Bootstrapping Weirds

draft-blanchet-weirds-bootstrap
draft-blanchet-weirds-bootstrap-ianaregistries
draft-blanchet-weirds-bootstrap-autonomous

Marc Blanchet
Viagénie
marc.blanchet@viagenie.ca

Vancouver IETF, November 2013

Context

- 2 bootstrap solutions were presented in Berlin:
  - DNS-based (draft-blanchet-weirds-bootstrap)
  - IANA registry based (draft-blanchet-weirds-bootstrap-ianaregistries)
- During the Berlin meeting, another solution was proposed, mainly to avoid going to IANA.
  - draft-blanchet-weirds-bootstrap-autonomous is an attempt to describe that another solution.
- Goal: to reach consensus on the direction
- Drafts are in good shape to get the idea, but not fully specified (on purpose). When consensus reached, will revise the draft(s)

DNS-based solution

- draft-blanchet-weirds-bootstrap
- names:
  - rdap query for example.com will result in DNS query of example.com.domain.rdap.arpa
- numbers:
  - rdap query for 192.9.200.0/24 generates a DNS request to 200.9.192.ip4.rdap.arpa.
  - rdap query for 2001:db8::/32 generates a DNS request to 8.b.d.0.1.0.0.2.ip6.rdap.arpa.
- requested RR could be A, AAAA, CNAME, SRV, NAPTR, with pros and cons.

IANA Registries based Solution

- draft-blanchet-weirds-bootstrap-ianaregistries
- names:
  - rdap query for example.com results in matching the content of the cell corresponding to the row for “com” in the IANA registry. The content is the rdap server url (http://rdap.mytld/rdap/...)
- numbers:
  - rdap query for 192.9.200.0/24 results in matching the content of the cell corresponding to the row for “192/8” in the IANA registry...

Example of Current IANA Registries

IANA IPv4 Address Space Registry

Last Updated
2013-05-20

Description
The allocation of Internet Protocol version 4 (IPv4) address space to various registries is listed here. Originally, all the IPv4 address spaces was managed directly by the IANA. Later parts of the address space were allocated to various other registries to manage for particular purposes or regional areas of the world. RFC 1466 [RFC1466] documents most of these allocations.

Alternative Formats

CSV Plain text

Prefix	Designation	Date	Whois	Status [1]	Note
058/8	APNIC	2004-04	whois.apnic.net	ALLOCATED	
059/8	APNIC	2004-04	whois.apnic.net	ALLOCATED	
060/8	APNIC	2003-04	whois.apnic.net	ALLOCATED	
061/8	APNIC	1997-04	whois.apnic.net	ALLOCATED	
062/8	RIPE NCC	1997-04	whois.ripe.net	ALLOCATED	
063/8	ARIN	1997-04	whois.arin.net	ALLOCATED	
064/8	ARIN	1999-07	whois.arin.net	ALLOCATED	
065/8	ARIN	2000-07	whois.arin.net	ALLOCATED	
066/8	ARIN	2000-07	whois.arin.net	ALLOCATED	
067/8	ARIN	2001-05	whois.arin.net	ALLOCATED	

Root Zone Database

The Root Zone Database represents the delegation details of top-level domains, including gTLDs such as .com, and country-code TLDs such as .uk. As the manager of the DNS root zone, IANA is responsible for coordinating these delegations in accordance with its policies and procedures.

Much of this data is also available via the WHOIS protocol at whois.iana.org.

Domain	Type	Sponsoring Organisation
.ac	country-code	Network Information Center (AC Domain Registry) c/o Cable and Wireless (Ascension Island)
.ad	country-code	Andorra Telecom
.ae	country-code	Telecommunication Regulatory Authority (TRA)
.aero	sponsored	Societe Internationale de Telecommunications Aeronautique (SITA INC
.af		
.ag		
.ai		
.al		

Delegation Record for .CA

ISO link for decoding the two-letter codes

Sponsoring Organisation

ty - AKEP

Canadian Internet Registration Authority (CIRA)
Autorite Canadienne pour les Enregistrements Internet (ACEI)
350 Sparks Street
Suite 306

Name Servers

Host Name	IP Address(es)
c.ca-servers.ca	192.228.28.9
e.ca-servers.ca	192.228.30.9
j.ca-servers.ca	198.182.167.1 2001:500:83:0:0:0:0:1
k.ca-servers.ca	156.154.100.4
l.ca-servers.ca	156.154.101.4
sns-pb.isc.org	192.5.4.1 2001:500:2e:0:0:0:0:1
z.ca-servers.ca	192.228.25.66
tld.isc-sns.net	63.243.194.3 2001:5a0:10:0:0:0:0:3
a.ca-servers.ca	199.253.251.2 2001:500:80:5000:0:0:0:2

Registry Information

URL for registration services: http://www.cira.ca/
WHOIS Server: whois.cira.ca

“Autonomous” Solution

- draft-blanchet-weirds-bootstrap-autonomous. No IANA involved.
- names:
  - rdap query for example.com will result in DNS query (SRV or NAPTR) of _rdap._tcp.com
- numbers:
  - RIRs have a daily updated file (large: 19M, 300K lines) containing a detailed compilation of all allocations (for the 5). An augmented file would include a new column pointing to the RDAP server for each allocaiton
  - rdap query for 192.9.200.0/24 generates a DNS query (SRV or NAPTR) to rdap.rirexample.net
  - rdap query for 2001:db8::/32 generates a DNS query (SRV or NAPTR) to rdap.rirexample.net

Comparing Solutions

	Names			Numbers		
	DNS $name.rdap.arpa RR	IANA $tld->rdap uri	Autonomous _rdap.$tld RR	DNS $ip.rdap.arpa RR	IANA $ip->rdap uri	Autonomous $ip->rdap uri
Registration authority involved	Y	Y	N	Y	Y	Y
URL Reply	with NAPTR	Y	with NAPTR	with NAPTR	Y	Y
$sld.$tld registries	Y	Possible	Possible	N/A	N/A	N/A
Redirection between parties	No need	No need	No need	Required	Required	Required
Can be secured?	DNSSEC	https	DNSSEC	DNSSEC	https	https

WG Direction

- Looking for concensus on direction to update the draft and to add more details (on the chosen solution).
- DNS-based? IANA-registry-based? Autonomous?

Backup Slides
(from Berlin IETF87)

DNS-based solution

- can be secured with DNSSEC
- highly scalable
- has expiration, caching, ...
- infrastructure already in place

IANA Registries based Solution

- Creation of new IANA registries
  - but based on current data and relationships
- Registries:
  - tld => rdap server url
    - similar to the current root zone database registry with a new “column”.
  - numbers => rdap server url
    - similar to the current IP address registries with a new column.
  - small single XML files
    - can be fetched in advance, locally cached, ...

ASN

- AS numbers are not hierarchical numberspace. flat.
- IANA allocations are done by ranges to RIR
- Both solutions can be mapped into the allocations
  - IANA registry-based solution would be identical to the addresses: match, column with the rdap url
  - DNS-based solution would be mostly a single flat space to a single entity (the RIR may agree to run a joint server/proxy for these.

Addresses

- Currently, RIR (only 5) usually:
  - know each other
  - know ranges for each RIR
  - therefore, redirect to the other server when they receive a request not for their own range.
- But:
  - we need to specify the list of these servers somewhere. (not in the RFC, IANA registry?)

Comparing solutions

- _Possible_ requirements/decision/differentiation points (was sent to the list)
  - require use of https on every request
  - specify per registry which of http/https is to be used by clients
  - provide delegation below the tld
  - same solution for both names and numbers
  - don't route all traffic through one point of attack (which is not the same as one point of failure)
  - base URL may have a prepended path (i.e. http://domain/my/own/path/query)
  - if DNS is used, only terminating DNS RR can be used (i.e. no CNAME, SRV, NAPTR
  - constrained to what Javascript offers in browsers
  - simplicity/easy to implement
  - does the client have a cache of "servers" to start with?
  - if a cache, how/when does it refresh the data?

HTTP vs HTTPS

- Support for both requires some signaling
  - DNS: “advanced” records (SRV/NAPTR)
  - IANA registries: a field saying which one is available.
- Single transport is easier for client. But https is heavier on servers and require one cert per TLD. But https gives us data integrity (and confidentiality and source verification)

Base URL and DNS

- If we want “http://example.com/rdap/mytld/” (instead of http://rdap.mytld), then
  - For DNS-based solution:
    - basic DNS RR (A, AAAA, CNAME) do not fill this
    - need to use SRV/NAPTR records which are more complex.
      - SIP had these records (as non mandatory) but almost nobody use them.
  - For IANA registry:
    - the base url is in the IANA registry.

Javascript

- Almost no DNS requests in the browser.
- But most JS use external APIs/AJAX/... to complement their code.
- JS in browsers should then, as typical, use some external API/AJAX for the purpose of bootstrapping.
  - could be a private service by the JS app provider
  - or a public service.
- Shall we restrict the specification to the only capabilities of the intersection of features on all JS browser implementations?

Impact on IANA

- We need IANA work for both solutions.
- IANA has already relationship with TLDs.
- DNS-based:
  - tlds tell IANA the RDAP DNS records for their tld. IANA put it in the related arpa zone.
  - DNS infrastructure already setup for this service.
- IANA registry-based:
  - tlds tell IANA the rdap server url for their tld. IANA put it in the IANA registry
  - IANA has to put some caching infrastructure to handle the load. (IANA is (preliminary) ok if this is what we need)

ICANN EWG Considerations

- ICANN EWG considering a centralized repository of (copied) registration data (copy received from the registries).
- Bootstrap
  - shall support this if that recommendation is put forward.
  - but also support at the same time other registration data repository (for example, cctld not going into EWG).
- DNS and IANA registries based approaches both support ICANN EWG direction.

Comparing Solutions

- DNS-based:
  - is more constrained (http-https, base url) if kept simple.
  - can be flexible if using more complex DNS records (SRV, NAPTR)
  - infrastructure already in place, scales, ...
- IANA-registry-based:
  - more flexible (full url with choice of http*)
  - infrastructure to be put in place

Comparing Solutions

- Mixed solution?
  - one solution for names, another solution for numbers
  - not simpler...