RDAP Technical Implementation Guide

12 August 2022
Version: 2.2

Contents

I. Introduction

The Registration Data Access Protocol (RDAP) provides "RESTful" web services to retrieve registration data from Domain Name registrars/registries and Regional Internet Registries. The RDAP base protocol is defined by IETF STD 95. The global set of RDAP RFCs and Internet Drafts are referred to as the RDAP Specifications. See Appendix A for a listing.

The purpose of this document is to encapsulate the operational requirements for RDAP specific to Registration Data Directory Services (RDDS) which, in conjunction with the RDAP Response Profile, defines RDAP implementation in an ICANN operating environment. This document neither creates nor modifies existing policy, rather it maps current policy requirements to the RDAP implementation with flexibility to incorporate future policy changes and the goal of minimal reengineering.

II. Implementation Instruction

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in

this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

1. RDAP protocol

1.1. An RDAP server MUST implement the following RFCs or their respective successors:

1.1.1. STD 95 - https://www.rfc-editor.org/info/std95

1.1.2. RFC8056 - Extensible Provisioning Protocol (EPP) and Registration Data Access Protocol (RDAP) Status Mapping

1.1.3. draft-ietf-regext-rdap-redacted (https://datatracker.ietf.org/doc/html/draft-ietf-regext-rdap-redacted-11) - Redacted Fields in the Registration Data Access Protocol (RDAP) Response, if not an RFC at the time of publication of this document, the latest version of the draft MUST be implemented. Once the specification is published as an RFC, the RDAP server MUST implement that version.

1.2. An *rdapConformance* object [RFC9083] MUST be present in the topmost object of every response, and it MUST contain the conformance level of the RDAP protocol and of any extensions, as specified in RFC9083.

1.3. A server MUST indicate compliance with this specification by including the literal string “icann_rdap_technical_implementation_guide_1” in the *rdapConformance* member for all responses provided by the server.

1.4. The RDAP service MUST be provided over HTTPS only as described in RFC9110 or its successors.

1.5. An RDAP server MUST use the best practices for secure use of TLS as described in RFC7525, RFC8446, and RFC8996 or successors.

1.6. An RDAP client SHOULD be able to successfully validate the TLS certificate used for the RDAP service with a TLSA record from the DNS (RFC6698 and RFC7671) published by the RDAP service provider. The certificate(s) for the RDAP service associated by DNS-Based Authentication of Named Entities (DANE) SHOULD satisfy the requirements of section 1.5.

1.7. The TLS certificate used for the RDAP service SHOULD be issued by a Certificate Authority (CA) trusted by the major browsers and operating systems

such as the ones listed in the Mozilla Included CA Certificate List (https://wiki.mozilla.org/CA:IncludedCAs). The TLS certificate used for the RDAP service SHOULD be issued by a CA that follows the latest CAB Forum Baseline Requirements (https://cabforum.org/baseline-requirements-documents).

1.8. The RDAP server MUST support both RFC7480 GET and HEAD types of HTTP methods.

1.9. The RDAP service MUST be available over both IPv4 and IPv6 transport as described in RFC791 and RFC8200, respectively, or their successors.

1.10. The DNS resource records for the RDAP service SHOULD be signed with DNSSEC, and if DNSSEC is enabled, the DNSSEC chain of trust from the root trust anchor to the name of the RDAP server MUST be valid.

1.11. The RDAP service MUST only use fully-qualified domain names (as defined in RFC8499) in RDAP responses.

1.12. Registry Bootstrap Requirements:

1.12.1. The base URL of Registry RDAP service for each TLD MUST be registered in the the IANA Bootstrap Service registry for Domain Name Space (https://www.iana.org/assignments/rdap-dns/rdap-dns.xhtml), as described in RFC9224.

1.12.2. When a Registry RDAP service base URL is changed, the current URL and the new URL MUST both remain in operation until: 1) the IANA's Bootstrap Service registry for Domain Name Space is updated, and 2) the date and time in the Expires HTTP header of a HTTP/GET request performed on the IANA's Bootstrap registry for Domain Name Space (after the new URL has been published) has elapsed.

1.13. Registrar Bootstrap Requirements

1.13.1. The base URL of Registrar RDAP services MUST be registered in the IANA Registrar IDs registry (https://www.iana.org/assignments/registrar-ids/registrar-ids.xhtml) .

1.13.2. When a Registrar RDAP service base URL is changed, the current URL and the new URL MUST both remain in operation until: 1) the registrar Bootstrap Service registry is updated, and 2) the date and time in the Expires HTTP header of a HTTP/GET request performed on the

then-current registrar Bootstrap (after the new URL has been published) has elapsed.

1.14. When responding to RDAP valid requests, an RDAP server MUST include the Access-Control-Allow-Origin response header, as specified by [W3C.REC-cors-20140116]. Unless otherwise specified, a value of "*" MUST be used.

1.15. RDAP extensions, if used, MUST be registered in the IANA RDAP Extensions registry (https://www.iana.org/assignments/rdap-extensions/rdap-extensions.xhtml), as defined in RFC7480.

2. RDAP Query Support

2.1. Domain name RDAP queries

2.1.1. The RDAP server MUST support Internationalized Domain Name (IDN) RDAP lookup queries using A-label and U-label format [RFC5890] for domain names.

2.2. Nameserver RDAP queries. This section applies only to Registries that support the host object model as described in RFC 5731.

2.2.1. The RDAP server MUST support Internationalized Domain Name (IDN) RDAP lookup queries using A-label and U-label format [RFC5890] for nameserver objects.

2.2.2. RDAP servers MUST support *nameserver* path queries based on the nameserver name as specified in 3.1.4 of RFC9082.

2.2.3. RDAP servers operated by Registries MUST support *nameserver* search queries based on IP address as defined in RFC9082 section 3.2.2, which, for clarity, does not require pattern matching.

2.2.4. RDAP servers operated by Registries MAY support nameserver search queries based on a “nameserver search pattern” as defined in RFC9082 section 3.2.2.

2.3. Contact object RDAP queries

2.3.1. Contact (object) lookups if supported MUST support RDAP lookup requests for *entities* with any role within other objects using the *handle* (as described in 3.1.5 of RFC9082).

2.4. Registrar object RDAP queries. This section applies only to Registries

2.4.1. Registry RDAP servers MUST support Registrar object lookup using an entity path request for *entities* with the *registrar* role using the *handle* (as described in 3.1.5 of RFC9082) where the *handle* of the *entity* with the *registrar* role is be equal to the IANA Registrar ID.

2.4.2. Registrar object lookup by an entity path request using the *fn* element as a handle (encoded according to RFC 3986) MUST be supported by an RDAP server

3. Responses to RDAP queries

3.1. An RDAP server that receives a query string (for domain name or nameserver objects) with a mixture of A-labels and U-labels SHOULD reject the query and return an HTTP 400 “Bad Request” response code with an RDAP error response body that indicates the type of error in the *description* member with an OPTIONAL “lang” (language) attribute. An RDAP server MAY process the query and return a response that contains both the *unicodeName* and the *ldhName* members.

3.2. If the Registrar's RDAP URL is registered in the IANA “Registrar IDs” registry ((https://www.iana.org/assignments/registrar-ids/registrar-ids.xhtml), a registry server RDAP response to a domain query MUST contain a *links* object as defined in [RFC9083] section 4.2., in the topmost JSON object of the response. The *links* object MUST contain the elements *rel:related* and *href* containing the Registrar's RDAP URL of the queried domain object and a *value* with the RDAP lookup path that generated the RDAP response.

3.3. Terms of Service

3.3.1. The terms of service of the RDAP service MUST be specified in the *notices* object in the initial JSON object of the response.

3.3.2. The *notices* object MUST contain a *links* object [RFC9083] containing the URL of the RDAP service provider’s terms of service in the *href*,

rel:terms-of-service, and a *value* with the RDAP lookup path that generated the RDAP response.

3.3.3. The RDAP service provider MUST provide a web page with the terms of service of the RDAP service at the URL contained in the *links* object (3.3.2) which MAY be the same as the terms or service in the *notices* object (3.3.1) or MAY expand upon them.

3.4. RDAP Help queries [RFC9082] MUST be answered and include a *links* member with a URL to a document that provides usage information, policy and other explanatory material.

3.5. Truncated RDAP responses MUST contain a *notices* member describing the reason for the truncation. The value of the *notices* object type MUST be of the form “Response truncated due to {authorization|load|unexplainable reason}”.

3.6. Truncated RDAP objects MUST contain a *remarks* member describing the reason for the truncation. The value of the *remarks* object type MUST be of the form “Result set truncated due to {authorization|load|unexplainable reason}”.

3.7. In the case where the RDAP service provider is querying its database directly, and therefore, using real-time data, the *eventAction* type *last update of RDAP database* member MUST show the timestamp of the response to the query.

3.8. If the RDAP response contains an entity object using jCard, the following applies:

3.8.1. An *entity* MUST use jCard [RFC7095, 3.3.1.3] structured addresses. If a street address has more than one line, it MUST be structured as an array of strings. Example:

```
["adr", {}, "text",
["", "", ["123 Main Street", "Suite 3305"],
"Any Town", "CA", "91921-1234", "U.S.A."]]
```

But if it has a single line or street address, it SHOULD be structured as a simple string. Example:

```
["adr", {}, "text",
["", "", "123 Main Street",
"Any Town", "CA", "91921-1234", "U.S.A."]]
```

3.8.2. In a contact *entity* [RFC9083], a phone number, if returned as part of a response, MUST be inserted as a *tel* property with a *voice* type parameter, as specified in RFC6350, the vCard Format Specification and its corresponding JSON mapping RFC7095.

3.8.3. In a contact *entity*, a fax number, if returned as part of a response, MUST be inserted as a *tel* property with a *fax* type parameter, as specified in RFC6350, the vCard Format Specification and its corresponding JSON mapping RFC7095.

3.9. All roles included in an RDAP query response MUST be registered at the IANA RDAP JSON Values Registry (https://www.iana.org/assignments/rdap-json-values/rdap-json-values.xhtml) as described in RFC9083.

3.10. If the RDAP response contains an entity object with the *registrar* role, the entity MUST contain a *publicIDs* member to identify the IANA Registrar ID from the IANA Registrar ID registry. The type value of the *publicID* object MUST be equal to the IANA Registrar ID

3.11. In the case of a Registry in which nameservers are specified as domain attributes, the existence of a nameserver used as an attribute for an allocated domain name MAY be treated as equivalent to the existence of a host object.

Appendix A: RDAP IETF Standards

STD 95 - RDAP
https://www.rfc-editor.org/refs/ref-std95.txt
https://www.rfc-editor.org/info/std95

RFC8056 – Extensible Provisioning Protocol (EPP) and Registration Data Access Protocol (RDAP) Status Mapping
https://tools.ietf.org/html/rfc8056
https://www.rfc-editor.org/info/rfc8056
Describes the mapping of the Extensible Provisioning Protocol (EPP) statuses with the statuses registered for us in the Registration Data Access Protocol (RDAP).

Registration Data Access Protocol (RDAP) Object Tagging
https://www.rfc-editor.org/info/rfc8521
Describes an update to RFC7484 by describing an operational practice that can be used to add structure to RDAP identifiers that makes it possible to identify the authoritative server for additional RDAP queries.

jCard: The JSON Format for vCard
https://tools.ietf.org/html/rfc7095
https://www.rfc-editor.org/info/rfc7095

vCard Format Specification
https://tools.ietf.org/html/rfc6350
https://www.rfc-editor.org/info/rfc6350

Appendix B: Other References

W3C.REC=cors-20140116 - Cross-Origin Resource Sharing
https://www.w3.org/TR/2014/REC-cors-20140116/
Defines a mechanism to enable client-side cross-origin requests

IANA RDAP JSON Values Registry
https://www.iana.org/assignments/rdap-json-values/rdap-json-values.xhtml
This registry defines valid values for RDAP JSON status, role, notices and remarks, event action, and domain variant relation, as defined in RFC9083.

IANA Bootstrap Service Registry for Domain Name Space
https://www.iana.org/assignments/rdap-dns/rdap-dns.xhtml

EPP Status Code (ICANN)
https://www.icann.org/epp

Draft Final Report from the Expert Working Group on Internationalized Registration Data
https://gnso.icann.org/en/issues/ird/ird-draft-final-10mar15-en.pdf

Study to Evaluate Available Solutions for the Submission and Display of Internationalized Contact Data
https://www.icann.org/en/system/files/files/transform-dnrd-02jun14-en.pdf

Mozilla Included CA Certificate List
https://wiki.mozilla.org/CA:IncludedCAs