IDN Deployment Test Results

Lars-Johan Liman
Autonomica AB

Abstract

Autonomica AB has, under a contract with ICANN, investigated whether the addition of top level domains containing encoded internationalized characters (so called IDNs) would have any impact on the operations of the root name servers providing delegations, or the iterative mode resolvers used to look up the information. No impact at all could be detected. All involved systems behaved exactly as expected.

$Id: idn-report.tex,v 1.9 2007/02/15 12:58:23 liman Exp $

Contents

1 Background

ICANN is responsible for the management of the name space in the highest level of the domain name system. ICANN wants to deploy a new type of top level domain in the public DNS system – domain names that contain encoded versions of names expressed with other characters than those in the English alphabet, so called “internationalized domain names” (IDNs).

ICANN has requested that two external parties test what the technical impact, on the DNS client side, of deploying such IDNs as top level domains in the public root would be. The scope of the test is limited to adding these domain names. There are no new record types or DNS “classes” involved, only the classical NS (name server) records for delegations, and the connected A (IPv4 address) records.

With IDNs, the domain names stored in the DNS servers are ordinary domain names just like before. The names stored have no special properties that makes it possible for the DNS servers to single out the IDN domains. There is no reason to believe that IDNs would make the DNS system as a whole behave different from its normal behaviour. Nevertheless, for prudence ICANN has asked that it be tested that this assumption is true.

It should also be noted, that the root name servers play a very small but crucial role in the DNS system. The root name servers can in principle only return two types of valid answers to valid DNS queries.

To queries about domain names in existing top level domains, the root name server will return a “referral” to a top level domain server (“go and talk to him over there”). For a query regarding a domain name in a non-existent top level domain, it will return a message indicating ”non-existent domain” (NXDOMAIN). This is the expected behaviour. There are of course a number of cases where the query is totally invalid (broken), and in these cases other return messages may appear, such as refusal to handle the query (REFUSED) or query format error (FORMERR).

In the tests we have not generated any queries with format errors, as they have no special relation to IDNs.

The tests were to be carried out in a closed lab environment and were expected to cover various implementations of server and client software to a degree that makes ICANN comfortable when making the decision whether to add these IDN TLDs to the public DNS or not.

Autonomica AB has been contracted to undertake such testing, and this is a presentation of the results of the tests. In the lab environment we were also able to measure the time it took to obtain the final response. With the minimal network delays involved, almost all delay must be related to the software, and unexpected delays would have been noted.

2 System Setup

The test setup is described in a separate document: “IDN Deployment Test – Test Setup”.

The actual address plan used in the test is appended in Appendix B.

The tests were carried out in Autonomica’s test facilities in Stockholm. The test system consisted of the following blocks

1. Two root name servers (root).
2. One top level domain authoritative server (TLD).
3. A number of different iterative mode resolvers (IMRs).
4. Query generator (client).
5. Connecting network.

Only the qualitative performance of the root name servers and the iterative mode resolvers were evaluated in this test, according to the demarcations in the contract.

2.1 Root Name Servers – root

The root name servers were installed on a Unix platform. Two major DNS server implentations were tested, known to be used by root server operators. They were

1. `BIND` version 9.3.2 from the Internet Systems Consortium, and
2. `NSD` version 2.3.5 from NLNet Labs.

To the best of our knowledge these versions are very close to what is actually used by the root server operators on the Internet.

We have tried to mimic the existing root name servers to the best of our ability. Two servers were operated, and they each served DNS from two IP addresses to ensure that the IMR software wouldn’t have problems based on the fact that only a single server is reachable.

The root servers contained a copy of the active root zone on the Internet, augmented with the IDN test domains supplied by ICANN (see Appendix A).

2.2 Top Level Domain Name Server – TLD

The TLD server was set up to “catch” the IMRs as they were referred by the root servers to the TLD servers. A normal DNS query is referred downwards from the root to create a full “path” from the root to an existing and properly configured name server that provides final (terminal) DNS records. The

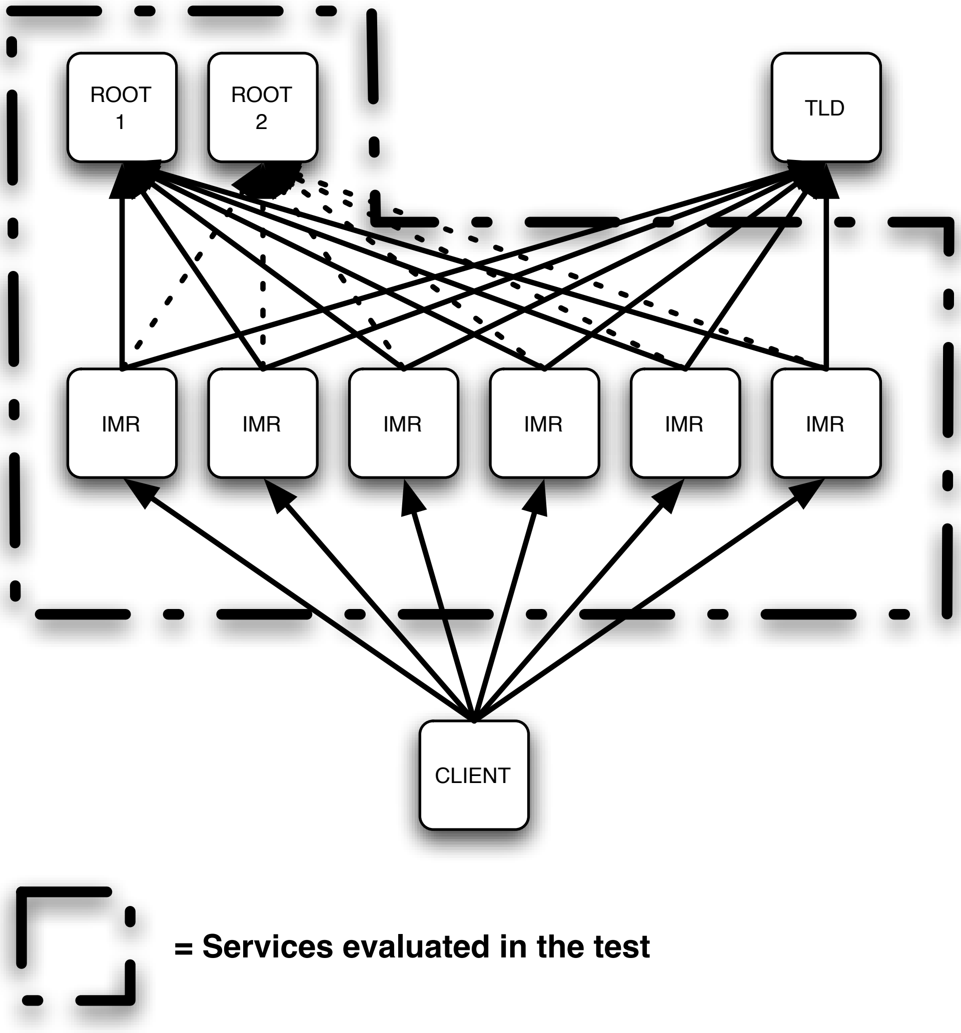

Figure 1: Service topology for the test

TLD service was provided from a separate IP address using a separate DNS configuration.

The TLD zones contained a few DNS records that could be queried for, to make the query process reach a final answer and come to full completion. All the TLD zones were stored on one and the same server, a name server serving multiple TLDs, as is commonly the case on the Internet.

2.3 Iterative Mode Resolver – IMR

The IMRs were set up side by side on parallel servers. They were set up as "empty" iterative mode resolvers, where the only substantial configuration is which root name servers to use. This configuration was changed between different phases of the test, (see below).

The IMRs were then queried for various DNS information, and the results were recorded for verification and comparison.

The IMR software versions tested were:

1. ISC BIND version 8.3
2. ISC BIND version 8.4
3. ISC BIND version 9.0
4. ISC BIND version 9.1
5. ISC BIND version 9.2
6. ISC BIND version 9.3
7. Microsoft DNS Server as shipped in Windows 2000 Server
8. Microsoft DNS Server as shipped in Windows 2003 Server

The number of implementations of IMR software is vast. There was no reasonable way one could test all versions of all software on all platforms. To make this at all feasible, we had to limit ourselves to the most common platforms, which are various versions of ISC BIND, and various versions of Microsoft DNS servers. Apple Macintosh uses BIND, and most, if not all, Unix vendors ship BIND as their primary DNS server. There is a plethora of alternative server platforms, but they are counted in far smaller numbers than those above above.

Since we were looking for possibly broken software, we chose not to test the most recent versions of the software, but the most *ancient* versions of the most common minor versions of BIND, and the basic installations – without any service packs – of the Windows 2000 Server, and the Windows 2003 Server in the belief that the service packs improve the software, and we really want to test ”worst case”.

Windows Vista server was not yet released when the test was performed, and Vista servers can therefore almost by definition not constitute a major part of the resolvers on the Internet.

3 The Test

3.1 Installation

The software platforms were standard installations of operating systems without any special configuration. In the Microsoft cases, plain installation from distribution CDs using default configurations was used, except that we checked the checkbox for “install DNS server software”, an obvious prerequisite for the test.

3.1.1 Root Servers

The root name servers were configured with a zone file based on a copy of the active (live) root zone file, as provided by VeriSign. It was modified to match the IP addresses in the lab setup, and delegations of the IDN TLD names specified by ICANN were appended using ordinary NS records and accompanying glue records.

Identical root zone files were of course used with the BIND and NSD root name servers.

3.1.2 The TLD Server

The authoritative TLD server was set up as one single server, serving all of the IDN TLDs delegated from the augmented root zone. The TLD service was provided from a separate IP address to avoid the untypical situation where a server serves multiple levels in the DNS tree from the same zone database, which would remove some of the properties of the DNS that we explicitly want to test.

Zone files were generated for all the IDN TLD zones in the test.

3.1.3 The Iterative Mode Resolvers

The various BIND versions were installed by compiling from source code (program text) according to the default installation procedure.

In a few cases, minor tweaks were necessary to make old versions of the code work on the more recent version of operating system of our choice. E.g., the names of some properties of the operating system have been changed, which made it necessary to make the same name changes to the program code. No functional changes were made.

The Windows nameserver code was included in the Windows system installation and required no extra steps.

3.1.4 The Client

The client used was an ordinary workstation connected to the same network as all the servers. The DNS debug tool “dig” (version 9.3.2-P1) was used, as it provides a plethora of information about the DNS traffic while it acts as any other application when sending a DNS query.

4 The Process

4.1 Root Servers

The first phase was to configure all the IMRs to use only the two BIND root name servers’ IP addresses. This way the interaction between all the IMR versions and a BIND root name server was specifically tested.

The second phase was to configure all the IMRs to use only the two NSD root name servers’ IP addresses. This way the interaction between all the IMR versions and an NSD root name server was specifically tested.

4.2 Iterative Mode Resolvers

All IMRs where restarted before each set of queries to ensure that their caches were empty and pristine.

4.3 Queries

Each of the IMRs was then sent several series of queries. The lists of domain names used for querying the IMRs were generated based on the list of IDN TLDs provided by ICANN. See below for more specific examples of queries.

4.4 Answer Processing

In all cases, the results were recorded on file and checked for any signals of bad process or bad data. Response times were noted in the process, to look for unexpected tardiness.

The tests were run several times using different TTL (Time To Live) values to investigate how the IMRs behave when records time out and are automatically cleared from the cache.

5 Results

5.1 Phase One: BIND Root Servers

The following results were noted during the first phase of the test, using BIND root name servers:

5.1.1 Existing Terminal Nodes – Host Name in ASCII

The first series of queries was made up by looping through the IDN top level domains, adding the prefix "www.", and query for the IPv4 address record type (A) in the Internet class (IN), thus making a list like:

```
www.xn--18-7g4a9f.         IN      A
www.hippo18potamus.        IN      A
www.xn--18-xf0jl42g.       IN      A
www.xn--18-h31ew85n.       IN      A
www.xn--flod18hst-12a.     IN      A
www.xn--18-xsdrfd6ex1e.    IN      A
...
```

This creates a list of existing terminal nodes in the delegated TLDs in the test system where the left label is in pure ASCII, and the right label is an encoded IDN. The intent was to ensure that the entire process of following the referral from the root to the TLD, and actually retrieving the final data, works as intended.

The expected result was that the IMR would be able to find and return the terminal A records in reasonable time.

Results: *All answers were consistent with expected behaviour, and no unexpected delays were discovered.*

5.1.2 Existing Terminal Nodes – Host Name in IDN

The second series of queries was made up by looping through the IDN top level domains and adding a prefix of the IDN name itself to its own TLD, thus generating two IDN labels in the same domain name. The record type used was still the IPv4 address (A) in the Internet (IN) class, thus creating the list:

```
xn--18-7g4a9f.xn--18-7g4a9f.            IN      A
hippo18potamus.hippo18potamus.          IN      A
xn--18-xf0jl42g.xn--18-xf0jl42g.        IN      A
xn--18-h31ew85n.xn--18-h31ew85n.        IN      A
xn--flod18hst-12a.xn--flod18hst-12a.    IN      A
```

```
xn--18-xsdrfd6ex1e.xn--18-xsdrfd6ex1e.  IN      A
...
```

The expected result was that the IMR would be able to find and return the terminal A records in reasonable time.

Results: *All answers were consistent with expected behaviour, and no unexpected delays were discovered.*

5.1.3 Non-existent Terminal Nodes – Host Name in ASCII

To identify possible failures in the lookup process, the IMRs were also queried for domain names with non-existent terminal nodes to make the IMRs follow the referral from the root but be unable to acquire the final data. They were all queried for

```
xyz.xn--18-7g4a9f.         IN      A
```

where the “xn–18-7g4a9f” TLD existed but the “xyz” record did not exist in the TLD.

The expected result was that the IMR would not be able to find and return any terminal records and would signal this with an NXDOMAIN response in reasonable time.

Results: *All answers were consistent with expected behaviour, and no unexpected delays were discovered.*

5.1.4 Non-existent IDN TLD

Another test to identify possible failures in the lookup process was performed by querying the IMRs for a host in a non-existent IDN top level domain thereby making the process terminate at an early stage, as the root server would give a “non-existent domain” (NXDOMAIN) return status. All the IMRs were queried for

```
www.xn--18-7e4a9f.         IN      A
```

The expected result was that the IMR would not be able to find and return any terminal records and would signal this with an NXDOMAIN response in reasonable time.

Results: *All answers were consistent with expected behaviour, and no unexpected delays were discovered.*

5.2 Phase Two: NSD Root Servers

The IMRs were reconfigured to use the NSD root name servers. All servers were restarted and the exact same queries were sent to the IMRs. For specifications of queries and expexted results, see section 5.1.

The following results were noted during this second phase of the test:

5.2.1 Existing Terminal Nodes – Host Name in ASCII

Results: *All answers were consistent with expected behaviour, and no unexpected delays were discovered.*

5.2.2 Existing Terminal Nodes – Host Name in IDN

Results: *All answers were consistent with expected behaviour, and no unexpected delays were discovered.*

5.2.3 Non-existent Terminal Nodes – Host Name in ASCII

Results: *All answers were consistent with expected behaviour, and no unexpected delays were discovered.*

5.2.4 Non-existent IDN TLD

Results: *All answers were consistent with expected behaviour, and no unexpected delays were discovered.*

5.3 Timing

Unexpected delay in the handling of the queries would have been percieved as a quality degrading property. During the tests, the longest response time noted was 24 ms (0.024 seconds), which is very well within the expected limits and percieved as “instantaneous” by the user.

5.4 Availability of Technical Details

The resulting output files from most of the test runs have been retained, and will be made available upon request. Please contact the author for access to these files. Also, if the reader has more specific technical questions, the author is happy to answer these to the best of his ability. Contact information below.

6 Conclusions

During these tests, we were unable to detect any deviation at all in any part of the system from the normal behaviour of DNS systems.

The addition of IDN strings seems to have had no measurable effect at all on the qualitative performance of the test systems.

7 Author's Contact Information

The author of this document is Lars-Johan Liman, Senior Systems Specialist at Autonomica AB, Stockholm, Sweden. He can be reached at:

Autonomica AB
Att: Lars-Johan Liman
Bellmansgatan 30
SE-118 47 Stockholm
Sweden

Tel: +46–8–6158572
Fax: +46–8–4420967

E-mail: liman@autonomica.se

APPENDICES

A IDN Test Strings

The list of TLD test strings to be used in the test, as provided by ICANN, is:

```
xn--18-7g4a9f
hippo18potamus
xn--18-xf0jl42g
xn--18-h31ew85n
xn--flod18hst-12a
xn--18-xsdrfd6ex1e
xn--18-dtd1bdi0h3ask
xn--18-28gg3ad5hl2fzb
xn--18-hmf0e1bza7dh8ioagd6n
xn--18-rjdbcud0neb9a8ce1ezef
xn--1818-63dcpd5be6bfqcecfadfad3dl
xn--1818-1goc0bacbac7eg2kh6ci9cj9bk4yla7ablb
xn--181818-qxecc5edd8aee8aebebecadeadead0fkkill5ymam
xn--flod18hstflod18hstflod18hstflod18hstflod18hstflod18-1iejjjj
hippo18potamushippo18potamushippo18potamushippo18potamus18hippo
```

B Address Plan

The IP addresses used in the test were the following:

IP address	**Service function**
Roots:	
10.10.10.10	BIND 9.3.2-P1 root
10.10.10.11	BIND 9.3.2-P1 root
10.10.10.20	NSD root
10.10.10.21	NSD root
TLD server:	
10.10.10.30	TLD server for all the IDN TLDs in the test.
Resolvers:	
10.10.10.83	IMR BIND version 8.3
10.10.10.84	IMR BIND version 8.4
10.10.10.90	IMR BIND version 9.0
10.10.10.91	IMR BIND version 9.1
10.10.10.92	IMR BIND version 9.2
10.10.10.93	IMR BIND version 9.3
10.10.10.200	IMR Microsoft Windows 2000 Server
10.10.10.203	IMR Microsoft Windows 2003 Server