

Registration Protocols Extensions
Internet-Draft
Intended status: Standards Track
Expires: 3 November 2022

M. Loffredo
M. Martinelli
IIT-CNR/Registro.it
2 May 2022

Registration Data Access Protocol (RDAP) Reverse search capabilities
draft-ietf-regext-rdap-reverse-search-11

Abstract

The Registration Data Access Protocol (RDAP) does not include query capabilities for finding the list of domains related to a set of entities matching a given search pattern. In the RDAP context, an entity can be associated with any defined object class. Moreover, other relationships between object classes exist and might be used for providing a reverse search capability. Therefore, a reverse search can be applied to other use cases than the classic domain-entity scenario. This document describes an RDAP extension that allow servers to provide a reverse search feature based on the relationship defined in RDAP between an object class for search and any related object class. The reverse search based on the domain-entity relationship is treated as a particular case.

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 <https://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 3 November 2022.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Revised BSD License.

Table of Contents

1.	Introduction	2
1.1.	Conventions Used in This Document	4
2.	RDAP Path Segment Specification	4
3.	RDAP Response Specification	5
4.	Reverse Searches Based on Entity Details	5
5.	RDAP Conformance	6
6.	Implementation Considerations	6
7.	Implementation Status	7
7.1.	IIT-CNR/Registro.it RDAP Server	7
7.2.	IIT-CNR/Registro.it RDAP Client	7
8.	IANA Considerations	8
9.	Privacy Considerations	8
10.	Security Considerations	9
11.	Acknowledgements	9
12.	References	9
12.1.	Normative References	9
12.2.	Informative References	10
Appendix A.	Paradigms to Enforce Access Control on Reverse Search in RDAP	11
Appendix B.	Change Log	12
	Authors' Addresses	13

[1.](#) Introduction

Reverse Whois is a service provided by many web applications that allows users to find domain names owned by an individual or a company starting from the owner's details, such as name and email. Even if it has been considered useful for some legal purposes (e.g. uncovering trademark infringements, detecting cybercrimes), its availability as a standardized Whois capability has been objected to for two main reasons, which now don't seem to conflict with an RDAP implementation.

The first objection concerns the potential risks of privacy violation. However, the domain name community is considering a new generation of Registration Directory Services [[ICANN-RDS1](#)] [[ICANN-RDS2](#)] [[ICANN-RA](#)], which provide access to sensitive data under

some permissible purposes and in accordance with appropriate policies for requestor accreditation, authentication and authorization. RDAP's reliance on HTTP means that it can make use of common HTTP-based approaches to authentication and authorization, making it more useful than Whois [[RFC3912](#)] in the context of such directory services. Since RDAP consequently permits a reverse search implementation complying with privacy protection principles, this objection is not well-founded.

The other objection to the implementation of a reverse search capability has been connected with its impact on server processing. However, the core RDAP specifications already define search queries, with similar processing requirements, so the distinction on which this objection is based is not clear.

Reverse searches, such as finding the list of domain names associated with contacts or nameservers, may be useful to registrars as well. Usually, registries adopt out-of-band solutions to provide results to registrars asking for reverse searches on their domains. Possible reasons for such requests are:

- * the loss of synchronization between the registrar database and the registry database;
- * the need for such data to perform bulk EPP [[RFC5730](#)] updates (e.g. changing the contacts of a set of domains, etc.).

Currently, RDAP does not provide any means for a client to search for the collection of domains associated with an entity [[RFC9082](#)]. A query (lookup or search) on domains can return the array of entities related to a domain with different roles (registrant, registrar, administrative, technical, reseller, etc.), but the reverse operation is not allowed. Only reverse searches to find the collection of domains related to a nameserver (ldhName or ip) can be requested. Since an entity can be in relationship with any RDAP object [[RFC9083](#)], the availability of a reverse search as largely intended can be common to all the object classes allowed for search. Through

a further step of generalization, the meaning of reverse search in the RDAP context can be extended to include any query for retrieving all the objects in relationship with another matching a given search pattern.

The protocol described in this specification aims to extend the RDAP query capabilities to enable reverse search based on the relationships defined in RDAP between an object class for search and a related object class. The reverse search based on the domain-entity relationship is treated as a particular case of such a generic query model.

[1.1.](#) Conventions Used in This Document

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.

[2.](#) RDAP Path Segment Specification

A generic reverse search path is described by the syntax:

```
{searchable-resource-type}/reverse_search_0/{related-resource-type}?<search-condition>
```

The path segments are defined as in the following:

- * searchable-resource-type: it MUST be one of the resource types for search defined in [Section 3.2 of \[RFC9082\]](#) (i.e. "domains", "nameservers" and "entities") or a resource type extension;
- * related-resource-type: it MUST be one of the resource types for lookup defined in [Section 3.1 of \[RFC9082\]](#) (i.e. "domain", "nameserver", "entity", "ip" and "autnum") or a resource type extension;
- * search-condition: a sequence of "property=search pattern" predicates separated by the ampersand character ('&', US-ASCII value 0x0026). Each "property" represents a JSON object property of the RDAP object class corresponding to "related-resource-type". Objects are only included in the search results if they satisfy

all included predicates. This includes predicates that are for the same property: it is necessary in such a case for the related object to match against each of those predicates. Based on their policy, servers MAY restrict the usage of predicates to make a valid search condition, by returning a 400 (Bad Request) response when a problematic request is received.

While related-resource-type is defined as having one of a number of different values, the only searches defined in this document are for a related-resource-type of "entity". Searches for the other resource types specified in [\[RFC9082\]](#) and resource type extensions may be defined by future documents.

Partial string matching in search patterns is allowed as defined in [section 4.1 of \[RFC9082\]](#).

[3.](#) RDAP Response Specification

Reverse search responses use the formats defined in [section 8 of \[RFC9083\]](#), which correspond to the searchable resource types defined in [Section 2](#).

[4.](#) Reverse Searches Based on Entity Details

Since in RDAP, an entity can be associated with any other object class, the most common kind of reverse search is one based on an entity's details. Such reverse searches arise from the query model by setting the related resource type to "entity".

By selecting a specific searchable resource type, the resulting reverse search aims at retrieving all the objects (e.g. all the domains) that are related to any entity object matching the search conditions.

This section defines the following reverse search properties servers SHOULD support regardless of the searchable resource type being selected:

Reverse search property: role
RDAP property: `$..entities[*].roles`
Reference: [Section 10.2.4 of \[RFC9083\]](#)

Reverse search property: handle
RDAP property: `$..entities[*].handle`
Reference: [Section 5.1 of \[RFC9083\]](#)

Reverse search property: fn
RDAP property: `$..entities[*].vcardArray[1][?(@[0]=='fn')][3]`
Reference: [Section 6.2.1 of \[RFC6350\]](#)

Reverse search property: email
RDAP property: `$..entities[*].vcardArray[1][?(@[0]=='email')][3]`
Reference: [Section 6.4.2 of \[RFC6350\]](#)

The mapping between the reverse search property and the corresponding RDAP response property is done through the use of a JSONPath expression [\[I-D.ietf-jsonpath-base\]](#).

The presence of a predicate on the reverse search property "role" means that the RDAP response property "roles" must contain at least the specified role.

The last two properties are related to jCard elements [\[RFC7095\]](#), but the field references are to vCard [\[RFC6350\]](#), since jCard is the JSON format for vCard.

Examples of reverse search paths based on the domain-entity relationship are presented in Figure 1.

```
/domains/reverse_search_0/entity?handle=CID-40*&role=technical  
  
/domains/reverse_search_0/entity?fn=Bobby*&role=registrant  
  
/domains/reverse_search_0/entity?handle=RegistrarX&role=registrar
```

Figure 1

Documents that deprecate or restructure RDAP responses such that one or more of the properties listed above becomes invalid MUST either note that the relevant reverse search is no longer available (in the case of deprecation) or describe how to continue supporting the relevant search by way of some new RDAP property (in the case of restructuring).

A server that includes additional fields in its objects in accordance with the extensibility provisions of [section 6 of \[RFC7480\]](#) MAY support the use of those fields in search conditions, in the same way as for the search conditions defined in this section. Support for such fields in the reverse search context MUST be documented in the extension specification.

[5.](#) RDAP Conformance

Servers complying with this specification MUST include the value "reverse_search_0" in the rdapConformance property of the help response [\[RFC9083\]](#). The information needed to register this value in the "RDAP Extensions" registry is described in [Section 8](#).

[6.](#) Implementation Considerations

To limit the impact of processing the search predicates, servers are RECOMMENDED to make use of indexes and similar functionality in their underlying data store. In addition, risks with respect to performance degradation or result set generation can be mitigated by adopting practices used for standard searches, e.g. restricting the search functionality, limiting the rate of search requests according to the user's authorization, truncating and paging the results, and returning partial responses.

[7.](#) Implementation Status

NOTE: Please remove this section and the reference to [RFC 7942](#) prior to publication as an RFC.

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in [\[RFC7942\]](#).

The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to [RFC 7942](#), "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

[7.1.](#) IIT-CNR/Registro.it RDAP Server

- * Responsible Organization: Institute of Informatics and Telematics of National Research Council (IIT-CNR)/Registro.it
- * Location: <https://rdap.pubtest.nic.it/>
- * Description: This implementation includes support for RDAP queries using data from the public test environment of .it ccTLD. Reverse search is allowed to authenticated users. Registrar users are allowed to perform reverse searches on their own domains and contacts. This is achieved by adding an implicit predicate to the search condition.
- * Level of Maturity: This is an "alpha" test implementation.
- * Coverage: This implementation includes all of the features described in this specification.
- * Contact Information: Mario Loffredo, mario.loffredo@iit.cnr.it

[7.2.](#) IIT-CNR/Registro.it RDAP Client

- * Responsible Organization: Institute of Informatics and Telematics of National Research Council (IIT-CNR)/Registro.it
- * Location: <https://web-rdap.pubtest.nic.it/>

- * Description: This is a Javascript web-based RDAP client. RDAP

responses are retrieved from RDAP servers by the browser, parsed into an HTML representation, and displayed in a format improving the user experience. Reverse search is allowed to authenticated users.

- * Level of Maturity: This is an "alpha" test implementation.
- * Coverage: This implementation includes all of the features described in this specification.
- * Contact Information: Francesco Donini, francesco.donini@iit.cnr.it

8. IANA Considerations

IANA is requested to register the following value in the RDAP Extensions Registry:

- * Extension identifier: reverse_search_0
- * Registry operator: Any
- * Published specification: This document.
- * Contact: IETF <iesg@ietf.org>
- * Intended usage: This extension describes reverse search query patterns for RDAP.

9. Privacy Considerations

The search functionality defined in this document may affect the privacy of entities in the registry (and elsewhere) in various ways: see [[RFC6973](#)] for a general treatment of privacy in protocol specifications. Registry operators should be aware of the tradeoffs that result from implementation of this functionality.

Many jurisdictions have laws or regulations that restrict the use of "Personal Data", per the definition in [[RFC6973](#)]. Given that, registry operators should ascertain whether the regulatory environment in which they operate permits implementation of the functionality defined in this document.

In general, given the sensitivity of this functionality, it SHOULD be accessible to authorized users only, and for specific use cases only.

Since reverse search requests and responses could contain Personally Identifiable Information (PII), reverse search functionality SHOULD be available over HTTPS only.

Providing reverse search in RDAP carries the following threats as described in [[RFC6973](#)]:

- * Correlation
- * Disclosure

* Misuse of information

Therefore, RDAP providers are REQUIRED to mitigate the risk of those threats by implementing appropriate measures supported by security services (see [Section 10](#)).

[10](#). Security Considerations

Security services required to provide controlled access to the operations specified in this document are described in [[RFC7481](#)]. A non-exhaustive list of access control paradigms an RDAP provider can implement is presented in [Appendix A](#).

The specification of the relationship within the reverse search path allows the RDAP servers to implement different authorization policies on a per-relationship basis.

[11](#). Acknowledgements

The authors would like to acknowledge the following individuals for their contributions to this document: Francesco Donini, Scott Hollenbeck, Francisco Arias, Gustavo Lozano, Eduardo Alvarez, Ulrich Wisser and James Gould.

Tom Harrison and Jasdip Singh provided relevant feedback and constant support to the implementation of this proposal. Their contributions have been greatly appreciated.

[12](#). References

[12.1](#). Normative References

- [OIDCC] OpenID Foundation, "OpenID Connect Core incorporating errata set 1", November 2014, <http://openid.net/specs/openid-connect-core-1_0.html>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC3912] Daigle, L., "WHOIS Protocol Specification", [RFC 3912](#), DOI 10.17487/RFC3912, September 2004, <<https://www.rfc-editor.org/info/rfc3912>>.
- [RFC5730] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)",

- [RFC6350] Perreault, S., "vCard Format Specification", [RFC 6350](#), DOI 10.17487/RFC6350, August 2011, <<https://www.rfc-editor.org/info/rfc6350>>.
- [RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", [RFC 6973](#), DOI 10.17487/RFC6973, July 2013, <<https://www.rfc-editor.org/info/rfc6973>>.
- [RFC7095] Kewisch, P., "jCard: The JSON Format for vCard", [RFC 7095](#), DOI 10.17487/RFC7095, January 2014, <<https://www.rfc-editor.org/info/rfc7095>>.
- [RFC7480] Newton, A., Ellacott, B., and N. Kong, "HTTP Usage in the Registration Data Access Protocol (RDAP)", STD 95, [RFC 7480](#), DOI 10.17487/RFC7480, March 2015, <<https://www.rfc-editor.org/info/rfc7480>>.
- [RFC7481] Hollenbeck, S. and N. Kong, "Security Services for the Registration Data Access Protocol (RDAP)", STD 95, [RFC 7481](#), DOI 10.17487/RFC7481, March 2015, <<https://www.rfc-editor.org/info/rfc7481>>.
- [RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", [BCP 205](#), [RFC 7942](#), DOI 10.17487/RFC7942, July 2016, <<https://www.rfc-editor.org/info/rfc7942>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in [RFC 2119](#) Key Words", [BCP 14](#), [RFC 8174](#), DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC9082] Hollenbeck, S. and A. Newton, "Registration Data Access Protocol (RDAP) Query Format", STD 95, [RFC 9082](#), DOI 10.17487/RFC9082, June 2021, <<https://www.rfc-editor.org/info/rfc9082>>.
- [RFC9083] Hollenbeck, S. and A. Newton, "JSON Responses for the

Registration Data Access Protocol (RDAP)", STD 95,
[RFC 9083](https://www.rfc-editor.org/info/rfc9083), DOI 10.17487/RFC9083, June 2021,
<<https://www.rfc-editor.org/info/rfc9083>>.

12.2. Informative References

[I-D.ietf-jsonpath-base]

Gössner, S., Normington, G., and C. Bormann, "JSONPath:
Query expressions for JSON", Work in Progress, Internet-

Loffredo & Martinelli

Expires 3 November 2022

[Page 10]

Internet-Draft

RDAP Reverse search

May 2022

Draft, [draft-ietf-jsonpath-base-03](https://www.ietf.org/archive/id/draft-ietf-jsonpath-base-03), 16 January 2022,
<<https://www.ietf.org/archive/id/draft-ietf-jsonpath-base-03.txt>>.

[I-D.ietf-regext-rdap-openid]

Hollenbeck, S., "Federated Authentication for the
Registration Data Access Protocol (RDAP) using OpenID
Connect", Work in Progress, Internet-Draft, [draft-ietf-regext-rdap-openid-08](https://www.ietf.org/archive/id/draft-ietf-regext-rdap-openid-08), 8 November 2021,
<<https://www.ietf.org/archive/id/draft-ietf-regext-rdap-openid-08.txt>>.

[ICANN-RA] Internet Corporation For Assigned Names and Numbers,
"Registry Agreement", July 2017,
<<https://newgtlds.icann.org/sites/default/files/agreements/agreement-approved-31jul17-en.pdf>>.

[ICANN-RDS1]

Internet Corporation For Assigned Names and Numbers,
"Final Report from the Expert Working Group on gTLD
Directory Services: A Next-Generation Registration
Directory Service (RDS)", June 2014,
<<https://www.icann.org/en/system/files/files/final-report-06jun14-en.pdf>>.

[ICANN-RDS2]

Internet Corporation For Assigned Names and Numbers,
"Final Issue Report on a Next-Generation gTLD RDS to
Replace WHOIS", October 2015,
<<http://whois.icann.org/sites/default/files/files/final-issue-report-next-generation-rds-07oct15-en.pdf>>.

[Appendix A](#). Paradigms to Enforce Access Control on Reverse Search in RDAP

Access control can be implemented according to different paradigms introducing increasingly stringent rules. The paradigms reported here in the following leverage the capabilities either supported natively or provided as extensions by the OpenID Connect [\[OIDCC\]](#):

- * **Role-Based Access Control:** access rights are granted depending on roles. Generally, this is done by grouping users into fixed categories and assigning static grants to each category. A more dynamic approach can be implemented by using the OpenID Connect "scope" claim;
- * **Purpose-Based Access Control:** access rules are based on the notion of purpose, being the intended use of some data by a user. It can be implemented by tagging a request with the usage purpose and

making the RDAP server check the compliance between the given purpose and the control rules applied to the data to be returned. The purpose can be stated within an out-of-band process by setting the OpenID Connect RDAP-specific "purpose" claim as defined in [\[I-D.ietf-regext-rdap-openid\]](#);

- * **Attribute-Based Access Control:** rules to manage access rights are evaluated and applied according to specific attributes describing the context within which data are requested. It can be implemented by setting within an out-of-band process additional OpenID Connect claims describing the request context and making the RDAP server check the compliance between the given context and the control rules applied to the data to be returned;
- * **Time-Based Access Control:** data access is allowed for a limited time only. It can be implemented by assigning the users with temporary credentials linked to access grants whose scope is limited.

[Appendix B](#). Change Log

- 00: Initial working group version ported from [draft-loffredo-regext-rdap-reverse-search-04](#)
- 01: Updated "Privacy Considerations" section.
- 02: Revised the text.
- 03: Refactored the query model.
- 04: Keepalive refresh.

- 05: Reorganized "Abstract". Corrected "Conventions Used in This Document" section. Added "RDAP Conformance" section. Changed "IANA Considerations" section. Added references to [RFC7095](#) and [RFC8174](#). Other minor edits.
- 06: Updated "Privacy Considerations", "Security Considerations" and "Acknowledgements" sections. Added some normative and informative references. Added [Appendix A](#).
- 07: Updated normative references.
- 08: Changed "Implementation Status" section. Updated informative references.
- 09: Extended the query model to represent a reverse search based on any relationship between the RDAP object classes. Changed the path segment "role" into a query parameter.
- 10: Updated "Reverse Searches Based on Entity Details" section to consider the use of JSContact format instead of jCard. Added references to JSContact documents.
- 11: Updated the document based on Tom Harrison and James Gould feedback:
 - * Updated section "RDAP Path Segment Specification":
 - Clarified how servers must evaluate a reverse search including predicates that are for the same property.

- Specified the error response servers must return when receiving a wrong reverse search request according to their policy.
- Clarified that searches for the related-resource-type values other than "entity" may be defined in future documents.
- * Reviewed text in section "Reverse Searches Based on Entity Details" about reverse searches based on custom response extensions.
- * Removed references to JSContact documents in section "Reverse Searches Based on Entity Details". Moved the mapping between jCard properties used in the RDAP response and JSContact counterparts to [draft-ietf-regext-rdap-jscontact](#).
- * Added section "RDAP Response Specification".
- * Changed the text to present reverse search as a single extension with multiple features.
- * Changed the definition of searchable-resource-type and related-resource-type to consider also the resource type extensions.
- * Replaced "reverse" with "reverse_search_0" in the generic

- reverse search path. Updated Figure 1 accordingly.
- * Removed the phrase "but with a special focus on its privacy implications" from both the "Abstract" and the "Introduction". Moved the mapping between jCard properties used in the RDAP response and JSContact counterparts to [draft-ietf-regext-rdap-jscontact](#).
 - * Reviewed the text of "Privacy Considerations" section.
 - * Text cleaning.

Authors' Addresses

Mario Loffredo
IIT-CNR/Registro.it
Via Moruzzi,1
56124 Pisa
Italy
Email: mario.loffredo@iit.cnr.it
URI: <http://www.iit.cnr.it>

Maurizio Martinelli
IIT-CNR/Registro.it
Via Moruzzi,1
56124 Pisa
Italy
Email: maurizio.martinelli@iit.cnr.it
URI: <http://www.iit.cnr.it>