IANA Registration of Enumservices: Guide, Template and IANA Considerations
draft-ietf-enum-enumservices-guide-13

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on May 7, 2009.

Abstract

This document specifies a revision of the IANA Registration Guidelines for Enumservices, describes corresponding registration procedures, and provides a guideline for creating Enumservice Specifications.
Table of Contents

1. Introduction ................................................. 4
2. Terminology ................................................... 4
3. Registration Requirements ................................. 5
   3.1. Functionality Requirements ......................... 5
   3.2. Naming Requirements ................................. 5
   3.3. Security Requirements ............................... 6
   3.4. Publication Requirements ............................. 7
4. Enumservice Creation Cookbook ............................ 7
   4.1. General Enumservice Considerations .................. 7
   4.2. Classification, Type and Subtype ................... 9
       4.2.1. General Type / Subtype Considerations ............. 9
       4.2.2. Protocol-Based Enumservices Class ............... 10
       4.2.3. Application-Based Enumservice Classes .......... 10
       4.2.4. Data Type-Based Enumservice Class ............... 12
       4.2.5. Other Enumservice ............................ 13
5. Required Sections and Information ...................... 13
   5.1. Introduction (MANDATORY) ............................ 13
   5.2. IANA Registration (MANDATORY) ..................... 13
   5.3. Examples (MANDATORY) ................................ 17
   5.4. Implementation Recommendations / Notes (OPTIONAL) .. 17
   5.5. Security Considerations (MANDATORY) ............... 17
   5.6. IANA Considerations (MANDATORY) ................... 18
   5.7. DNS Considerations (MANDATORY) .................... 18
   5.8. Other Sections (OPTIONAL) .......................... 19
6. The Process of Registering New Enumservices ........... 19
   6.1. Step 1: Read this Document in Detail ............... 20
   6.2. Step 2: Write and Submit Registration Document ..... 20
   6.3. Step 3: Request Comments from the IETF Community 21
       6.3.1. Outcome 1: No Changes Needed ................. 21
       6.3.2. Outcome 2: Changes, but no further Comments 21
               Requested .................................... 21
       6.3.3. Outcome 3: Changes and further Comments Requested 21
   6.4. Step 4: Submit Registration Document ............... 22
   6.5. Step 5: Expert Review ............................... 23
       6.5.1. Outcome 1: Experts Approve the Registration 23
               Document .................................. 23
       6.5.2. Outcome 2: Changes Required ................. 23
       6.5.3. Outcome 3: Experts Reject the Registration Document 23
   6.7. Step 7: Adding Enumservice to IANA Registry ........ 23
1. Introduction

E.164 Number Mapping (ENUM) [I-D.ietf-enum-3761bis] provides an identifier mapping mechanism to map E.164 numbers [ITU.E164.2005] to Uniform Resource Identifiers (URIs) [RFC3986]. One of the primary concepts of ENUM is the definition of "Enumservices", which allows for providing different URIs for different applications of said mapping mechanism.

The IETF’s ENUM Working Group has encountered an unnecessary amount of variation in the format of Enumservice Specifications. The ENUM Working Group’s view of what particular fields and information are required and/or recommended has also evolved, and capturing these best current practices is helpful in both the creation of new Enumservice Specifications, as well as the revision or refinement of existing Enumservice Specifications.

This document specifies a revision of the IANA Registry for Enumservices, which was originally described in [RFC3761]. This document obsoletes Section 3 of RFC 3761.

The new registration processes have been specifically designed to be decoupled from the existence of the ENUM working group. Compared to RFC 3761, the main changes are:

- For an Enumservice to be inserted to the IANA Registry, 'Expert Review' and 'Specification Required' according to "Guidelines for Writing an IANA Considerations Section in RFCs" [RFC5226] are now sufficient.

- The IANA Registration Template contains new fields, i.e. "Enumservice Class" and "Enumservice Specifications(s)".

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

For the purpose of this document:

- ‘Registration Document’ refers to a draft specification that defines an Enumservice and proposes its registration following the procedures outlined herein.
3. Registration Requirements

As specified in the ABNF found in [I-D.ietf-enum-3761bis], an Enumservice is made up of Types and Subtypes. For any given Type, the allowable Subtypes (if any) must be defined in the Enumservice Specification. There is currently no concept of a registered Subtype outside the scope of a given Type.

While the combination of each Type and all of its Subtypes constitutes the allowed values for the ‘Enumservice’ field, it is not sufficient to simply list the allowed values of those fields. To allow interoperability, a complete Enumservice Specification MUST document the semantics of the Type and Subtype values to be registered, and MUST contain all sections listed in Section 5 of this document.

Furthermore, in order for an Enumservice to be registered, the entire Registration Document requires approval by the experts according to the “Expert Review” process defined in "Guidelines for Writing an IANA Considerations Section in RFCs" [RFC5226].

All Enumservice Specifications are expected to conform also to various requirements laid out in the following sections.

3.1. Functionality Requirements

A registered Enumservice must be able to function as a selection mechanism when choosing one NAPTR resource record [RFC3403] from another. That means that the Enumservice Specification MUST specify what is expected when using that very NAPTR record, and the URI which is the outcome of the use of it.

Specifically, a registered Enumservice MUST specify the URI Scheme(s) that may be used for the Enumservice, and, when needed, other information that will have to be transferred into the URI resolution process itself.

3.2. Naming Requirements

An Enumservice MUST be unique in order to be useful as a selection criterion:
The Type MUST be unique.

The Subtype (being dependent on the Type) MUST be unique within a given Type.

Types and Subtypes MUST conform to the ABNF specified in [I-D.ietf-enum-3761bis].

The ABNF specified in [I-D.ietf-enum-3761bis] allows the "-" (dash) character for Types and Subtypes. To avoid confusion with possible future prefixes, a "-" MUST NOT be used as the first nor as the second character of a Type nor a Subtype. Furthermore, a "-" MUST NOT be used as the last character of a Type nor a Subtype. In addition, Types and Subtypes are case insensitive and MUST be specified in small letters.

To avoid confusion with Enumservice fields using an obsolete syntax, any identifying tag of any Enumservice MUST NOT be set to nor start with "E2U".

The Subtype for one Type MAY have the same identifier as a Subtype for a different registered Type but it is not sufficient to simply reference another Type’s Subtype. The functionality of each Subtype MUST be specified in the context of the Type being registered.

Section 4 contains further naming recommendations.

3.3. Security Requirements

An analysis of security issues is REQUIRED for all registered Enumservices. (This is in accordance with the basic requirements for all IETF protocols.)

All descriptions of security issues MUST be as accurate and extensive as feasible. In particular, a statement that there are "no security issues associated with this Enumservice" must not be confused with "the security issues associated with this Enumservice have not been assessed".

There is no requirement that an Enumservice must be completely free of security risks. Nevertheless, all known security risks MUST be identified in an Enumservice Specification.

The security considerations section of Enumservice Specifications is subject to continuing evaluation and modification, in accordance with Section 11.1.5.

Some of the issues to be looked at in a security analysis of an
Enumservice are:

1. Complex Enumservices may include provisions for directives that institute actions on a user's resources. In many cases provision can be made to specify arbitrary actions in an unrestricted fashion which may then have devastating results (especially if there is a risk for a new ENUM look-up, and because of that an infinite loop in the overall resolution process of the E.164 number).

2. Complex Enumservices may include provisions for directives that institute actions which, while not directly harmful, may result in disclosure of information that either facilitates a subsequent attack or else violates the users' privacy in some way.

3. An Enumservice might be targeted for applications that require some sort of security assurance but do not provide the necessary security mechanisms themselves. For example, an Enumservice could be defined for storage of confidential security services information such as alarm systems or message service passcodes, which in turn require an external confidentiality service.

3.4. Publication Requirements

Enumservices Specifications MUST be published according to the requirements for 'Specification Required' set in "Guidelines for Writing an IANA Considerations Section in RFCs" [RFC5226]. RFCs fulfill these requirements. Therefore, it is strongly RECOMMENDED Enumservice Specifications be published as RFCs.

In case the Enumservice Specification is not published as an RFC, sufficient information that allows to uniquely identify the Enumservice Specification MUST be provided.

4. Enumservice Creation Cookbook

4.1. General Enumservice Considerations

ENUM is an extremely flexible identifier mapping mechanism, using E.164 (phone) numbers as input identifiers, and returning URIs as output identifiers. Because of this flexibility, almost every use case for ENUM could be implemented in several ways.

Section 2 of "Guidelines for Writing an IANA Considerations Section in RFCs" [RFC5226] provides motivation why management of a name space might be necessary. Even though the namespace for Enumservices is rather large (up to 32 alphanumeric characters), there are reasons to
manage this in accordance with Section 2 of [RFC5226]. The following is a list of motivations applying to Enumservices:

- Prevent hoarding or wasting of values: Enumservice Types are not an opaque identifier to prevent collisions in the namespace, but rather identify the use of a certain technology in the context of ENUM. Service Types might also be displayed to end users in implementations, so meaningful Type strings having a clear relation to the protocols and applications used are strongly RECOMMENDED. Therefore, preventing hoarding, wasting, or "hijacking" of Enumservice Type names is important.

- Sanity check to ensure sensible or necessary requests: This applies to Enumservices, since especially various Enumservices for the same purpose would reduce the chance of successful interoperability, and unnecessarily increase the confusion among implementers.

- Delegation of namespace portions: Theoretically, the Type and/or Subtype structure of Enumservices would allow for delegations of Type values, and self-supporting management of Subtype values by a delegate within the Type value. Such delegates could for example be other standardization bodies. However, this would require clear policies regarding publication and use of such Subtypes. Delegation of Enumservice namespace portions is therefore currently not supported.

- Interoperability: Since the benefit of an Enumservice rises with the number of supporting clients, the registration and use of several services for a similar or identical purpose clearly reduces interoperability. Operational circumstances suggest to keep the space occupied by all services published in the NAPTR RRSet at any owner in the e164.arpa domain bounded. Registration of nearly identical services and subsequent competing or parallel use could easily increase the DNS operational complexity.

Generally, before commencing work on a new Enumservice registration, the following should be considered:

- Is there an existing Enumservice that could fulfill the desired functionality without overloading it? Check the IANA Enumservice Registry at <http://www.iana.org/assignments/enum-services>.

- Is there work in progress, or previous work, on a similar Enumservice? Check the <enum@ietf.org> mailing list archives at <http://www.ietf.org/mail-archive/web/enum/index.html>, and search the Internet-Drafts Archive at <http://tools.ietf.org/>. As some Internet-Drafts may have expired and no longer be available in the
Section 4.2 provides three general categories for Enumservice classification. In some cases, there might be several options for designing an Enumservice. For example, a mapping service using HTTP could be considered a "protocol Type" Enumservice (using HTTP as the protocol), while it could also be viewed as an "application Type" Enumservice, with the application being access to mapping services. In such a case where several options are available, defining use cases before commencing work on the Enumservice itself might be useful before making a decision on which aspect of the Enumservice is more important.

4.2. Classification, Type and Subtype

Because of its flexibility, Enumservices can be and are used in a lot of different ways. This section contains a classification of Enumservices, and provides guidance for choosing suitable Type and Subtype strings for each individual Enumservice Class.

The Classification of each Enumservice MUST be listed in the Enumservice Specification (see Section 5.2). If the Enumservice cannot be assigned to one of the classes outlined below, the Enumservice Specification MUST contain a section on the difficulties encountered while trying to classify the service to help the experts in their decision.

4.2.1. General Type / Subtype Considerations

To avoid confusion, the name of a URI Scheme MUST NOT be used as a Type name for an Enumservice which is not specifically about the respective protocol or URI Scheme. For example, the Type name 'imap' would be inadequate for use in an Enumservice about "Internet mapping" services, because it corresponds to an existing URI Scheme or protocol for something different.

If Subtypes are defined, the minimum number SHOULD be two (including the empty subtype, if defined). The choice of just one possible Subtype for a given Type does not add any information when selecting a ENUM record, and hence can be left out completely. However, potential future expansion of a Type towards several Subtypes may justify the use of Subtypes, even in the case just one is currently defined, as noted in Section 9.

It is perfectly legal under a certain Type to mix the Enumservice
without a Subtype ("empty Subtype") with Enumservices containing a Subtype. In that case, however, the Enumservice with an empty Subtype SHOULD be specified to reflect the base service, while the other Enumservices SHOULD be specified to reflect variants.

4.2.2. Protocol-Based Enumservices Class

Such an Enumservice indicates that an interaction using the named protocol will result for use of this NAPTR. The expected behavior of a system using this Enumservice MUST be clear from the protocol.

A good indication that an Enumservice belongs to this Class is the fact that a client does not need to understand the actual application to make use of an instance of this Enumservice.

Examples of such Enumservices include XMPP [RFC4979] and SIP [RFC3764].

4.2.2.1. Protocol-Based Enumservice "Type" Strings

A protocol-based Enumservice SHOULD use the lowercase name of the protocol as its Type name.

4.2.2.2. Protocol-Based Enumservice "Subtype" Strings

Where there is a single URI Scheme associated with this protocol, a Subtype SHOULD NOT be specified for the Enumservice.

Where there are a number of different URI Schemes associated with this protocol, the Enumservice Specification MAY use the empty Subtype for all URI Schemes that it specifies as mandatory to implement. For each URI Scheme that is not mandatory to implement a distinct Subtype string MUST be used.

If Subtypes are defined, it is RECOMMENDED to use the URI Scheme name as the Subtype string.

4.2.3. Application-Based Enumservice Classes

Application-based Enumservices are used when the kind of service intended is not fully defined by a protocol specification. There are three cases here:

- Common Application Enumservice:

  The application reflects a kind of interaction that can be realized by different protocols, but where the intent of the publisher is the same. From a user’s perspective, there is a
common kind of interaction - how that interaction is implemented is not important. The Enumservice Specification MUST describe the interaction and expected behavior in enough detail that an implementation can decide if this activity is one in which it can engage. However, it is RECOMMENDED that the Enumservice is defined in a way that will allow others to use it at a later date. An Enumservice that defines a generalized application is preferred to one that has narrow use.

An example of this flavor of Enumservice is email. Whilst this might appear to be a "pure" protocol scheme, it is not. The URI Scheme is 'mailto', and does not identify the protocol used by the sender or the recipient to offer or retrieve emails.

Another example is SMS, where the existence of such an Enumservice indicates that the publishing entity is capable of engaging in sending or receiving a message according to the Short Messaging Service specifications. The underlying protocol used and the URI Scheme for the addressable end point can differ, but the "user visible" interaction of sending and receiving an SMS is similar.

- Subset Enumservice:

The application interaction reflects a subset of the interactions possible by use of a protocol. Use of this Enumservice indicates that some options available by use of the protocol will not be accepted or are not possible in this case. Any such Enumservice Specification MUST define the options available by use of this NAPTR in enough detail that an implementation can decide whether or not it can use this Enumservice. Examples of this kind of Enumservice are voice:tel and fax:tel. In both cases the URI holds a telephone number. However, the essential feature of these Enumservices is that the telephone number is capable of receiving a voice call or of receiving a Facsimile transmission, respectively. These form subsets of the interactions capable of using the telephone number, and so have their own Enumservices. These allow an end point to decide if it has the appropriate capability of engaging in the advertised user service (a voice call or sending a fax) rather than just being capable of making a connection to such a destination address. This is especially important where there is no underlying mechanism within the protocol to negotiate a different kind of user interaction.

- Ancillary Application Enumservice

Another variant on this is the Ancillary Application. This is one in which further processing (potentially using a number of different protocols or methods) is the intended result of using
this Enumservice. An example of this kind of application is the PSTN:tel Enumservice. This indicates that the NAPTR holds Number Portability data. It implies that the client should engage in number portability processing using the associated URI. Note that this Enumservice usually does not itself define the kind of interaction available using the associated URI. That application is negotiated with some other "out of band" means (either through prior negotiation, or explicitly through the number portability process, or through negotiation following the selection of the final destination address).

4.2.3.1. Application-Based Enumservice "Type" Strings

It is RECOMMENDED that Application-class Enumservices use the lowercase well known name of the abstract application as Type name.

4.2.3.2. Application-Based Enumservice "Subtype" Strings

It is RECOMMENDED to use the URI Scheme(s) which the application uses, as Subtype name(s). Subtype names MAY be shared between URI Schemes, if all the URI Schemes within the same Subtype are mandatory to implement.

If it is foreseen that there is only one URI Scheme ever to be used with the application, the empty Subtype string MAY be used.

4.2.4. Data Type-Based Enumservice Class

"Data Type" Enumservices typically refer to a specific data type or format, which may be addressed using one or more URI Schemes and protocols. It is RECOMMENDED to use a well known name of the data type or format as the Enumservice Type. Examples of such Enumservices include ‘vpim’ [RFC4238] and ‘vCard’ [RFC4969].

4.2.4.1. Data Type-Based Enumservice "Type" Strings

It is RECOMMENDED to use the lowercase well known name of the data or format as the Type name.

4.2.4.2. Data Type-Based Enumservice "Subtype" Strings

It is RECOMMENDED to use the URI Schemes used to access the service as Subtype name. Subtype names MAY be shared between URI Schemes, if all the URI Schemes within the same Subtype are mandatory to implement.

If there is only one URI Scheme foreseen to access the data or format, the empty Subtype string MAY be used.
4.2.5. Other Enumservice

In case an Enumservice proposal cannot be assigned to any of the classes mentioned above, the "Classification" field in the IANA Registration Template (see Section 5.2) MUST be populated with "Other". In that case, the Enumservice Specification MUST contain a section elaborating why the Enumservice does not fit into the classification structure.

5. Required Sections and Information

There are several sections that MUST appear in an Enumservice Specification. These sections are as follows, and SHOULD be in the given order.

The following terms SHOULD begin with a capital letter, whenever they refer to the IANA Registration:

- Class
- Type
- Subtype
- URI Scheme

5.1. Introduction (MANDATORY)

An introductory section MUST be included. This section will explain, in plain English, the purpose of and intended use of the proposed Enumservice registration.

The Introduction SHOULD start with a short sentence about ENUM, introduce the protocol used in the Enumservice, and discuss the Enumservice as it refers from the E.164 number to the protocol or service.

5.2. IANA Registration (MANDATORY)

This section MUST be included in an Enumservice Specification. Where a given Enumservice Type has multiple Subtypes, there MUST be a separate ‘IANA Registration’ section for each Subtype. The following lists the fields and order of an ‘IANA Registration’ section.

- Enumservice Class:

  This field contains the Class of the Enumservice as defined in Section 4.2. It’s value MUST be one of (without quotes):
* "Protocol-Based": The Enumservice belongs to the Protocol-based class as described in Section 4.2.2.

* "Application-Based, Common": The Enumservice is a "common" case of the Application-based class as described in Section 4.2.3.

* "Application-Based, Subset": The Enumservice belongs to the "subset" case of the Application-based class as described in Section 4.2.3.

* "Application-Based, Ancillary": The Enumservice is an "ancillary" case of the Application-based class, as described in Section 4.2.3.

* "Data Type-Based": The Enumservice belongs to the Data Type-Based class as described in Section 4.2.4.

* "Other": The majority of the functionality of the Enumservice does not fall into one of the classes defined.

  e.g. Protocol-Based

  o Enumservice Type:

  The Type of the Enumservice. All Types SHOULD be listed in lowercase. The choice of Type depends on the Enumservice Class. The Type string MUST be placed between double quotes. Please find further instructions in Section 4.

  e.g. "foo"

  o Enumservice Subtype:

  The Subtype of the Enumservice. All Subtypes SHOULD be listed in lowercase. The choice of Subtype depends on the Enumservice Class. The Subtype string MUST be placed between double quotes. Should the Enumservice not require a Subtype, then "N/A" MUST be used. If a given Enumservice Type has multiple Subtypes, then there MUST be a separate ‘IANA Registration’ section for each Subtype. Please find further instructions in Section 4.

  e.g. "bar"

  e.g. N/A
o URI Scheme(s):

The URI Schemes that are used with the Enumservice. The selection of URI Schemes often depends on the Enumservice Class, Type, and/or Subtype. A colon MUST NOT be placed after the URI Scheme. The URI Scheme string MUST be placed between single quotes. If there is more than one URI Scheme, then a comma MUST be used as a separator, as shown below. Please find further instructions in Section 4.

  e.g.
  'bar', 'sbar'

Note: A client cannot choose a specific ENUM record in a record set based on the URI Scheme - the selection is only based on Type and Subtype, in accordance with [RFC3402]

o Functional Specification:

The Functional Specification describes how the Enumservice is used in connection with the URI to which it resolves.

  e.g.
  This Enumservice indicates that the resource identified can be addressed by the associated URI in order to foo the bar. [...] Where the terms used are non-obvious, they should be defined in the Enumservice Specification, or a reference to an external document containing their definition should be provided.

o Security Considerations:

An internal reference to the 'Security Considerations' section of a given Enumservice Specification.

  e.g.
  See Section 10

o Intended Usage:

One of the following values (without quotes):

* "COMMON": Indicates that the Enumservice is intended for widespread use on the public Internet, and that its scope is not limited to a certain environment.
* "LIMITED USE": Indicates that the Enumservice is intended for use on a limited scope, for example in private ENUM-like application scenarios. The use case provided in the Enumservice Specification should describe such a scenario.

* "OBSOLETE": Indicates that the Enumservice has been declared obsolete (Section 11.1.5) and is not to be used in new deployments. Applications SHOULD however expect to encounter legacy instances of this Enumservice.

  e.g.
  COMMON

- Enumservice Specification(s):


  e.g.
  [RFC 9999]

  e.g.
  [RFC 7777] (Obsoleted by RFC 8888)
  [RFC 8888] (Updated by RFC 9999)
  [RFC 9999]

  e.g.

- Authors:

  The authors of the Enumservice Specification.

  e.g.
  John Doe, Jane Dale

  Note: If there is more than one author, use a comma as separator.

  Note: You MUST NOT put email addresses in the authors field of an IANA Registration.

- Further Information:

  Any other information the authors deem interesting.
5.3. Examples (MANDATORY)

This section MUST show at least one example of the Enumservice being registered, for illustrative purposes. The example(s) shall in no way limit the various forms that a given Enumservice may take, and this should be noted at the beginning of this section of the document. The example(s) MUST show the specific formatting of the intended NAPTRs (according to [RFC3403] and [I-D.ietf-enum-3761bis]), including one or more NAPTR example(s), AND a brief textual description, consisting of one or more sentences written in plain English, explaining the various parts or attributes of the record(s).

The example(s) SHOULD contain a brief description how a client supporting this Enumservice could behave, if that description was not already given in e.g. the Introduction or the Functional Specification.

The example(s) SHOULD follow any relevant IETF guidelines on the use of domain names, phone numbers, and other resource identifier examples, such as [RFC2606].

5.4. Implementation Recommendations / Notes (OPTIONAL)

If at all possible, recommendations that pertain to implementation and/or operations SHOULD be included. Such a section is helpful to someone reading an Enumservice Specification and trying to understand how best to use it to support their network or service.

5.5. Security Considerations (MANDATORY)

A section explaining any potential security threats that are unique to the given registration MUST be included. This MUST also include any information about access to Personally Identifiable Information (PII).
An Enumservice Specification SHOULD NOT include general and obvious security recommendations, such as securing servers with strong password authentication.

[RFC3552] provides guidance to write a good Security Considerations section, Section 10.2 of this document contains guidance specific to Enumservice registration.

5.6. IANA Considerations (MANDATORY)

Describe the task IANA needs to fulfill processing the Enumservice Registration Document.

e.g.
This document requests the IANA registration of the Enumservice "Foo" with Type "foo" and Subtype "bar" according to the definitions in this document, RFC XXXX [Note for RFC Editor: Please replace XXXX with the RFC number of this document before publication] and [I-D.ietf-enum-3761bis].

e.g.
This document requests an update of the IANA registration of the Enumservice Type "foo" with Subtype "bar", according to the definitions in this document, RFC XXXX [Note for RFC Editor: Please replace XXXX with the RFC number of this document before publication] and [I-D.ietf-enum-3761bis]. Therefore, in the existing IANA registration for this Enumservice, the field "Enumservice Specification(s)" is enhanced by adding a supplementary reference that points to this document.

e.g.
This document requests an update of the IANA registration of the Enumservice Type "foo" with all its Subtypes, in order to declare it obsolete. Therefore, in the existing IANA registration for this Enumservice, the field "Intended Usage" is changed to "OBSOLETE", and the field "Enumservice Specification(s)" is enhanced by adding a supplementary reference that points to this document.

5.7. DNS Considerations (MANDATORY)

In case the inclusion of protocols and URI Schemes into ENUM specifically introduces new DNS issues, those MUST be described within this section.

Such DNS issues include, but are not limited to:
5.8. Other Sections (OPTIONAL)

Other sections beyond those required above MAY be included in an Enumservice Specification. These sections may relate to the specifics of the intended use of the Enumservice registration, as well as to any associated technical, operational, administrative, or other concerns.

A use case SHOULD be included by the authors of the proposal, so that experts can better understand the problem the proposal seeks to solve (intended use of the Enumservice). The inclusion of such a use case will both accelerate the Expert Review Process, as well as make any eventual registration easier to understand and implement by other parties.

6. The Process of Registering New Enumservices

This section is an illustration of the process by which a new Enumservice Registration Document is submitted for review and comment, how such proposed Enumservices are reviewed, and how they are published.

Figure 1 shows, what authors of a Registration Document describing an Enumservice MUST carry out, before said Registration Document can be formally submitted to IANA for Expert Review. Figure 2 shows the process from Expert Review onwards.
6.1. Step 1: Read this Document in Detail

This document describes all of the necessary sections required and recommended, and makes suggestions on content.

6.2. Step 2: Write and Submit Registration Document

An Internet-Draft (or another specification as appropriate) MUST be written and made publicly available (submitted). The Registration Document MUST follow the guidelines according to Section 4 and Section 5 of this document.
6.3. Step 3: Request Comments from the IETF Community

The authors MUST send an email to <enum@ietf.org>, in which comments on the Registration Document are requested. A proper public reference (a URL is RECOMMENDED) to the Registration Document MUST be included in this email.

The authors SHOULD allow a reasonable period of time to elapse, such as two to four weeks, in order to collect any feedback. The authors then consider whether or not to take any of those comments into account, by making changes to the Registration Document and submitting a revision, or otherwise proceeding. The following outcomes are open to the authors. The choice of path is left to the authors’ judgement.

Note: Whatever that outcome is, the Experts are not bound to any decision during this phase.

6.3.1. Outcome 1: No Changes Needed

No changes to the Registration Document are made, and the authors proceed to Step 4 below.

This outcome is recommended when the feedback received does not lead to a new revision of the Registration Document.

6.3.2. Outcome 2: Changes, but no further Comments Requested

The authors update the Registration Document and is/are confident that all issues are resolved and do not require further discussion. The authors proceed to Step 4 below.

This outcome is recommended when minor objections have been raised, or minor changes have been suggested.

6.3.3. Outcome 3: Changes and further Comments Requested

The authors update and submit the Registration Document, and proceed to Step 3 above, which involves sending another email to <enum@ietf.org> to request additional comments for the updated version.

This outcome is recommended when substantial objections have been raised, or substantial changes have been suggested.
6.4. Step 4: Submit Registration Document

If the Registration Document is to be published as RFC, the normal IETF publication process applies (see [instructions2authors]), i.e. the Registration Document is submitted to the RFC Editor in the form of an Internet Draft. For Independent Submission the guidelines in Independent Submissions to the RFC Editor [RFC4846] apply.

For publications as RFC Steps 6 below does not apply.

If the Registration Document is not published as RFC, the authors submit the Registration Document to IANA for Expert Review via the http://iana.org/ website.

The Step 6 below does only apply in case the Registration Document is to be published in a specification other than RFC.

R-D: Registration Document
6.5. Step 5: Expert Review

After the Registration Document arrives at IANA, they will conduct an Expert Review according to [RFC5226]. The authors MUST be prepared for further interaction with IANA and the experts.

6.5.1. Outcome 1: Experts Approve the Registration Document

No (more) changes to the Registration Document are made. IANA will inform the authors, who then will proceed to Step 6 below.

6.5.2. Outcome 2: Changes Required

The experts might require changes before they can approve the Registration Document. The authors update and submit the Registration Document. The authors inform the experts about the available update, who then continue the Expert Review Process.

6.5.3. Outcome 3: Experts Reject the Registration Document

The expert might reject the Registration, which means the Expert Review Process is discontinued. For appeals, see Section 7.3.

6.6. Step 6: Publication of the Registration Document

This Step 5 only applies in case the Registration Document is to be published in a specification other than RFC. (In the RFC case the RFC publication process ensures that the Enumservice Specification is published.)

The authors are responsible that the Registration Document is published according to ‘Specification Required’ as defined in [RFC5226].

6.7. Step 7: Adding Enumservice to IANA Registry

In case the Registration Document is to be published as an RFC, the RFC publication process ensures that IANA will add the Enumservice to the Registry.

In case the Registration Document is to be published in a specification other than RFC, the authors MUST inform IANA, as soon as the Enumservice Specification has been published according to ‘Specification Required’ as defined in [RFC5226]. The ‘Enumservice Specification(s)’ field in the IANA Template MUST contain a unambiguous reference to the Enumservice Specification (see also...
Section 5.2). In addition, the authors MUST provide IANA with a stable URL to the Enumservice Specification, in order that IANA may obtain the information included in the Enumservice Specification. IANA will then add the Enumservice to the Registry.

7. Expert Review

7.1. Expert Selection Process

According to Section 3.2 of [RFC5226], experts are appointed by the IESG upon recommendation by the RAI Area Directors. The RAI area directors are responsible for ensuring that there is always a sufficient pool of experts available.

7.2. Review Guidelines

Generally, the Expert Review Process of an Enumservice MUST follow the guidelines documented in Section 3.3 of "Guidelines for Writing an IANA Considerations Section in RFCs" [RFC5226].

The experts MUST evaluate the criterion as set out in [RFC5226], as well as consider the following:

- Verify conformance with the ENUM specification [I-D.ietf-enum-3761bis].
- Verify that the requirements set in this document (Section 3, Section 5) are met. This includes check for completeness and whether all the aspects described in Section 3 and Section 5 are sufficiently addressed.
- If a use case is provided, the experts SHOULD verify whether the proposed Enumservice does actually match the use case. The experts SHOULD also determine whether the use case could be covered by an existing Enumservice.
- Verify that the Enumservice proposed cannot be confused with identical (or similar) other Enumservices already registered.
- If the Enumservice is classified according to Section 4.2, the experts MUST verify that the principles of the Class in question are followed.
- In case the Enumservice is not classified, the experts MUST verify whether a convincing reason for the deviation is provided in the Registration Document.
o Investigate whether the proposed Enumservice has any negative side effects on existing clients and infrastructure, particularly the DNS.

o If the output of processing an Enumservice may be used for input to more ENUM processing (especially services returning 'tel' URIs), the experts SHOULD verify that the authors have adequately addressed the issue of potential query loops.

In case of conflicts between [RFC5226] and the guidelines in this section, the former remains authoritative.

7.3. Appeals

Appeals of Expert Review decisions follow the process described in section 7 of [RFC5226] and section 6.5 of [RFC2026].

8. Revision of Pre-Existing Enumservice Specifications

Many Enumservice Registrations, published via IETF RFCs, already exist at the time of the development of this document. These existing Enumservice Specifications MAY be revised to comply with the specifications contained herein. All revisions of Enumservice Specifications MUST follow the specifications contained herein.

9. Extension of Existing Enumservice Specifications

There are cases where it is more sensible to extend an existing Enumservice registration rather than proposing a new one. Such cases include adding a new Subtype to an existing Type. Depending on the nature of the extension, the original Enumservice Specification needs to be extended (Updates) or replaced (Obsoletes) [RFC2223]. Specifically, an update is appropriate when a new subtype is being added without changes to the existing repertoire. A replacement is needed if there is a change to the default, or changes to the assumptions of URI support in clients.

10. Security Considerations

10.1. Considerations Regarding This Document

Since this document does not introduce any new technology, protocol, or Enumservice Specification, there are no specific security issues to be considered for this document. However, as this is a guide to authors of new Enumservice Specifications, the next section should be
considered closely by authors and experts.

10.2. Enumservice Security Considerations Guideline

[I-D.ietf-enum-3761bis] already outlines security considerations affecting ENUM as a whole. Enumservice Specifications do not need to and SHOULD NOT repeat considerations already listed in that document. However, Enumservice Specifications SHOULD include a reference to that section.

ENUM refers to resources using existing URI Schemes and protocols. Enumservice Specifications do not need to and SHOULD NOT repeat security considerations affecting those protocols and URI Schemes themselves.

However, in some cases, the inclusion of those protocols and URI Schemes into ENUM specifically could introduce new security issues. In these cases, those issues or risks MUST be covered in the ‘Security Considerations’ section of the Enumservice Specification.

Authors should pay particular attention to any indirect risks that are associated with a proposed Enumservice, including cases where the proposed Enumservice could lead to the discovery or disclosure of Personally Identifiable Information (PII).

11. IANA Considerations

11.1. Enumservice Registrations

IANA will update the registry "Enumservice Registrations" according to (this) Section 11.1, which will replace the old mechanism as defined in RFC 3761 [RFC3761].

It is noted that the process described herein applies only to ordinary Enumservice registrations (i.e. the registration process of ‘X-’ Enumservices is beyond the scope of this document).

11.1.1. IANA Registration Template

The IANA Registration Template consists of the following fields that are specified in Section 5.2:

- Enumservice Class:
- Enumservice Type:
11.1.2. Location

Approved Enumservice registrations are published in the IANA Registry named "Enumservice Registrations", which is available at the following URI:
< http://www.iana.org/assignments/enum-services >.

In this Registry, only the filled IANA Registration Template as listed in Section 11.1.1 and specified in Section 5.2 is published.

Where the Enumservice Specification is NOT an RFC, IANA MUST hold an escrow copy of that Enumservice Specification. Said escrow copy will act as the master reference for that Enumservice Registration.

11.1.3. Structure

IANA maintains the Enumservice Registry sorted in alphabetical order. The first sort field is Type, the second is Subtype.

Each Enumservice starts with a caption, which is composed of Type and Subtype, separated by a colon; e.g. if the Type is "foo" and the Subtype "bar", the resulting caption is "foo:bar".

[I-D.hoeneisen-enum-enumservices-transition] updates the existing Enumservices into the new IANA Registration Template.
11.1.4. Registration Procedure

There is a difference in process depending on whether or not the Enumservice Specification will be published as RFC. In case of RFC, the normal IETF procures (according to [RFC5226]) apply. In case of a specification other than RFC, there is a slight difference to [RFC5226] (see below). The reason for this lies in the complexity of Enumservice Specifications. Registration Documents will most likely undergo changes during Expert Review, so that in most cases it will not be published by the time the Expert Review is carried out.

11.1.4.1. Published as RFC

As soon as IANA receives the Registration Document from the RFC Editor, IANA will take care of the ‘Expert Review Process’ according to "Guidelines for Writing an IANA Considerations Section in RFCs" [RFC5226].

After successful Expert Review IANA will register the Enumservice, i.e. add the Enumservice to the IANA "Enumservice Registrations" Registry (see also Section 11.1.2).

The RFC Editor will now take care of the publication of the RFC.

11.1.4.2. Published as generic Specification

Whenever a Registration Document is submitted via the IANA website, IANA will take care of the ‘Expert Review Process’ according to "Guidelines for Writing an IANA Considerations Section in RFCs" [RFC5226].

Once the experts have approved the Enumservice, IANA will inform the authors. This information SHOULD also include a reminder, that the authors are now responsible for publication of the Registration Document (see also Section 6.6) and that the Enumservice will be added to the IANA Registry only after its Enumservice Specification is published according to ‘Specification Required’ as defined in [RFC5226] (see also Section 6.7). The Registration process will now be on hold until the authors inform IANA about the publication of the Enumservice Specification (see also Section 6.7).

Once the authors have informed IANA about the publication, IANA MUST ensure that the requirements to "Specification Required" as defined in [RFC5226] are met, the reference to the specification is unambiguous, and the content of the Enumservice Specification is identical to the Registration Document as approved by the Experts. IANA will then register the Enumservice, i.e. add the Enumservice to the IANA "Enumservice Registrations" Registry (see also
Change control of any Enumservice Registrations is done by "Expert Review" and "Specification Required" according to [RFC5226]. Updates of Enumservice Specifications MUST comply with the guidelines described in this document. Updates are handled the same way as initial Enumservice Registrations.

Authorized Change Controllers are the experts and the IESG.

Enumservice registrations MUST NOT be deleted. An Enumservice that is believed no longer appropriate for use, can be declared obsolete by publication of a new Enumservice Specification changing its "Intended Usage" field to "OBSOLETE"; such Enumservices will be clearly marked in the lists published by IANA. As obsoletions are updates, they are also handled the same way as initial Enumservice Registrations.

11.1.6.  Restrictions

As stated in Section 3.2, a "-" (dash) MUST NOT be used as the first nor as the second character of a Type nor a Subtype. Furthermore, any identifying tag of any Enumservice MUST NOT be set to nor start with "E2U". Any Enumservice registration requests covered by these restrictions MUST be rejected by IANA, and the 'Expert Review Process' SHOULD NOT be initiated.

Section 5.2 contains examples for Enumservice registrations. Therefore, IANA MUST NOT register an Enumservice with Type or Subtype set to "foo", "bar", or "sbar", unless the experts explicitly confirm an exception.

12.  Acknowledgements

The authors would like to thank the following people who have provided feedback or significant contributions to the development of this document: Lawrence Conroy, Alfred Hoenes, Peter Koch, Edward Lewis, and Jon Peterson

Lawrence Conroy has provided extensive text for the Enumservice Classification section.

Section 3 of RFC 3761 [RFC3761], which was edited by Patrik Faltstrom and Michael Mealling, has been incorporated to this document. Please see the Acknowledgments section in RFC 3761 for additional
13. References

13.1. Normative References


13.2. Informative References


Appendix A. Changes Overview

This section lists the changes applied to the Enumservice registration process and the IANA registry definition, compared to RFC 3761.

- While RFC 3761 required "Standards track or Experimental" RFCs for an Enumservice to be registered, this document mandates "Expert Review" and "Specification Required".

- This document defines the classification of Enumservices. The IANA Registration Template has been complemented to contain a "Classification" field.
A new field "Enumservice Registration(s)" has been added to the IANA Registration Template.

The former field "Any other information that the author deems interesting" of the IANA Registration Template has been shortened to "Further Information".

The Enumservice "Name" field has been removed from the IANA Registration Template.

Appendix B. Document Changelog

[RFC Editor: This section is to be removed before publication]

draft-ietf-enum-enumservices-guide-13:
- alex: Some minor changes - the only real open issue is whether or not we should go to an XML template instead of the plain text one. IANA provided a "chunk", but gave no feedback about schema, namespace, etc. so it is deemed not "normative" enough yet.
- bernie: Implemented IANA Feedback: made difference between RFC and no-RFC specs more clear; now the both variants slightly differ in process.
- bernie: Implemented more feedback of Peter Koch:
  * Terminology updated throughout the document: Enumservice Specification / Registration Document
  * Changed IANA Template field ‘Registration Document(s) to ‘Enumservice specification(s)’
  * Disallow dash ‘-‘ as last char of Type or Subtype
  * Removed XML2RFC template and referencing sections
- bernie: changed "Subtype names MAY be shared between URI Schemes that the Registration specifies as mandatory to implement for a given Subtype." to "Subtype names MAY be shared between URI Schemes, if all the URI Schemes within the same Subtype are mandatory to implement."
- bernie: Cleared out independent submission and added reference to RFC 4846
- jason: Per the co-chair and Peter Koch, doc changed to BCP. Doc doesn’t specify a protocol but a process. Both RFC 2026, section 5, and section 4.3 of RFC 5226 suggest that process documents, and IANA Guidelines in particular, usually are published as BCP RFCs. Also, there’s little to implement independently in this draft that could help advance it on the Standards Track.
- jason: various nits clean-up suggested by Peter Koch.

draft-ietf-enum-enumservices-guide-12:
bernie: Refined process, i.e. separation of Expert Review and addition to IANA Registry (only after publication of spec):
  * Split up "Further Steps" into three new sections
  * Extended ASCII Art
  * Adjusted IANA considerations
bernie: Updated Open Issues
alex: Added reference to RFC3552 (security considerations guidance)
alex: Added instructions2author as informative reference - i don’t see another way (revision 439, closing ticket 25)
alex: Moved text about use cases from Review Guidelines up to "other sections", slightly reworded it (revision 438, closing ticket 66)
bernie: Updated own contact details
bernie: Implemented editorial feedback from Alfred Hoenes
bernie: Added some clarifications to IANA consideration as proposed by Michelle Cotton (IANA)
bernie: Edited appendix "Changes Overview", moved stuff from "Introduction" to "Changes Overview"
bernie: Updated IANA section "Change Control":
  * Authorized Change controllers are experts and IESG
  * Removed field "Authorized Change Controller" (was introduced in -11)
bernie: Replaced "number blocks" by "wildcards" (DNS Considerations) to avoid conflict with RFC3761
bernie: Replaced reference rfc2434bis with rfc5226
bernie: Moved terminology related paragraph from Introduction to Terminology Section
bernie: Added reference to transition document
jason: Updated my author address
jason: Closed out active tickets at http://ietf.enum.at/cgi-bin/trac.cgi/report/1
jason: Section 8, review of pre-existing enumservices, updated with IETF 72 feedback that this must take place
jason: Ticket 39: Added text to section 4.1, general enumservice considerations, section 2, bullet 2 to address comment by Lawrence Conroy about expired I-Ds
jason: Ticket 45: Added text to section 7.1, expert review / review guidelines, bullet 3, to indicate that a use case SHOULD be included. Also added related text to section 5.8, other sections, to address this. This resolves comments by Lawrence Conroy
o jason: Ticket 55: Replaced 'repository' with 'registry' throughout the document to normalize this text and make it uniform.

o jason: Ticket 52: Checked references to ensure rfc5226 is cited instead of rfc2434bis, which Bernie seems to have mainly covered. I also added a reference in the header for rfc5226, since it is a normative reference.

o jason: Ticket 25: Removed reference to rfc2223bis-08 as this I-D is now listed as dead.

o jason: Ticket 49: Have updated section 5.2, IANA registration, bullet on authors addresses, to say that email addresses MUST NOT be included in the IANA Registry. I opened a related ticket. Seems there are some email addresses in the registry. Also simplified author(s) and expert(s) to authors and experts throughout.

o jason: Ticket 28: Minor changes to Section 10.1 and 10.2, Security Considerations

o jason: Ticket 30: Updated section 6.4, 6.5, on IANA registration to include that submission must be in XML format for IANA and that the Enumservice must have an RFC number, per discussion at IETF 72

o jason: Ticket 42: Cleaned up section 5.7, DNS considerations, per comments from Lawrence.

o jason: Updated definitions to reflect IANA Designated Experts per RFC 5226, and clean up of IANA-related terms (Registry, Template, etc.)

o jason: Ticket 51: added section to describe the need to have a contact listed for updating a registration, per RFC 5226, section 5.2.

draft-ietf-enum-enumservices-guide-10:

o bernie: No longer empty field for IANA Registration ('N/A' must be used in this case)

o bernie: Adjusted IANA Registration Template:
  * Registration Document -> Registration Document(s)
  * Author -> Author(s)

o bernie: IANA repository in alphabetical order by Type and Subtype

o bernie: Class, Type, Subtype and URI Schema to begin with capital

o bernie: Captions for each Enumservice

o bernie: Consistent use of "field" for fields within IANA registration template (no longer used are "item" or "section")

o bernie: URI Schemes without colons and between single quotes, no longer email address in author(s) field

o bernie: Adjusted IANA Registration Section of XML2RFC template

o alex: Added List of Classes to choose from

draft-ietf-enum-enumservices-guide-09:

o alex: Removed Enumservice "Name" as decided at IETF 71
o alex: Reworded registration requirements
o alex: Explained possible values for "Intended Usage"
o bernie: Rewrite of section 'Change Control'
o bernie: Cleared out scope of this document (only ordinary, but no 'X-' registrations)
o bernie: Cleared out naming restrictions in IANA section
o bernie: Changed section name from 'ENUM Service Registration’ to 'IANA Registration'
o bernie: Combined Expert Review related sections
o bernie: Partly implemented feedback Alfred Hoenes and added him to Acknowledgments
o bernie: Enhanced examples for "Registration Document"
o bernie: Enhanced examples for "IANA Considerations" (feedback from Alfred Hoenes)
o bernie: Removed Note about RFC3761bis obsoleting RFC3761 (does not belong to this doc)
o bernie: Rewrote Naming Requirements section (impact to IANA Considerations - Restrictions)

draft-ietf-enum-enumservices-guide-08:
o alex: new text for Subtypes of protocol class enumservices
("mandatory to implement" stuff)
o alex: added "to be foreseen" to the application Type Subtype recommendation
o alex: added "lowercase" recommendation to the Type names
o bernie: Corrected various typos, clarifications, and other editorial stuff (feedback from Lawrence Conroy)
o bernie: IANA Registry ftp -> http (feedback from Lawrence Conroy)
o bernie: Made steps prior to IANA submission mandatory (feedback from Lawrence Conroy)
o bernie: Shortened abstract

draft-ietf-enum-enumservices-guide-07:
o bernie: Section DNS considerations made mandatory
o bernie: Complete rewrite of IANA considerations
o bernie: XML2RFC template will be downloadable at IANA
o bernie: Complete re-write of process
o alex: Adjusted Cook-book / classification
o bernie: Take over chapter "Registration mechanism for Enumservices" from RFC 3761bis
o bernie: Changed title to adjust to new purpose
o bernie: Intended status changed to Standards Track (was bcp)
o bernie: Obsoletes (partly) RFC 3761
o bernie: Adjusted section "Registration mechanism for Enumservices"
o bernie: Updated most RFC 3761 references to either RFC3761bis or new (internal) section
o bernie: Acknowledgment for RFC3761 contributors
o bernie: Shortened bullet point in IANA Registration Template:
  "Any other information that the author deems interesting"
  => "Further Information"
o alex: Rewritten Abstract, Introduction to be consistent with with
  new goal (IANA Registry description)
o alex: Add obsoletes section 3 of RFC 3761 to Introduction
o alex: Changed section 3 to "registration requirements", Simplified
  structure
o alex: Added examples for protocol Enumservice classification
o alex: Added text about "other" classification

draft-ietf-enum-enumservices-guide-06:
o alex: updated Class Schemes.
o alex: updated expert’s tasks
o alex: added experts review considerations
o bernie: Moved Terminology section in XML2RFC template (now after
  Introduction)
o bernie: Class is now part of the Enumservice registration in the
  IANA template
o bernie: Individual Submission relaxed (comment Peter Koch)
o bernie: updated vcard Ref (now RFC)

draft-ietf-enum-enumservices-guide-05:
o bernie/alex: added text for sections ‘The Enumservice Expert
  Selection Process’ and ‘The Process for Appealing Expert Review
  Decisions’
o bernie: added ASCII-art figure for registration process
o bernie: adjusted registration process
o jason: proposed registration process

draft-ietf-enum-enumservices-guide-04:
o bernie: added section about Extension of existing Enumservice RFCs
o bernie: added open issue about future registration process
o bernie: added category (bcp)
o bernie: clean up in Security Considerations
o bernie: editorial stuff (mainly XML issues)

draft-ietf-enum-enumservices-guide-03:
o alex: moved terminology section
o alex: removed note asking for feedback
o bernie: added DNS consideration section
o bernie: added Acknowledgments section
o bernie: editorial stuff (nicer formatting, fixing too long lines)
o alex: added security considerations from vcard draft.

draft-ietf-enum-enumservices-guide-02:
Appendix C.  Open Issues

[RFC Editor: This section should be empty before publication]

o Decision on whether to go on with XML-Registry at IANA
o Review XML2RFC template

Authors’ Addresses

Bernie Hoeneisen
Swisscom
Hardturmstrasse 3
CH-8005 Zuerich
Switzerland

Phone: +41 44 2747111
Email: bernie@ietf.hoeneisen.ch (bernhard.hoeneisen AT swisscom.com)
URI: http://www.swisscom.ch/

Alexander Mayrhofer
enum.at GmbH
Karlsplatz 1/9
Wien A-1010
Austria

Phone: +43 1 5056416 34
Email: alexander.mayrhofer@enum.at
URI: http://www.enum.at/
Full Copyright Statement

Copyright (C) The IETF Trust (2008).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.