

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: December 31, 2014

C.M. Martinez, Ed.
LACNIC
L. Zhou, Ed.
CNNIC
G. Rada
LACNIC
July 2014

**Redirection Service for Registration Data Access Protocol
draft-ietf-weirds-redirects-04**

Abstract

The traditional WHOIS protocol has several important shortcomings, and over the past few years several approaches to a better Registration Data Access Protocol (RDAP) have been discussed and proposed.

It is worth noting that the term WHOIS is sometimes used interchangeably to mean either (a) the registration data itself or (b) the protocol used to access registration data

Among these shortcomings, different registries operate different WHOIS services. For users this means that several WHOIS queries to different registries may be necessary in order to obtain data for a given resource.

This document describes a redirection service for RDAP queries. This service allows clients to query a single RDAP service and expect either an authoritative answer or a redirection hint pointing to another, possibly authoritative, RDAP server.

The solution implemented proposed here applies to Regional Internet Registries(RIRs) and Domain Name Registries(DNRs).

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 <http://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 December 31, 2014.

Copyright Notice

Copyright (c) 2014 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 (<http://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 Simplified BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
1.1.	Requirements Language	3
1.2.	The REST Approach to Web Services	3
1.3.	Request Redirection for RDAP Queries	3
1.4.	The Redirection Table. The Bootstrap Problem.	3
2.	Architectural Use Cases of Redirects in RDAP	3
2.1.	A Joint RDAP Tree through HTTP Redirection	3
2.2.	Helper Service for Constrained RDAP Clients	5
3.	Security Considerations	5
3.1.	Loops in Redirection	6
4.	References	6
4.1.	Normative References	6
4.2.	Informative References	6
	Authors' Addresses	7

[1.](#) Introduction

A user interested in obtaining registration information for a given number or domain resource normally uses the WHOIS service provided by the RIRs and DNRs.

In order to avoid having to query several databases until obtaining an answer, some approaches have been discussed and implemented in the past, most notably the Joint WHOIS [[lacnic-joint-whois](#)] initiative. However, among other shortcomings, Joint WHOIS is implemented using proxies and server-side referrals.

The RDAP protocol ([draft-ietf-weirds-using-http](#) [I-D.ietf-weirds-using-http]) makes it comparatively easy to implement client-side redirects based on normal HTTP 1.1 semantics and behavior.

The goal of this I-D is to describe an implementation of an RDAP redirection service and to encourage discussion on the topic of

redirects in this problem domain.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

1.2. The REST Approach to Web Services

While a full introduction to REST and RESTful interfaces is out of the scope of this document it is important to note that these interfaces employ the verbs defined in HTTP (GET, POST, HEAD, DELETE) and HTTP response codes to signal the semantics and outcomes of an operation.

As WHOIS is a read-only service only the GET and HEAD verbs are usually implemented.

HTTP status codes provide signaling for errors and other conditions, including the concept of "client-side redirection" as outlined below.

1.3. Request Redirection for RDAP Queries

Each RDAP server should answer directly only those queries for which it is authoritative. In this case, being authoritative equals "having direct access to a given registry database".

For all other queries, a RDAP server could provide a 301 MOVED PERMANENTLY redirect answer pointing to an URL hosted on a different RDAP server.

As all requests are to be performed employing HTTP GETs, a user agent can transparently follow the HTTP 30x redirection hints ([[RFC2616](#)]) until obtaining a non-error answer (HTTP 20x) or an unrecoverable error condition (HTTP 40x or 50x).

1.4. The Redirection Table. The Bootstrap Problem.

For the redirection table lookup function, the redirector can either have pre-populated local table or have access to a service provided by some form of directory service. How either this local table or directory service is fed is known as the "bootstrap problem".

RDAP Bootstrap is described in [draft-ietf-weirds-bootstrap](#) [I-D.ietf-weirds-bootstrap]

2. Architectural Use Cases of Redirects in RDAP

2.1. A Joint RDAP Tree through HTTP Redirection

In an scenario where a client does not know which registry can
provide authoritative answers***TBC

When an RDAP server receives a query for which it does not have an authoritative answer to provide, it MAY provide an HTTP 30x redirection message pointing the client to a redirection-only RDAP server, which in turn can provide further redirections guiding the client to an authoritative server.

The redirect-only server is responsible for tracking and returning the authoritative sources for IP, AS, domain name, name server or entity queries. All the query format are described in the [draft-ietf-weirds-rdap-query](#) [I-D.ietf-weirds-rdap-query]. We call this redirect server "the redirector".

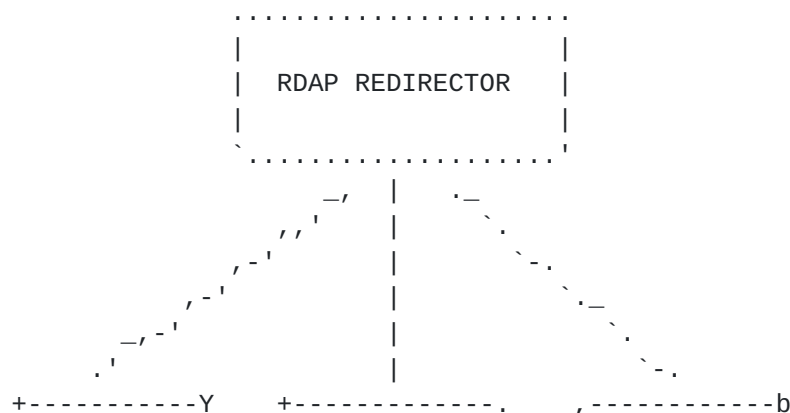
The redirect server needs access to data sources that, given a queried resource, provide a pointer to the authoritative RDAP server. For lack of a better name, we will call this data source the "redirection table".

Assuming the redirector has access to a redirection table, the following pseudo code describes its expected behaviour:

```
while(true) {
    query = read_query_from_network()
    auth_rdap_svr = redirect_table_lookup (query.resource)
    if (auth_rdap_svr != null) {
        write_http_301(auth_rdap_svr)
    } else {
        write_http_404("resource not in redirect table")
    }
}
```

Redirector state machine

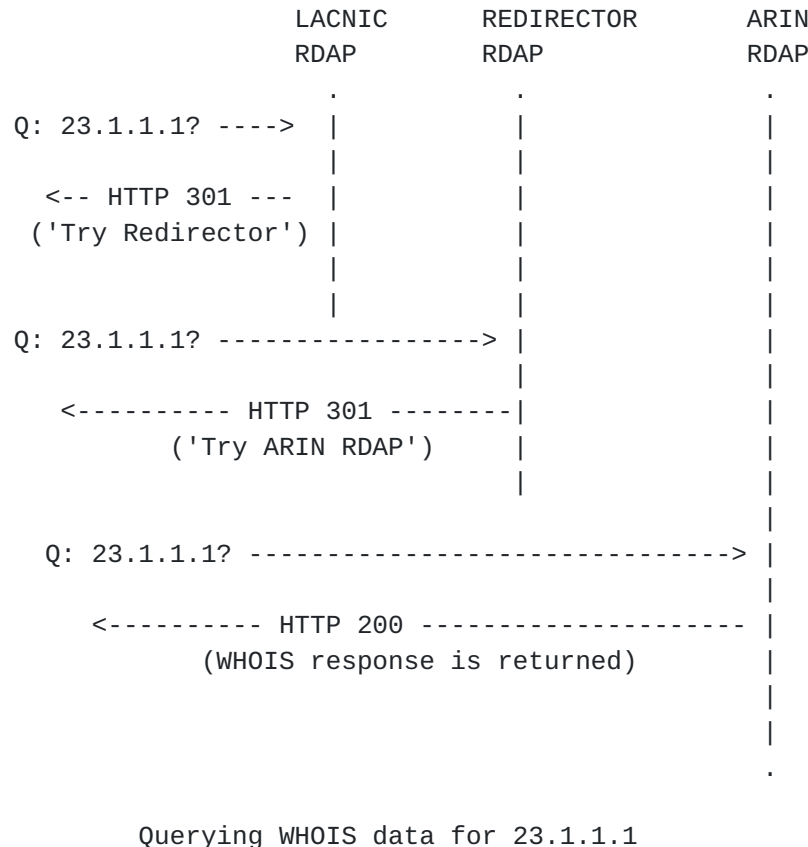
Figure 2 shows the general scheme of a single RDAP Redirection Service serving three different RIRs standalone RDAPs while providing a seamless query interface to clients.



	REGY1			REGY2			REGY3	
	

RDAP Joint WHOIS Tree.

Figure 3 shows how HTTP 301 redirection hints guide a client looking for registration data for the IPv4 address 23.1.1.1 (administered by ARIN) from LACNIC's WHOIS, the redirector and finally ARIN's WHOIS.



2.2. Helper Service for Constrained RDAP Clients

It is expected that a significant portion of RDAP clients will be written for and operate under constrained environments. For example, simple Javascript clients written to run inside a web browser's sandbox cannot perform arbitrary DNS queries nor open sockets, thus limiting the ability of the client to actually access bootstrapping data

TBD

3. Security Considerations

HTTP 30x-based redirection could offer an attack vector for a Man-in-the-Middle type of attack, where the adversary modifies the redirection URL offered by the server to the client.

For example, an attacker able to modify HTTP traffic could modify the redirect URL from http://www.labs.lacnic.net/restwhois/rwhois_redir/ip/23.1.1.1 and change it into <http://www.labs.somenic.net/restwhois/>

[rwhois_redir/ip/23.1.1.1](#), where bogus information can be offered to the client.

This particular type of attack can be prevented by using HTTPS for the RDAP connection. However, this certainly places a load burden upon the servers.

While security practices are outside the scope of this document, the authors believe it is important to identify such problematic use cases to any DNR or RIR that may implement the redirection WHOIS service.

3.1. Loops in Redirection

When redirection is used there is always the risk that bogus user-agents and applications or malicious user can create loops that in turn may become Denial of Service attacks.

Commonly used user agents (including HTTP libraries) have loop detection features that are deemed sufficient for breaking loops in RDAP.

4. References

4.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.

4.2. Informative References

[I-D.ietf-weirds-bootstrap]
Blanchet, M., "Finding the Authoritative Registration Data (RDAP) Service", Internet-Draft [draft-ietf-weirds-bootstrap-03](#), June 2014.

[I-D.ietf-weirds-rdap-query]
Newton, A. and S. Hollenbeck, "Registration Data Access Protocol Query Format", Internet-Draft [draft-ietf-weirds-rdap-query-02](#), December 2012.

[I-D.ietf-weirds-using-http]
Newton, A., Ellacott, B. and N. Kong, "Using the Registration Data Access Protocol (RDAP) with HTTP", Internet-Draft [draft-ietf-weirds-using-http-01](#), December 2012.

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P. and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", [RFC 2616](#), June 1999.

[lacnic-joint-whois]

LACNIC, "LACNIC Joint WHOIS Implementation", 2005, <<ftp://anonymous@ftp.registro.br/pub/gter/gter20/02-jwhois-lacnic.pdf>>.

Authors' Addresses

Carlos M. Martinez, editor
LACNIC
Rambla Mexico 6125
Montevideo, 11400
Uruguay

Phone: +598-2604-2222
Email: carlos@lacnic.net

Linlin Zhou, editor
CNNIC
No. 4, South 4th Steet, Zhongguancun
Beijing, 100190
China

Phone: +8610-5881-2677
Email: zhoulinlin@cnnic.cn

Gerardo Rada
LACNIC
Rambla Mexico 6125
Montevideo, 11400
Uruguay

Phone: +598-2604-2222
Email: gerardo@lacnic.net

