actor may need to send a request to a Registry rather than a CSP. Different parties may be authorized to receive different information about TNs.

As an example, a CSP might use the acquisition interface to acquire a chunk of numbers from a Registrar. Users might then provision administrative data associated with those numbers at the CSP through the management interface, and query for service data relating to those numbers through the retrieval interface of the CSP.

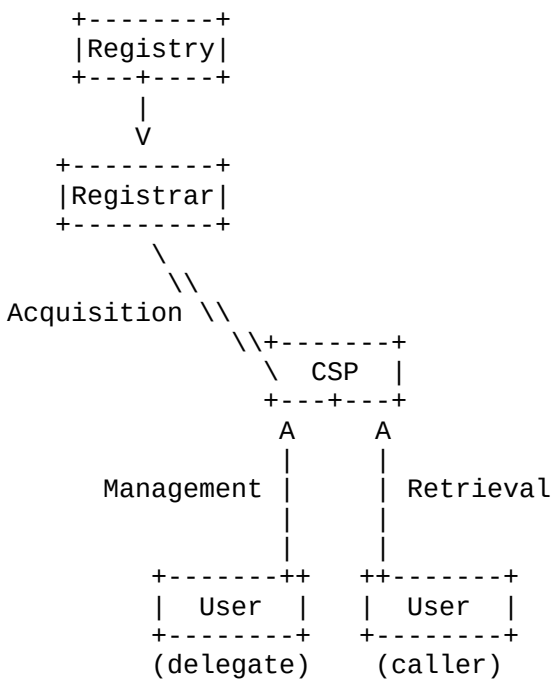

Figure 2: Example of the Three Interfaces

4. Use Cases

The high-level use cases in this section will provide an overview of the expected operation of the three interfaces in the MODERN problem space:

4.1. Acquisition

There are various scenarios for how TNs can be acquired by the relevant actors, that is, a CSP, Service Enabler, and a User. There are three actors from which numbers can be acquired: a Registrar, a CSP and a User (presumably one who is delegating to another party). It is assumed that Registrars are either the same entity as Registries, or that Registrars have established business relationships with Registries that enable them to distribute the numbers that the Registries administer. In these use cases, a User may acquire TNs either from a CSP or a Registry, or from an intermediate delegate.

4.1.1. Acquiring TNs from Registrar

The most traditional number acquisition use case is one where a CSP, such as a carrier, requests a block of numbers from a Registrar to hold as inventory or assign to customers.

Through some out-of-band business process, a CSP develops a relationship with a Registrar. The Registrar maintains a profile of the CSP and assesses whether or not CSPs meet the policy restrictions for acquiring TNs. The CSP may then request TNs from within a specific pool of numbers in the authority of the Registry; such as region, mobile, wireline, or freephone. The Registrar must authenticate and authorize the CSP, and then either grant or deny a request. When an assignment occurs, the Registry creates and stores administrative information related to the assignment such as TN status and Registrar contact information, and removes the specific TN(s) from the pool of those that are available for assignment. As a part of the acquisition and assignment process, the Registry provides to the Registrar any tokens or other material needed by a Credential Authority to issue credentials (for example, STIR certificates [17]) used to attest the assignment for future transactions. Depending on the policies of the Numbering Authorities, Registrars may be required to log these operations.

Before it is eligible to receive TN assignments, per the policy of a Numbering Authority, the CSP may need to have submitted (again, through some out-of-band process) additional qualifying information such as current utilization rate or a demand forecast.

There are two scenarios under which a CSP requests resources; they are requesting inventory, or they are requesting for a specific User or delegate. For the purpose of status information, TNs assigned to a User are always considered assigned, not inventory. The CSP will associate service information for that TN, e.g., service address, and make it available to other CSPs to enable interconnection. The CSP may need to update the Registrar regarding this service activation; this is part of the "TN status" maintained by the Registrar.

There are also use cases in which a User can acquire a TN directly from a Registrar. Today, a user wishing to acquire a freephone number may browse the existing inventory through one or more Registrars, comparing their prices and services. Each such Registrar either is a CSP, or has a business relationship with one or more CSPs to provide services for that freephone number. In this case, the User must establish some business relationship directly with a Registrar, similarly to how such functions are conducted today when Users purchase domain names. In this use case, after receiving a number assignment from the Registrar, a User will then obtain

communications service from a CSP, and provide to the CSP the TN to be used for that service. The CSP will associate service information for that TN, e.g., service address, and make it available to other CSPs to enable interconnection. The user will also need to inform the Registrar about this relationship.

4.1.2. Acquiring TNs from CSPs

Today, a User typically acquires a TN from CSP when signing up for communications service or turning on a new device. In this use case, the User becomes the delegate of the CSP. A reseller or a service bureau might also acquire a block of numbers from a CSP to be issued to Users.

Consider a case where a User creates or has a relationship with the CSP, and subscribes to a communications service which includes the use of a TN. The CSP collects and stores administrative data about the User. The CSP then activates the User on their network and creates any necessary service data to enable connectivity with other CSPs. The CSP could also update public or privileged databases accessible by other Actors. The CSP provides any tokens or other material needed by a Credential Authority to issue credentials to the User (for example, a STIR certificate [17]) to prove the assignment for future transactions. Such credentials could be delegated from the one provided by the Credential Authority to the CSP to continue the chain of assignment. CSPs may be required to log such transactions, if required by the policy of the Numbering Authority.

Virtually the same flow would work for a reseller: it would form a business relationship with the CSP, at which point the CSP would collect and store administrative data about the reseller and give the reseller any material needed for the reseller to acquire credentials for the numbers. A user might then in turn acquire numbers from the reseller: in this case, the delegate re-delegating the TNs would be performing functions done by the CSP, e.g., providing any credentials, collecting administrative data, or creative service data.

The CSP could assign a TN from its existing inventory or it could acquire a new TN from the Registrar as part of the assignment process. If it assigns it from its existing inventory, it would remove the specific TN from the pool of those available for assignment. It may also update the Registrar about the assignment so the Registrar has current assignment data. If a reseller or delegate CSP is acquiring the numbers, it may have the same obligations to provide utilization data to the Registry as the assignee, per Section 4.1.1.

4.2. Management

The management protocol mechanism is needed to associate administrative and service data with TNs, and may be used to refresh or rollover associated credentials.

4.2.1. Management of Administrative Data

Administrative data is primarily related to the status of the TN, its administrative contacts, and the actors involved in providing service to the TN. Protocol interactions for administrative data will therefore predominantly occur between CSPs and Users to the Registrar, or between Users and delegate CSPs to the CSP.

Some administrative data may be private, and would thus require special handling in a distributed data store model. Access to it does not require real-time performance therefore local caches are not necessary. And it will include sensitive information such as user and contact data.

Some of the data could lend itself to being publicly available, such as CSP and TN assignment status. In that case it would be deemed public information for the purposes of the retrieval interface.

4.2.1.1. Managing Data at a Registrar

After a CSP acquires a TN or block of TNs from the Registrar (per Section 4.1.1 above), it then provides administrative data to the Registrar as a step in the acquisition process. The Registrar will authenticate the CSP and determine if the CSP is authorized to provision the administrative data for the TNs in question. The Registry will update the status of the TN, i.e., that it is unavailable for assignment. The Registrar will also maintain administrative data provided by the CSP.

Changes to this administrative data will not be frequent. Examples of changes would be terminating service (see Section 4.2.3.2), changing the name or address of a User or organization, or changing a CSP or delegate. Changes should be authenticated by a credential to prove administrative responsibility for the TN.

In some cases, such as the freephone system in North America today, the User has a direct relationship with the Registrar. Naturally, these users could provision administrative data associated with their TNs directly to the Registrar, just as a freephone provider today maintains account and billing data. While delegates may not ordinarily have a direct relationship to a Registrar, some environments as an optimization might want to support a model where

the delegate updates the Registrar directly on changes, as opposed to sending that data to the CSP or through the CSP to the Registrar. As stated already, the protocol should enable Users to acquire TNs directly from a Registrar, which Registrar may or may not also act as a CSP. In these cases the updates would be similar to that described in Section 4.2.1.1.

In a distributed Registry model, TN status, e.g., allocated, assigned, available, unavailable, would need to be provided to other Registries in real-time. Other administrative data could be sent to all Registries or other Registries could get a reference address to the host Registry's data store.

4.2.1.2. Managing Data at a CSP

After a User acquires a TN or block of TNs from a CSP, the User will provide administrative data to the CSP. The CSP commonly acts as a Registrar in this case, maintaining the administrative data and only notifies the Registry of the change in TN status. In this case, the Registry maintains a reference address (see Section 2.3) to the CSP/ Registrar's administrative data store so relevant actors have the ability to access the data. Alternatively, a CSP could send the administrative data to an external Registrar to store. If there is a delegate between the CSP and user, they will have to ensure there is a mechanism for the delegate to update the CSP as change occurs.

4.2.2. Management of Service Data

Service data is data required by an originating or intermediate CSP to enable communications service to a User: a SIP URI is an example of one service data element commonly used to route communications. CSPs typically create and manage service data, however, it is possible that delegates and Users could as well. For most use cases involving individual Users, it is anticipated that lower-level service information changes (such as an end-user device receiving a new IP address) would be communicated to CSPs via existing protocols. For example, the baseline SIP REGISTER [2] method, even for bulk operations [13], would likely be used rather than through any new interfaces defined by MODERN.

4.2.2.1. CSP to other CSPs

After a User enrolls for service with a CSP, in the case where the CSP was assigned the TN by a Registrar, the CSP will then create a service address such as a SIP URI and associate it with the TN. The CSP needs to update this data to enable service interoperability. There are multiple ways that this update can occur, though most commonly service data is exposed through the retrieval interface (see

Section 4.3). For certain deployment architectures, like a distributed data store model, CSPs may need to provision data directly to other CSPs.

If the CSP is assigning a TN from its own inventory it may not need to perform service data updates as change occurs because the existing service data associated with inventory may be sufficient once the TN is put in service. They would however likely update the Registry on the change in status.

4.2.2.2. User to CSP

Users could also associate service data to their TNs at the CSP. An example is a User acquires a TN from the Registrar (as described in Section 4.1.1) and wants to provide that TN to the CSP so the CSP can enable service. In this case, once the user provides the number to the CSP, the CSP would update the Registry or other actors as outlined in Section 4.2.2.1.

4.2.3. Managing Change

This section will address some special management use cases that were not covered above.

4.2.3.1. Changing the CSP for an Existing Service

Consider the case where a User who subscribes to a communications service, and received their TN from that CSP, wishes to retain the same TN but move their service to a different CSP.

In the simplest scenario, where there’s an authoritative combined Registry/Registrar that maintains service data, the User could provide their credential to the new CSP and let the CSP initiate the change in service. The new CSP could then provide the new service data with the User’s credential to the Registry/Registrar, which then makes the change. The old credential is revoked and a new one is provided. The new CSP or the Registrar would send a notification to the old CSP, so they can disable service. The old CSP will undo any delegations to the User, including contacting the Credential Authority to revoke any cryptographic credentials (e.g., STIR certificates [17]) previously granted to the User. Any service data maintained by the CSP must be removed, and similarly, the CSP must delete any such information it provisioned in the Registry.

In a model similar to common practice in environments today, the User could alternatively provide their credential to the old CSP, and the old CSP initiates the change in service. Or, a User could go

directly to a Registrar to initiate a port. This framework should support all of these potential flows.

Note that in cases with a distributed Registry that maintained service data, the Registry would also have to update the other Registries of the change.

4.2.3.2. Terminating a Service

Consider a case where a user who subscribes to a communications service, and received their TN from the CSP, wishes to terminate their service. At this time, the CSP will undo any delegations to the User, which may involve contacting the Credential Authority to revoke any cryptographic credentials (e.g., STIR certificates [17]) previously granted to the User. Any service data maintained by the CSP must be removed, and similarly, the CSP must delete any such information it provisioned in the Registrar. However, per the policy of the Numbering Authority, Registrars and CSPs may be required to preserve historical data that will be accessible to Government Entities or others through audits, even if it is no longer retrievable through service interfaces.

The TN will change state from assigned to unassigned, the CSP will update the Registry. Depending on policies the TN could go back into the Registry, CSP, or delegate’s pool of available TNs and would likely enter an ageing process.

In an alternative use case, a User who received their own TN assignment directly from a Registrar terminates their service with a CSP. At this time, the User might terminate their assignment from the Registrar, and return the TN to the Registry for re-assignment. Alternatively, they could retain the TN and elect to assign it to some other service at a later time.

4.3. Retrieval

Retrieval of administrative or service data will be subject to access restrictions based on the category of the specific data: public, semi-restricted or restricted. Both administrative and service data can have data elements that fall into each of these categories. It is expected that the majority of administrative will fall into the semi-restricted category: access to this information may require some form of authorization, though service data crucial to reachability will need to be accessible. In some environments, it’s possible that none of the service data necessary to initiate communications will be useful to an entity on the public Internet, say, or that all that service data will have dependencies on the origination point of calls.

The retrieval protocol mechanism for semi-restricted and restricted data needs a way for the receiver of the request to identify the originator of the request and what is being requested. The receiver of the request will process that request based on this information.

4.3.1. Retrieval of Public Data

Either administrative or service data may be made publicly available by the authority that generates and provisions it. Under most circumstances, a CSP wants its communications service to be publicly reachable through TNs, so the retrieval interface supports public interfaces that permit clients to query for service data about a TN. Some service data may however require that the client be authorized to receive it, per the use case in Section 4.3.3 below.

Public data can simply be posted on websites or made available through a publicly available API. Public data hosted by a CSP may have a reference address at the Registry.

4.3.2. Retrieval of Semi-restricted Administrative Data

Consider a case in which a CSP is having service problems completing calls to a specific TN, so it wants to contact the CSP serving that TN. The Registry authorizes the originating CSP to access this information. It initiates a query to the Registry, the Registry verifies the requestor and the requested data and Registry responds with the serving CSP and contact data. However, CSPs might not want to make those administrative contact points public data: they are willing to share them with other CSPs for troubleshooting purposes, but not to make them available to general communication.

Alternatively that information could be part of a distributed data store and not stored at a monolithic Registry. In that case, the CSP has the data in a local distributed data store and it initiates the query to the local data store. The local data store responds with the CSP and contact data. No verification is necessary because it was done when the CSP was authorized to receive the data store.

4.3.3. Retrieval of Semi-restricted Service Data

Consider a case where a User on a CSP’s network calls a TN. The CSP initiates a query for service data associated with the TN to complete the call, and will receive special service data because the CSP operates in a closed environment where different CSPs receive different responses, and only participating CSPs can initiate communications. This service data would be flagged as semi-restricted. The query and response have real-time performance requirements in that environment.

Semi-restricted service data also works in a distributed data store model, where each CSP distributes its updated service data to all other CSPs. The originating CSP has the service data in its local data store and queries it. The local data store responds with the service data. The service data in the response can be a reference address to a data store maintained by the serving CSP, or it can be the service address itself. In the case where the response gives a reference address, a subsequent query would go to the serving CSP, who would in turn authorize the requestor for the requested data and respond appropriate. In the case where the original response contains the service address, the requestor would use that service address as the destination for the call.

In some environments, aspects of the service data may reside at the Registry itself (for example, the assigned CSP for a TN), and thus the query may be sent to the Registry. The Registry verifies the requestor and the requested data and responds with the service data, such as a SIP URI containing the domain of the assigned CSP.

4.3.4. Retrieval of Restricted Data

A Government Entity wishes to access information about a particular User, who subscribes to a communications service. The entity that operates the Registry on behalf of the Numbering Authority in this case has some pre-defined relationship with the Government Entity. When the CSP acquired TNs from the Numbering Authority, it was a condition of that assignment that the CSP provide access for Government Entities to telephone numbering data when certain conditions apply. The required data may reside either in the CSP or in the Registrar.

For a case where the CSP delegates a number to the User, the CSP might provision the Registrar (or itself, if the CSP is composed with a Registrar) with information relevant to the User. At such a time as the Government Entity needs information about that User, the Government Entity may contact the Registrar or CSP to acquire the necessary data. The interfaces necessary for this will be the same as those described in Section 4.3; the Government Entity will be authenticated, and an authorization decision will be made by the Registrar or CSP under the policy dictates established by the Numbering Authority.

5. Acknowledgments

We would like to thank Henning Schulzrinne and Adam Roach for their contributions to this problem statement and framework, and to thank Pierce Gorman for detailed comments.

6. IANA Considerations

This memo includes no instructions for the IANA.

7. Privacy Considerations

This framework defines two categories of information about telephone numbers: service data and administrative data. Service data describes how telephone numbers map to particular services and devices that provide real-time communication for users. As such, service data could potentially leak resource locations and even lower-layer network addresses associated with these services, and in rare cases, with end-user devices. Administrative data more broadly characterizes who the administrative entities are behind telephone numbers, which will often identify CSPs, but in some layers of the architecture could include personally identifying information (PII), even WHOIS-style information, about the end users behind identifiers. This could conceivably encompass the sorts of data that carriers and similar CSPs today keep about their customers for billing purposes, like real names and postal addresses. The exact nature of administrative data is not defined by this framework, and it is anticipated that the protocols that will perform this function will be extensible for different use cases, so at this point, it is difficult to characterize exactly how much PII might end up being housed by these services.

As such, if an attacker were to compromise the registrar services in this architecture which maintain administrative data, and in some cases even service data, this could leak PII about end users. These interfaces, and the systems that host them, are a potentially attractive target for hackers and need to be hardened accordingly. Protocols that are selected to fulfill these functions must provide the security features described in [Sec Cons].

Finally, this framework recognizes that in many jurisdictions, certain government agencies have a legal right to access service and administrative data maintained by CSPs. This access is typically aimed at identifying the users behind communications identifiers in order to enforce regulatory policy. Those legal entities already have the power to access the existing data held by CSPs in many jurisdictions, though potentially the administrative data associated with this framework could be richer information.

8. Security Considerations

The acquisition, management, and retrieval of administrative and service data associated with telephone numbers raises a number of security issues.

Any mechanism that allows an individual or organization to acquire telephone numbers will require a means of mutual authentication, of integrity protection, and of confidentiality. A Registry as defined in this document will surely want to authenticate the source of an acquisition request as a first step in the authorization process to determine whether or not the resource will be granted. Integrity of both the request and response is essential to ensuring that tampering does not allow attackers to block acquisitions, or worse, to commandeer resources. Confidentiality is essential to preventing eavesdroppers from learning about allocations, including the personally identifying information associated with the administrative or technical contracts for allocations.

A management interface for telephone numbers has similar requirements. Without proper authentication and authorization mechanisms in place, an attack could use the management interface to disrupt service data or administrative data, which could deny service to users, enable new impersonation attacks, prevent billing systems from operating properly, and cause similar system failures.

Finally, a retrieval interfaces has its own needs for mutual authentication, integrity protection, and for confidentiality. Any CSP sending a request to retrieve service data associated with a number will want to know that it is reaching the proper authority, that the response from that authority has not been tampered with in transit, and in most cases the CSP will not want to reveal to eavesdroppers the number it is requesting or the response that it has received. Similarly, any service answering such a query will want to have a means of authenticating the source of the query, and of protecting the integrity and confidentiality of its responses.

9. Informative References

[1] Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 4474, DOI 10.17487/RFC4474, August 2006, <https://www.rfc-editor.org/info/rfc4474>.

[2] 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>.

[3] 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>.

[4] Channabasappa, S., Ed., "Data for Reachability of Inter-/Intra-NetworK SIP (DRINKS) Use Cases and Protocol Requirements", RFC 6461, DOI 10.17487/RFC6461, January 2012, <https://www.rfc-editor.org/info/rfc6461>.

[5] Watson, M., "Short Term Requirements for Network Asserted Identity", RFC 3324, DOI 10.17487/RFC3324, November 2002, <https://www.rfc-editor.org/info/rfc3324>.

[6] Jennings, C., Peterson, J., and M. Watson, "Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks", RFC 3325, DOI 10.17487/RFC3325, November 2002, <https://www.rfc-editor.org/info/rfc3325>.

[7] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <https://www.rfc-editor.org/info/rfc6698>.

[8] Elwell, J., "Connected Identity in the Session Initiation Protocol (SIP)", RFC 4916, DOI 10.17487/RFC4916, June 2007, <https://www.rfc-editor.org/info/rfc4916>.

[9] Schulzrinne, H., "The tel URI for Telephone Numbers", RFC 3966, DOI 10.17487/RFC3966, December 2004, <https://www.rfc-editor.org/info/rfc3966>.

[10] Rosenberg, J. and C. Jennings, "The Session Initiation Protocol (SIP) and Spam", RFC 5039, DOI 10.17487/RFC5039, January 2008, <https://www.rfc-editor.org/info/rfc5039>.

[11] Peterson, J., Jennings, C., and R. Sparks, "Change Process for the Session Initiation Protocol (SIP) and the Real-time Applications and Infrastructure Area", BCP 67, RFC 5727, DOI 10.17487/RFC5727, March 2010, <https://www.rfc-editor.org/info/rfc5727>.

[12] Newton, A. and S. Hollenbeck, "Registration Data Access Protocol (RDAP) Query Format", RFC 7482, DOI 10.17487/RFC7482, March 2015, <https://www.rfc-editor.org/info/rfc7482>.

[13] Roach, A., "Registration for Multiple Phone Numbers in the Session Initiation Protocol (SIP)", RFC 6140, DOI 10.17487/RFC6140, March 2011, <https://www.rfc-editor.org/info/rfc6140>.

[14] Hollenbeck, S., "Generic Registry-Registrar Protocol Requirements", RFC 3375, DOI 10.17487/RFC3375, September 2002, <https://www.rfc-editor.org/info/rfc3375>.

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

[16] Peterson, J., "An Architecture and Information Model for Telephone-Related Information (TeRI)", draft-peterson-modern-teri-03 (work in progress), July 2017.

[17] Peterson, J. and S. Turner, "Secure Telephone Identity Credentials: Certificates", RFC 8226, DOI 10.17487/RFC8226, February 2018, <https://www.rfc-editor.org/info/rfc8226>.

[18] Barnes, M., Jennings, C., Rosenberg, J., and M. Petit-Huguenin, "Verification Involving PSTN Reachability: Requirements and Architecture Overview", draft-jennings-vipr-overview-06 (work in progress), December 2013.

[19] Wendt, C. and H. Bellur, "Distributed Registry Protocol (DRiP)", draft-wendt-modern-drip-02 (work in progress), July 2017.

[20] 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>.

Authors’ Addresses

Jon Peterson
Neustar, Inc.
1800 Sutter St Suite 570
Concord, CA 94520
US

Email: jon.peterson@neustar.biz

Tom McGarry
Neustar, Inc.
1800 Sutter St Suite 570
Concord, CA 94520
US

Email: tom.mcgarry@neustar.biz

Network Working Group
Internet-Draft
Intended status: Informational
Expires: September 6, 2018

J. Peterson
Neustar
C. Wendt
Comcast
March 5, 2018

Using Telephone Related Information (TeRI) with the Distributed Registry Protocol (DRiP)
draft-peterson-modern-drip-teri-00.txt

Abstract

The Distributed Registry Protocol (DRiP) allows a set of nodes to implement a decentralized registry function. This document explores how Telephone Related Information (TeRI) Records can be shared by DRiP, and a decentralized registry approaches the operations necessary to assign, provision, and route for telephone numbers.

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 September 6, 2018.

Copyright Notice

Table of Contents

1. Introduction

The Distributed Registry Protocol (DRiP) [I-D.wendt-modern-drip] is a protocol that enables a distributed set of nodes to synchronize information in real-time with a minimal amount of delay. DRiP assumes a peer-to-peer gossip network that shares key-value pairs. The protocol is intended to carry information related to personal communication, including identifiers like telephone numbers and related identity information about the participants in communication.

The Telephone Related Information [I-D.peterson-modern-teri] information model provides a framework for distributing Records that convey service or administrative data about telephone numbers. TeRI Records are signed by entities such as CSPs or Registrars who possess credentials which enable relying parties to trust that Records have been created or modified by the appropriate parties for a particular telephone number, such as STIR [RFC8226] certificates. In TeRI parlance, anyone holding such a credential attesting authority over telephone number resources is called an "Authority." TeRI Records containing service data provide routing information for telephone numbers, and may be retrieved from local caches, remote services, or even a distributed network, as relying parties trust Authorities rather than services. The TeRI Record format therefore seems suitable for distribution via DRiP.

Following the MODERN framework [I-D.ietf-modern-problem-framework], TeRI usages for centralized registries support three fundamental operations on telephone numbers: acquisition, management, and

retrieval. There are at a high level a couple of potential ways to approach using TeRI with DRiP: for example, the gossip protocol could be used as a transport layer to pass client-server requests to what is effectively a centralized Authority that actually creates and signs TeRI Records; or, more interestingly, nodes in the gossip network might all act as Authorities, possessing credentials that enable them to create and sign TeRI Records themselves and then vote on their validity to prevent conflicts and race conditions. The possibility of a decentralized registry based on the latter principle largely motivates this exploration of the intersection between TeRI and DRiP.

This initial draft explores some key use cases for TeRI over DRiP, and how they differ from the use cases already given in the baseline MODERN framework [I-D.ietf-modern-problem-framework].

2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in [RFC2119].

3. Use Cases

3.1. Assumptions

It is assumed in the MODERN system that before any actor can interact with a Registry, a Credential Authority provides certificate credentials to individual actors corresponding to their identity and role: such as the CSPs and Users of the MODERN framework [I-D.ietf-modern-problem-framework]. For the purposes of this document, we assume those credentials are STIR [RFC8226] certificates associated with telephone number blocks that can be managed by the distributed Registry.

These use cases explore the possibility that there could be more than one distinct administrative entity who holds credentials authorizing them to generate TeRI Records for the same numbering resources: effectively, that set of Authorities functions as a distributed Registry. Imagine, for example, that in the North American Numbering Plan an experimental area code were created that enabled any CSP authorized by the distributed Registry to reserve a new telephone number on a first come first serve basis - with some policy controls.

For these use cases we assume that a policy governs the acquisition of telephone numbers in the distributed Registry. For example, there could be a financial cost of acquiring numbers that goes up

dramatically if CSPs acquire resources too greedily; or alternatively, there could be some policy enforcement of the ratio of allocated to assigned numbers a CSP has claimed to maintain an agreed reasonable level of number inventory per CSP. In the DRiP gossip network, there could moreover be a "policy node" that simply votes "no" when new TeRI Records that conflict with the policy are propagated. These initial use cases are however "sunny day" cases where we assume that CSPs scrupulously adhere to policy.

3.2. CSP Acquires a Block

In the following case, a CSP performs acquires a block of telephone numbers, placing it under their control using their credentials.

First, assume that a Numbering Authority has unallocated number blocks that are eligible for allocation, and that these resources have been made available to the distributed Registry.

Then, a CSP participating in the distributed Registry declares its intention to allocate one such block of Numbers. In centralized MODERN architectures, the CSP would send a TeRI acquisition operation query to the Registry, and receive a Record for the Block (along with associated credentials) from the Registry response. In this distributed architecture, the CSP simply creates a new administrative TeRI Record for the block and signs it with its own credential. It then propagates that Record through the gossip network. If no node in the network votes against the Record, it is cached by all nodes, and that Record becomes a new administrative Record for that block.

Note that per the MODERN framework [I-D.ietf-modern-problem-framework], a CSP can act directly as a Registrar itself, or it can use a third-party Registrar to effect these transactions.

3.3. CSP Acquires a Single Number

In the following case, a CSP acquires a single available number in a block. Imagine, for example, that a new freephone area code 8yy were allocated in the North American Numbering Plan that allowed any Responsible Organization to acquire numbers on a first-come-first-serve basis under some governing policy.

First, assume that there are numbers under 8yy available for assignment.

Then, a CSP acting as a RespOrg participating in the distributed Registry declares its intention to allocate and assign a number to a customer. In this distributed architecture, the CSP simply creates a

new administrative TeRI Record for the individual TN and signs it with its own credential, marking it as assigned. It then propagates that Record through the gossip network. If no node in the network votes against the Record, it is cached by all nodes, and that Record becomes a new administrative Record for that block.

3.4. CSP Assigns a Single Number within its Block

In the following case, a CSP had already allocated a block to itself per Section 3.2. Now, it intends to assign a single number in that block.

In centralized MODERN architectures, the CSP would contact the Registry with a TeRI management operation, notifying the Registry that the number's status had changed to assigned. In this distributed Registry, the CSP creates a new TeRI record for that individual number, marking it as assigned, signs it, and then propagates that Record through the gossip network.

The same would apply for marking a sub-block within the block as assigned: the CSP creates a new TeRI record for that individual sub-block, marking it as assigned, signs it, and then propagates that Record through the gossip network.

3.5. Number Porting

The most difficult use cases for the distributed Registry are ones where control of resources has been allocated or assigned to one CSP but must now move to a new CSP. Number portability is the most common cause of this, though various other business reasons might result in changes of control over allocated and/or assigned numbers.

Suppose that CSP B has allocated and assigned a block to itself, and that TeRI records for that block are cached throughout the gossip network. Now, CSP A declares its intention to assign a particular TN within the block of CSP B. CSP A does so by creating a new TeRI Record for the number which CSP A signs, allocating the number to itself and marking it as assigned. As this Record propagates through the gossip network, CSP B recognizes this transaction and does not vote "no", in effect authorizing the transfer. If CSP B's customer were not porting the number to CSP A, then CSP B would vote "no."

From a policy oversight perspective, this could require a "policy node" or similar actor in the network to make sure it is not abused.

4. Identity Elements in TeRI

[Future versions of this specification will explore extensions to baseline TeRI for DRiP use cases.]

5. Acknowledgments

We would like to thank you for your contributions to this problem statement and framework.

6. IANA Considerations

This document contains no instructions for the IANA.

7. Security Considerations

TBD.

8. Informative References

[I-D.ietf-acme-acme]
Barnes, R., Hoffman-Andrews, J., McCarney, D., and J. Kasten, "Automatic Certificate Management Environment (ACME)", draft-ietf-acme-acme-09 (work in progress), December 2017.

[I-D.ietf-acme-service-provider]
Barnes, M. and C. Wendt, "ACME Identifiers and Challenges for VoIP Service Providers", draft-ietf-acme-service-provider-02 (work in progress), October 2017.

[I-D.ietf-acme-star]
Sheffer, Y., Lopez, D., Dios, O., Pastor, A., and T. Fossati, "Support for Short-Term, Automatically-Renewed (STAR) Certificates in Automated Certificate Management Environment (ACME)", draft-ietf-acme-star-03 (work in progress), March 2018.

[I-D.ietf-acme-telephone]
Peterson, J. and R. Barnes, "ACME Identifiers and Challenges for Telephone Numbers", draft-ietf-acme-telephone-01 (work in progress), October 2017.

[I-D.ietf-modern-problem-framework]
Peterson, J. and T. McGarry, "Modern Problem Statement, Use Cases, and Framework", draft-ietf-modern-problem-framework-03 (work in progress), July 2017.

[I-D.ietf-stir-certificates]
Peterson, J. and S. Turner, "Secure Telephone Identity Credentials: Certificates", draft-ietf-stir-certificates-18 (work in progress), December 2017.

[I-D.ietf-stir-passport]
Wendt, C. and J. Peterson, "Personal Assertion Token (PASSporT)", draft-ietf-stir-passport-11 (work in progress), February 2017.

[I-D.ietf-stir-rfc4474bis]
Peterson, J., Jennings, C., Rescorla, E., and C. Wendt, "Authenticated Identity Management in the Session Initiation Protocol (SIP)", draft-ietf-stir-rfc4474bis-16 (work in progress), February 2017.

[I-D.peterson-modern-teri]
Peterson, J., "An Architecture and Information Model for Telephone-Related Information (TeRI)", draft-peterson-modern-teri-03 (work in progress), July 2017.

[I-D.rescorla-stir-fallback]
Rescorla, E. and J. Peterson, "STIR Out of Band Architecture and Use Cases", draft-rescorla-stir-fallback-02 (work in progress), June 2017.

[I-D.wendt-modern-drip]
Wendt, C. and H. Bellur, "Distributed Registry Protocol (DRiP)", draft-wendt-modern-drip-02 (work in progress), July 2017.

[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>.

[RFC7340] Peterson, J., Schulzrinne, H., and H. Tschofenig, "Secure Telephone Identity Problem Statement and Requirements", RFC 7340, DOI 10.17487/RFC7340, September 2014, <https://www.rfc-editor.org/info/rfc7340>.

[RFC7515] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Signature (JWS)", RFC 7515, DOI 10.17487/RFC7515, May 2015, <https://www.rfc-editor.org/info/rfc7515>.

[RFC7519] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Token (JWT)", RFC 7519, DOI 10.17487/RFC7519, May 2015, <https://www.rfc-editor.org/info/rfc7519>.

[RFC8226] Peterson, J. and S. Turner, "Secure Telephone Identity Credentials: Certificates", RFC 8226, DOI 10.17487/RFC8226, February 2018, <https://www.rfc-editor.org/info/rfc8226>.

Authors' Addresses

Jon Peterson
Neustar, Inc.
1800 Sutter St Suite 570
Concord, CA 94520
US

Email: jon.peterson@neustar.biz

Chris Wendt
Comcast
One Comcast Center
Philadelphia, PA 19103
USA

Email: chris-ietf@chriswendt.net

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 6, 2018

J. Peterson
Neustar, Inc.

March 5, 2018

An Architecture and Information Model for Telephone-Related Information (TeRI)
draft-peterson-modern-teri-04

Abstract

As telephone services migrate to the Internet, Internet applications require tools to access and manage information about telephone numbers. This document specifies a protocol-independent framework and information model for managing service and administration data related to telephone numbers.

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 September 6, 2018.

Copyright Notice

Table of Contents

1. Terminology

In this document, the key words "MAY", "MUST, "MUST NOT", "SHOULD", and "SHOULD NOT", are to be interpreted as described in [RFC2119]. This document also incorporates the terminology of the MODERN Framework [I-D.ietf-modern-problem-framework].

2. Motivation

Telephone numbers remain the worldwide standard identifier for routing calls and text messages over the Public Switched Telephone Network (PSTN). Increasingly, real-time communications is migrating to the Internet, and bringing telephone numbers with it. As identifiers, however, telephone numbers differ fundamentally from those commonly used by Internet applications. Email, the web and native Voice over IP (VoIP) systems such as SIP ([RFC3261]) use identifiers that rely on the Domain Name System (DNS) to resolve a domain portion of the identifier to a particular IP address; commonly, Uniform Resource Indicators (URIs) with a user and host

component serve this purpose. SIP, for example, quickly developed a convention for using a TEL URI in the user part of its URIs. To help telephone numbers work similarly on the Internet, a number of efforts have specified mechanisms to manage and retrieve information about telephone numbers via network services.

The ENUM ([RFC6116]) effort originally specified a public DNS profile for translating telephone numbers into URIs. Due to the difficulty of coordinating the public administration of telephone numbers in the DNS, this work transitioned to "infrastructure" ENUM ([RFC5067]), which assumed private DNS implementations, each of which could give a different answer to the same request to translate a telephone number depending on who asked, or other internal factors. The framework of the SPEERMINT working group ([RFC6406]), expanding on these requirements, differentiated the mapping of a telephone number to a target network (the "Look-up Function") from the mapping made by the originating network to the proper next-hop to reach such a target network (the "Location Routing Function"). To provision the data associated with telephone numbers, the DRINKS working group ([RFC6461]) designed systems for uploading back-end data to the services that would answer ENUM queries.

None of the preceding efforts, however, encompassed the entire lifecycle of a telephone number as an Internet identifier. They focused largely on service data, on how to "resolve" a telephone number to a location on the Internet, rather than on administrative questions of how numbers are acquired, how the entities associated with telephone numbers are authorized to provision data, and what kinds of systems need to be in place to allow a diverse community of devices, applications and users to rely on telephone numbers. Early considerations were moreover based on overlapping, but not entirely consistent, information models: intrinsic limitations in the DNS kept the queries and responses of ENUM relatively simple, whereas the DRINKS provisioning system considered a much richer syntax.

The need for solutions in this space is pressing, as many carriers worldwide contemplate migrating their entire PSTN infrastructure onto the Internet within the next decade. Further pressures come from emerging Internet communications providers who never invested in PSTN infrastructure in the first place, but want access to services related to telephone numbers. This includes devices, services, and applications on the Internet that make use of telephone numbers and need to distribute and manage numbering inventory: for example, an Internet-enabled PBX that might want to automate the process for allowing new connected phones to acquire numbers and provision contact information for their users. Ultimately, the resources identified by telephone numbers must also be reachable on the Internet, and different applications might want to use different

protocols to retrieve information about numbers. In some environments, there are performance constraints that would require a very lightweight binary protocol; in others, applications might prefer human-readable markup languages suitable for interfacing with existing APIs. The use cases associated with these functions are detailed in [I-D.ietf-modern-problem-framework].

Therefore, this document proposes a reconsideration of telephone service and administration data on the Internet, based on an information model that allows records associated with telephone number to be created, modified and accessed through network interfaces. This document specifies no particular syntax or encoding for queries or responses, but instead describes an extensible information model for the semantics of provisioning and querying operations associated with a telephone number.

3. The Information Model

The fundamental building block of the TeRI model is the Record. A Record is created by an Authority who has authority over a particular telephone number or a set of numbers. There may be more than one Authority who is authorized to create Records for a particular telephone number, and a TeRI service may have multiple Records corresponding to a single telephone number, including potentially overlapping Records associated with a range of numbers that encompasses a particular telephone number. Under various circumstances detailed in Section 5, participants in the numbering ecosystem may create, read, update, and modify Records.

Records contain Elements that hold data about the telephone number. Elements in this information model have a Name, which may optionally be associated with a Type and Value. Records are divided into two broad categories: Administrative Records and Service Records. Administrative Records hold data about how records have been allocated that is typically generated by a Registrar or similar entity that distributes numbers; they include information on the administrative contacts for telephone numbers, and so on. Service Records hold data required to initiate communication with the resources reachable at a telephone number; these Records are typically generated by an assignee or delegate such as a CSP.

The distinction between Administrative and Service Records exists because different parties might only need acces to one sort of information instead of another: moreover, some actors may be authorized to view Service Records for a particular telephone number but not Administrative Records, or vice versa. In practice, a Record may contain both Administrative and Service Elements, but the creators of Records may find it useful to keep the two types of

information separate. If a Record contains both Administrative and Service Elements, it may be returned in a Retrieval Query for either, provided the Client is authorized to receive the Elements within.

3.1. Record Elements

A Record is made up of Elements, which may contain Service or Administrative Data. All Records can contain the following generic Elements.

3.1.1. Identifier

Every Record has an Identifier, which is a globally unique identifier of the Record. The Identifier will typically be created at the same time as the Record itself, at a time when an assignment or delegation has occurred (as described in [I-D.ietf-modern-problem-framework]).

3.1.2. Authority

Every Record contains an Authority Element indicating the source of the data: either the entity that provisioned the data with the Service, or the external source from which the Service collected the data. The Authority element ideally gives a logical identity of the source of the data. A public key value may also be associated with an Authority element.

3.1.3. Access

Every Record contains an Access Element indicating the conditions under which Retrieval Requests can acquire the Record. The Access Element is set by the Authority generating the Record.

3.1.4. Subject

Every Record has a Subject. As TeRI Records concern telephone numbers, the Subject of a Record is an array of either a telephone number type or a telephone number range type. The simplest Record Subject is an array with one element consisting of a single telephone number.

3.1.5. Signature

Optionally, a Record contains a Signature element. The Signature element contains a signature over the concatenation of the other elements given the Record. Signatures are provided by the Authority responsible for the Record.

[Syntax TBD]

3.1.6. Administrative Elements

Records that contain Administrative Elements are Administrative Records. The baseline TeRI specification sets only one Administrative Element, the Contact.

3.1.6.1. Contact

Every Administrative Record has at least one Contact. The Contact contains administrative data about the assignee of the telephone number, though additionally Contacts may contain information about delegates (as defined in [I-D.ietf-modern-problem-framework]). Typically, this information would be set by the Registrar; policies outside the scope of this specification dictate the sorts of entities that may be designated as Contacts in Records.

3.1.7. Service Elements

Records that contain a Service Element are Service Records. The most important Service Element is simply called Service, and it contains an identifier for a communications resource reachable through a telephone number. More than one Service Element can appear in a given Record. Other Service Elements may be defined by later specifications.

3.1.7.1. Service

Records optionally have one or more Service entries. A Service may be of any Service Type, as given in Section 3.2.1. Optionally, subelements modify how a Service Element should be retained.

3.1.7.1.1. Priority

Optionally, a Service may specify a weighted Priority associated with a Record. Priorities are between 0 and 1, with a value of 1 having the highest priority.

3.1.7.1.2. Expiration

Optionally, a Service may specify an absolute time at which a Record will no longer be valid, should a client or intermediary wish to cache a Record. In the absence of an Expiration element, Records may be cached for a maximum of twenty-four hours.

3.2. Element Value Types

The remainder of a Record is made up of Elements. Elements types are specified in this section. Every Element Type has a Type Code. A Type Code is used as a short form for the Element in a Record.

3.2.1. Service Types

3.2.1.1. Telephone Number Type

The telephone number type conforms to the telephone number syntax given in [RFC3966] Section 3, in the ABNF for "telephone-subscriber."

Type Code: T

[TBD - need for subtying? E.164, Service Code, Short Code, Prefix, Nationally-Specific and Unknown.]

3.2.1.1.1. TN Range Type

The TN range type consists of a prefix of a telephone number (per [RFC3966] "telephone-subscriber"), and is semantically equivalent to all syntactically-valid telephone numbers below that prefix. For example, in the North American Numbering plan, the prefix 157143454 would be equivalent to all numbers ranging from 15714345400 to 15714345499.

[TBD - identify alternative ways of specifying ranges, potentially as separate element types]

Type Code: R

3.2.1.2. Domain Name Type

The domain name type conforms to the syntax of RFC1034 Section 3.5 and Section 2.1 of [RFC1123].

Type Code: D

3.2.1.3. Uniform Resource Indicator (URI) Type

The Uniform Resource Indicator (URI) type conforms to the syntax for URIs given in [RFC3986] (see Section 3).

Type Code: U

3.2.1.4. Internet Protocol (IP) Address Type

The IP Address type conforms to the ABNF syntax of either the IPv4address given in RFC3986 (Appendix A) or the IPv6reference of [RFC5954].

Type Code: I

3.2.1.5. Trunk Group Type

The trunk group type conforms to the "trunk-group-label" ABNF given in [RFC4904] (Section 5).

Type Code: G

3.2.1.6. Service Provider Identifier (SPID) Type

The SPID type consists of a four-digit number.

[TBD - introduce other elements for alternative SPID syntaxes]

Type Code: ?

3.2.2. Public Key Type

The Credential type consists of a public key [encoding TBD].

Type Code: C

3.2.3. Contact Type

The contact type follows the conventions of jCard [RFC7095].

Type Code: C

3.2.4. Access Type

The access type consists of a string, which is set to the values "Public," "Semi-restricted" or "Restricted." If either "Semi-restricted" or "Restricted" appears as the access type, the Element will need to be accompanied by a Permissions Element. [TBD - work to be done here]

Type Code: A

3.2.5. Expiry Type

The Expiry type is an absolute time conformant to the syntax of [RFC3339].

Type Code: E

3.2.6. Priority Type

The Priority type contains a number between 0 and 1, conforming to the specification of the "q" parameter of the Contact header field in [RFC3261].

Type Code: P

3.2.7. Record Identifier Type

The Record Identifier Type consists of a unique identifier for a record [format TBD].

Type Code: U

3.2.8. Signature

[Syntax TBD]

Type Code: S

3.2.9. Extension Type

This code is reserved for future use.

Type Code: X

4. Relationship to the MODERN Framework

The MODERN Framework [I-D.ietf-modern-problem-framework] enumerates a series of actors and use cases related to telephone number administration on the Internet. In terms of actors, it details interactions between Users, Communications Service Providers (CSPs), Registries, Registrars, and Government Entities. These actors acquire, manage, or retrieve telephone numbers, implementing various interfaces in support of different use cases. Registries in MODERN may be centralized or decentralized. The TeRI Operations discussed in this document pertain largely to centralized Registries: the creation and propagation of Records for decentralized Registries is outside the scope of this document. For centralized Registries, client-server operations are conducted to acquire, manage, and

retrieve telephone numbers with TeRI. Typically, Users, CSPs, and Government Entities act as TeRI Clients, and CSPs, Registries, and Registrars act as TeRI Services.

In the MODERN framework, the lifecycle of a number begins with a Registry. Registrars acquire telephone numbers from Registries, and make those numbers available for allocation. Thus, an Acquisition Operation is used by a Registrar that acquires numbers from a Registry, and this Request, if successful, will result in the creation of a Record that is returned in the Response. That Record renders the Registrar an Authority for the telephone numbers in question, but that Record will contain exclusively Administrative Data, with no Service Data.

In some cases, that Registrar will also fulfil the role of a CSP, and as a CSP, it will allocate those numbers to Users and generate any associated Records itself. Alternatively, a Registrar that does not act as a CSP may in turn act as a TeRI Service to which CSPs, and potentially Users, will send Acquisition Requests to acquire number blocks or individual numbers. Through that process, CSPs and Users can also become Authorities for telephone numbers. New Records containing Administrative Data indicating the contact information and so forth of the CSP or the User will be generated when that allocation occurs; those Records will be stored at the Registrar. The Registrar may also house a "glue" Record of Service Data that indicates the servicing CSP for the telephone number, and in particular the Retrieval interface of that CSP where Records with further Service Data can be found.

The Authorities who create and propagate Records of Service Data are typically CSPs and Users. Most commonly, CSPs will store these Service Data Records, and make them accessible through a Retrieval interface. CSPs may also propagate these Records to various external directories; the signature of the CSP and expiry data in the Record will prove its integrity and freshness to any relying party. It is envisioned that multiple Authorities may create Records for different services that are associated with a given telephone number.

Finally, CSPs and Users may query a Retrieval interface at a CSP to acquire Records containing Service Data that will enable them to route communications. The Retrieval interface will enable Clients to ask for Records associated with particular services, though Retrieval can present Clients with a number of service options. Entities may also query the Retrieval Interface of Registrars to acquire Administrative Data about a telephone number, though it is likely that authorization policies will restrict access to that data. Government Entities may have legal relationships with Registrars that

grant them authorization privileges with regard to Administrative Data.

5. TeRI Client-Server Operations

In TeRI, Clients use Operations to acquire, manage, or retrieve Records, which are typically stored at Services. Every Operation consists of a Request and a Response. Requests may pass directly from a Client to a Service, or they may pass through one or more Request Intermediaries; Request Intermediaries can modify Requests and Responses in transit. A Response will contain a Response Code indicating the status of the requested Operation. Both Requests and Responses can, in certain Operations, carry Records. TeRI does not specify any specific data format or underlying protocol to instantiate Requests, Responses, or Records: TeRI is an abstract architecture that must be implemented with concrete bindings and encodings (see Section 5.6).

The TeRI information model (see Section 3) specifies the baseline contents of Records, though Records are designed to be extended by future specifications for particular use cases or environments. Records provide information related to telephone numbers; a Record may apply to one telephone number, a block of numbers, or several discrete blocks of numbers. There may be multiple Records stored at a Service which cover a single telephone number: this may include multiple Records that apply only to that one telephone number, which probably have been provisioned by different Authorities, as well as Records applying to a telephone number range which contains that one telephone number. Authorities sign Records, and Clients typically have a trust relationship with those Authorities.

The three TeRI Operations are as follows:

The Acquisition Operation enables a Client to request the allocation of unallocated telephone numbers that are held by a Service on behalf of an Authority. A Service makes an authorization decision before allocating the telephone number(s) in accordance with the policy of the Authority. One or more new Records may be created as a result of a successful Acquisition Operation, and the Service will pass any such Record(s) to the acquiring Client as well as retaining them locally at the Service. As a result of a successful Acquisition Operation, the administrative entity operating the Client will typically become a new Authority for the allocated telephone numbers.

The Management Operation enables a Client to push new values for a Record to a Service. In the baseline Operation described in this document, the Client pushes the entire value of the Record to the

Service. The Service then makes an authorization decision to determine whether or not the Client is permitted to upload the Record in question. The policy behind those authorization decisions is outside the scope of this document, though at a high-level, the Client must be an Authority for a telephone number in order to publish and modify Records associated with that number. However, outside of hierarchical Authorities, Clients will not be able to modify or delete Records related to that number that have been provisioned by other Authorities.

The Retrieval Operation enables a Client to request one or more Records that are stored at a Service. Some Records may contain public information, and some may contain information that requires an authorization decision to be made before it is shared with a Client. Note that Services may have trust relationships with Request Intermediaries, and that the Response may depend on that trust relationship rather than on the Service's trust relationship with the Client. Although a Client acquires Records from a Service, a Client need not have a trust relationship with it - typically, the Client trusts the Record because it trusts the Authority which signed the Record rather than the Service that holds or delivers the Record.

All entities that act as TeRI Services will offer at least the Management and Retrieval interfaces, and some will also offer the Acquisition interface. All entities that act as TeRI Clients will implement at least the Retrieval Operation; others may implement the client side of one or both of the Management and Acquisition Interfaces.

5.1. Elements Common to All Operations

All Operations in the TeRI model consist of Requests and Responses. A Request from a TeRI Client to a Service may attempt to create, read, update, or delete TeRI Records. Requests may use Restrictions to focus only on particular parts of a TeRI Record. A Response gives the result of the Operation back to the Client, which may indicate success of failure.

5.1.1. Requests

All TeRI Requests have a Source, a Subject, and optionally a set of Restrictions which further specify the nature of the Request. Some Requests will contain the Identifier of the Record they concern; others will query for all Records matching a given Subject.

5.1.1.1. Source

The Source is a required element in all Requests. In this specification, two categories of Sources are defined: Request Source and Request Intermediary. At least one of these Sources must be present in a Retrieval Request, and multiple Sources are permitted. Responses do not contain a Source.

Future specifications may extend the set of Source types.

5.1.1.1.1. Request Source

Every Request generated by a Client has a Request Source, which identifies the originator of the Request. This represents the logical identity of the user or service provider who first sent the Request, rather than the identity of any Intermediate entity. This field is provided in the Source to authenticate the poser of the Request, so that the Service can make any necessary authorization decisions as it formulates a Response.

In some service deployments, an Intermediary may wish to mask the Request's Source from a Service. The removal of the Request's Source by an Intermediary is permitted by TeRI, but any Intermediary that removes the Request Source must provide a Request Intermediary for the Source element.

A Request Source element has a Type, which indicates how the logical identity of the originator of the Request has been represented. The Type field of the Request Source is extensible. Initial values include a domain name, a URI and a telephone number.

The Type element of the Request Source is followed by a Value, which contains the identity. The format of the identity is determined by the Type.

5.1.1.1.2. Request Intermediary

Optionally, Requests may contain one or more Request Intermediary elements in the Source. A Request Intermediary resides between the originator of the Request (the Client) and the Service, where it may aggregate queries, proxy them, transcode them, or provide any related relay function to assist the delivery of Requests to the Service.

The Request Intermediary element, like the Request Source, contains the logical identity of the service that relayed the Request. This field is provided in the Source for those deployments in which the Service makes an authorization decision based on the identity of the Intermediary rather than a Request Source.

A Request Intermediary element has a Type, which indicates how the logical identity of the Intermediary has been represented. The Type element of the Request Intermediary is extensible. Initial values include a domain name, an X.509 certificate subject, or a URI.

The Type of the Request Intermediary element is followed by a Value, which contains the identity. The format of the identity is determined by the Type.

5.1.1.2. Subject

All Requests have a Subject. The Subject identifies the resource that the Request concerns. Responses only contain a Subject if the Subject of the Response differs from that of the original Request, which may occur when (for example) the Subject contains a broad range, and the Service replies with a more narrow Subject. Future specifications, including Profiles, may define alternative Subject elements.

5.1.1.2.1. Request Restrictions

TeRI Request Restrictions consist of a Name with an optional Type and an Optional Value. Most Restrictions are specific to the Operation.

5.1.2. Responses

All TeRI Responses will have a Responde Code, and may contain one or more Records.

5.1.2.1. Response Code

All Responses contain a Response Code.

Response Codes defined by this document include: Success, Subject Does Not Exist, Subject Conflict, No Suitable Records Exist for Subject, Subject Syntax Error, No Suitable Records Exist for Restriction, Unauthorized Source, Route Source Topology Unavailable.

[TBD]

5.2. The Acquisition Operation

An Acquisition Request has a Source and a Subject, and may have one or more Restrictions. An Acquisition Response has a Response Code, and will contain one Record if it is successful.

The Subject of an Acquisition Request always specifies a Telephone Number Type or a Telephone Number Range Type. If the Subject

contains a particular telephone number, then the Acquisition Request is a Request to acquire that particular telephone number. If it is a range, the Acquisition Request should be considered to be for the entire range, but future Restrictions defined for this the Request might limit the scope of the resources requested. The Service will determine whether or not the Client is authorized to acquire the resources in question based on the Source of the Acquisition Request.

The Response to an Acquisition Request will contain a Success Response Code if the resource can be allocated. The Subject of a Success Response will always contain the Telephone Number Type or Telephone Number Range that has been allocated. A successful Acquisition Response must contain a Record with a Identifier Element; that Record may also contain an Element containing tokens or other material that the Client might use to acquire credentials from a Credential Authority (see [I-D.ietf-modern-problem-framework]). By default, this Record will contain only Administrative Elements, without Service Elements. If a requested telephone number (or range) is already allocated, or a telephone number in the specified range is not available, then a Subject Conflict Response Code is returned.

5.3. The Management Operation

A Management Request comprises a Source, a Subject, and one or more Records; it also may contain one or more Restrictions. A Management Response contains a Response Code, and optionally may contain a Record.

The Subject of a Management Request always specifies a Telephone Number Type or a Telephone Number Range Type. In almost all circumstances, however, the Service will locate that Record(s) that a Management Request modifies through the Identifier Restriction on each Record in the Management Request.

A Management Request contains at least one Record; it may contain multiple Records. Each Record in the Management Request must contain a Record Identifier Element which designates the Record that the Client is requesting that the Service provision as or replace with the Record included in the Management Request. The Service will authorize whether or not the Client is authorized to modify the Record in question via the Source of the Management Request.

The Management Operation not only provisions Records at a Service, but also provisions at the Service any information needed by the Service to make authorization and policy decisions when responding to Retrieval Requests. This information is tied to the Access Element of the Record.

5.3.1. Service-to-Service Record Distribution

TeRI Records contain the signature of the Authority who generated them, and as such, a relying party trusts a Record based on that signature rather than based on the Service from which a Record was retrieved. This permits architectures that allow a Records to be duplicated across a distributed Service. Distribution protocols are left to future specifications.

5.4. The Retrieval Operation

Every Retrieval Request comprises a Source and a Subject, and may have one or more Restrictions. A Retrieval Response has a Response Code, optionally one or more Records, and optionally a Subject, if the Subject differs from that of the Request.

Retrieval Requests optionally contain Restrictions; a Request with no specified Restrictions requests that the Service return any Records associated with the Subject. In a Request, the presence of one or more Restrictions limits the scope of the Request to Records about the Subject containing those Elements, or the Restrictions otherwise qualify the Request. Typically a Restriction will specify a Service or Service Type that the Client seeks Records for.

Successful Retrieval Responses always contain one or more Records; unsuccessful Responses never contain Records.

5.5. Common Restrictions

Restrictions are broadly structured around Elements, typically the Service, Contact, and Identifier Elements. A TeRI Request may contain a Restriction based on any Element, be it a baseline Element or a Service or Administrative Element, including Elements that are defined in future specifications. Semantically, a Restriction may target Records that contain a particular Element, or only Elements with a particular subtype or even value. Multiple Restrictions may appear in a Request, and Restrictions are always additive, which is to say that Restriction always narrow then target of a Request.

Restrictions may either name a target Element, or both an Element and a value. For example, a Management Request replacing an existing Record must name as its target with a Restriction both the Identifier Element and the value, which is the identifier for the Record. A Retrieval Request for a particular Subject might restrict itself to Service elements, or even Service elements that have a particular subtype, such as a URI.

5.5.1. Route Source

Optionally, Retrieval Requests may contain a Route Source which functions in much the same way as a Restriction. A Route Source identifies a reference point in the network from which any Service Elements in the response should be calculated. It therefore always designates a network element, though depending on the circumstances, it may be an endpoint, a gateway, a border device, or any other agent that makes forwarding decisions for telephone calls and related services. A Route Source is a subelement of the Source element.

A Route Source element has a Type, which indicates how the network element has been represented. The Type field of the Request Source is extensible. Initial values include a domain name, an IP address or a trunk group.

The Type of the Route Source element is followed by a Value, which designates the network element. The format of the identity is determined by the Type.

5.6. Implementing Operations

This framework specifies an abstract Request/Response protocol that enables a Client to send Requests to a Service about telephone numbers or related telephone services. Requests may pass through one or more Intermediaries on their way from a Client to a Service; for example, through aggregators or service bureaus. A Client establishes the Subject of a Request, and optionally includes one or more Restrictions to focus the scope of the Request. When a Service receives a Request, it performs any necessary authorization and policy decisions based on the Source. If policy permits, the Service generates a Response, which will consist of a Response Code and one or more Records associated with the Subject. The Service then sends the Response through the same path that the Request followed; transactional identifiers set by the Client and Service correlate the Request to the Response and assist any intermediary routing.

5.6.1. Transport Independence

The information model provided for Requests and Responses in this framework is independent of any underlying transport or encoding. Future specifications will define Bindings that specify particular transports and Encodings for Requests and Responses. In some deployment environments, for example, a binary encoding and lightweight transport might be more appropriate than the use of a web protocol. This specification provides a template of requirements that must be addressed by any encoding scheme.

It is a design goal of this work that the semantics of Requests and Responses survive interworking through translations from one encoding to another; for example, when an Intermediary receives a binary Request from a Client, it should be able to transcode it to an XML format to send to a Service without discarding any of the original semantics.

5.6.2. Bindings

A TeRI Binding is an underlying protocol that carries Requests and Responses. Future specifications may define Bindings in accordance with the procedures in the IANA Considerations sections of this document.

By underlying protocol, this specification means both transport-layer protocols as well as any application-layer protocols that the Binding requires. Thus an example Binding might specify a combination of TCP, TLS, HTTP and SOAP as the underlying transport for TeRI. Alternatively, it might only specify a very lightweight underlying protocol like UDP. A Binding may be specific to a particular Encoding, or it may be independent of any Encoding.

Bindings must specify whether they are continuous, transactional or non-transactional. A continuous Binding creates a persistent connection between two TeRI entities over which many, potentially unrelated, Requests and Responses might flow. Many Bindings defined for use between an Intermediary and a Service will have this property, as Intermediaries may aggregate on behalf of many Clients, and opening a separate transport-layer connection for each new Request would be inefficient. A transactional Binding creates a temporary connection between two TeRI entities for the purpose of fulfilling a single Request; any Responses to the Request will use the same connection to return to the sender of the Request. Finally, a non-transactional Binding does not rely on any sort of connection semantics: the senders of Requests and Responses will always initiate a new instance of the Binding to send a message.

This document makes no provision for discovering the Bindings supported by a TeRI Client, Intermediary or Service. Intermediaries may transcode between Bindings if necessary when acting to connect a Client and a Service, especially if the Client and Service support no Bindings in common.

A Binding specification must enumerate all categories of metadata required to establish a connection using a Binding. For some Bindings, this might comprise solely an IP address and a port; for other Bindings, this might instead require higher-layer application identifiers like a URI. This metadata includes any identifiers

necessary for correlating Requests to Responses in a continuous or non-transactional Binding; any Encoding making use of these Bindings must specify how it carries those elements.

Bindings must also describe the security services they make available. Bindings must have a means of providing mutual authentication, integrity and confidentiality between Clients, Intermediaries and Services. If a Binding supports TLS, for example, these features can be provided by using TLS in an appropriate deployment environment.

5.6.3. Encodings

A TeRI Encoding specifies how the Request and Response are constructed syntactically. An Encoding may be specific to a particular Binding, or it may be specified independently of any Binding.

An Encoding may define an object format; for example, an XML or JSON object, described with any appropriate schemas, or an ABNF description. An Encoding might alternatively specify a mapping of the semantic elements of Requests and Responses on to the existing fields of headers of a protocol, especially when that protocol has been defined as an underlying protocol Binding. Encodings must also define whether or not they provide a bundling feature that allows multiple Requests to be carried within particular objects or mappings.

Every Encoding must specify how each semantic Element Type of a Request and Response will be represented. For all baseline TeRI Restrictions and Element Types, the Encoding specifies whether values will be text or binary, how they will be encoded. Many baseline Element Types (such as telephone numbers) can appear in different places in a TeRI message; Encodings need only specify these common element types once. Due to the extensibility of TeRI, however, future specifications might define Element Types that an Encoding does not address. Profiles using those extensions and Encodings must explain their interaction.

Encodings must also describe the security services they make available. In particular, encodings must describe a means of providing authentication of the Sources and Authorities of Requests and Responses respectively, as well as an integrity check over critical elements including the Subject of Requests and the Record of Responses.

[TBD - we may define more about the computation of this signature, including canonicalization of elements, in this framework, and make it a requirement for encodings to support this mechanism]

5.6.4. Profiles and Extension Elements

For particular deployment environments, only one Binding, Encoding and set of Restrictions or other extended elements may be meaningful. Future specifications may therefore define TeRI Profiles, which describe a particular deployment environment and the Binding, Encoding and set of Elements and Restrictions it requires.

Profiles may encompass extensions to baseline TeRI, and any new Elements or Restrictions necessary may be defined within the Profile. It is not necessary for a TeRI Service to understand extension Elements that appear in Records or as Restrictions in a Query: if a Service receives a Query with a Restriction, it can search Records with the target Subject for Elements matching the Restriction and return only those that apply. As such there is no formal capability negotiation for extensions in the TeRI model: a Record may contain Elements beyond baseline TeRI that a particular Client does not understand and must ignore; similarly, a Service may receive a Query with a Restriction that applies to no Records collected at the Service, in which case the Service returns a "No Suitable Records Exist for Restriction" Response Code.

6. Security Considerations

The framework of this document differs from previous efforts to manage telephone numbers on the Internet largely by offering a much richer set of security services. In particular, it requires that three entities be capable of authenticating themselves to one another at the layer of a binding: Clients, Intermediaries and Services. It furthermore requires object security at the encoding layer so that Sources and Authorities can sign data in order to authenticate Requests and Responses that may pass through Intermediaries, and moreover so that Authorities can prove to Clients that their Records are authoritative even when the Authority does not operate the Service. The requirements that bindings and encodings must satisfy to meet these security needs are specified in Section 5.6.1.

[TBD - more]

7. IANA Considerations

This specification defines several registries: A registry of Elements, a registry of Element Types, and a registry of Response Codes.

This document creates a registry of Elements for use with this framework. This registry is extensible, with an IANA Registration policy of Specification Required. Any new Element registered must supply the name of the Element, the name of the parent Element in the information model, and a code point. [TBD]

This specification pre-provisions the Element Types registry with the entries given in Section 6. These elements are indexed by their Type Code. This registry is extensible, with an IANA Registration policy of Specification Required. Any new Element Type registered must supply the name of the Element Type, the name of the parent element in the information model, and a Type Code.

This document furthermore creates a registry of Response Codes. This registry is pre-provisioned with the values given in Section 5.5. [TBD]

8. Acknowledgements

The authors would like to thank Chris Wendt, Paul Kyzviat and Dale Worley for their input into this specification.

9. Informative References

[I-D.ietf-modern-problem-framework] Peterson, J. and T. McGarry, "Modern Problem Statement, Use Cases, and Framework", draft-ietf-modern-problem-framework-03 (work in progress), July 2017.

[RFC1123] Braden, R., Ed., "Requirements for Internet Hosts - Application and Support", STD 3, RFC 1123, DOI 10.17487/RFC1123, October 1989, <https://www.rfc-editor.org/info/rfc1123>.

[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>.

[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>.

[RFC3324] Watson, M., "Short Term Requirements for Network Asserted Identity", RFC 3324, DOI 10.17487/RFC3324, November 2002, <https://www.rfc-editor.org/info/rfc3324>.

[RFC3325] Jennings, C., Peterson, J., and M. Watson, "Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks", RFC 3325, DOI 10.17487/RFC3325, November 2002, <https://www.rfc-editor.org/info/rfc3325>.

[RFC3339] Klyne, G. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, DOI 10.17487/RFC3339, July 2002, <https://www.rfc-editor.org/info/rfc3339>.

[RFC3966] Schulzrinne, H., "The tel URI for Telephone Numbers", RFC 3966, DOI 10.17487/RFC3966, December 2004, <https://www.rfc-editor.org/info/rfc3966>.

[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>.

[RFC4474] Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 4474, DOI 10.17487/RFC4474, August 2006, <https://www.rfc-editor.org/info/rfc4474>.

[RFC4904] Gurbani, V. and C. Jennings, "Representing Trunk Groups in tel/sip Uniform Resource Identifiers (URIs)", RFC 4904, DOI 10.17487/RFC4904, June 2007, <https://www.rfc-editor.org/info/rfc4904>.

[RFC4916] Elwell, J., "Connected Identity in the Session Initiation Protocol (SIP)", RFC 4916, DOI 10.17487/RFC4916, June 2007, <https://www.rfc-editor.org/info/rfc4916>.

[RFC5039] Rosenberg, J. and C. Jennings, "The Session Initiation Protocol (SIP) and Spam", RFC 5039, DOI 10.17487/RFC5039, January 2008, <https://www.rfc-editor.org/info/rfc5039>.

[RFC5067] Lind, S. and P. Pfautz, "Infrastructure ENUM Requirements", RFC 5067, DOI 10.17487/RFC5067, November 2007, <https://www.rfc-editor.org/info/rfc5067>.

[RFC5727] Peterson, J., Jennings, C., and R. Sparks, "Change Process for the Session Initiation Protocol (SIP) and the Real-time Applications and Infrastructure Area", BCP 67, RFC 5727, DOI 10.17487/RFC5727, March 2010, <https://www.rfc-editor.org/info/rfc5727>.

[RFC5954] Gurbani, V., Ed., Carpenter, B., Ed., and B. Tate, Ed., "Essential Correction for IPv6 ABNF and URI Comparison in RFC 3261", RFC 5954, DOI 10.17487/RFC5954, August 2010, <https://www.rfc-editor.org/info/rfc5954>.

[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>.

[RFC6406] Malas, D., Ed. and J. Livingood, Ed., "Session PEERing for Multimedia INTerconnect (SPEERMINT) Architecture", RFC 6406, DOI 10.17487/RFC6406, November 2011, <https://www.rfc-editor.org/info/rfc6406>.

[RFC6461] Channabasappa, S., Ed., "Data for Reachability of Inter-/Intra-NetworK SIP (DRINKS) Use Cases and Protocol Requirements", RFC 6461, DOI 10.17487/RFC6461, January 2012, <https://www.rfc-editor.org/info/rfc6461>.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <https://www.rfc-editor.org/info/rfc6698>.

[RFC6950] Peterson, J., Kolkman, O., Tschofenig, H., and B. Aboba, "Architectural Considerations on Application Features in the DNS", RFC 6950, DOI 10.17487/RFC6950, October 2013, <https://www.rfc-editor.org/info/rfc6950>.

[RFC7095] Kewisch, P., "jCard: The JSON Format for vCard", RFC 7095, DOI 10.17487/RFC7095, January 2014, <https://www.rfc-editor.org/info/rfc7095>.

[RFC7340] Peterson, J., Schulzrinne, H., and H. Tschofenig, "Secure Telephone Identity Problem Statement and Requirements", RFC 7340, DOI 10.17487/RFC7340, September 2014, <https://www.rfc-editor.org/info/rfc7340>.

Author’s Address

Jon Peterson
Neustar, Inc.

Email: jon.peterson@team.neustar