A Survey of the Resiliency of the Domain Name System

By Michael Schiraldi, under the supervision of Henning Schulzrinne

{mgs21, hgs}@cs.columbia.edu

Contents

1 Introduction

The Domain Name System may well be the most critical service on the Internet, as virtually every other protocol and service relies upon it. Due to its need for authoritive sources for its information, some centralization is a necessary part of the protocol. This is a potential weakness in the protocol – without careful design, an organization may develop critical points in its DNS infrastructure such that a single failure could severely disrupt its DNS, and thereby all of its external and possibly even internal network operations.

With this in mind, we set out to study the prevalence of such critical points and to determine the general stability of the DNS system.

1.1 Areas studied

The specific areas measured are as follows:

- Uptime percentage How often are servers operational? How often are domains resolvable?
- Domain consolidation How many domains does the typical nameserver host? How many nameservers does the typical domain have?
- Server dispersal Are all of a domain's nameservers typically on the same subnet, or are they located on far-flung networks?

2 Uptime percentage

2.1 Rationale

Our first objective was to measure how often domains are resolvable, to get a coarse idea of the general state of the DNS system.

2.2 Methodology

In order to have a manageable workload, approximately 1 out of every 20,000 domains was randomly selected from VeriSign's master list of all domains in the com, net, and org zones. This resulted in a sample set of 2,369 domains.

Approximately every four hours from November 16 through December 15, 2002 (a total of 174 trials), a query to each domain's authoritative nameserver asking for the domain's A records was attempted. Any results, positive or negative, from any of the domain's authoritative nameservers, was considered successful – the domain was recorded as being "up" at that time. However, if no response was received, or if the gTLD servers reported that the domain did not exist, it was considered "down."

2.3 Results

The results of the one-month survey were as follows:

```
 426 domains were never up
  25 domains have a DNS uptime of between 0% and 4%
   5 domains have a DNS uptime of between 5% and 9%
   0 domains have a DNS uptime of between 10% and 14%
  10 domains have a DNS uptime of between 15% and 19%
   4 domains have a DNS uptime of between 20% and 24%
   5 domains have a DNS uptime of between 25% and 29%
   5 domains have a DNS uptime of between 30% and 34%
   5 domains have a DNS uptime of between 35% and 39%
   3 domains have a DNS uptime of between 40% and 44%
   3 domains have a DNS uptime of between 45% and 49%
   3 domains have a DNS uptime of between 50% and 54%
   6 domains have a DNS uptime of between 55% and 59%
   6 domains have a DNS uptime of between 60% and 64%
   9 domains have a DNS uptime of between 65% and 69%
   4 domains have a DNS uptime of between 70% and 74%
   3 domains have a DNS uptime of between 75% and 79%
  11 domains have a DNS uptime of between 80% and 84%
  14 domains have a DNS uptime of between 85% and 89%
  23 domains have a DNS uptime of between 90% and 94%
 368 domains have a DNS uptime of between 95% and 99%
1431 domains were never down
```

Clearly, there is a gulf between the up servers and the down ones. There exist domains that “care” and ones that do not, and very few in between. A bit of manual examination was performed, and it appears that domains with DNS uptime below 90% just “don’t care” – for example, if they had any web presence, it was often the web server’s default welcome page.

This was not the case with domains whose uptime was above 90%. They mostly appeared to be making an effort to run a site and keep it online.

Also studied were downtime “streaks” – the number of consecutive probings in which a domain was found to be down. For each domain, we calculated the longest downtime streak seen during our survey. The results were as follows:

```
 391 domains were never up
   2 domains have a longest-downtime streak of 170-173
  19 domains have a longest-downtime streak of 160-169
  12 domains have a longest-downtime streak of 150-149
  13 domains have a longest-downtime streak of 140-149
  11 domains have a longest-downtime streak of 130-139
   9 domains have a longest-downtime streak of 120-129
   8 domains have a longest-downtime streak of 110-119
   7 domains have a longest-downtime streak of 100-109
   5 domains have a longest-downtime streak of 90-99
   3 domains have a longest-downtime streak of 80-89
  11 domains have a longest-downtime streak of 70-79
   7 domains have a longest-downtime streak of 60-69
  10 domains have a longest-downtime streak of 50-59
   4 domains have a longest-downtime streak of 40-49
   7 domains have a longest-downtime streak of 30-39
  12 domains have a longest-downtime streak of 20-29
   9 domains have a longest-downtime streak of 10-19
  25 domains have a longest-downtime streak of 5-9
  17 domains have a longest-downtime streak of 4
  55 domains have a longest-downtime streak of 3
  34 domains have a longest-downtime streak of 2
 267 domains have a longest-downtime streak of 1
1431 domains were never down
```

Again there was a gap between domains which cared and domains which did not. Manual observation showed that the line should be drawn somewhere near the “10-19” mark – domains which were down for more than ten consecutive probings (or about one and a half days) typically did not appear to be “production-level” sites, but instead appeared to be either experimental, unfinished, or retired.

Finally, we attempted to develop a concept of the “average” downtime. All downtimes, across all domains, were grouped together so that aggregate measurements could be taken.

We found a total of 1683 downtimes, the mean being 52 trials (8.5 days), and the median 2 trials (four hours).

A large portion of these downtimes were for the entire timespan – the domains were down for all 174 trials. After removing these, there were 1292 downtimes remaining with a mean of 15 trials (about 2.5 days) and a median of 1 trial (a single instant).

We further removed the downtimes of length 1, which represented but a single moment in time. At this point, only a mere 523 downtimes remained, with a mean of 35 trials (about 5.5 days) and a median of 8 trials (just over one day).

3 Domain consolidation

3.1 Rationale

Oftentimes, a single machine will be known by multiple names. Similarly, many different domains, representing many different organizations' Internet presence, are often hosted on a single machine or set of machines. We studied this both to gauge the number of domains which could drop off the Internet through the failure of a single set of DNS servers and also to help us locate domain squatters and other such bulk-registrants whom we may want to remove from our dataset.

3.2 Methodology

For each of the 2,369 domains in the random sample described in the previous section, we grouped together domains which resolve to the same set of IP addresses. This was done through a series of simple recursive A record requests made to the local nameserver.

3.3 Results

The results are graphed in Figure 1. Each pie slice with label n represents the domains who share a server with n other domains. For example, there is a pink pieslice labelled "11-50" which takes up about 6.5% of the graph. This is because 154 of the domains in our set meet the criterion: "This domain shares an IP address (or set of addresses) with a group of between 11 and 50 other domains in our set." (154 / 2369 is about 6.5%). The pieslice labelled "N/A" represents the domains that did not resolve.

Because such a large portion of the domains did not resolve, we ran the trial again, this time looking for "www.domain.com" instead of just "domain.com". Figure 2 shows these results.

4 Server dispersal

4.1 Rationale

To see how many domains concentrate all of their DNS servers into a single subnet or group of nearby networks, we mapped out the DNS servers for each domain in our random sample and counted the number of servers found plus the leading bits the IP addresses all had in common. This is the same as the "slash" network containing them all. For example, if a domain was served by 128.59.1.1 and 128.59.200.200, it would have a value of 16. This represents that the set of nameservers have the leading 16 bits in common, or to put it another way, they all reside within the same /16 network. If a domain only had one nameserver, it would get a value of 32 – all the nameservers (i.e. just the one) reside within the same /32.

4.2 Methodology

As in the Domain Consolidation section, a series of simple recursive A record requests were made to the local nameserver. The resulting IP addresses were grouped and tabulated.

4.3 Results

Figure 3 is a histogram showing the number of A records returned for each domain (in other words, the number of nameservers which serve it).

Clearly, the vast majority of domains have exactly two nameservers. Part of the reason for this is that, as seen in the previous sections, there are certain "placeholder" sites that are pointed to by hundreds or thousands of domains. A number of these placeholder sites (such as the 78 in our sample set run by futuresite.register.com or the 68 at VeriSign's 64.225.154.175) happen to have exactly two nameservers, and the physical machines are counted many times – once for each of their domains.

Figure 4 is a histogram showing the number of leading IP address bits shared by all the nameservers for each domain. It forms a sort of inverse bell curve. The values closest to 0 are the best – they represent the most widely dispersed nameservice. The "N/A" datum represents the 104 domains which did not have any nameservers at the time statistics were gathered – they all expired between the time they were selected and the time this test was run.

While it is positive to see so many domains clustered around the top edge of the graph, it is troublesome to see that nearly 40% of the domains surveyed had all their DNS servers within the same /24 network. Presumably, any one of these domains could be entirely taken off the Internet by the failure of a single subnet.

5 Appendix A: Missteps and areas for improvement

We initially hoped to gather and cross-reference all of our data, such that we could determine, for example, the average uptime for all domains which had exactly three nameservers and did not belong to a "placeholder" service. However, when envisioning this, we had forgotten that DNS configurations tend to be complex and varied, often with multiple levels of indirection. This is why BIND's resolver and recursive nameserver are such hugely complex programs.

The project quickly grew too complex for its resources, and in an effort to simplify, it was broken into small, simple, independent units. However, if greater resources were available, we believe it would be fruitful to have a single large database of all the gathered data, which would enable all sorts of cross-referencing features like the ones described above.

We also looked into several different possibilities for isolating the "real" domains – not the parked ones, not the forgotten ones, but the domains that ostensibly are trying to eke out an existence on the Internet. One method we attempted was to find the most-visited domains, as reported in such lists as the Alexa 1000. Incorporating this information into our surveys became untenable after the project was broken up. However, with the suggested large database, this addition would have been much easier to implement.

Another area for improvement is the querying rate for the uptime measurements. Four hours between queries was too coarse to precisely measure most downtimes (a third of the measured downtimes lasted for just a single query at our level of granularity – perhaps many others were missed altogether because they lasted less than four hours), and it additionally limited the number of samples we could take in the time available for this project. The four-hour limitation was due to waiting for the slow-to-resolve domains, or the ones which simply timed out. If the surveying program were multithreaded, and running on a fast computer with a high-bandwidth connection to the Internet, trials could be run every fifteen minutes.

It would also be interesting to focus more closely on the domains with uptimes between 90% and 99%, as these are the ones which really want to be up, but sometimes struggle.

Finally, counting the common leading IP address bits is not the best way to determine whether two machines are on the same network. A better approach would have been to look at the route to each machine, and count the number of routers in common, or the number of routers not in common.

6 Appendix B: Source and data

See HTML version.