         Registration Data Access Protocol Basic Search Processing
                  draft-hollenbeck-weirds-rdap-search-02

Abstract

   This document describes path segments and query parameters needed to
   construct HTTP URLs that may be used to search for and retrieve
   registration information from registries (including both Regional
   Internet Registries (RIRs) and Domain Name Registries (DNRs)) using
   "RESTful" web access patterns. It also describes a method of
   encoding responses using Javascript Object Notation (JSON).

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 http://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 December 20, 2013.

the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.

Table of Contents

1. Conventions Used in This Document

   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].

1.1. Acronyms and Abbreviations

      DNR: Domain Name Registry
      IDN: Internationalized Domain Name
      RDAP: Registration Data Access Protocol
      RIR: Regional Internet Registry

2. Introduction

   This document describes a specification for registration data search
   functions using a RESTful web service. The search functions are
   implemented using the Hypertext Transfer Protocol (HTTP) [RFC2616].

   Lookup processing as specified in [I-D.ietf-weirds-rdap-query] is
   used when a client wishes to retrieve information associated with a
   data object represented by a character string that exactly matches a
   particular key. There is no provision for partial string pattern
   matching to represent unknown characters or multiple result
   possibilities. Experience with WHOIS [RFC3912] implementation and

operation has shown that people are often unsure of exact spellings
and they often want to receive multiple results that match a
particular pattern. This specification is intended to meet that need
for the Registration Data Access Protocol (RDAP).

3. Query and Search Parameters

   RDAP search path segments are formed using a concatenation of the
   plural form of the object being searched for, a forward slash
   character ('/', ASCII value 0x002F), and an HTTP query string. The
   HTTP query string is formed using a concatenation of the question
   mark character ('?', ASCII value 0x003F), the JSON object value
   associated with the object being searched for, the equal sign
   character ('=', ASCII value 0x003D), and the search pattern. For the
   domain and entity objects described in this document the plural
   objects forms are "domains" and "entities". The JSON object value is
   "name". One could construct a query string for an entity email
   address using the "email" object name, but that is beyond the scope
   of this specification.

3.1. Domain Search

   Syntax: domains/?name=<domain search pattern>

   Searches for domain information are of the form /domains/?name=XXXX,
   where XXXX is a search pattern representing a fully-qualified domain
   name [RFC4343] in a zone administered by the server operator of a
   DNR. The following path would be used to find DNR information for
   domain names matching the "example*.com" pattern:

   /domains/?name=example*.com

   Internationalized Domain Names (IDNs) in U-label format [RFC5890] can
   also be used as search patterns (see Section 4).

   Note that this search is relevant to DNRs, not RIRs.

3.2. Entity Search

   Syntax: entities/?name=<entity search pattern>

   Searches for entity information are of the form /entities/?name=XXXX,
   where XXXX is a search pattern representing an entity name as
   specified in Section 7.1 of [I-D.ietf-weirds-json-response]. The
   following path would be used to find DNR information for DNR entity
   names matching the "Bobby Joe*" pattern.

   /entities/?name=Bobby%20Joe*

   URLs MUST be properly encoded according to the rules of [RFC3986].
   In the example above, "Bobby Joe*" is encoded to "Bobby%20Joe*".

4. Search Processing

   Searching occurs either on whole strings or on partial strings.
   Partial string searching uses the asterisk ('*', ASCII value 0x002A)
   character to match zero or more characters. The location or number
   of occurrences of the asterisk character is not dictated by this
   specification. Some servers might support the asterisk in a trailing
   location only (e.g. "/domains/example*.com") while others may allow
   it a leading location (e.g. "/domains/*example.com"), or within
   strings, or even in multiple locations.

   If a server receives a search request but cannot process the request
   because it does not support a particular style of partial match
   searching, it SHOULD return an HTTP 422 [RFC4918] error. When
   returning a 422 error, the server MAY also return an error response
   body as specified in Section 12 of [I-D.ietf-weirds-json-response] if
   the requested media type is one that is specified in
   [I-D.ietf-weirds-using-http].

   Because Unicode characters may be combined with another Unicode
   character or characters, partial matching is not feasible across
   combinations of Unicode characters. Servers SHOULD NOT partially
   match combinations of Unicode characters where a Unicode character
   may be legally combined with another Unicode character or characters.
   Clients MUST NOT issue a partial match search of Unicode characters
   where a Unicode character may be legally combined with another
   Unicode character or characters. Partial match searches with
   incomplete combinations of characters where a character must be
   combined with another character or characters are invalid. Partial
   match searches with characters that may be combined with another
   character or characters are to be considered non-combined characters
   (that is, if character x maybe combined with character y but
   character y is not submitted in the search string then character x is
   a complete character and no combinations of character x are to
   searched).

   Because Unicode characters may be combined with another Unicode
   character or characters, partial matching requires that a server
   maintain a list of valid character combinations to be considered a
   match. When comparing DNS U-labels, servers SHOULD use the code
   points specified in [RFC5892] to determine partial matches. When
   comparing entity names, servers SHOULD use the normalization rules
   and code points specified by [I-D.ietf-precis-nickname] to determine
   partial matches.

   Clients SHOULD NOT submit search requests with partial matching for
   DNS A-labels. A-labels represent an encoding that can only be
   reconstructed properly when the label is complete.

5. Search Results

   The method to return search results described here is not limited to
   the /domains or /entities searches defined in this document. It can
   also be used by future specifications to define search results for
   other types of registration data.

   Search results are returned in a JSON object. This object contains
   data structures as outlined in Section 5 of
   [I-D.ietf-weirds-json-response] (e.g. "rdapConformance", "notices",
   etc...) and an array called "results" containing the objects that are
   a result of the search. For the /domains (Section 3.1) and /entities
   (Section 3.2) searches, the entity object class and the domain object
   class are defined in [I-D.ietf-weirds-json-response].

   To identify the type of object returned in the "results" array, each
   object SHOULD contain a JSON string named "objectClass" (see
   Section 4.2 of [I-D.ietf-weirds-json-response] regarding the
   inclusion of new JSON data in object classes). For domains, the
   string MUST be "domain" and for entities the string MUST be "entity".

   Servers SHOULD signify their compliance with this specification by
   including the string "domain_entity_search_level_0" in the
   "rdapConformance" array.

```
   {
     "rdapConformance" :
     [
       "rdap_level_0",
       "domain_entity_search_level_0"
     ],
     ...
     "results" :
     [
       {
         "handle" : "1-XXXX",
         "name" : "1.example.com",
         "objectClass" : "domain",
         ...
       },
       {
         "handle" : "2-XXXX",
         "name" : "2.example.com",
         "objectClass" : "domain",
```

```
      ...
     }
    ]
   }
```

                                Figure 1

6. Internationalization Considerations

   TBD. Give guidelines for how a local policy could work for
   searching.

7. IANA Considerations

   This document does not specify any IANA actions.

8. Security Considerations

   Security services for the operations specified in this document are
   described in "Security Services for the Registration Data Access
   Protocol" [I-D.ietf-weirds-rdap-sec]. Additional considerations that
   are specific to search functionality are described here.

   Search functionality typically requires more server resources (such
   as memory, CPU cycles, and network bandwidth) when compared to basic
   lookup functionality. This increases the risk of server resource
   exhaustion and subsequent denial of service due to abuse. This risk
   can be mitigated by developing and implementing controls to restrict
   search functionality to identified and authorized clients. If those
   clients behave badly, their search privileges can be suspended or
   revoked. Rate limiting as described in Section 5.5 of "Using the
   Registration Data Access Protocol (RDAP) with HTTP"
   [I-D.ietf-weirds-using-http] can also be used to control the rate of
   received search requests. Server operators can also reduce their
   risk by restricting the amount of information returned in response to
   a search request.

9. Acknowledgements

   The authors would like to acknowledge the following individuals for
   their contributions to this document: Ning Kong, Linlin Zhou.

10. References

10.1. Normative References

   [I-D.ietf-precis-nickname]

                 Saint-Andre, P., "Preparation and Comparison of
                 Nicknames", draft-ietf-precis-nickname-05 (work in
                 progress), November 2012.

     [I-D.ietf-weirds-json-response]
                 Newton, A. and S. Hollenbeck, "JSON Responses for the
                 Registration Data Access Protocol (RDAP)", draft-ietf-
                 weirds-json-response-04 (work in progress), June 2013.

     [I-D.ietf-weirds-rdap-query]
                 Newton, A. and S. Hollenbeck, "Registration Data Access
                 Protocol Lookup Format", draft-ietf-weirds-rdap-query-05
                 (work in progress), June 2013.

     [I-D.ietf-weirds-rdap-sec]
                 Hollenbeck, S. and N. Kong, "Security Services for the
                 Registration Data Access Protocol", draft-ietf-weirds-
                 rdap-sec-04 (work in progress), June 2013.

     [I-D.ietf-weirds-using-http]
                 Newton, A., Ellacott, B., and N. Kong, "HTTP usage in the
                 Registration Data Access Protocol (RDAP)", draft-ietf-
                 weirds-using-http-05 (work in progress), May 2013.

     [RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate
                 Requirement Levels", BCP 14, RFC 2119, March 1997.

     [RFC2616]   Fielding, R., Gettys, J., Mogul, J., Frystyk, H.,
                 Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext
                 Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

     [RFC3986]   Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform
                 Resource Identifier (URI): Generic Syntax", STD 66, RFC
                 3986, January 2005.

     [RFC4343]   Eastlake, D., "Domain Name System (DNS) Case Insensitivity
                 Clarification", RFC 4343, January 2006.

     [RFC4918]   Dusseault, L., "HTTP Extensions for Web Distributed
                 Authoring and Versioning (WebDAV)", RFC 4918, June 2007.

     [RFC5890]   Klensin, J., "Internationalized Domain Names for
                 Applications (IDNA): Definitions and Document Framework",
                 RFC 5890, August 2010.

     [RFC5892]   Faltstrom, P., "The Unicode Code Points and
                 Internationalized Domain Names for Applications (IDNA)",
                 RFC 5892, August 2010.

10.2. Informative References

   [RFC3912] Daigle, L., "WHOIS Protocol Specification", RFC 3912,
              September 2004.

Appendix A. Change Log

   Initial -00: First draft individual submission.
   -01: Changed HTTP error code from 500 to 422 in Section 4. Added a
      normative reference to RFC 4918.
   -02: Updated Security Considerations. Added IDN processing text in
      Section 3.1.

Authors' Addresses

   Scott Hollenbeck
   Verisign Labs
   12061 Bluemont Way
   Reston, VA 20190
   US

   Email: shollenbeck@verisign.com
   URI: http://www.verisignlabs.com/

   Andrew Lee Newton
   American Registry for Internet Numbers
   3635 Concorde Parkway
   Chantilly, VA 20151
   US

   Email: andy@arin.net
   URI: http://www.arin.net