General (Networking) Working Group

Internet Draft
T. **Glassey**
Document: draft-glassey-dns-rzp-00.txt
ServerWerks Inc.
Expires: 00/2003
June 2002

ROOT ZONE PROTOCOL
draft-glassey-dns-rzp-00.txt

Some thoughts on a proposed change to DNS

Status of this Memo

Abstract

The changing structure and size of today's Internet has taxed the
existing name services and their architecture to the breaking
point. The DNS system of today was unfortunately architected to
have a single root zone and limited set of Top Level Domains. This
is defined usually in the set of root servers any resolving
request uses to lookup an address.

This proposal then is an attempt to lessen the impact on end-users
and registrars and to allow IP owners a greater freedom in
representing namespace to their customers.

The elevator description is that this I-D proposes the
restructuring of domain names more fully along the lines of a

telephone number by creating a ROOT ZONE PROTOCOL as an addition
to multiply the existing DNS services times the number of ROOT
ZONES.

Table of Contents

[1](#). Document Scope
This document addresses a method or reducing the Intellectual
property 'collisions', the confusion, pain and suffering with
allocating names in DNS land, and does so by creating a discovery
protocol for what has until now been a static part of the DNS
infrastructure. The DNS Root Zone Tables.

The Intent of the Root Zone Protocol (RZP) is to allow for the
dynamic mapping of DNS Root Zones at the DNS Server Level and it
identifies most all the components that the new Syntax Model would
affect as well.

This document is a very high level document and will require a
number of iterations and the support of other documents to
implement its proposed technology including but not limited to a
modified version of [RFC1034](#) and 3538 to support its changes at the
very least, and likely some changes to the email world to make use
of these capabilities.

1.1 Document Audience
I assume you know the history of Names and Hostfile formats as per
the original works starting at RFC608 and proceeding onward from
there. I also assume you know Mockapetris' work on the basic DNS
RFC's (starting with RFC1034 and others) and their evolution
through what is in place today.

I also assume that you understand how zone management works and
how DNS is globally administered by the Internet Registrars and
how the current Zone Tables are managed OOB.

2. Conventions used in this document

2.1 Key Words
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 [ ].

2.2 Special terms
The following TERMS are coined in this Document:

2.2.1   ZONE's
One specific change is the difference in the use of the term ZONE
relative to RFC-2136's use. Their use of the term ZONE refers to
which zone within any given root hierarchy. Our use of the term
ZONE here is particular to which the Root Hierarchy is being used
to resolve any domain name in question.

Zones at the use level are represented by Root Tags. The Root tag
may be a string of up to 24 characters allowing for the creation
of 26 ^^ 24 instances of possible zone names.

2.2.2   ZONE Publication
All registrar's supporting Multi-Root operations will make
available online an accurate representation of the ROOT ZONES they
publish domains in. For each ZONE they add, the registrar and the
registrars alone will be responsible for maintaining the accuracy
of the Internet's Zone Mapping table(s).

Further, this ROOT ZONE publication table will be made available
in whatever form the implementation of this protocol extensions
demands including through active LDAP lookup or just a simple
recursive DNS type model.

2.2.3   Globally Qualified Domain Name (GQDN)
The GQDN is a "root zone" enhanced notation model using what was
known as a Fully Qualified Domain Name and adding the root zone

tag as a prefix to the domain name.

2.2.4   Locally Qualified Domain Name (LQDN)
The Locally Qualified Domain Name is identical to what we were calling the Fully Qualified Domain Name. It is called locally qualified since it is assumed to be a part of the system's default root zone.

2.2.5   Registrars ZONE TABLES
A locally produced DB.CACHE file and listing of all the ZONES this registrar resolves for.

2.2.6   Zone-Lookup
The Zone Lookup is the process of querying the master ZONE TABLE SERVER or a Registrar's local ZONE TABLES for an associated set of ROOT ZONES for any supported set of ZONES.

The Operating Authority and all the ZONE-Enabled Registrars maintain ZONE Tables.

2.2.7   Zone-Failure
The Failure of the ZONE SERVER to have the correct lookup tables for the requested ZONE(s). It is Zone-Failure that causes the local server to start a ZONE RESOLUTION transaction with the master ZONE TABLE Server.

2.3 DNS ROOT Servers
This list of servers defines all of the representations and who will resolve them for this "DNS Hierarchy". Traditionally this is distributed as the DB.CACHE file that BIND uses at boot time. Other Named implementations place it in various other places including as a part of the Active Directory in Microsoft Land.

2.3.1   Single DNS Root
This refers to the first implementation rounds at the DNS protocol. It was extended to conform to any number of lookup and record transmittal models all based on the defined hostname and Domain Name structures. In the original DNS service models there is only one root zone and it defines the scope of what DNS can and can't define.

2.3.2   Root Zone Database (RZDB)
The RZDB is a local database maintained by the DNS Server regarding which roots it knows how to resolve addresses from. The ROOT ZONE DB or RZDB is a policy centric cache of locally stored DNS ROOT SERVER lists.

The Lists are stored locally in textual form and are flushed on a
DNS Policy Timeout basis as defined in the ROOT-CACHE-LIFE
parameter being added to the Config file (named.boot).

3. Goals of this I-D
The goal of this I-D is to propose a new system for the dynamic
exchange of ROOT SERVER Data and thus the opening for the ICANN
Registrars to support any number of roots. We clearly acknowledge
that there are other ways to accomplish aspects of what the Root
Zone Protocol enables, but not all of the features can be
reproduced through other methods so it is considered a real
potential

This new system and nomenclature for representing DNS addresses in
textual form with the addition of a ZONE or "Root Server"
resolution protocol will increase the number of available DOMAIN
NAMES in a common format such that any Domain Name can take on a
more simple and easier to understand format.

The net effect of this will be the increasing of the potential
number of ROOT ZONES, ad infinitum. This author feels that this is
the best way to meet the commercial expansion of the Internet.

4. Existing DNS constraints û the starting point.
Existing DNS services are architected around a common root set of
name servers. These common servers act as an anchor, but
specifically limits the root instance to that of the original
ARPA. Root's. Today's ROOT Server Lists are maintained by the
various Registrars and distributed to their clients out of band.

This means that at this time, for the totality of the public
Internet, that there is only one of any given name and that the
names are parsed from the bottom up (See RFC608. RFC810, and
RFC1033/1034, RFC2308, RFC2826 and RFC3258 for more detail).

4.1 Domain Names, IP and the UDRP Issues
This facility it is felt will greatly relieve the IP Congestion
that today's "One and Only Naming Convention" put in place.

This document acknowledges that there are Intellectual Property
(IP) similarities between textual, network addresses and the
absolute or ASCII representation of the text inside a registered
trade or service mark, but that at this time (06/2002), the law
surrounding this is still somewhat vague and untested, and as such
that the ICANN's UDRP (Uniform domain Dispute Resolution Protocol)
was developed to protect ICANN and the registrars from IP suits.

To date the UDRP hasn't worked to well, but much of the need for
it is based in the "there can only be one of any network name" and
for the most part this is based on that there is only one dot com,
net or orgà We hope to address that.

The other issue is which parts of the domain names are allowed by
design to carry any customer definable content, and that is at the
Second Level Domain Name (SLDN) only. So this further restricts
the possible combinations from "phrase style identity tags" to the
current resolving structures.

4.2 RFC2826 and its limitations
THE IAB published a document in May of 2000 complaining of the
problems of addressing the dynamic ROOT management of the
Internet. Thy cited three cause why this was not doable today, and
these were:

1)   The need to maintain a common naming and nomenclature
standard
2)   The complexity of the NAME ZONE update processes
3)   And finally that traditional ROOT TABLES were poorly
maintained and were distributed in the existing DNS mode
out-of-band (OOB).

None of these are insurmountable problems and in fact the second
one has no bearing on the issues of ROOT ZONE MANAGEMENT.

4.3 Need to service a more compartmentalized Internet
Nowadays, and for whatever reason, the Internet is becoming more
and more compartmentalized. This may be due to concepts like
eBorders and jurisdictional dilemmas being resolved in this
manner, or that the economics of the Internet are directly tied to
the economic outlook of the world and that since the slowdown of
last year and 9/11 that drastic changes in what the Internet
physically consists of are now in play. Even the Internet
Architecture Board in their report of May 2000 criticized the
existence of only a single DNS root [rfc-2826].

4.4 The limitations of existing naming conventions
The real issue with textual name space is how small the usable
portion of it is. There are two key constraints defining the scope
of the namespace that is relevant in the current model and that is
the totality of the second level domain names. There is no way for
the Domain Client to define the Top Level domains, and since these
are the same for all Registrars pretty much, this means that the
totality for the space they can sell is that of the 2nd level
domains.

Inside of this Second Level Domain envelope is the further

limitation of that relatively very few of the possible
combinations of letters actually form words or symbols that are
identifiable, further constricting the totality of the number of
realistically representable names that and TLD can support.

And when you compound that with existing domain name needs, there
is a meltdown coming of available names.

5. The New proposal û ROOT ZONE Extended DNS Services.
This new proposal for bringing ROOT ZONE MANAGEMENT under the
scope of what is provided from common DNS Services, and through
the dynamic opening of ROOT TAG SPACE, will address the creation
of a more wide-open namespace.

The new DNS Nomenclature proposal is based around the addition of
a ROOT ZONE TAG or RZT's as a zone identifier code, which is
prepended like an Area Code, to the front of the existing FQDN.

This system supports the existing FQDN name resolution, but as the
default set of ROOT SERVERS that the DNS SERVER uses to resolve
queries to it.

The Globally Qualified Domain Name (GQDN) will be constructed as
follows:

<[ROOT ZONE TAG]><LQDN>.

And the LQDN (a.k.a. the FQDN) is deconstructed as follows:

<Hostname>.<second level domain>.<TLD>.

o-   Where [ROOT ZONE TAG] is the optional bracket enclosed text
string indicating which group of ROOT ZONE servers to use for
this query.
        o-   Where Hostname = any reasonable RFC compliant Hostname
o-   Where Second Level Domain = the existing settable or user
definable portion of the FQDN, and;
o-   Where the final root of the name is the anchoring TLD or DNS
Zone Table

5.1 LQDN based processing
The LQDN structure is parsed and processed exactly as the FQDN is
today. The name is sent to the NAME SERVER for resolution and the
name server's parser will see that it is a LQDN and so this name
server will use its Default Set of ROOT SERVERS.

5.2 GQDN based processing
If a ROOT ZONE TAG (RTZ) is prepended onto the Host name, the DNS
Server will process that tag to see if it is either the default

tag name or that of an already known root zone. For any root zone
it already "knows" there will be a local copy of that root zone in
the DNS Server's ROOT ZONE Database.

[6](). Software Affected by the proposed changes
It is important to understand that this change and new requirement
may break some existing resolver/parser code since they may be
hardwired to the existing domain structure. But any good
programmer knows that it is easy to create a secondary footprint
and set of parsing rules such that if there is a ROOT ZONE TAG
(RZT) detected in the host's domain address specification, that it
would be resolvable.

The following Services are expected to require some work to
address this new set of ZONE management facilities.

[6.1]()   End-User Clients and their local client-side Resolvers
Depending on how much parsing the client side applications do,
there may be some trivial extending of the parsers to accept the
bracket delimited "[ROOT ZONE TAG]" as the prepended extension to
the LQDN's address.

But in many instances it is more likely that the OS's resolver is
the totality of what the application uses and the apps just pass
text strings to the local resolver input. Therefore extending it
to support the "[ROOT ZONE TAG]" extensions is not such a big
deal, especially since it is easily cut in as an optional feature
in most all Resolvers.

[6.2]()   DNS Servers and their local server-side Resolvers
DNS Resolvers themselves may be there largest part of the recoding
effort, but this next generation syntax support can be rolled
within a major BIND release, so this will not be such a painful
effort. Once at least one generation of RZP enhanced DNS is
released the rest of the infrastructure will be deployed as well.

[6.2.1]()   Changing sets of Root Servers dynamically
The key issues with the DNS Servers that needs to be upgraded to
support root ZONE management is the process within a running DNS
server of changing its root pointers.

Most DNS Servers' root pointers are currently read from static
files as text strings, and in most instances, at least UNIX ones,
there is a need to create a dynamic set of pointer indirections
that the actual values pointed to by the ROOT SERVER Template, can
be mapped in and out of at will (and without a hang-up or HUP
signal).

This is easily accomplished by creating a dummy list, or template
as a global and then loading the named selected ROOT SERVER
Entries into it at will.

6.2.2   Need to add the Root Zone Database (RZDB)à and it's processing.
Somehow the local server needs to create a ROOT CACHE for each
ROOT ZONE it is to resolve for. These zones can remain pretty
static or expire depending on who and what. The only real reasons
that they would change once downloaded would be that registrar was
significantly changing their service topology or going out of
business. So its no big deal to cache the zone tables at any DNS
server for 6 months at a time or more;

6.2.3   Other TOOLS for DNS management - Dig, nslookup, etc.
This is the real heartbreaker. For tools that do not rely on the
OS's underlying resolver but supply their own, there is a
significant amount of recoding here at the toolsmith level to deal
with the expanded namespace and ROOT ZONE resolution.

Oh well, that's life is my only response here. We as a culture
need the ROOT SPACE too much to not suffer whatever pain this
causes.

6.3 How this all fits togetherà
The Use Modelsà   Client requests the resolving of a GQDN's
address from the Name Server. The server parses the ROOT ZONE TAG
(RZT) from the GQGN and looks up in its local ROOT ZONE TAG Cache
to see if the TAG and its equates exist there. If so - Then the
server checks the expiry of the equate list to make sure it can
use the entry. If the expiry is still good then it resolves the
name and passes the address back to the client's Application
Otherwise ZONE FAIL occurs.

ROOT ZONE FAILs can be addressed through policy in a number of
manners, and as a default mode, we propose that the server
attempts to recover a new copy of the ZONE TABLE from the master
server or from the previous supplier of this ZONE TABLE, which was
also cached locally in the Server. If the ZONE recovery is
completed then the DNS Server caches the updated ZONE Table entry,
updates its ZONE EXPIRY and continues to process the name
resolution.

To continue, with the new ZONE TABLE's list of root servers, the
DNS Server looks up the address and assuming a successful
response, sends the address onward into the client. If there is a
lookup failure then the user is told that there was a standard DNS
resolution error, or the process times out and the client barfs
appropriately, just as with today's SINGLE ROOT service model.

If the ZONE TABLE transfer fails then the DNS server has the
choice of continuing to use the old table if it works and
continuing the attempt to resolve the name.

7. Interim Operations

The problem with Interim operations is that there may be name
collisions between ROOT 'a' and ROOT 'b' such that addresses that
exist in both are never reached in the second ROOT because those
in the first root will take precedence.

To this end we see ROOT ZONES and the fact that there are now
several mutually exclusive ones up and functioning on the Internet
as a serious problem without this bridge process to allow for
selective ROOT ZONE management.

8. Security Considerations

There are no real security considerations with this change or
augmentation of existing DNS Services.

9. References

[RFC-830]   Z. Su, "A Distributed System for Internet Name
                                        Service", IETF RFC-830,
Network Information
                                        Center, SRI International,
October 1982.
                                        Contains early thoughts on the
design of the
                                        domain system.

[RFC-882]   P. Mockapetris, "Domain names - Concepts and
                                Facilities," RFC-882, USC/Information
Sciences
                Institute, November 1983.

[RFC-883]   P. Mockapetris, "Domain names - Implementation and
                Specification," RFC-883, USC/Information

                    Sciences Institute, November 1983.

[RFC-920]   J. Postel and J. Reynolds, "Domain Requirements",
                                    RFC-920, USC/Information Sciences
Institute,
                                October 84.

[RFC-952]   K. Harrenstien, M. Stahl, E. Feinler, "DoD
Internet
                                    Host Table Specification", RFC-952, SRI,
October
85.

[RFC -1033]   M. Lottor, DOMAIN ADMINISTRATORS OPERATIONS GUIDE
                                    SRI, International, November 1987

[RFC-1034}   P. Mockapetris DOMAIN NAMES - CONCEPTS AND
                                    FACILITIES, ISI November 1987

[RFC 2136]   Vixie, Et al., DNS Update, April 1997

[RFC 2308]   M. Andrews, DNS NCACHE (Negative Caching of DNS),
                                    March 1998

[RFC-2782]   A. Gulbrandsen, P. Vixie, L. Esibov, DNS SRV
Resource
                                    Records, February 2000

[RFC-2826]   IAB Technical Comment on the Unique DNS Root, May
                                    2000

[RFC-3258]   Distributing Authoritative Name Servers April 2002

10. Acknowledgments

Vint Cerf û For taking all the crap I have dished out to him for
all these years. My friends in the IETF including those PKIX and
POISSION WG Chairs and its Secretariat who I also gave merde to
for so long.

Those inventors of DNS and the Host Naming Conventions named in
the references section of this document and those critical
Telephone Pioneers (John Draper and others).

11. Author's Addresses

Todd S. Glassey
ServerWerks
PO Box 0026, Menlo Park, Ca., 94026
Phone: 650-926-9609

Email: Todd.Glassey@ServerWerks.CC