Network Working Group
Internet-Draft
Intended status: Informational
Expires: December 7, 2019

M. Blanchet
Viagenie
June 5, 2019

RDAP Deployment Findings and Update
draft-blanchet-regext-rdap-deployfindings-03

Abstract

Registration Access Data Protocol(RDAP) is being deployed in domain and IP address registries. This document describes issues and findings while interfacing with the known server implementations and deployments. It also provides recommendations for the specifications.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

While developing various tools and software related to RDAP, issues have been found and are documented below. This document should help in writing future version of the specifications and provide better conformant deployment. It is split in various sections based on where the fix should be applied. Obviously, there are different levels of severity of the issues, including nits or very minor. The actual instances and organisations running the RDAP servers where the issues were found are not listed.

2. IANA RDAP Registries Related Issues

This section describes issues related to the IANA non-Bootstrap registries as specified in [RFC7483].

2.1. Values not Registered or Similar

The IANA RDAP JSON Values registry [1] contains various values expected in JSON responses. The following table shows values not registered in the registry but seen in the field. The second column shows the possible corresponding values already registered.

Recommendation: implementations should replace their custom values with the registered ones, when one exist. Implementors should register their values when there is no corresponding registered one.

Remarks Type

Unregistered Values	Possibly Corresponding Registered Values
object truncated due to server policy	object truncated due to authorization
Response truncated due to authorization	object truncated due to authorization
Object truncated due to authorization	object truncated due to authorization
object redacted due to authorization	object truncated due to authorization

Event Action

Unregistered Values	Possibly Corresponding Registered Values
delegation check	
last correct delegation check	
last update	last changed

Status Value

Unregistered Values	Possibly Corresponding Registered Values
server deleted prohibited	server delete prohibited
ok	active

Role Value

Unregistered Values	Possibly Corresponding Registered Values
owner	registrant

2.1.1. Registry Entity

The (domain or IP) registry itself is currently not modeled in entities in RDAP. In an whois query for a TLD itself, the Remarks contains the URL of the registry entity (for registration information) and the whois entry of the registry is returned. In RDAP context, the RDAP server URL of the TLD registry should also be returned. Therefore, IANA RDAP server should send this data for the TLDs as part of its RDAP response. These semantics are currently not modeled.

This document proposes that RDAP servers may send an entity with role "registry" in the top-level of the RDAP response. This entity would have embedded [links] to its web server ("rel": "self", "type": "text/html") and rdap server ("rel": "self", "type": "application/rdap+json").

IANA Action: add a new row "registry", "role" to the RDAP JSON Values registry.

2.2. RDAP Extensions not Registered

The IANA RDAP Extensions registry [2] contains various extensions values expected in RDAP JSON responses in the rdapCconformance member. It is our understanding from [RFC7483] section 4.1 and [RFC7480] section 8.1 that only the prefix of the extension (i.e. "rdap_ObjectTag"), not the whole string ("rdap_objectTag_level_0"), need to be registered in the IANA registry. However, some entries in

the IANA RDAP extensions registry seem to imply a 0 version as part of the registered value.

The following table shows values seen in the field in the first column, corresponding prefix (guessed as there is no clear delimiter) in the second column and if the prefix is registered in IANA registry in the third column.

This registry may end up listing all names of all registries if each one has his own extension. Moreover, there is no clear delimiter of the prefix in the full string, which may not help the RDAP client to interpret correctly. As with [RFC6350], we may instead use the First Come First Serve(FCFS) private enterprise numbers (PEN) registry to automatically have an organisation prefix defined without creating another set of org names within this registry and have the delimiter be "_" following the PEN.

Recommendation (short term): implementations should replace their custom values with the registered ones, when one exist. Implementors should register their values when there is no corresponding registered one.

Values Seen	Corresponding Assumed Prefix	Prefix Already Registered in IANA
rdap_objectTag_level_0	rdap_objectTag	Y
fred_version_0	fred	Y
rdap_openidc_level_0	rdap_openidc	N
icann_rdap_technical_implementation_guide_0	icann_rdap_technical_implementation_guide	N
icann_rdap_response_profile_0	icann_rdap_response_profile	N
itNic_level_0	itNic	N
nicbr_level_0	nicbr	N
ur_domain_check_level_0		N
history_version_0		N
registrar_api_0		N

3. RDAP Responses

This section discusses issues found related to RDAP responses, specified in [RFC7483].

3.1. Cross-origin resource sharing(CORS)

As specified in [RFC7480], the HTTP "Access-Control-Allow-Origin: *" header should be included in the responses, to enable Web clients to work properly. Some RDAP servers do not set this header. RFC7480 says "it is RECOMMENDED that servers". It should be updated to "for any public Internet deployment, servers MUST".

3.2. Object Class Name empty

A non-conformant server sends the following answer, where the value of "objectClassName" is an empty string (as well as "handle" also empty). As per [RFC7483] section 4.9, this "objectClassName" value is required. Extract of the seen response:

```
{
  entities: [
   {
   "entities": [
     {
     "objectClassName": "",
     "handle": "",
     }
   ],
 ],
}
```

3.3. Links Relation Values

The links relation values as specified in [RFC7483] section 4.3 refer to [RFC5988] which creates the IANA Link Relations registry [3]. This registry contains a large number of values where most of them do not apply to the RDAP deployment. As seen with other values above that are similar to registered ones but not used, we list here the ones we have seen. It would be appropriate to further describes the main ones in the RFC so implementors focus on ones that are expected instead of picking the wrong ones in the IANA registry or to define new ones and do not register them.

Links Relation Values Seen

Values	Registered in IANA registry
about	Y
alternate	Y
copyright	Y
describedBy	Y
help	Y
related	Y
self	Y
terms-of-service	Y
up	Y
https://restOfURLRedacted	N

As shown in the table, an implementation put an URL as the value of the "rel", instead of an actual registered value.

3.4. Related link pointing to self causes infinite loop

An RDAP server returns a link of "rel": "related" is pointing to itself, therefore causing the RDAP client to fetch the object again, then read the related link and then fetch again, creating an infinite loop. Extract of the seen response:

```
{
  "links": [
    {
    "title": "Self",
    "rel": "self",
    "type": "application/rdap+json",
    "href": "https://rdapserver.example.com/domain/example.net"
    },
    {
    "title": "Registrar Data for this object",
    "rel": "related",
    "href": "https://rdapserver.example.com/domain/example.net",
    "type": "application/rdap+json"
    }
  ],
}
```

Recommendation: do not put related link same as self. RFC7483 section 4.2 should be updated to add the following text: "A link of "rel": "related" should not have the "href" value the same as the value of "href" of link of "rel": "self".

3.5. Registrant Entity Too Deep

An RDAP server returns the registrant entity in a subentity, which makes difficult to parse given the expectation is the registrant would be at the top level. Extract of the seen response:

```
{
 entities: [
  {
    "objectClassName": "entity",
    "handle": "HANDLE1",
    "roles": [ "abuse" ],
    "vcardArray": [ ... ],
    "entities": [
       {
       "objectClassName": "entity",
       "handle": "HANDLE2",
       "roles": [ "registrant" ],
       "vcardArray": [ ... ],
       }
    ],
],
```

Recommendation: put the registrant in the top-level entities as follows:

```
{
 entities: [
  {
    "objectClassName": "entity",
    "handle": "HANDLE1",
    "roles": [ "abuse" ],
    "vcardArray": [ ... ]
  },
  {
    "objectClassName": "entity",
    "handle": "HANDLE2",
    "roles": [ "registrant" ],
    "vcardArray": [ ... ],
  }
],
```

4. Queries

This section talks about support of RFC7482 queries and the RDAP server behaviors seen.

4.1. URL encoding of :

For RIR registries, the ip query may include an IPv6 address which then includes one or many ":". Clients may decide to do percent-encoding of the query. In one RDAP server, the server rejected the percent-encoded query of an IPv6 address. For example, https://rdapserver.example.com/ip/2001%3Adb8%3A0%3A%3A/48 is rejected, while https://rdapserver.example.com/ip/2001:db8:0::/48 is accepted.

Recommendation: accept both percent-encoded queries or non-percent encoded queries.

5. Domain Registrar RDAP Server Location

The ICANN RDAP Profile [4] section 3.2 requires the domain registries who do not have registrant information (so-called thin registries) to put a specific link of "rel": "related" pointing to the domain registrar responsible for the domain being queried, so that a client can get the registrant information using a second query to the related link. However, the semantics seems ambiguous as other RDAP servers may use the "rel": "related" for other related means, but not the specific semantic of finding the registrant data. Therefore, a possible mitigation is to define a new "rel" type of "registrantInfo" (mnemonic TBD) to carry the specific semantic of registrant info.

6. Issues related to RFC7482

6.1. Search patterns that are not

Section 3.2.1 of [RFC7482] says: "domains?nsIp=ZZZZ. ZZZZ is a search pattern representing an IPv4 [RFC1166] or IPv6 [RFC5952] address.". Search pattern has been used throughout the document as something that can include '*', while here, it does not. The syntax statement is also misleading. Similarly, section 3.2.2 says: "nameservers?ip=YYYY YYYY is a search pattern representing an IPv4 [RFC1166] or IPv6 [RFC5952] address."

Recommendation: in [RFC7482], replace: "ZZZZ is a search pattern representing an IPv4" by "ZZZZ is an IPv4", "Syntax: domains?nsIp=<domain search pattern>" by "Syntax: domains?nsIp=<nameserver IP address>", "YYYY is a search pattern representing an IPv4" by "YYYY is an IPv4", "Syntax: nameservers?ip=<nameserver search pattern>" by "Syntax: nameservers?ip=<nameserver IP address>"

7. IANA RDAP Bootstrap Registries Related Issues

This section describes issues related to the IANA Bootstrap registries as specified in [RFC7484].

7.1. Missing Trailing Char in Bootstrap Registries

[RFC7484] section 3 says: "Base RDAP URLs MUST have a trailing "/" character". However, some values in the various IANA Bootstrap registries do not have the trailing "/" character. These should be added to provide consistency.

7.2. Single target value

[RFC7484] provides a way to list multiple RDAP servers for an entry. This flexibility was designed initially to support multiple URI types, such as http: and https, and to provide some level of redundancy. However, given that security deployment policy is to use https everywhere and redundancy can be accomplished in other ways, deployment has shown that all entries in all bootstrap registries have a single target RDAP URL value. Therefore, we can consider updating the RFC to provide only one target value. However, this should be done carefully to avoid breaking current deployed clients.

8. Security Considerations

Proper conformance to specifications helps security. However, no security issues have been found in the context of this draft.

9. IANA Considerations

This document request IANA to add the following values to this registry. TBD. See 'IANA Action:' within the document.

10. Acknowledgements

Audric Schiltknecht, Mario Loffredo, Justin Mack have provided input and suggestions to this document.

11. References

11.1. Normative References

[RFC7480] Newton, A., Ellacott, B., and N. Kong, "HTTP Usage in the Registration Data Access Protocol (RDAP)", RFC 7480, DOI 10.17487/RFC7480, March 2015, <https://www.rfc-editor.org/info/rfc7480>.

[RFC7482] 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>.

[RFC7483] Newton, A. and S. Hollenbeck, "JSON Responses for the Registration Data Access Protocol (RDAP)", RFC 7483, DOI 10.17487/RFC7483, March 2015, <https://www.rfc-editor.org/info/rfc7483>.

[RFC7484] Blanchet, M., "Finding the Authoritative Registration Data (RDAP) Service", RFC 7484, DOI 10.17487/RFC7484, March 2015, <https://www.rfc-editor.org/info/rfc7484>.

11.2. Informative References

[RFC5988] Nottingham, M., "Web Linking", RFC 5988, DOI 10.17487/RFC5988, October 2010, <https://www.rfc-editor.org/info/rfc5988>.

[RFC6350] Perreault, S., "vCard Format Specification", RFC 6350, DOI 10.17487/RFC6350, August 2011, <https://www.rfc-editor.org/info/rfc6350>.

11.3. URIs

[1] https://www.iana.org/assignments/rdap-json-values/rdap-json-values.xhtml

[2] https://www.iana.org/assignments/rdap-extensions/rdap-extensions.xhtml

[3] https://www.iana.org/assignments/link-relations/link-relations.xhtml

[4] https://www.icann.org/en/system/files/files/rdap-technical-implementation-guide-15feb19-en.pdf

Author's Address

Marc Blanchet
Viagenie
246 Aberdeen
Quebec, QC G1R 2E1
Canada

Email: marc.blanchet@viagenie.ca
URI: http://viagenie.ca