An Update on Anomalous DNS Behavior

Duane Wessels
The Measurement Factory, and
CAIDA
wessels@measurement-factory.com

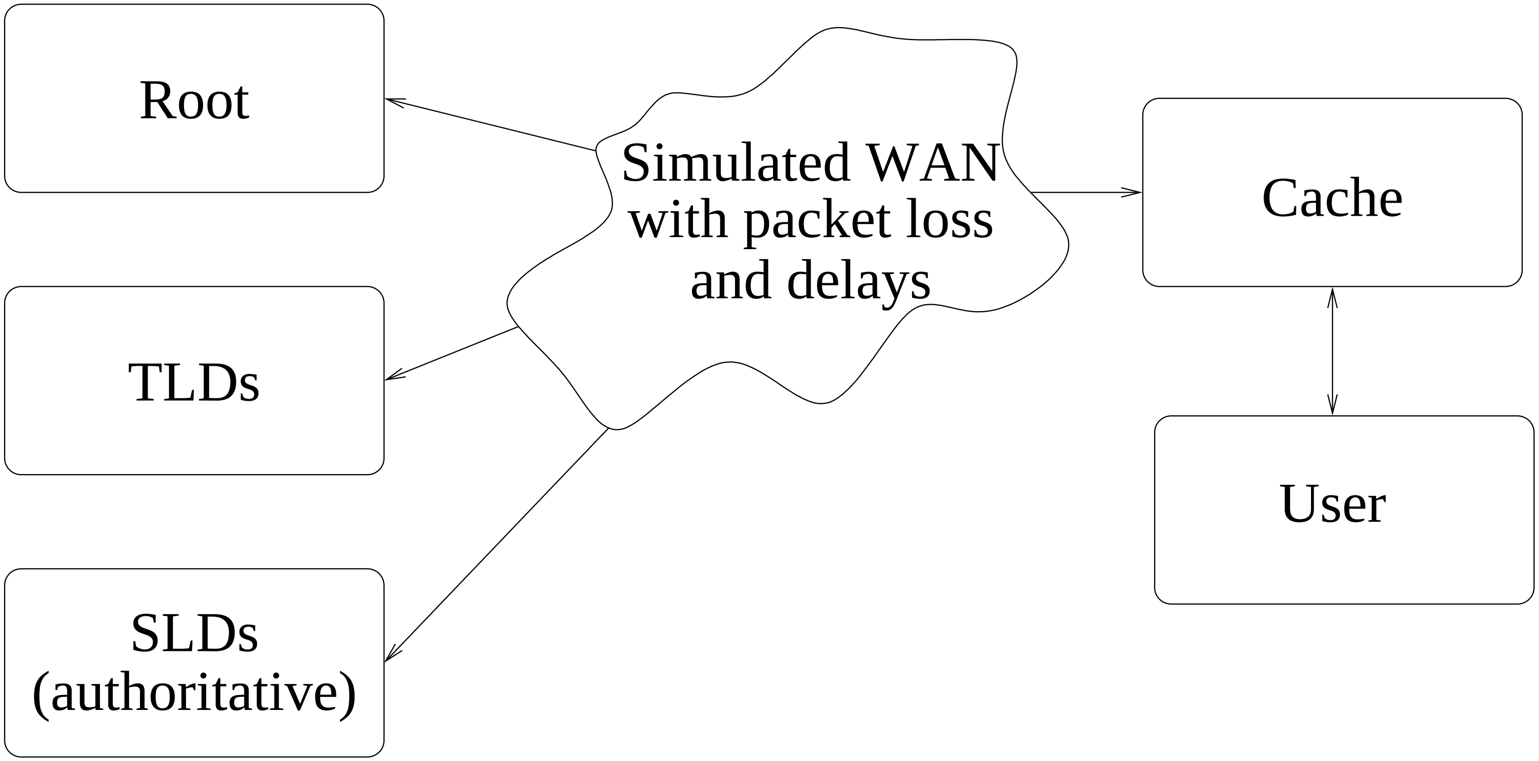
NANOG 29
October 2003

Motivation

- Why are root servers getting slammed?
- Are caching/forwarding DNS servers doing the right thing?
- How does a caching nameserver distribute its load to multiple nameservers?
- Do some implementations abuse the system more than others?

The Setup

A Test Network



Workload

- Hostnames taken from 12 hours worth of caching proxy logs
- 5,532,641 Million DNS requests
- 107,777 unique hostnames
- 70,365 second-level domains
- 431 top-level domains
- 1 Root

Synthetic Zone Files

- Root & TLD zones use real values for:
  - Number of nameservers
  - NS, glue TTLs
- SLD zones use random values derived from sample of real zone data for:
  - Number of A records per name
  - A, NS, CNAME TTLs
- Each SLD zone has two nameservers (ns0, ns1)
- Global 35% probability that a name is a CNAME record

Example SLD Zone

```
$ORIGIN org.
@       8640    IN      SOA     org.    root.org.
        ( 1 720 360 604800 8640 )

@       51840   IN      NS      ns0.org.
        51840   IN      NS      ns1.org.
        51840   IN      NS      ns2.org.
        51840   IN      NS      ns3.org.
        51840   IN      NS      ns4.org.
        51840   IN      NS      ns5.org.
        51840   IN      NS      ns6.org.
        51840   IN      NS      ns7.org.
        51840   IN      NS      ns8.org.

ns0     51840   IN      A       192.168.3.41
ns1     51840   IN      A       192.168.3.42
ns2     51840   IN      A       192.168.3.43
ns3     51840   IN      A       192.168.3.44
ns4     51840   IN      A       192.168.3.45
ns5     51840   IN      A       192.168.3.46
ns6     51840   IN      A       192.168.3.47
ns7     51840   IN      A       192.168.3.48
ns8     51840   IN      A       192.168.3.49
```

```
$ORIGIN 0-vip.org.
@       30      IN      NS      ns0.0-vip.org.
        30      IN      NS      ns1.0-vip.org.
ns0     2143    IN      A       192.168.4.215
ns1     2143    IN      A       192.168.4.216

$ORIGIN 0xdeadbeef.org.
@       1440    IN      NS      ns0.0xdeadbeef.org.
        1440    IN      NS      ns1.0xdeadbeef.org.
ns0     8640    IN      A       192.168.4.95
ns1     8640    IN      A       192.168.4.96

$ORIGIN 1000traveltips.org.
@       30      IN      NS      ns0.1000traveltips.org.
        30      IN      NS      ns1.1000traveltips.org.
ns0     8640    IN      A       192.168.4.27
ns1     8640    IN      A       192.168.4.28

$ORIGIN 1128.org.
@       1440    IN      NS      ns0.1128.org.
        1440    IN      NS      ns1.1128.org.
ns0     90      IN      A       192.168.4.127
ns1     90      IN      A       192.168.4.128
```

Caching NS Software Tested

- BIND 8.4.3
- BIND 9.2.1
- DJBDNS 1.05 (a.k.a. dnscache)
- Windows 2000 (v5.0.49664)
- Windows 2003 (v5.2.3790.0)

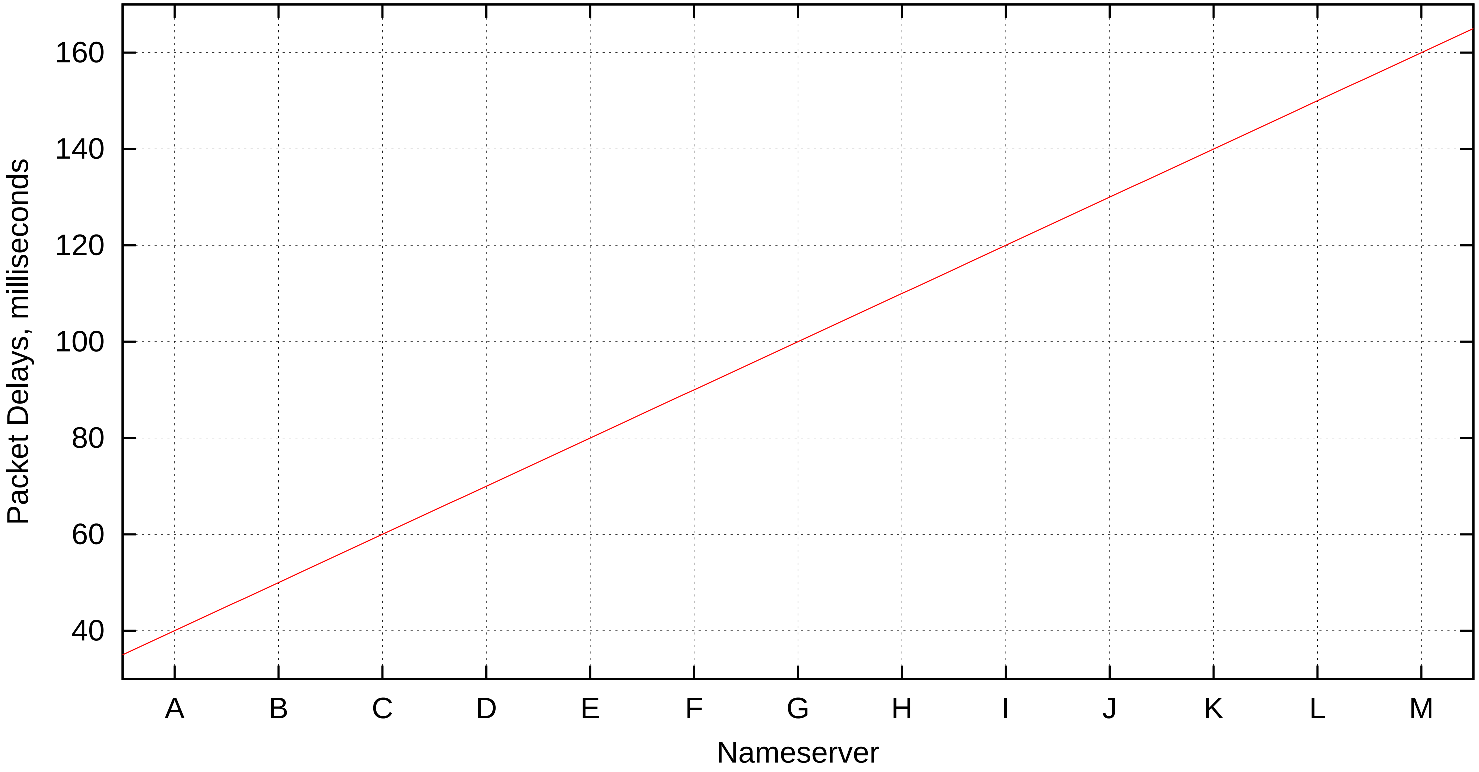
Root, TLD, SLD servers always run BIND 8.4.3.

Test Configurations

1. No delay, no packet loss
2. 100ms delay, no packet loss
3. Linear delays, no packet loss
4. Linear delays, 5% packet loss
5. Linear delays, 25% packet loss
6. No delay, 100% packet loss

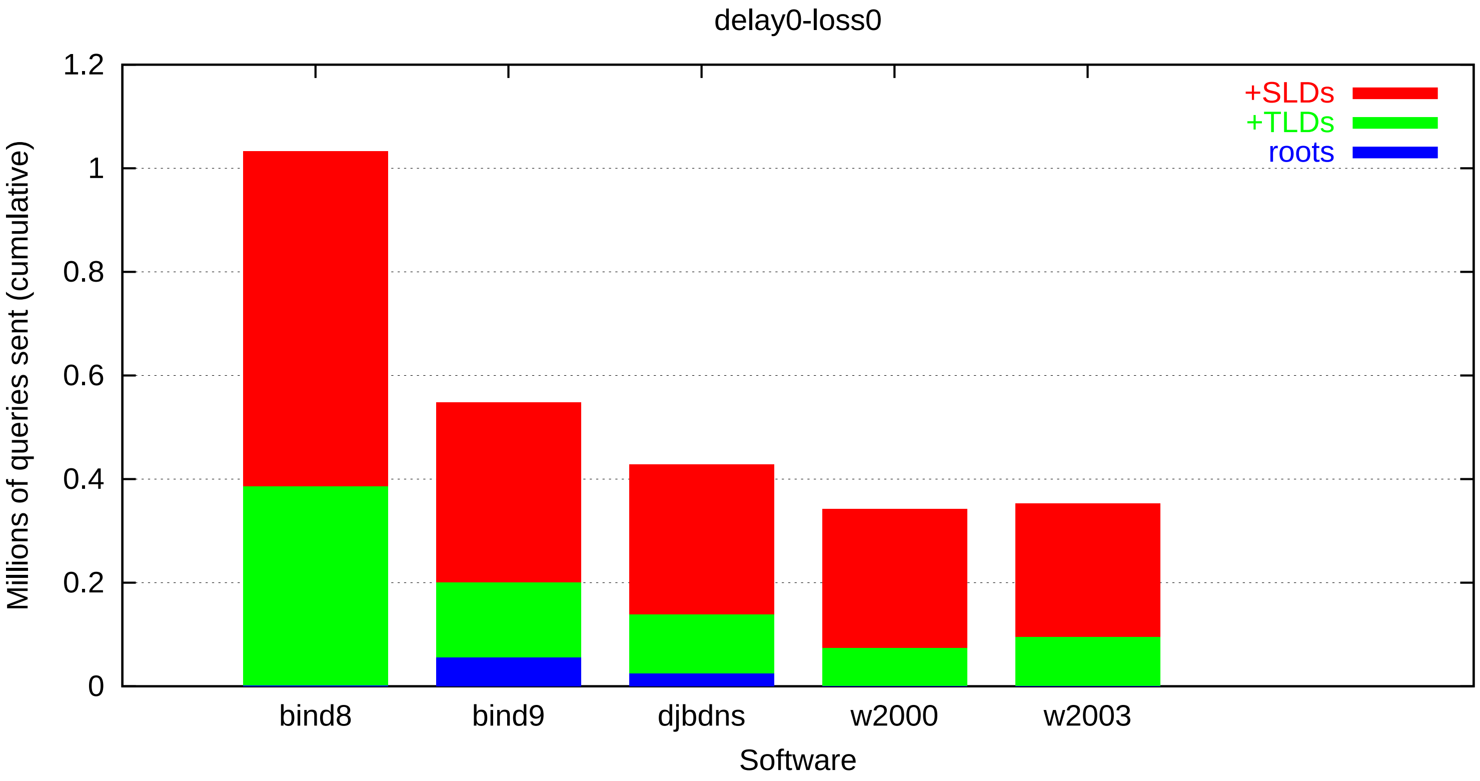
Delays and packet loss are implemented using FreeBSD's *Dummynet*. They are placed on the Root, TLD, and SLD servers.

Linear Delays



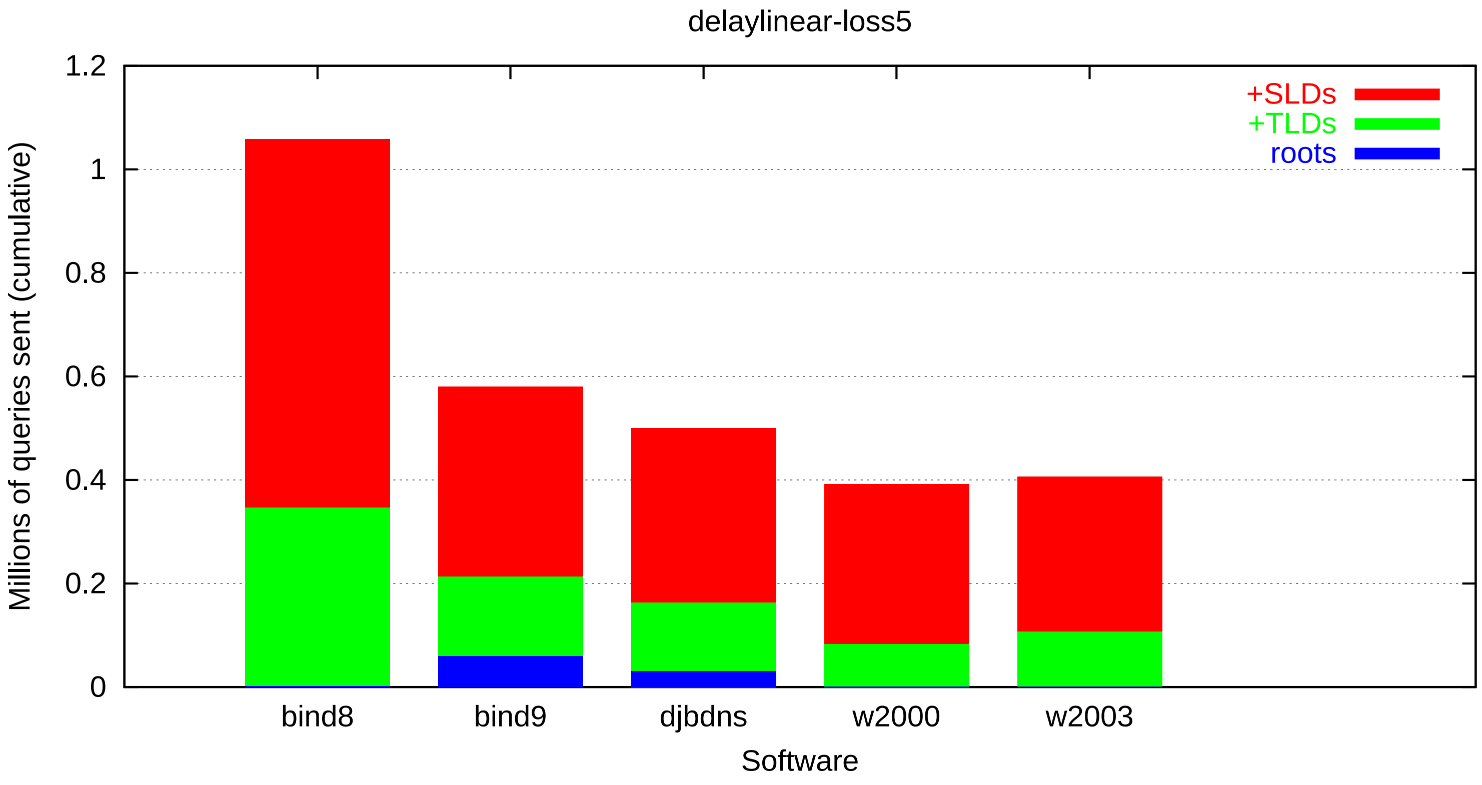
Some Results

No delays, no packet loss



For example, BIND 9 sent 548,671 total queries, 55,329 to the Roots, 144,723 to the TLDs, and 348,619 to the SLDs.

Linear delays, 5% packet loss



About the same as the previous slide; just slightly higher in most categories.

BIND 8 Sends The Most Queries

- Sends A, AAAA, and A6 queries to Roots, SLDs, and TLDs for expired NS addresses.
- Forwards cache misses for pending hits.
  - (djbdns, w2000, w2003 do this too)

Forwarding Cache Misses for Pending Hits

```
16:43:19.489271 USER.1756  > BIND8.53:        7+  A? www.popularsite.com.
16:43:19.491460 USER.1756  > BIND8.53:       11+  A? www.popularsite.com.
16:43:19.494532 BIND8.1041 > ROOT.11.53:  25426   A? www.popularsite.com.
16:43:19.495513 BIND8.1041 > ROOT.11.53:  32628   A? www.popularsite.com.
16:43:19.495855 ROOT.11.53 > BIND8.1041:  25426-  0/13/14
16:43:19.497446 ROOT.11.53 > BIND8.1041:  32628-  0/13/14
16:43:19.497648 TLD.68.53  > BIND8.1041:   1217-  0/2/3
16:43:19.504748 BIND8.1041 > TLD.3.53:    21305   A? www.popularsite.com.
16:43:19.505638 TLD.3.53   > BIND8.1041:  21305-  0/2/3
16:43:19.506783 BIND8.1041 > SLD.118.53:  55782   A? www.popularsite.com.
16:43:19.507983 BIND8.1041 > SLD.118.53:  39986   A? www.popularsite.com.
16:43:19.509285 USER.1756  > BIND8.53:       13+  A? www.popularsite.com.
16:43:19.509522 SLD.118.53 > BIND8.1041:  55782*- 1/2/3 A 25.240.249.31
16:43:19.509743 SLD.118.53 > BIND8.1041:  39986*- 1/2/3 A 25.240.249.31
16:43:19.510225 BIND8.1041 > SLD.118.53:  38069   A? www.popularsite.com.
16:43:19.511444 USER.1756  > BIND8.53:       16+  A? www.popularsite.com.
16:43:19.511729 SLD.118.53 > BIND8.1041:  38069*- 1/2/3 A 25.240.249.31
16:43:19.511729 SLD.118.53 > BIND8.1041:  38069*- 1/2/3 A 25.240.249.31
16:43:19.516474 BIND8.53   > USER.1756:       7*  1/2/2 A 25.240.249.31
16:43:19.516907 BIND8.53   > USER.1756:      11*  1/2/2 A 25.240.249.31
16:43:19.521322 BIND8.53   > USER.1756:      13*  1/2/2 A 25.240.249.31
16:43:19.522298 BIND8.53   > USER.1756:      16   1/2/2 A 25.240.249.31
```

Why So Many BIND 9 Queries to Roots?

- bind9 re-queries for expired glue starting at the root.
- Sends A and A6 queries for both SLD nameservers.

```
21:18:35.998866 BIND9.1041 > ROOT.1.53:  17639  A? ns0.iastate.edu.
21:18:36.000520 BIND9.1041 > ROOT.1.53:   7798 A6? ns0.iastate.edu.
21:18:36.002224 BIND9.1041 > ROOT.1.53:  51091  A? ns1.iastate.edu.
21:18:36.003895 BIND9.1041 > ROOT.1.53:  36953 A6? ns1.iastate.edu.
```

- Real-world:

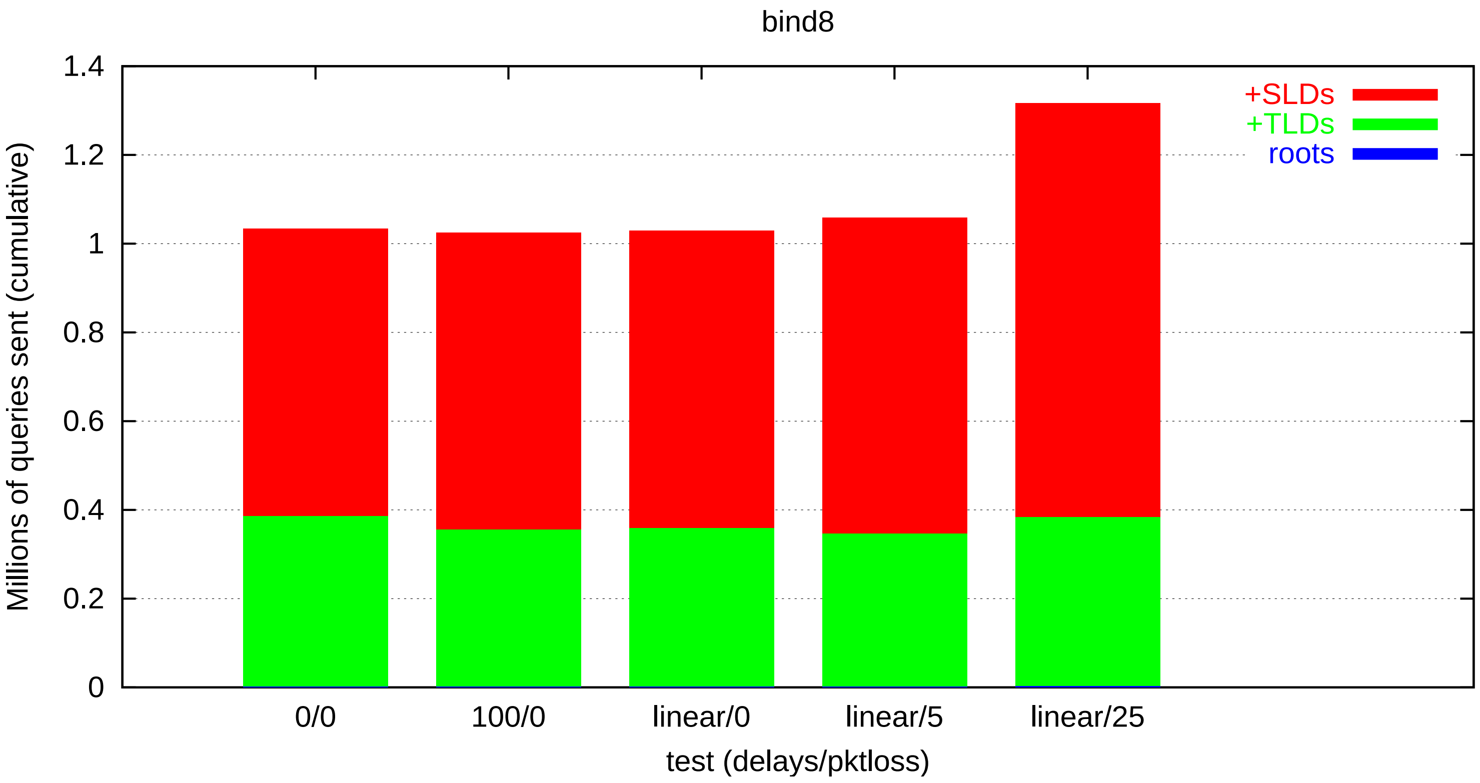
```
;; ANSWER SECTION:
microsoft.com.          1H IN NS        dns1.cp.msft.net.
microsoft.com.          1H IN NS        dns1.sj.msft.net.
microsoft.com.          1H IN NS        dns1.tk.msft.net.
microsoft.com.          1H IN NS        dns3.uk.msft.net.

;; ADDITIONAL SECTION:
dns1.cp.msft.net.       1H IN A         207.46.138.20
dns1.sj.msft.net.       1H IN A         65.54.248.222
dns1.tk.msft.net.       1H IN A         207.46.245.230
dns3.uk.msft.net.       1H IN A         213.199.144.151
```

djbdns Also has a lot of Root Queries

- djbdns re-queries for expired glue starting at the root.
- But only for first nameserver
- Only A queries
- djbdns does not trust “additional” answers with TTL 0 and re-queries for them starting at the root.

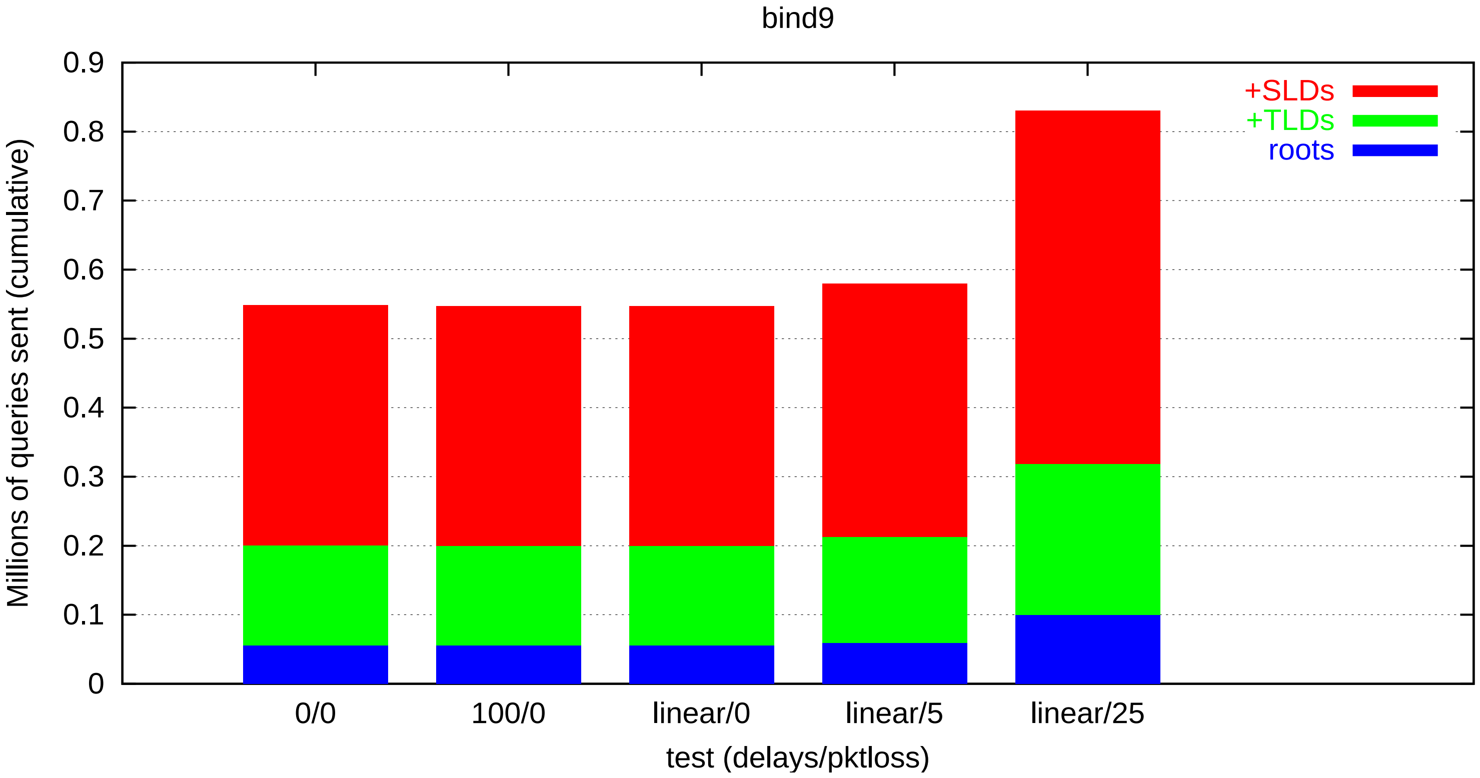
```
20:33:55.902072 USER.1805    > DJBDNS.53:       295+  A? www3.kwyjibo.com.
20:33:55.906091 DJBDNS.7410  > TLD.9.53:       1206   A? www3.kwyjibo.com.
20:33:55.906704 TLD.9.53     > DJBDNS.7410:    1206-  0/2/2
20:33:55.908634 DJBDNS.10380 > ROOT.13.53:     2361   A? ns0.kwyjibo.com.
20:33:55.909236 ROOT.13.53   > DJBDNS.10380:   2361-  0/13/13
20:33:55.913527 DJBDNS.5244  > TLD.12.53:     32554   A? ns0.kwyjibo.com.
20:33:55.914155 TLD.12.53    > DJBDNS.5244:   32554-  1/2/2 A SLD.67
20:33:55.925028 DJBDNS.60070 > SLD.68.53:     19539   A? ns0.kwyjibo.com.
20:33:55.925751 SLD.68.53    > DJBDNS.60070:  19539*- 1/2/2 A SLD.67
20:33:55.954061 DJBDNS.44857 > SLD.68.53:     30676   A? www3.kwyjibo.com.
20:33:55.954797 SLD.68.53    > DJBDNS.44857:  30676*- 1/2/2 A 50.233.168.16
20:33:55.961632 DJBDNS.53    > USER.1805:       295   1/0/0 A 50.233.168.16
```

BIND 8



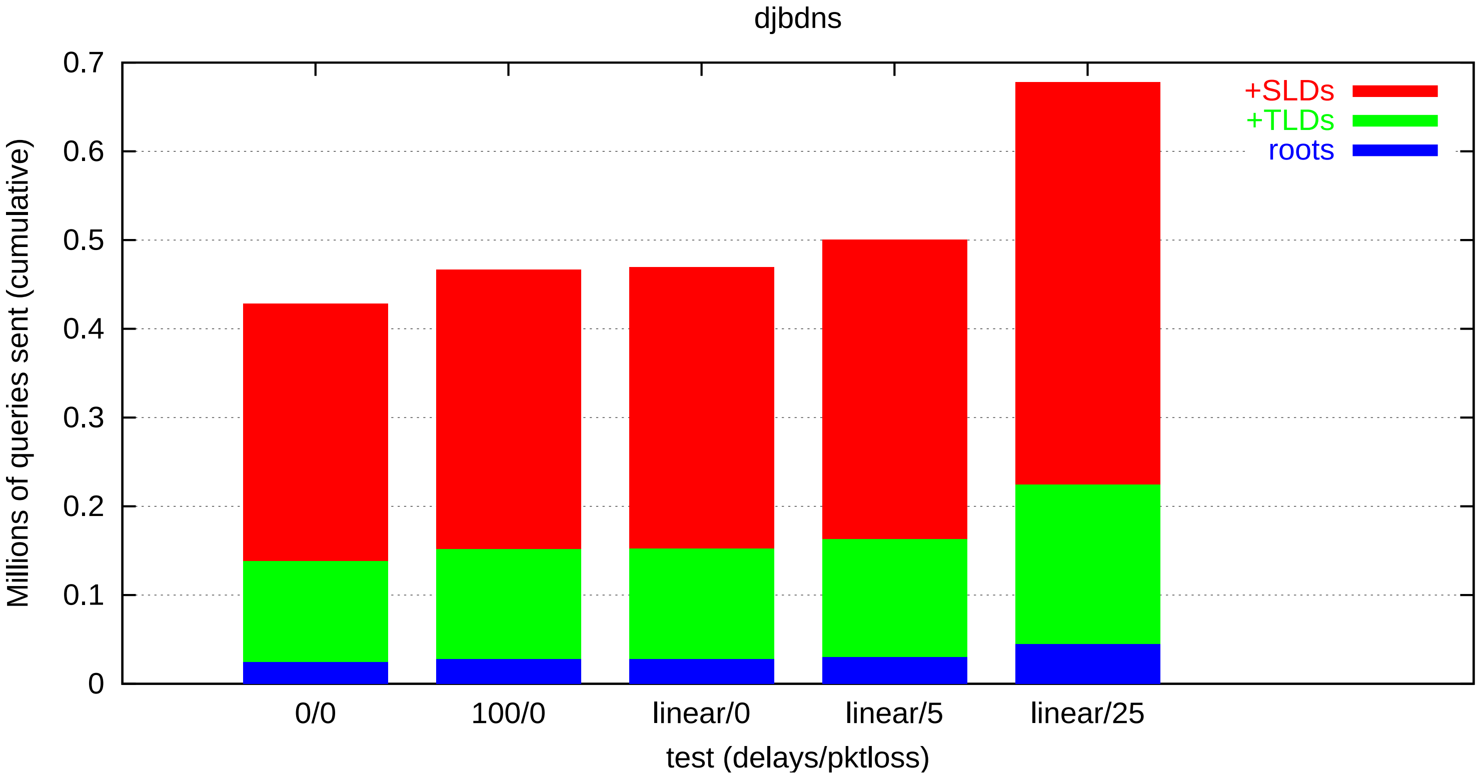
BIND8 actually sends slighly fewer queries in “harsher” conditions. For example, 383,995 TLD queries in 0ms/0% test, but only 344,628 in linear/5% test.

BIND 9

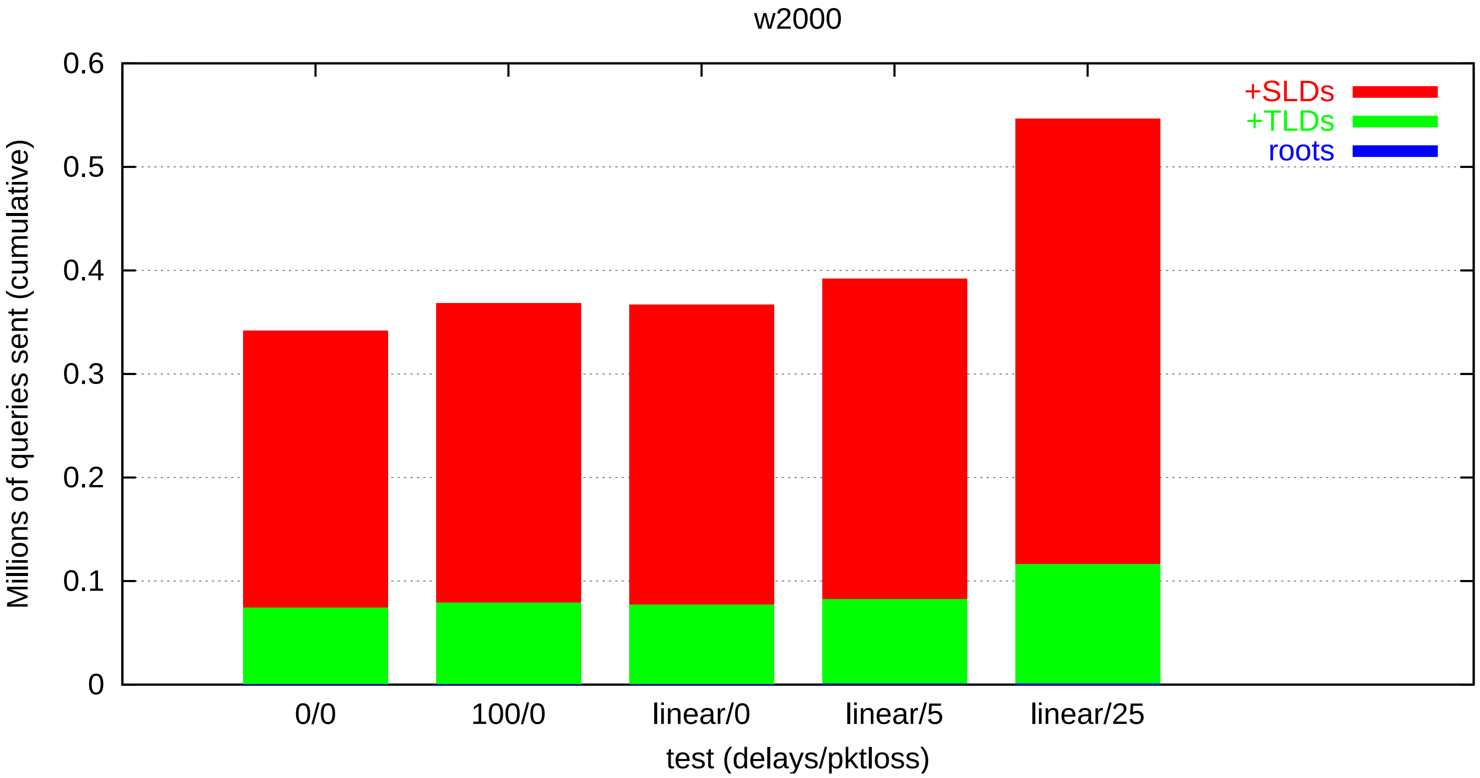


Unaffected by delays, only by loss.

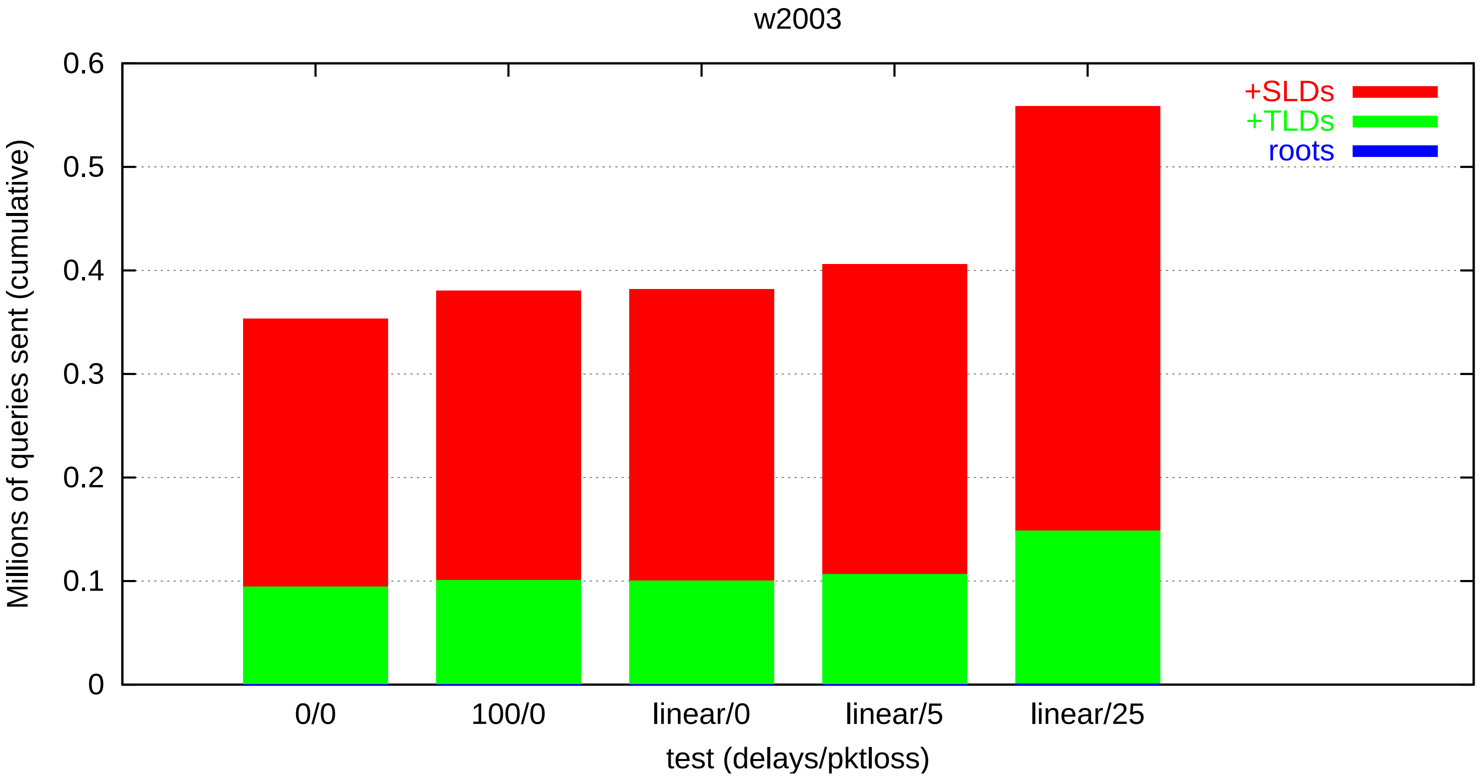
DJBDNS


djbdns
Millions of queries sent (cumulative)
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0/0
100/0
linear/0
linear/5
linear/25
test (delays/pktloss)
+SLDs
+TLDs
roots

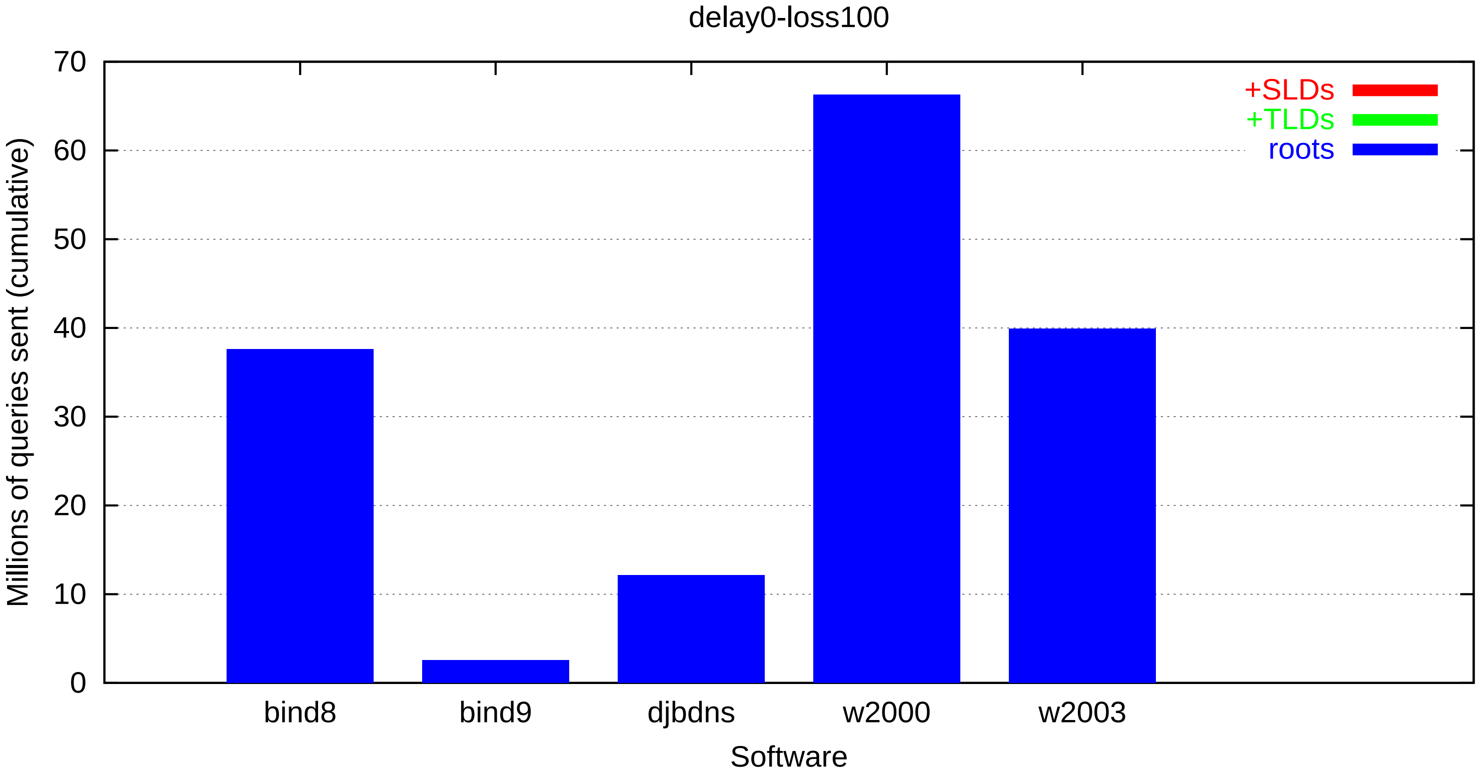
Windows 2000



Windows 2003

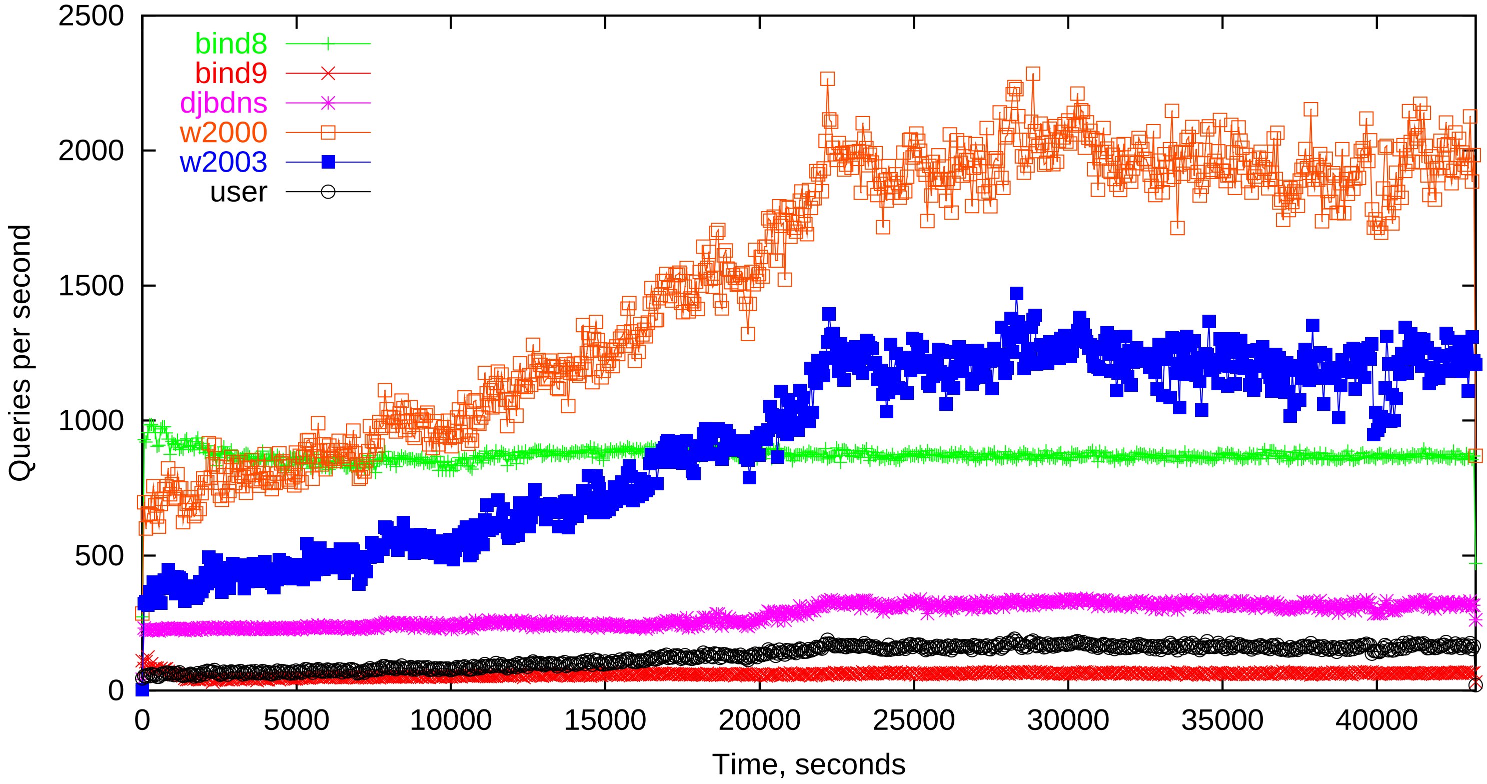


No delays, 100% packet loss



Wow!

No delays, 100% packet loss

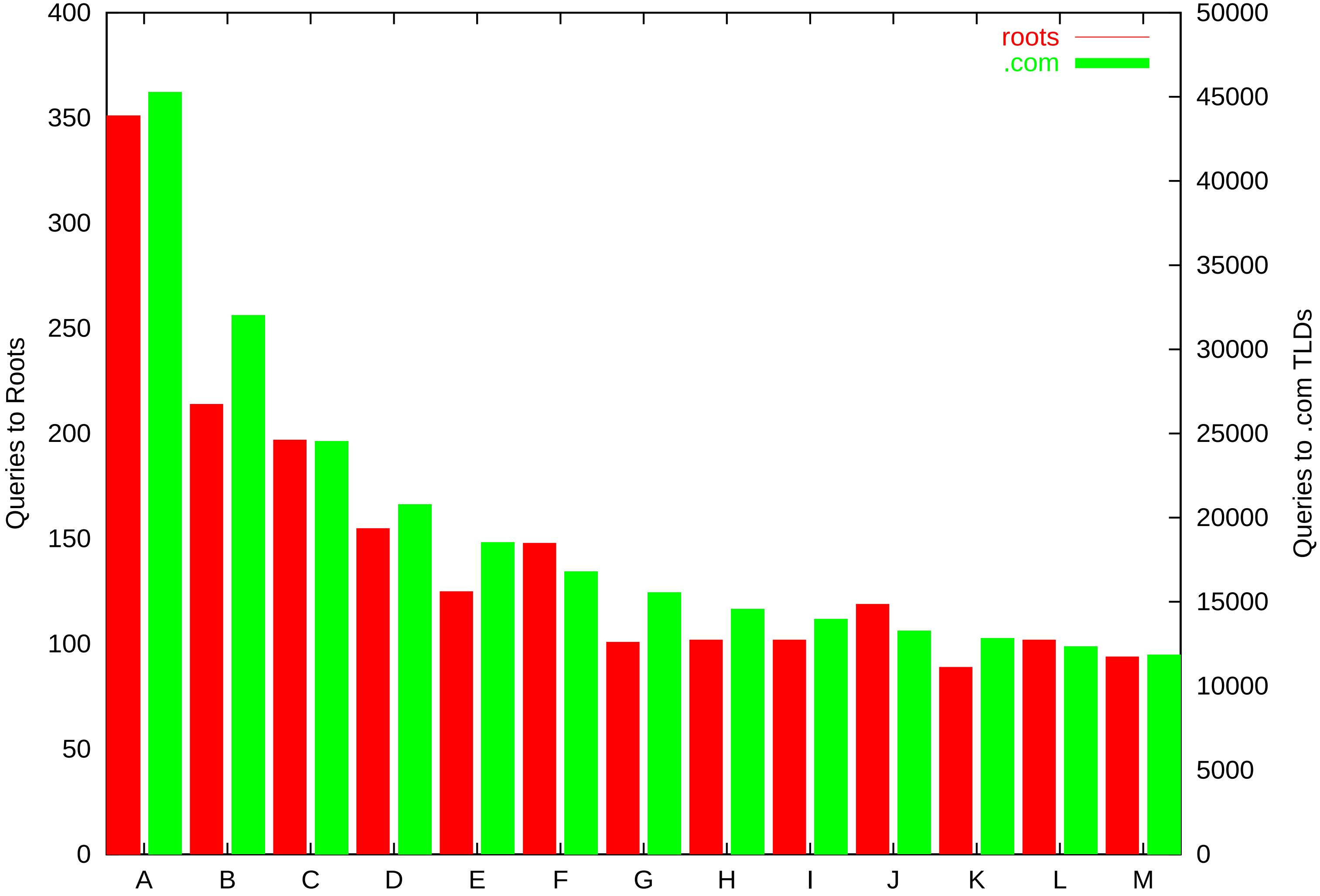


Most software amplifies the user query rate, but BIND9 attenuates.

“DNS servers on the other hand track RTTs for query responses and really *know* which server is the fastest rather than guess based on third hand routing information.”

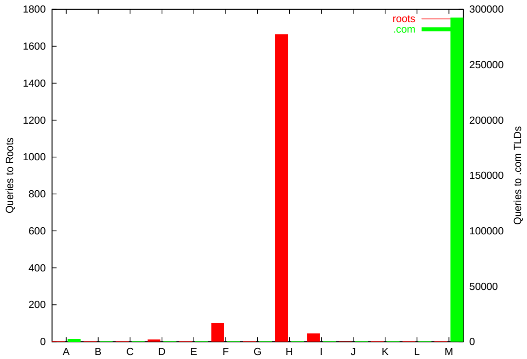
–Iljitsch van Beijnum, 18 Sep 2003

Distribution of Queries to Nameservers

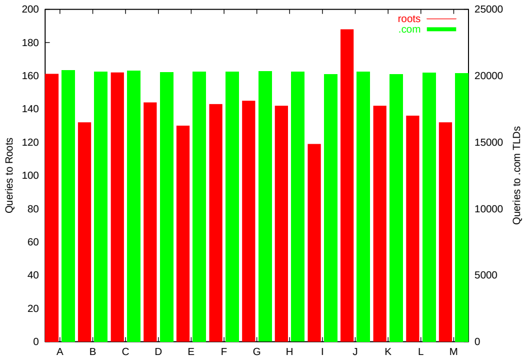


BIND 8

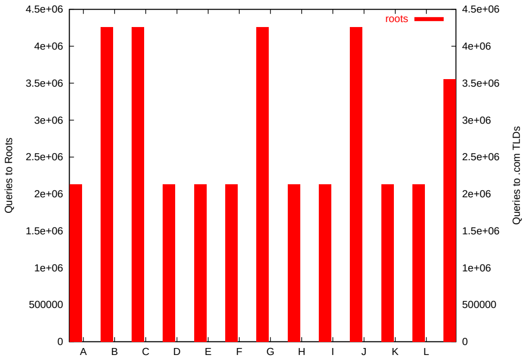
0ms/0%



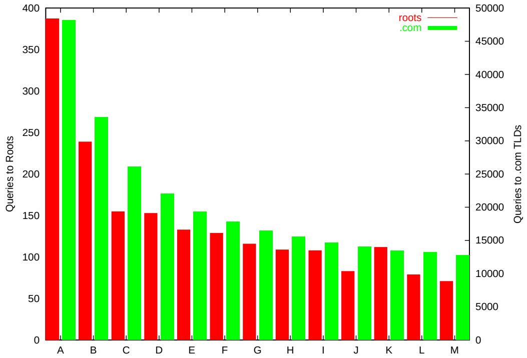
100ms/0%



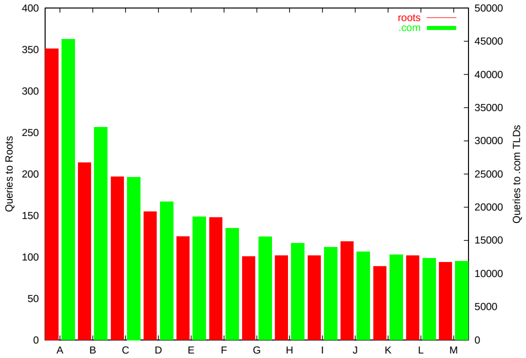
0ms/100%



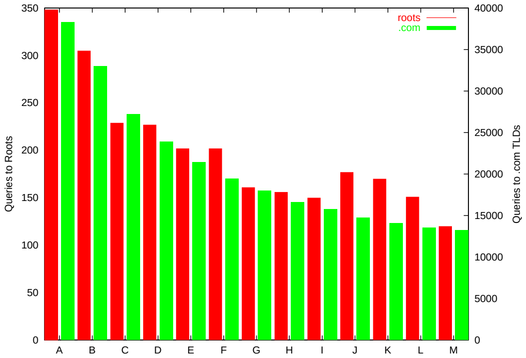
linear/0%



linear/5%

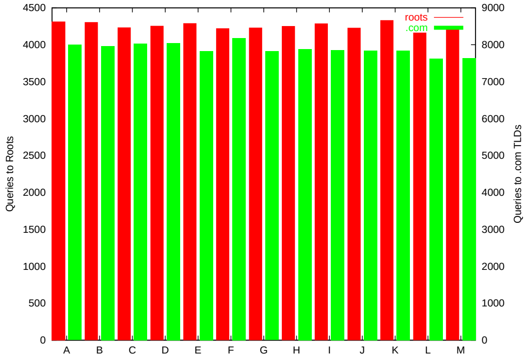


linear/25%

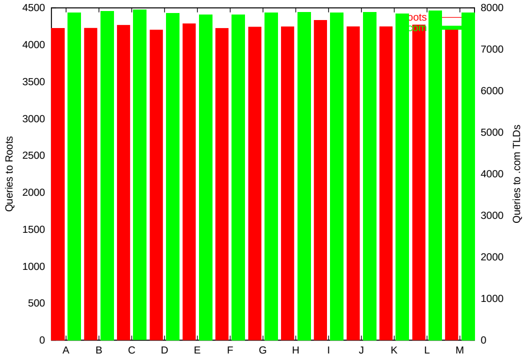


BIND 9

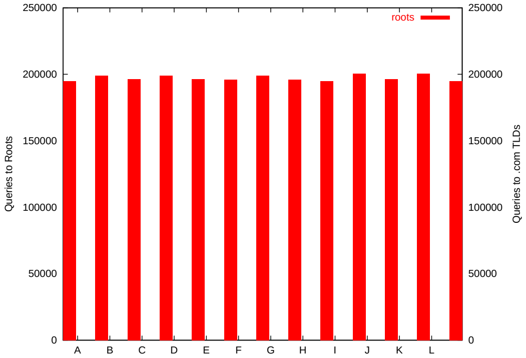
0ms/0%



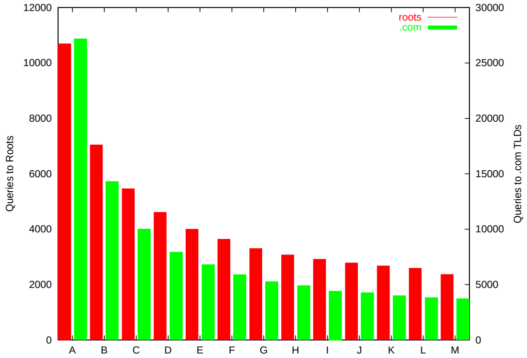
100ms/0%



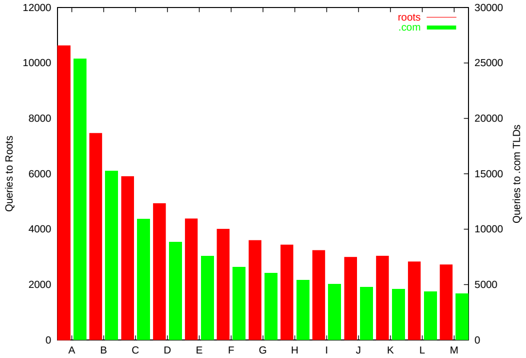
0ms/100%



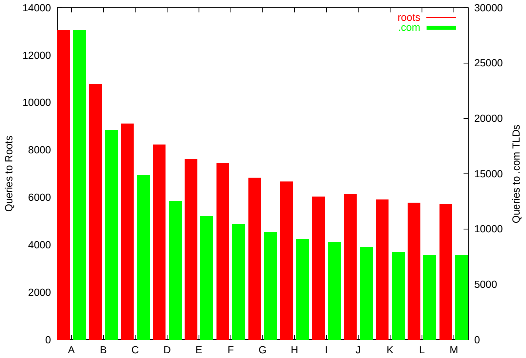
linear/0%



linear/5%

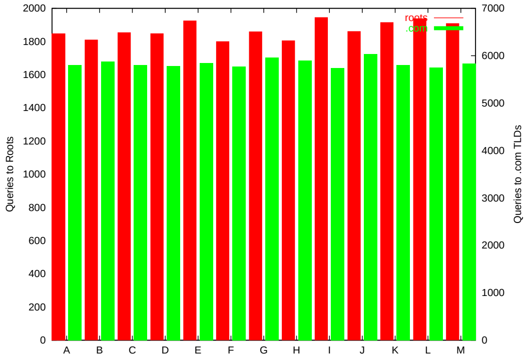


linear/25%

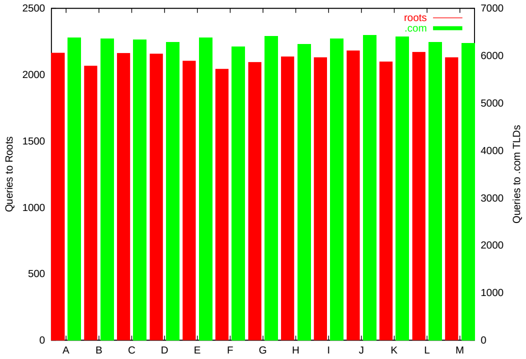


DJBDNS

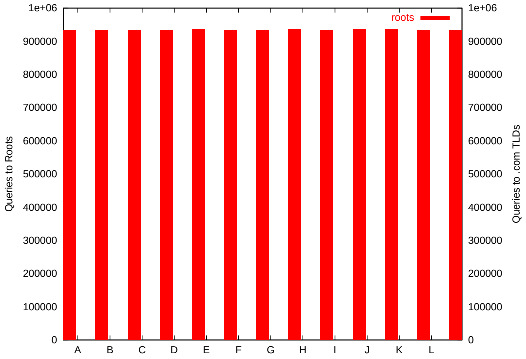
0ms/0%



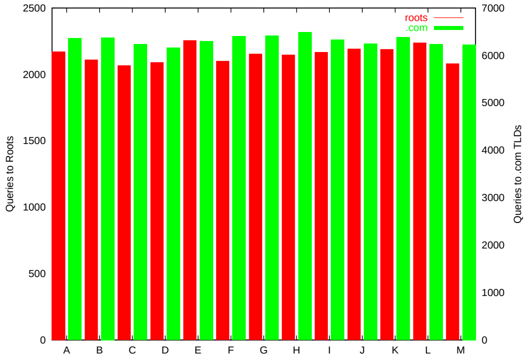
100ms/0%



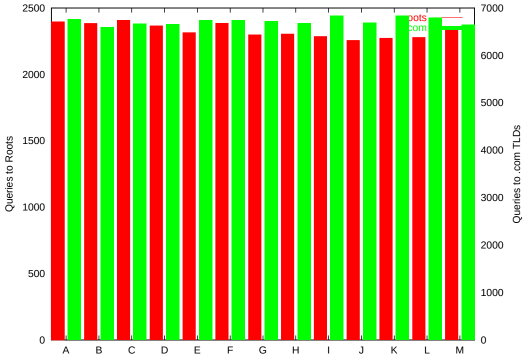
0ms/100%



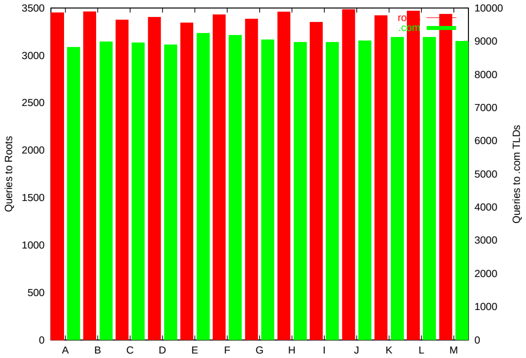
linear/0%



linear/5%

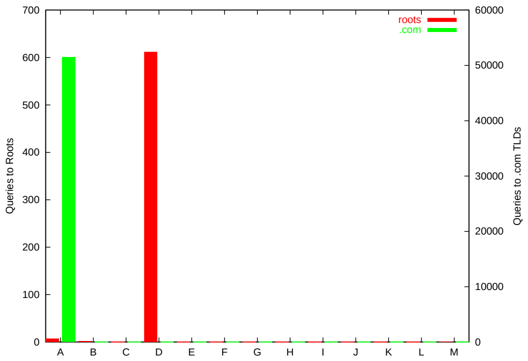


linear/25%

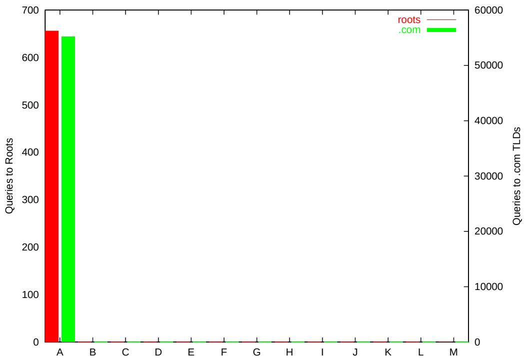


Windows 2000

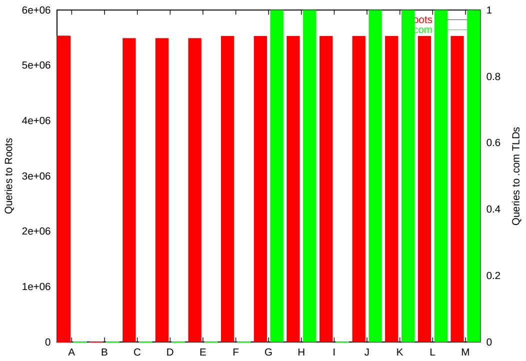
0ms/0%



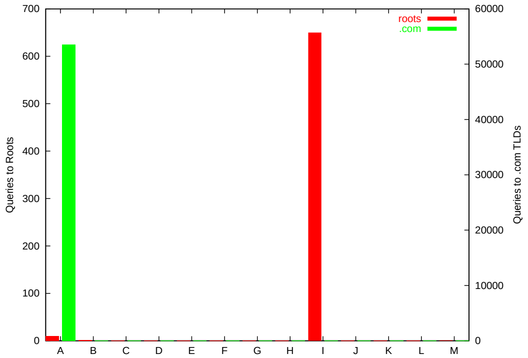
100ms/0%



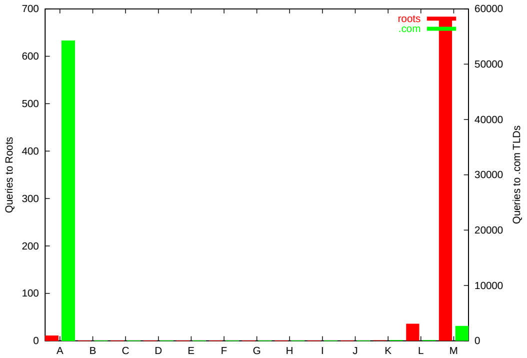
0ms/100%



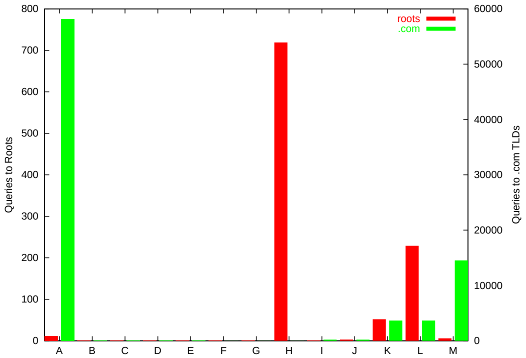
linear/0%



linear/5%

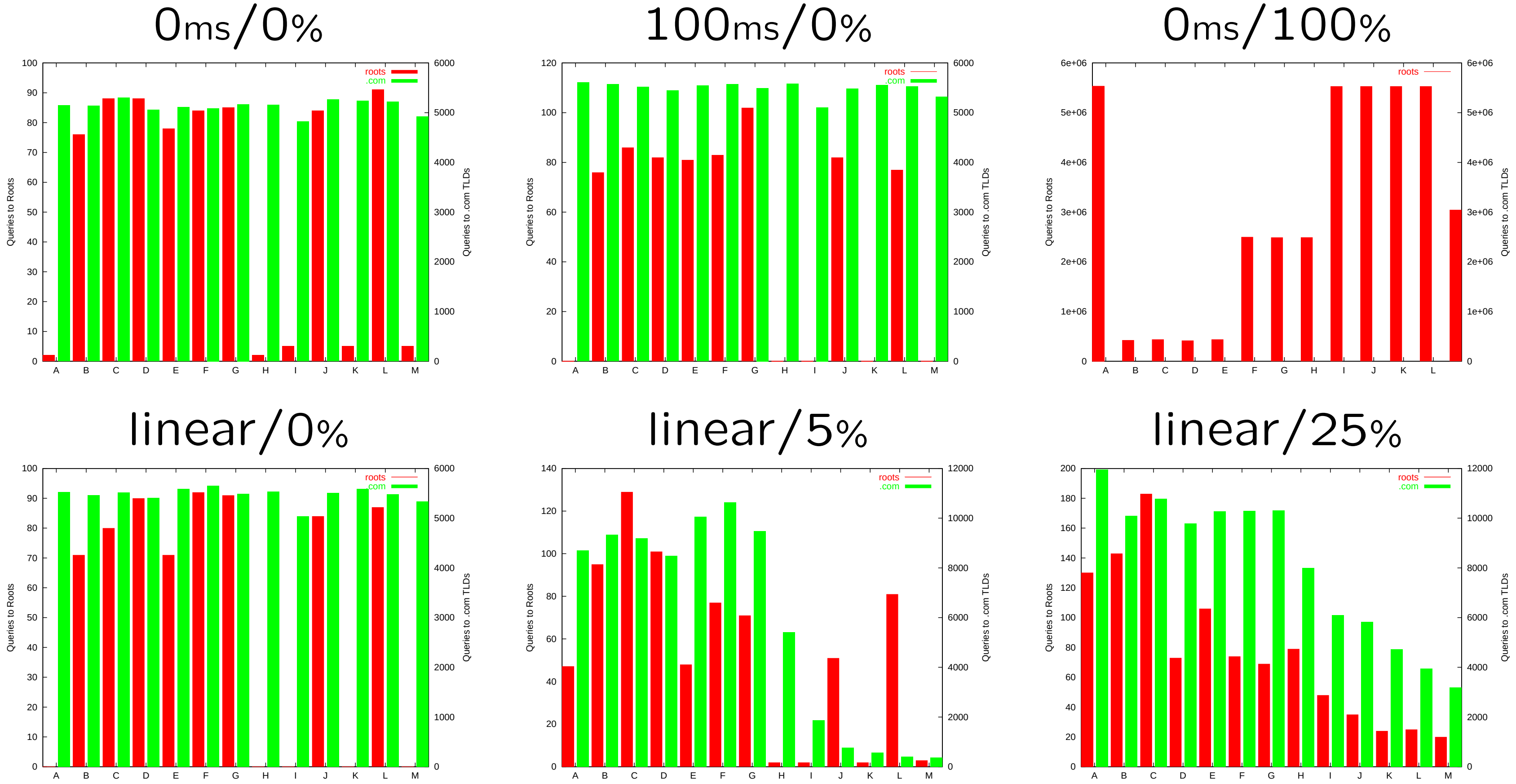


linear/25%



Seems to always choose first *.com* nameserver.

Windows 2003



0ms/0%

100ms/0%

0ms/100%

linear/0%

linear/5%

linear/25%

“Scientists at the San Diego Supercomputer Center found that 98% of the Slashdot comments at the root level are unnecessary.”

–Anonymous Coward

Punchline from Last Year's Talk

Type	Count	Percent
Repeated QNAME	68,610,091	44.9
Repeat Query	38,838,688	25.4
Unknown TLD	19,165,840	12.5
A for A	10,739,857	7.03
Referral Not Cached	6,653,690	4.36
Legitimate	3,284,569	2.15
Nonprintable in QNAME	2,962,471	1.94
rfc1918 PTR	2,452,806	1.61
Unused Query Class	36,313	.024

Run Simulations Through Earlier Root Server Analysis Tools

What happens when we run one of the simulated Root server traces through the tools used to analyze real root server activity in last year's talk?

	bind8	bind9	djbdns	w2000	w2003
unknown-tld	20.2	0.7	1.2	73.0	64.3
repeated-query	0.0	0.0	0.0	0.0	0.8
repeated-qname	23.1	31.4	51.7	2.5	0.8
referral-not-ca	21.1	64.2	43.4	0.9	7.8
legit	35.6	3.7	3.7	23.6	26.4

oops. Learned that some software always goes back to the roots for expired NS addresses.

Run Simulations Through Earlier Root Server Analysis Tools

What happens if we exclude queries for (expired) NS addresses?

	bind8	bind9	djbdns	w2000	w2003
unknown-tld	39.7	64.3	69.0	73.0	65.9
repeated-query	0.0	0.0	0.0	0.0	0.8
repeated-qname	17.9	0.0	0.0	2.5	0.8
referral-not-ca	6.1	0.9	2.4	0.9	7.1
legit	36.3	34.8	28.6	23.6	25.4

oops? Some TLDs have short TTLs on NS addresses. Earlier study assumed TLD TTLS 24 hours or greater. But is it as simple as that?

Conclusions — Balancing

- BIND 8 can latch on to a single nameserver in low loss/latency conditions, but we didn't determine the threshold.
- DJBDNS uses uniform distribution among nameservers regardless of conditions.
- W2000 server selection sucks.
- W2003 server selection only slightly better.

Conclusions — Abuse

- A6 and AAAA queries on IPv6-enabled machines may be abusing Root, TLD, SLD nameservers.
- BIND9 does not forward cache misses for pending hits. cool.
- BIND9 is the only software tested that attenuates the user queries in the event of 100% packet loss.
- Need to improve our models for analyzing root server abuse. Must consider BIND9 and DJBDNS's techniques to prevent cache poisoning.

The End