Workgroup: Registration Protocols Extensions
Internet-Draft:
draft-ietf-regext-rdap-reverse-search-09
Published: 10 February 2022
Intended Status: Standards Track
Expires: 14 August 2022
Authors: M. Loffredo
IIT-CNR/Registro.it
M. Martinelli
IIT-CNR/Registro.it

Registration Data Access Protocol (RDAP) Reverse search capabilities

Abstract

The Registration Data Access Protocol (RDAP) does not include query capabilities to find 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 RDAP query extensions 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 but with a special focus on its privacy implications.

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 14 August 2022.

Table of Contents

1. Introduction

Reverse Whois is a service provided by many web applications that allow 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 has been caused by the potential risks of privacy violation. However, TLDs 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 according to adequate policies to enforce

the requestor accreditation, authentication, authorization, and terms and conditions of data use. It is well known that such security policies are not implemented in Whois [RFC3912], while they are in RDAP [RFC7481]. Therefore, RDAP permits a reverse search implementation complying with privacy protection principles.

The other objection to the implementation of a reverse search capability has been connected with its impact on server processing. Since RDAP supports search queries, the impact of both standard and reverse searches is equivalent and can be mitigated by servers adopting ad hoc strategies. Furthermore, the reverse search is almost always performed by specifying an entity role (e.g. registrant, technical contact) and this can contribute to restricting the result set.

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 massive 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 the reverse search based on the relationships 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 of such a generic query model but with a special focus on its privacy implications. The extension is implemented by adding new path

segments (i.e. search paths) and using a RESTful web service [REST]. The service is implemented using the Hypertext Transfer Protocol (HTTP) [RFC7230] and the conventions described in [RFC7480].

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

The new search paths are OPTIONAL extensions of those defined in [RFC9082]. A generic reverse search path is described by the syntax:

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

The path segments are defined as in the following:

* searchable-resource-type: it MUST be one of resource types for search defined in Section 3.2 of [RFC9082], i.e. "domains", "nameservers" and "entities";

* 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";

* 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". All the predicates are joined by the AND logical operator. Based on their policy, servers MAY restrict the usage of predicates to make a valid search condition.

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

2.1. 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 searches are based on the entity details. Such reverse searches arise from the above 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 condition.

This section defines the following reverse search properties to be used regardless of the searchable resource type being selected:

Reverse search property: role
RDAP property: $..entities[*].roles
RFC reference: Section 10.2.4 of [RFC9083]
Reverse search property: handle
RDAP property: $..entities[*].handle
RFC reference: Section 5.1 of [RFC9083]
Reverse search property: fn
RDAP property: $..entities[*].vcardArray[1][?(@[0]=='fn')][3]
RFC reference: Section 6.2.1 of [RFC6350]
Reverse search property: email
RDAP property: $..entities[*].vcardArray[1][?(@[0]=='email')][3]
RFC reference: Section 6.4.2 of [RFC6350]

Regarding the definitions above, it must be noted that:

* 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.

* Some of the properties are related to jCard elements [RFC7095] but, being jCard the JSON format for vCard [RFC6350], the corresponding RFC reference is to the vCard specification [RFC6350].

Servers MAY implement other properties than those defined in this section.

Examples of reverse search paths based on the domain-entity relationship are presented below:

```
/domains/reverse/entity?handle=CID-40*&role=technical

/domains/reverse/entity?fn=Bobby*&role=registrant

/domains/reverse/entity?handle=RegistrarX&role=registrar
```

Figure 1

3. 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 6.

4. Implementation Considerations

The implementation of the proposed extension is technically feasible. To limit the impact of processing the search predicates, servers are RECOMMENDED to mandate the use of at least one property among those mapped to indexed fields of the registry database. Other properties, such as "role", MAY be allowed to further restrict the set of possible results. In addition, the risks to degrade the performance or to generate huge result sets can be mitigated by adopting the same policies valid for handling searches (e.g. restricting the search functionality, limiting the rate of search requests according to the user profile, truncating and paging the results, returning partial responses).

5. 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".

5.1. 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 condition to the search pattern.
*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

5.2. 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

6. 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.

7. Privacy Considerations

The use of the capability described in this document whenever a contact detail is taken MUST be compliant with the rules about privacy protection each RDAP provider is subject to. Sensitive registration data MUST be protected and accessible for permissible purposes only. This feature SHOULD be only accessible to authorized users and only for a specified use case.

Since the request for this feature could contain Personal Identifiable Information, it SHOULD only be accessible to authorized users and available over HTTPS.

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 8).

8. 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.

9. 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 and Ulrich Wisser.

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

10. References

10.1. 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)", STD 69, RFC 5730, DOI 10.17487/RFC5730, August 2009, <https://www.rfc-editor.org/info/rfc5730>.

[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>.

[RFC7230] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", RFC 7230, DOI 10.17487/RFC7230, June 2014, <https://www.rfc-editor.org/info/rfc7230>.

[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, DOI 10.17487/RFC9083, June 2021, <https://www.rfc-editor.org/info/rfc9083>.

10.2. Informative References

[I-D.ietf-jsonpath-base] Gössner, S., Normington, G., and C. Bormann, "JSONPath: Query expressions for JSON", Work in Progress, Internet-Draft, 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, 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>.

[REST] Fielding, R., "Architectural Styles and the Design of Network-based Software Architectures", 2000, <http://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.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 each category with static grants. 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 which means the intended usage 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 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 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.

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