

RDAP Response Profile

~~15 February 2019~~ [21 February 2024](#)

Version: ~~2.12.2~~

Contents

I. Introduction	2
II. Policy Mapping	2
III. Access Requirements	3
IV. Display Requirements	4
General	4
Responses to Domain name RDAP queries	5
Responses to Registrar RDAP queries	12
Responses to Nameserver RDAP queries	13
Appendix A: RDAP IETF Standards	14
Appendix B: Other Technical References	14
Appendix C: Policy References	19
Appendix D: RDS Fields (data element mappings)	20
Domain Name Responses:	30
Registrar Responses:	35
Name Server Responses:	36

I. Introduction	1
II. Policy Mapping	2
III. Response Requirements	2
1. General	2
2. Responses to Domain name RDAP queries	3
3. Responses to Registrar Entity RDAP queries	8
4. Responses to Nameserver RDAP queries	9
Appendix A: RDAP IETF Standards	10
Appendix B: Other Technical References	11
Appendix D: Data Element Mappings	13
Domain Name Responses	13
Name Server Responses	15
Appendix E: Redacted Fields in the Registration Data Access Protocol (RDAP) Response	

<u>“redacted name” JSON Values Registry Registrations</u>	<u>16</u>
<u>Appendix F: Data Element Mappings</u>	<u>21</u>
<u>Registrar Responses:</u>	<u>21</u>

I. Introduction

~~In 2012, The Internet Engineering Task Force (IETF) chartered the WEIRDS (Web Extensible Internet Registration Data Services) working group to replace the WHOIS protocol with a RESTful data service that supports internationalization, a formal data model, and differential services. This working group concluded in early 2015 with the publication of RFC7480, RFC7481, RFC7482, RFC7483, and RFC7484 that define the Registry Data Access Protocol (RDAP) as a standardized replacement for WHOIS. RDAP supports both Regional Internet Registries (RIRs) and Domain Name Registries (DNRs). Since 2015 other RDAP internet drafts and RFCs have been created including RFC8056, draft-ietf-regext-rdap-object-tag, and draft-hollenbeck-regext-rdap-openid, and draft-lozano-rdap-nameservers-sharing-name~~
[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 ~~specific to Registration Data Services (RDS) in a single document~~[for RDAP](#) which, in conjunction with the RDAP Technical Implementation Guide ~~define a domain registry RDAP implementation, 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 ~~with minimal reengineering.~~

~~Additionally, the process of creating these two documents has been memorialized in the RDAP Pilot Working Group Report, which is available for download on the page where this document is hosted. The Report contains important information about the process by which these specifications were developed including the rationale for certain decisions (both controversial and not), the consideration of public comments, input provided by ICANN Org, items where dissent was registered by participants, and areas for future consideration.~~

~~H.~~ [and the goal of minimal reengineering.](#)

II. Policy Mapping

This document specifies the RDAP Policy requirements ~~originating from the ICANN Temporary Specification for gTLD Registration Data (the “Temporary Specification”) effective 25 May 2018 which builds upon the existing legacy Whois~~[Registration Data Policy effective \(date TBD\) which builds upon existing RDDS policy and contractual](#) requirements. For clarity, the ~~Temporary Specification~~[Registration Data Policy](#) has precedence in any conflict between this document and the ~~Temporary Specification~~. ~~The following source material forms additional basis for the policy mapping~~[Registration Data Policy](#). See [Appendix C for a listing of source material](#) used to create ~~the~~[this](#) RDAP Response Profile.

~~gTLD Base Registry Agreement (RA):~~

~~<https://newgtlds.icann.org/sites/default/files/agreements/agreement-approved-31jul17-en.pdf>~~

~~2013 Registrar Accreditation Agreement~~

~~<https://www.icann.org/resources/pages/approved-with-specs-2013-09-17-en>~~

~~Additional Whois Information Policy (AWIP), <https://www.icann.org/resources/pages/policy-awip-2014-07-02-en>~~

~~Registry Registration Data Directory Services Consistent Labeling and Display Policy (CL&D), <https://www.icann.org/resources/pages/rdds-labeling-policy-2017-02-01-en>~~

~~Temporary Specification for gTLD Registration Data—~~

~~<https://www.icann.org/en/system/files/files/gtld-registration-data-temp-spec-17may18-en.pdf>~~

~~III.~~ **Access Requirements**

~~The RDAP implementation based on ICANN’s Temporary Specification assumes multiple layers of access to RDS data. Based on the policy set forth by the Temporary Specification, the current implementation provides public access to certain registration data. Access to non-public data is out of scope for the current specification.~~

~~As specified in the Temporary Specification, data from the registration data set can optionally be provided in the public layer provided that certain conditions are met (e.g., a registrant consents to full publication, or the registrant is not in the European Economic Area and the registrar optionally publishes additional data).~~

~~IV.~~ **Display**

III. Response Requirements

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. General

1.1. These requirements represent the minimum baseline for RDAP query responses. RDAP server operators MAY output additional RDDS fields, RDAP *events* or RDAP *roles* without further approval by ICANN.

~~1.1.1.2.~~ A server MUST indicate compliance with this specification by including the literal string "icann_rdap_response_profile_1" in the rdapConformance member for all responses provided by the server.

~~1.2.1.3.~~ RDAP extensions

~~1.2.1.3.1.~~ RDAP extensions to responses, if used, MUST be registered in the IANA's RDAP Extensions registry (<https://www.iana.org/assignments/rdap-extensions/rdap-extensions.xhtml>), as defined in [RFC7480](#).

~~1.2.2.~~ RDAP extensions MUST NOT add browser executable code (e.g., Javascript) to the response.

~~1.3.~~ An RDAP server that conforms to this specification MUST include the string literal "icann_rdap_response_profile" as a prefix in the "rdapConformance" member for all responses provided by the server and the suffix of "0", concatenated according to [\[RFC7483\]](#) section 4.1. For clarity, conformance to the current document MUST be noted with a value of "icann_rdap_response_profile_0". Note: at the time of publication, "icann_rdap_response_profile" is pending registration in the IANA RDAP Extensions Registry.

~~1.4.~~ ISO 3166-1 alpha 2

~~1.4.1.~~ If the ISO-3166-1-alpha-2 parameter has been published in the vCard Properties registry defined in Section 10.3.1 of [RFC 6350 \[RFC6350\]](#), then the country name parameter of the *adr* structure in an entity object MUST be empty and the ISO-3166-1-alpha-2 property MUST be populated according to the specification

1.3.2. If the ISO-3166-1-alpha-2 parameter has not been published in the vCard Properties registry defined in Section 10.3.1 of [RFC 6350 \[RFC6350\]](#),

~~then the country name parameter of the *adr* structure in an entity object MAY be populated with data if that data is an ISO 3166 country code~~

1.4. In an entity object with an *adr* structure, the *country name* property MUST be empty and the *cc* property MUST be populated by a value from ISO 3166-1 alpha-2.

1.5. The topmost object in the RDAP response MUST contain an event of *eventAction type last update of RDAP database* with a value equal to the timestamp when the RDAP database was last updated.

1.6. Contact representation

1.6.1. An RDAP server MUST use jCard [RFC7095] to represent contact information absent an indication from the RDAP client preferring an alternative format in the RDAP response.

1.6.2. An RDAP server MAY use alternative formats, approved as “Proposed Standard” (as opposed to Informational or Experimental specifications) through the applicable IETF processes, in addition to jCard, to represent contact information in the RDAP response.

~~4.4.2.~~1.6.3. If a mechanism exists to allow the RDAP client to request a preferred format to represent contact information and the RDAP client indicates its preference for an alternative format, the RDAP server MUST use one of the following to represent contact information: 1) jCard, 2) the requested alternative format, or 3) both.

2. Responses to Domain name RDAP queries

2.1. Domain Name - In response to a non-IDN domain query, the returned RDAP response MUST include a *domain* object and contain a *ldhName* member. In response to an IDN domain query, if the queried domain is an A-label, then the returned RDAP response MUST include a domain object and MUST contain an *ldhName* member and MAY contain a *unicodeName* member. In response to an IDN domain query, if the queried domain is an U-label, then the returned RDAP response MUST include a domain object and MUST contain an *unicodeName* member and MAY contain a *ldhName* member.

2.2. Registry Domain ID - The *domain* object *handle* in the RDAP response MUST contain the Repository Object Identifier (ROID of the domain object, `<domain:roid>` as defined in ~~RFC5731~~ [RFC5731](#)) for the domain name object.

2.3. Event ~~Actions (Updated, Creation, Registry Expiry, Registrar Registration Expiration, Transfer dates)~~ and Event Actions - This section describes Events and Event Actions related to Domain name responses

2.3.1. The domain object in the RDAP response MUST contain the following events:

2.3.1.1. Event of *eventAction* type *registration*

2.3.1.2. Event of *eventAction* type *expiration*

~~Event of *eventAction* type *last update of RDAP database with a value equal to the timestamp when the RDAP database was last updated*~~

2.3.2. The domain object in the RDAP response MAY contain the following events:

2.3.2.1. An event of *eventAction* type *registrar expiration*.

2.3.2.2. Event of *eventAction* type *last changed* - The event of *eventAction* type *last changed* MUST be omitted [from the response](#) if the domain [nameobject](#) has not been updated since it was created.

2.3.2.3. An event of *eventAction* type *transfer*, with the last date and time that the domain was transferred ([for clarity: change of sponsoring registrar](#)). The event of *eventAction* type *transfer* MUST be omitted [from the response](#) if the domain [nameobject](#) has not been transferred since it was created.

2.4. Registrar (Registrar Entity)

2.4.1. Registrar - The *domain* object in the RDAP response MUST contain an *entity* with the *registrar* role (called registrar entity in this section) and a valid *fn* member MUST be present.

2.4.2. Registrar IANA ID - The *handle* of the *entity* MUST be equal to the IANA Registrar ID.

2.4.3. Registrar IANA ID - The *entity* with the *registrar* role in the RDAP response MUST contain a *publicIDs* member [~~RFC7483~~[RFC9083](#)] to identify the IANA Registrar ID from the IANA's Registrar ID registry (<https://www.iana.org/assignments/registrar-ids/registrar-ids.xhtml>). The type value of the *publicID* object MUST be equal to IANA Registrar ID.

2.4.4. Other members MAY be present in the *entity* (as specified in ~~RFC6350~~ [RFC6350](#), the vCard Format Specification and its corresponding JSON mapping ~~RFC7095~~ [RFC7095](#)).

2.4.5. Abuse Contact (email, phone) - An RDAP server MUST include an *entity* with the *abuse* role within the registrar *entity* which MUST include *tel* and *email* members, and MAY include other members.

2.4.6. Registrar URL - The entity with the *registrar* role in the RDAP response MUST contain a *links* member [RFC9083]. The *links* object MUST contain the elements: *value*, identical to the the RDAP Base URL for the Registrar as provided in the IANA “Registrar IDs” registry (i.e., <https://www.iana.org/assignments/registrar-ids>); *rel:about*, and *href* containing the Registrar URL. Note: in cases where the Registry Operator acts as sponsoring Registrar (e.g., IANA Registrar ID 9999), the *href* shall contain a URL from the Registry.

2.5. Reseller - The returned *domain* object in the RDAP response MAY contain an entity with the *reseller* role, if the domain name was registered through a reseller.

2.6. Domain Status

2.6.1. The top-level domain object in the RDAP response MUST contain at least one *status* member [~~RFC7483~~RFC9083] contained in the IANA RDAP JSON Values registry (<https://www.iana.org/assignments/rdap-json-values/rdap-json-values.xhtml>) of *status* type.

2.6.2. The *status* member value MUST conform to the Extensible Provisioning Protocol (EPP) and Registration Data Access Protocol (RDAP) Status Mapping [~~RFC8056~~RFC8056].

~~2.6.3. A domain name RDAP response MUST contain a *notices* member with a *title* “Status Codes”, a *description* containing the string “For more information on domain status codes, please visit <https://icann.org/epp>” and a *links* member with the <https://icann.org/epp> URL. Note: at the time of publication, the aforementioned URL is expected to be updated by ICANN to include information on RDAP status codes.~~

~~2.7. Contacts~~

~~2.7.1. If the RDAP service is provided by a registry that does not support contacts (for example thin registries), then the contact entities described in this section are not REQUIRED.~~

~~2.7.2. Processing where subject to the GDPR is defined in the Temporary Specification Appendix A Section 2 and processing where not subject~~

to the GDPR is defined in the Temporary Specification Appendix B—Section 3.

~~2.7.3. Registrant, Administrative, Technical, Other—Subject to the Temporary Specification, the domain object in the RDAP response MUST contain entities with the registrant, administrative and technical roles and MAY contain other entities with corresponding roles (such as billing) with a handle (ROID of the contact object, <contact:roid>, as defined in RFC5733) and valid members fn, adr, tel, email (as specified in RFC6350, the vCard Format Specification and its corresponding JSON mapping RFC7095).~~

~~2.7.3.1. The following RDDS fields MUST be included in the RDAP response: Street, City, Country (complying with section 1.4 above).~~

2.6.3. The in the *href, rel:glossary*, and a *value* with the RDAP lookup path that generated the RDAP response.

2.7. Contacts

2.7.1. The *domain* object in the RDAP response MAY contain entity instances, each of which includes a role signifying the relationship to the *domain* object.

2.7.2. The RDAP response by a registrar server to a domain query MUST contain an *entity* with the *registrant* role.

2.7.3. Except for the *entity* that includes the *registrar* role, the handle of an *entity* MUST be the ROID of the contact object, <contact:roid>, as defined in RFC5733.

~~2.7.3.2. The entity that includes the registrant role MAY include valid members *fn, org, adr, tel, and email*. Subject to the redaction requirements below, the following RDDS ~~fields~~elements MUST be included in the RDAP response if the data exists: Organization, State/Province, Postal Code, Phone Ext, Fax, Fax Ext. If no data exists, the fields SHOULD NOT be included in the RDAP response.~~

~~2.7.4. Redaction~~

~~2.7.4.1. Registrant—Where processing is subject to the GDPR, the following MUST be omitted unless consent to publish has been provided and where processing is not subject to the GDPR MAY be omitted—the *handle, fn* and *tel* members of the (registrant)~~

contact entity and the Street, City, Postal Code, Phone Ext, Fax and Fax Ext fields of the *adr* member in the RDAP response.

~~2.7.4.2. Administrative, Technical, Other—Where processing is subject to the GDPR, the following MUST be omitted unless consent to publish has been provided and where processing is not subject to the GDPR MAY be omitted—the *handle*, *fn* and *tel* members of the (administrative, technical, other) contact entity and the Organization, Street, City, State/Province, Postal Code, Country, Phone Ext, Fax and Fax Ext fields of the *adr* member in the RDAP response (complying with section 1.4 above).~~

~~2.7.4.3. In an RDAP response where elements of the contact entity have been omitted for privacy considerations, the contact entity MUST include a remarks element containing a *title* member with a value substantially similar to “REDACTED FOR PRIVACY” and a description member with a value “Some of the data in this object has been removed” and a *type* member with a value “object redacted due to authorization”.~~

~~2.7.5. Email—~~

~~2.7.5.1. The *email* property MUST be omitted~~

~~2.7.5.2. Email (Registrar Only)—the value of the CONTACT-URI member in the entity object of the RDAP response MUST be an email address or link to a web form to facilitate email communication with the Registrant but MUST NOT identify the contact email address or the contact itself. Note: at the time of publication, the CONTACT-URI property is pending registration in the IANA vCard Elements Registry~~

~~2.7.5.3. Email (Registry Only)—The registry MUST include a *remarks* element containing a *title* member with a value “EMAIL-REDACTED FOR PRIVACY”, a *description* member with a value substantially similar to the following “Please query the RDDS service of the Registrar of Record identified in this output for information on how to contact the Registrant of the queried domain name” and a *type* member with a value “object redacted due to authorization”.~~

2.7.4. The RDAP server response to a Contact query MUST include an eventAction type “*last update of RDAP database*” with a value equal to the timestamp when the RDAP data source was last updated [fn, org, adr, tel, and email members based on the mappings defined in Appendix D](#) when section 9.1 of the Registration Data Policy calls for the Publication of the RDDS element:

2.7.4.1. Registrant Name

2.7.4.2. Registrant Organization

2.7.4.3. Registrant Street

- [2.7.4.4. Registrant City](#)
- [2.7.4.5. Registrant State/Province](#)
- [2.7.4.6. Registrant Postal Code](#)
- [2.7.4.7. Registrant Country](#)
- [2.7.4.8. Registrant Phone](#)
- [2.7.4.9. Registrant Email](#)

[2.7.5. Subject to the redaction requirements below, the *entity* that includes the *registrant* role MAY include the following RDDS data elements:](#)

- [2.7.5.1. Registrant Phone Ext](#)
- [2.7.5.2. Registrant Fax](#)
- [2.7.5.3. Registrant Fax Ext](#)

[2.7.6. The entity that includes the *technical* role MAY include valid members *fn*, *tel*, and *email*. Subject to the redaction requirements below, the following RDDS elements MUST be included in the *fn*, *tel*, and *email* members based on the mappings defined in Appendix D when section 9.1 of the Registration Data Policy calls for the Publication of the RDDS element:](#)

- [2.7.6.1. Tech Name](#)
- [2.7.6.2. Tech Phone](#)
- [2.7.6.3. Tech Email](#)

[2.7.7. When applying the redaction requirements in the Registration Data Policy, the redacted RDDS elements MUST be indicated using the Redacted Fields in the Registration Data Access Protocol \(RDAP\) Response \(draft-ietf-regext-rdap-redacted, as defined by the Technical Implementation Guide 1.1.3\) and using the redacted "name" member values included in Appendix E.](#)

[2.7.8. When applying the redaction requirements for an email address in the Registration Data Policy, the Registrar \(1\) MUST use the Redaction by Replacement Value Method in the Registration Data Access Protocol \(RDAP\) Response \(draft-ietf-regext-rdap-redacted, as defined by the Technical Implementation Guide 1.1.3\) for the email property, and \(2\) MUST comply with one of the following:](#)

- [2.7.8.1. The email property MUST contain a syntactically valid email address as defined in RFC5322, and\) the entity object MUST NOT contain a *contact-uri* member.](#)
- [2.7.8.2. The value of the *contact-uri* member in the entity object MUST be a syntactically valid HTTP URL as defined in RFC9110, and\) the entity object MUST NOT contain an *email* member.](#)

~~2.2.8.2.7.9.~~ Notwithstanding [the](#) redaction requirements ~~stated in Section 2.7, registries and registrars MAY provide unredacted registration data as described in Temporary Specification for gTLD Registration Data, Appendix A, section 4 via RDAP~~ [section 2.7.7 and 2.7.8 where the Registered Name Holder provides its consent to publish data that would otherwise be redacted, Registrar MUST NOT redact such RDDS data elements.](#)

~~2.2.2.8.~~ Name Server(s) - The domain object in the RDAP response MUST contain the name servers of the domain in the *nameservers* member.

~~2.2.9.2.8.1.~~ Each nameserver object MUST contain the following member: *ldhName*.

~~2.2.9.2.8.2.~~ The following members are Optional: *ipAddresses* [~~RFC7483~~[RFC9083](#)], *unicodeName*, *handle* [~~RFC7483~~[RFC9083](#)] (ROID of the host object, <host:roid> as defined in RFC5732), and *status*.

~~2.2.9.2.8.3.~~ ~~In the case of a TLD~~ [This section 2.8.3 only applies to Registries](#) in which name servers are specified as domain attributes, ~~the~~ [The](#) nameserver object MUST NOT contain the following members: *handle* and *status*, but the nameserver object MUST contain an *ipAddresses* member listing all IPv4 and IPv6 glue records for the ~~nameserver~~ [in-domain name server \(see RFC 8499\)](#).

~~2.7.2.9.~~ DNSSEC - The domain object in the RDAP response MUST contain a *secureDNS* member [~~RFC7483~~[RFC9083](#)] including at least a *delegationSigned* element. Other elements (e.g. *dsData*) of the *secureDNS* member MUST be included, if the domain name is signed and the elements are stored in the Registry or Registrar database, as the case may be.

~~2.10.~~ RDDS Inaccuracy - A domain name RDAP response MUST contain a *notices* member with a title "RDDS Inaccuracy Complaint Form", a description containing the string "URL of the ICANN RDDS Inaccuracy Complaint Form: <https://icann.org/wicf>" and a *links* member with the <https://icann.org/wicf> URL.

~~2.10.~~ [- in the href, rel:help, and a value with the RDAP lookup path that generated the RDAP response.](#)

~~2.8.2.11.~~ Registrar-only requirements - the following requirements apply to registrars only.

~~2.8.1.2.11.1.~~ A Registrar [RDAP service](#) MUST return an HTTP 404 response to a domain name request when the Registrar is not the Sponsoring Registrar for the domain name.

~~2.8.2~~2.11.2. The domain object *handle member* in the RDAP response MUST contain the Repository Object Identifier (ROID of the domain object, <domain:roid> as defined in RFC5731) for the Domain Name object. For example, a Registrar could obtain the ROID from the Registry via EPP and cache the information locally after creating or gaining a domain name via a transfer.

~~2.8.3~~2.11.3. The entity *handle member* in the RDAP response MUST contain the Repository Object Identifier (ROID of the contact object, <contact:roid>, as defined in RFC5733) for the Contact object. For example, a Registrar could obtain the ROID from the Registry via EPP and cache the information locally. The RAA 2013 defines that this information MUST be shown if available from the Registry. If this information is not available from the Registry (e.g., a "thin" Registry), the handle MUST contain the contact object's unique identifier within the Registrar.

~~2.8.4~~2.11.4. The *eventAction* type *last changed* MUST reflect the date and time of the latest successful update known to the Registrar. Registrars are not required to constantly refresh this date from the Registry.

~~2.8.5~~2.11.5. The *status* element MUST reflect the latest known set of statuses in the Registry. Registrars are not required to constantly refresh the statuses from the Registry.

3. Responses to Registrar Entity RDAP queries

This section only applies to Registries

3.1. Registrar (name, address, phone number, email) - In response to a query for a registrar ~~queries~~entity, the returned RDAP response MUST be an *entity* with *registrar* role, with a *handle* and valid elements *fn*, *adr*, *tel*, *email*.

3.1.1. Registrar (Street, City, Country) - The *adr* member in the RDAP response for a Registrar query MUST at least contain the following RDDS fields: Street, City, Country.

3.1.2. Registrar (State/Province, Postal Code, Fax Number) - the following fields are optional in the *adr* member of the RDAP response: State/Province, Postal Code, Fax Number.

3.2. Contacts (Admin, Technical) - The RDAP response SHOULD contain at least two ~~entities~~entity objects, with the *administrative* and *technical* roles respectively

within the *entity* with the *registrar* role. The *entities* with the *administrative* and *technical* roles MUST contain valid *fn*, *tel*, *email* members, and MAY contain a *handle* and a valid *adr* element.

~~3.3. The RDAP response to a Registrar query MUST include an *eventAction* type *last update of RDAP database* with a value equal to the timestamp when the RDAP database was last updated. Appendix F contains non-normative information to assist in the implementation of this section.~~

4. Responses to Nameserver RDAP queries

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

4.1. Name Server (Name) - In response to a non-IDN Nameserver query the returned RDAP response MUST include a *nameserver* object and contain a *ldhName* member. In response to an IDN Nameserver query, if the queried Nameserver is an A-label, then the returned RDAP response MUST include a *nameserver* object and MUST contain an *ldhName* member and MAY contain a *unicodeName* member. In response to an IDN Nameserver query, if the queried Nameserver is an U-label, then the returned RDAP response MUST include a *nameserver* object and MUST contain an *unicodeName* member and MAY contain a *ldhName* member.

4.2. IP Address(es) - If the name server record includes IP addresses then the *nameserver* object MUST contain an *ipAddresses* member listing all IPv4 and IPv6 glue records for the *in-domain* Nameserver.

4.3. Registrar (Name, IANA ID) - The Registrar RDDS field is Optional; if present in the response, it MUST be represented as an entity with the registrar role. The *handle* of the entity with the registrar role MUST be equal to the IANA Registrar ID. If the Registrar does not have an IANA ID then the *handle* of the entity with the registrar role MUST equal "not applicable". If the Registrar has an IANA ID, then the entity with the registrar role in the RDAP response MUST contain a *publicIDs* member with a *type* value equal to the IANA Registrar ID. If the Registrar does not have an IANA ID then the RDAP response MUST NOT contain a *publicIDs* member.

~~4.4. The RDAP response to a Name Server query MUST include an *eventAction* type *last update of RDAP database* with a value equal to the timestamp when the RDAP database was last updated.~~

~~Appendix A: RDAP IETF Standards~~

~~RDAP standards are a set of specifications, which together provide a complete RDAP service. Each specification is briefly described below.~~

~~RFC7480—HTTP Usage in the Registration Data Access Protocol (RDAP)~~

~~<https://www.rfc-editor.org/rfc/rfc7480.txt>~~

~~Describes usage of HTTP transport for RDAP, error messages, RDAP extensions, rate limiting and internationalization with URIs.~~

~~RFC7481—Security Services for the Registration Data Access Protocol (RDAP)~~

~~<https://www.rfc-editor.org/rfc/rfc7481.txt>~~

~~Covers access control, authentication, authorization, privacy, data confidentiality and RDAP services availability considerations.~~

~~RFC7482—Registration Data Access Protocol (RDAP) Query Format~~

~~<https://www.rfc-editor.org/rfc/rfc7482.txt>~~

~~Defines the URL patterns for networks, autonomous systems, reverse DNS, name servers, registrars and entities queries. Also covers help requests, search (wildcards) and internationalization in requests.~~

~~RFC7483—JSON Responses for the Registration Data Access Protocol (RDAP)~~

~~<https://www.rfc-editor.org/rfc/rfc7483.txt>~~

~~Defines JSON object classes for domains, name servers, entities, IP networks and autonomous system numbers. Describe answers to help queries, searches, JSON-embedded error codes and truncated answers.~~

~~RFC7484—Finding the Authoritative Registration Data (RDAP) Service~~

~~<https://www.rfc-editor.org/rfc/rfc7484.txt>~~

~~Describes a method to find the authoritative server for RDAP data.~~

~~Appendix B: Other Technical References~~

~~RFC7485—Inventory and Analysis of WHOIS Registration Objects~~

~~<https://www.rfc-editor.org/rfc/rfc7485.txt>~~

~~RFC8056—Extensible Provisioning Protocol (EPP) and Registration Data Access Protocol (RDAP) Status Mapping~~

~~<https://tools.ietf.org/html/rfc8056>~~

~~Describes the mapping of the Extensible Provisioning Protocol (EPP) statuses with the statuses registered for us in the Registration Data Access Protocol (RDAP).~~

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.ietf.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).

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>

EPP Status Code (ICANN)

<https://www.icann.org/epp>

Appendix B: Other Technical References

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 [RFC7483](#)[RFC9083](#).

IANA Bootstrap Service Registry for Domain Name Space

<https://www.iana.org/assignments/rdap-dns/rdap-dns.xhtml>

~~draft-lozano-[rdap-nameservers-sharing-name](#)—Nameserver objects sharing the same name, support for the Registration Data Access Protocol (RDAP)~~

~~[rdap-nameservers-sharing-name](https://tools.ietf.org/html/draft-lozano-https://tools.ietf.org/html/draft-lozano-[rdap-nameservers-sharing-name](#)~~

~~Describes a Registration Data Access Protocol (RDAP) extension that may be used to retrieve the registration information of a particular nameserver object sharing the name with other nameserver objects.~~

~~draft-ietf-[regext-\[rdap-object-tag\]\(#\)](#)—Registration Data Access Protocol (RDAP) Object Tagging~~

~~[regext-\[rdap-object-tag\]\(#\)">https://tools.ietf.org/html/draft-ietf-\[regext-\\[rdap-object-tag\\]\\(#\\)\]\(#\)](https://tools.ietf.org/html/draft-ietf-<a href=)~~

~~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.~~

~~Federated Authentication for the Registration Data Access Protocol (RDAP) using OpenID~~

~~Connect [regext-\[rdap-openid\]\(#\)">https://tools.ietf.org/html/draft-hollenbeck-\[regext-\\[rdap-openid\\]\\(#\\)\]\(#\)](https://tools.ietf.org/html/draft-hollenbeck-<a href=)~~

~~Describes a federated authentication system for RDAP based on OpenID Connect.~~

jCard: The JSON Format for vCard

<https://tools.ietf.org/html/rfc7095>

vCard Format Specification

<https://tools.ietf.org/html/rfc6350>

EPP Status Code (ICANN)

<https://www.icann.org/epp>

Draft Final Report from the Expert Working Group on Internationalized Registration Data

<https://gns0.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>

[Redacted Fields in the Registration Data Access Protocol \(RDAP\) Response](https://datatracker.ietf.org/doc/html/draft-ietf-regext-rdap-redacted-16)
<https://datatracker.ietf.org/doc/html/draft-ietf-regext-rdap-redacted-16>
Describes an RDAP extension for explicitly identifying redacted RDAP response fields, using JSONPath as the default expression language.

Appendix C: Policy References

gTLD Base Registry Agreement

<https://newgtlds.icann.org/sites/default/files/agreements/agreement-approved-09jan14-en.htm>

2013 Registrar Accreditation Agreement

<https://www.icann.org/resources/pages/approved-with-specs-2013-09-17-en>

Registry Registration Data Directory Services Consistent Labeling and Display Policy (CL&D),

<https://www.icann.org/resources/pages/rdds-labeling-policy-2017-02-01-en>

Temporary Specification for gTLD Registration Data –

<https://www.icann.org/en/system/files/files/gtld-registration-data-temp-spec-17may18-en.pdf>

ICANN Advisories

<https://www.icann.org/resources/pages/advisories-2012-02-25-en>

Advisory: Clarifications to the Registry Agreement, and the 2013 Registrar Accreditation Agreement (RAA) regarding applicable Registration Data Directory Service (Whois) Specifications (RDDS clarification Advisory)

<https://www.icann.org/resources/pages/registry-agreement-raa-rdds-2015-04-27-en>

Advisory: Registrar Implementation of the 2013 RAA's Whois Requirements

<https://www.icann.org/news/announcement-2013-07-31-en>

ICANN Consensus Policies

<https://www.icann.org/resources/pages/registrars/consensus-policies-en>

Additional Whois Information Policy

<https://www.icann.org/resources/pages/policy-awip-2014-07-02-en>

Final Report on the Thick Whois Policy Development Process

<https://gns0.icann.org/en/issues/whois/thick-final-21oct13-en.pdf>

ICANN Whois Marketing Restriction Policy

<https://www.icann.org/resources/pages/registrars/consensus-policies/wmrp-en>

~~Appendix D: RDDS Fields (data element mappings)~~

~~In the tables below, all elements described as being an “RDDS Field” make reference to the output fields as defined in Specification 4 of the Base Registry Agreement, as amended from time to time. The inclusion of a field in this appendix defines the mapping of the RDDS field to an RDAP response element. Inclusion of a field~~[Registration Data Policy \(EPDP Phase 1\)](https://www.icann.org/resources/pages/registration-data-policy-implementation-resources-2024-02-21-en)
~~<https://www.icann.org/resources/pages/registration-data-policy-implementation-resources-2024-02-21-en>~~

Appendix D: Data Element Mappings

The tables below show the mapping between the data elements in the registration data policy and the RDAP response element. Including a data element in this appendix does not imply its inclusion in any RDAP response.

Domain Name Responses

<u>Data Element</u>	<u>RDAP Response Element</u>
<u>Domain Name</u>	<u>ldhName</u>
<u>Registry Domain ID</u>	<u>handle</u>
<u>Updated Date</u>	<u>events.eventAction "last changed"</u>
<u>Creation Date</u>	<u>events.eventAction "registration"</u>
<u>Registry Expiry Date</u>	<u>events.eventAction "expiration"</u>
<u>Domain Status</u>	<u>status object</u>
<u>Name Server</u>	<u>nameservers.lhname</u>
<u>DNSSEC Elements</u>	<u>secureDNS object</u>
<u>Last update of RDDS</u>	<u>Events.eventAction "last update of RDAP database"</u>
<u>Registrar</u>	<u>Entities.role registrar</u>
<u>Registrar</u>	<u>Entities.roles.registrar</u>

Registrar IANA ID	publicIDs.identifier
Registrar Abuse Contact Email	Entities.role abuse email
Registrar Abuse Contact Phone	Entities.role abuse phone
Registrar Registration Expiration Date	events.eventAction "registrar expiration"
Reseller	Entities.roles reseller
Registrant	Entities.role registrant
Registry Registrant ID	Entity.handle
Registrant Name	jCard "fn"
Registrant Organization	Org
Registrant Street	Grouped into adr member while complying with section 1.4 above
Registrant City	
Registrant State/Province	
Registrant Postal Code	
Registrant Country	
Registrant Phone	Tel type parameter voice

Registrant Phone Ext	Ext
Registrant Fax	Tel type parameter Fax
Registrant Fax Ext	Ext
Registrant Email	Email
Technical Contact	Entities.role Technical
Registry Tech ID	Entity.handle
Tech Name	jCard "fn"
Tech Phone	Tel type parameter voice
Tech Email	Email

Name Server Responses

<u>Data Element</u>	<u>RDAP Response Element</u>
Name Server(s)	nameserver.ldhName
Name Server IP Address(es)	nameserver.ipAddresses
Registrar	Entities.roles registrar
Last Update of RDDS	events.eventAction "last update of RDAP database"

|

Appendix E: Redacted Fields in the Registration Data Access Protocol (RDAP) Response “redacted name” JSON Values Registry Registrations

After the Redacted Fields in the Registration Data Access Protocol (RDAP) Response draft (draft-ietf-regext-rdap-redacted) becomes an RFC, the following is the list of “redacted name” registrations for use in redacting the fields defined in the gTLD RDAP Profile.

Value: Registry Domain ID

Type: redacted name

Description: Redacted domain object class “handle” member. The “removal” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.handle”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Registry Registrant ID

Type: redacted name

Description: Redacted entity object class, with “registrant” role, “handle” member. The “removal” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.entities[?(@.roles[0]==‘registrant’)].handle”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Registrant Name

Type: redacted name

Description: Redacted entity object class, with “registrant” role, name property. When using jCard, redacting the “vcard” “fn” property. The “emptyValue” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.entities[?(@.roles[0]==‘registrant’)].vcardArray[1][?(@[0]==‘fn’)][3]”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Registrant Organization

Type: redacted name

Description: Redacted entity object class, with “registrant” role, organization property. When using jCard, redacting the “vcard” “org” property. The “removal” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.entities[?(@.roles[0]==‘registrant’)].vcardArray[1][?(@[0]==‘org’)]”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Registrant Street

Type: redacted name

Description: Redacted entity object class, with “registrant” role, street properties. When using jCard, the “vcard” “adr” street properties (vCard “ADR-component-pobox”, “ADR-component-ext”, and “ADR-component-street”). The “emptyValue” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.entities[?(@.roles[0]==‘registrant’)].vcardArray[1][?(@[0]==‘adr’)][3][:3]”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Registrant City

Type: redacted name

Description: Redacted entity object class, with “registrant” role, city property. When using jCard, the “vcard” “adr” locality property (vCard “ADR-component-locality”). The “emptyValue” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.entities[?(@.roles[0]==‘registrant’)].vcardArray[1][?(@[0]==‘adr’)][3][3]”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Registrant Postal Code

Type: redacted name

Description: Redacted entity object class, with “registrant” role, postal code property. When using jCard, the “vcard” “adr” code property (vCard “ADR-component-code”). The “emptyValue” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.entities[?(@.roles[0]==‘registrant’)].vcardArray[1][?(@[0]==‘adr’)][3][5]”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Registrant Phone

Type: redacted name

Description: Redacted entity object class, with “registrant” role, voice phone property. When using jCard, the “vcard” “tel” property with type “voice”. The “removal” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.entities[?(@.roles[0]==‘registrant’)].vcardArray[1][?(@[1].type==‘voice’)]”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Registrant Phone Ext

Type: redacted name

Description: Redacted entity object class, with “registrant” role, voice phone extension property. When using jCard, the “vcard” “tel” property extension value with type “voice”. The “removal” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is

“\$.entities[?(@.roles[0]==‘registrant’)].vcardArray[1][?(@[1].type==‘voice’)]”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Registrant Fax

Type: redacted name

Description: Redacted entity object class, with “registrant” role, fax phone property. When using jCard, the “vcard” “tel” property with type “fax”. The “removal” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is

“\$.entities[?(@.roles[0]==‘registrant’)].vcardArray[1][?(@[1].type==‘fax’)]”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Registrant Fax Ext

Type: redacted name

Description: Redacted entity object class, with “registrant” role, fax phone extension property. When using jCard, the “vcard” “tel” property with type “fax”. The “removal” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is

“\$.entities[?(@.roles[0]==‘registrant’)].vcardArray[1][?(@[1].type==‘fax’)]”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Registrant Email

Type: redacted name

Description: Redacted entity object class, with “registrant” role, email property. When using jCard, the “vcard” “email” property. The “removal” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is

“\$.entities[?(@.roles[0]==‘registrant’)].vcardArray[1][?(@[0]==‘email’)]”. The “replacementValue”

redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is

“\$.entities[?(@.roles[0]==‘registrant’)].vcardArray[1][?(@[0]==‘email’)][3]” or using the

“replacementPath” to the “vcard” “contact-uri” property.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Registry Tech ID

Type: redacted name

Description: Redacted entity object class, with “technical” role, “handle” member. The “removal” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is

“\$.entities[?(@.roles[0]==‘technical’)].handle”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Tech Name

Type: redacted name

Description: Redacted entity object class, with “technical” role, name property. When using jCard, redacting the “vcard” “fn” property. The “emptyValue” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.entities[?(@.roles[0]==technical)].vcardArray[1][?(@[0]=='fn')][3]”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Tech Phone

Type: redacted name

Description: Radacted entity object class, with “technical” role, voice phone property. When using jCard, the “vcard” “tel” property with type “voice”. The “removal” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.entities[?(@.roles[0]=='technical)].vcardArray[1][?(@[1].type=='voice')]”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Tech Phone Ext

Type: redacted name

Description: Redacted entity object class, with “technical” role, voice phone extension property. When using jCard, the “vcard” “tel” property extension value with type “voice”. The “removal” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.entities[?(@.roles[0]=='technical)].vcardArray[1][?(@[1].type=='voice')]”.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org

Value: Tech Email

Type: redacted name

Description: Redacted entity object class, with “technical” role, email property. When using jCard, the “vcard” “email” property. The “removal” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.entities[?(@.roles[0]=='technical)].vcardArray[1][?(@[0]=='email')]”. The “replacementValue” redacted “path” member JSONPath for Figure “Unredacted RDAP Lookup Response” of draft-ietf-regext-rdap-redacted is “\$.entities[?(@.roles[0]=='technical)].vcardArray[1][?(@[0]=='email')][3]” or using the “replacementPath” to the “vcard” “contact-uri” property.

Registrant Name: ICANN

Registrant Contact Information: globalsupport@icann.org]

|

Appendix F: Data Element Mappings

The tables below show the mapping between data elements and the RDAP response element. Including a data element in this appendix does not necessarily imply its inclusion in any RDAP response.

Domain Name Responses:

Registrar Responses:

RDDS Field	RDAP Response Element
Domain Name	ldhName
Domain ID	handle
Updated Date	events.eventAction "last changed"
Creation Date	events.eventAction "registration"
Registry Expire Date	events.eventAction "expiration"
Domain Status	status object
Name Server	nameservers.lidhname
DNSSEC	secureDNS object
Internationalized Domain Name	unicodeName
Last update of WHOIS Database	Events.eventAction "last update of RDAP database"

Registrar	Entities.role-registrar
Sponsoring-Registrar	Entities.roles-registrar
Sponsoring-Registrar IANA-ID	publicIDs.identifier
Registrar-Abuse-Contact Email	Entities.role-abuse-email
Registrar-Abuse-Contact Phone	Entities.role-abuse-phone
Registrar-Registration Expiration-Date	events.eventAction "registrar-expiration"
Registrar-RDS-Server	Links.object-with-rel:related
Reseller	Entities.roles-reseller
Registrant	Entities.role-registrant
Registrant-ID	Entity.handle
Registrant-Name	jCard "fn"
Registrant-Organization	Org
Registrant-Street	Grouped into adr member while complying with section 1.4 above
Registrant-City	
Registrant-State/Province	

Registrant Postal Code	
Registrant Country	
Registrant Phone Number	Tel type parameter voice
Registrant Phone Number Ext	Ext
Registrant Fax	Tel type parameter Fax
Registrant Fax Ext	Ext
Registrant email	Email
Admin Contact	Entities.role Administrative
Admin ID	Entity.handle
Admin Name	jCard "fn"
Admin Organization	Org
Admin Street	Grouped into adr member, while complying with section 1.4 above
Admin City	
Admin State/Province	
Admin Postal Code	
Admin Country	
Admin Phone Number	Tel type parameter voice

Admin-Phone-Number-Ext	Ext
Admin-Fax	Tel type parameter Fax
Admin-Fax-Ext	Ext
Admin-email	Email
Technical-Contact	Entities.role-Technical
Tech-ID	Entity.handle
Tech-Name	jCard "fn"
Tech-Organization	Org
Tech-Street	Grouped into adr member, while complying with section 1.4 above
Tech-City	
Tech-State/Province	
Tech-Postal-Code	
Tech-Country	
Tech-Phone-Number	Tel type parameter voice
Tech-Phone-Number-Ext	Ext
Tech-Fax	Tel type parameter Fax
Tech-Fax-Ext	Ext

Tech-email

Email

Data Element

RDAP Response Element

Registrar	jCard fn
Registrar Street	Grouped into the adr member, while complying with section 1.4 above
Registrar City	
Registrar State/Province	
Registrar Postal Code	
Registrar Country	
Registrar Phone	Tel with a type parameter voice
Registrar Fax	Tel with a type parameter fax
Registrar Email	email
Registrar Admin/Tech Contact	Entity.role administrative or technical
Administrative/Technical Contact	jCard fn
Contact Phone	Tel with a type parameter voice
Contact Fax	Tel with a type parameter fax
Contact Email	email

Registrar Responses:

RDDS Field

RDAP Response Element

RDDS Field	RDAP Response Element
Registrar	jCard fn
Registrar Street	Grouped into the adr member, while complying with section 1.4 above
Registrar City	
Registrar State/Province	
Registrar Postal Code	
Registrar Country	
Registrar Phone	Tel with a type parameter voice
Registrar Fax	Tel with a type parameter fax
Registrar Email	email
Registrar admin/tech contact	Entity.role administrative or technical
administrative/technical contact	jCard fn
Contact Phone Number	Tel with a type parameter voice
Contact Fax Number	Tel with a type parameter fax
Contact Email	email
WHOIS Server /Referral URL	n/a

Name Server Responses:

RDDS Field

RDAP Response Element

RDDS Field	RDAP Response Element
Server Name	nameserver.IdhName
IP Address	nameserver.ipAddresses
Registrar	Entities.roles-registrar
WHOIS Server /Referral URL	n/a
Last update of WHOIS database	events.eventAction "last update of RDAP database"