Death to whois

An exploratory look at RDAP

20C TWENTIETH CENTURY

MATT GRISWOLD

<grizz@20C.com>

INTRODUCTION

While writing code for PeeringDB, I did extensive work querying and comparing RDAP results across the RIRs and was surprised by the inconsistencies. It's a great tool to replace whois, but it still needs work and communication to come to fruition—luckily the RIRs have all been very receptive to complaints and are quick to fix bugs.

GOALS

- **Get Schwifty — USE RDAP!**
- Share experiences/solutions—the community is very receptive to input
- Contribute to OSS RDAP projects

WHAT IS WHOIS?

A simple query protocol to discover information about network objects.

If you don’t use whois multiple times a day, this talk might not be for you :)

WHAT IS RWHOIS?

“RWhois (Referral Whois) is a directory services protocol which extends and enhances the WHOIS concept in a hierarchical and scalable fashion.”

Allows for delegation of the authoritative source,

“server0 knows this much information, server1 knows more”

WHAT’S WRONG WITH WHOIS

(From RFC7485)

- It does not support user authentication or access control for differentiated access.
- It has not been internationalized and thus does not consistently support Internationalized Domain Names (IDNs) as described in [RFC5890].

WHAT’S WRONG WITH WHOIS

No normalization, each server uses a different format and different field names.

To find the org that owns as ASN:

```
whois -h whois.arin.net as63311 | grep OrgName
OrgName:    20C, LLC


whois -h whois.ripe.net as15562 | grep org-name
org-name:   Job Snijders
```

WHAT’S WRONG WITH WHOIS

Incredibly “hacky” to find authoritative sources, whois clients all have them hardcoded and the packages need to be updated as things change.

https://github.com/rfc1036/whois/blob/next/as_del_list

```
248     251     ripe
306     371     whois.nic.mil
379     508     whois.nic.mil
```

WHAT’S WRONG WITH WHOIS

Current whois on CentOS 7:

```
$ rpm -qa | grep whois
whois-5.1.1-2.el7.x86_64

$ whois as306
getaddrinfo(whois.nic.mil): Name or service not known
```

WHAT'S WRONG WITH WHOIS

Of course, you can just simply look it up at iana.org now as well:

```
$ whois -h $(whois -h whois.iana.org as306 | \
  grep -e '^whois:' | \
  sed 's/^whois: *\(.*\)$/\1/') as306
```

WHAT’S RIGHT WITH WHOIS

Client and usage is very well known.

A single tool to use to get information on any network object.

WHAT IS RDAP?

Registration **D**ata **A**ccess **P**rotocol

A collection of RFCs for using HTTP and JSON as a replacement to whois.

RFCs 7480, 7481, 7482, 7483, 7484, 7485

WHAT IS RDAP?

Covers all object types covered by whois, DNS, IP, ASN, et al.

This deck only covers ASNs from RIRs.

WHY HTTP+JSON?

HTTP freely on everything—widely used and understood

302 redirects fit perfectly for rwhois style delegation

JSON also widely used and understood

Quite easy to implement a server for delegated records, to go from a database to JSON over HTTP is less than 10 lines of code with most languages.

RDAP 101

Produces (in theory) the same information as whois, only using standards?!?!

- **Nested JSON structure** (RFC7159)
- **Standard field names** (RFC7483)
- **JSON vCard** (RFC6350/RFC7095)
- Coming: **JSON RDAP JCR** (draft-newton-rdap-jcr)

RDAP 101

Authoritative sources are discovered from the remote server, client never needs to update

HTTP redirects delegate information

RDAP 101

Example simple query

```
$ curl https://rdap.arin.net/registry/autnum/63311
```

Returns a json dictionary with info

RDAP 101

Has entities with handles similar to whois, they are either nested inside the original request or need to be separately queried.

SEPARATE ENTITY QUERY

curl https://rdap.db.ripe.net/autnum/8283

```
  "entities" : [ {
    "handle" : "CLUE1-RIPE",
    "roles" : [ "administrative", "technical" ],
    "objectClassName" : "entity"
  },
```

curl https://rdap.db.ripe.net/entity/CLUE1-RIPE

QUEST FOR ORG FROM ASN

Problem: PeeringDB wanted to automate network signups

Possible solutions:

- Scrape whois data
- RDAP

Guess which we picked?

QUEST FOR ORG FROM ASN

We required:

- email addresses associated with ASN
- org info for ASN registrant

QUEST FOR ORG FROM ASN

AFRNIC: https://rdap.afrinic.net/rdap/

- Has email addresses
- No role information on any entities
- No name on org entity

QUEST FOR ORG FROM ASN

ARIN: https://rdap.arin.net/

- Has email addresses with roles
- Org data in nested vCard under `roles`: "registrant"

QUEST FOR ORG FROM ASN

APNIC: https://rdap.apnic.net/

- Has email addresses
- Has roles, but not registrant
- Data is inconsistent, changes per request

QUEST FOR ORG FROM ASN

LACNIC: https://rdap.lacnic.net/bootstrap/

- Has email addresses
- No org info
- Brazilian ASNs redirect to nic.br
    - IPv4 has email addresses, IPv6 does not

QUEST FOR ORG FROM ASN

RIPE: https://rdap.db.ripe.net/

- Has email addresses with recursive queries
- No org data

PROGRESS!

- LACNIC added email addresses
- LACNIC / nic.br is fixing the IPv6 issue
- RIPE has addressed adding org info
    - Needs RFC and RIR coordination to not be a one off

THINGS WE STILL NEED TO DO

- Normalize all RIRs to give the correct information
- Get RIRs to expose all whois data
- rm `which whois`

Which brings us back to...

GOALS

- **USE RDAP!**
- Share experiences/solutions—the community is very receptive to input
- Contribute to OSS projects

PEERINGDB WHOIS CLIENT

- Authenticated -- when logged in, it gives contact information for networks

Same syntax as normal whois, only uses HTTPS. Server side it’s the exact same code. Running the following 2 commands are equal, except if you’re logged into PeeringDB you also get email addresses.

`peeringdb whois as63311` is the same as

`whois -h peeringdb.com as63311`

PEERINGDB WHOIS CLIENT

With PeeringDB 2.0, all email addresses were hidden unless the user was authenticated, which changed whois.

We added whois to our client, using HTTPS+JSON - **not** RDAP, just the same concept. Uses the same syntax as whois, produces the same output.

PEERINGDB WHOIS CLIENT

Running the following 2 commands are equal, even uses the same code on the whois server to deliver. With the peeringdb command, if you’re logged in, you also get email addresses.

`peeringdb whois as63311` is the same as

`whois -h peeringdb.com as63311`

RDAP CLIENTS

Python client
pip install rdap

https://github.com/20c/rdap

Ruby client from ARIN
gem install nicinfo

https://github.com/arineng/nicinfo

RDAP CLIENTS

Clients should produce the same output as whois, but also provide the same information to scripts in a programmatic way.

GOING FORWARD

Draft to check RDAP responses, will be a repeatable way to check for RIR data integrity

https://datatracker.ietf.org/doc/draft-newton-rdap-jcr/

https://datatracker.ietf.org/doc/draft-newton-json-content-rules/

GOING FORWARD

IETF Registration Protocols Extensions list discusses RDAP changes, join and let people know if you want to be involved.

https://www.ietf.org/mailman/listinfo/regext

USE RDAP

Let’s all work together to get rid of whois!

Questions?

Death to whois

An exploratory look at RDAP

20C TWENTIETH CENTURY

MATT GRISWOLD
<grizz@20C.com>