DNS Measurements at a Root Server

Nevil Brownlee
The University of Auckland and CAIDA, SDSC, UC San Diego, e-mail: nevil@caida.org
kc Claffy
CAIDA, SDSC UC San Diego, e-mail: kc@caida.org
Evi Nemeth
University of Colorado and CAIDA, SDSC, UC San Diego, e-mail: evi@caida.org

Abstract—**The Domain Name System (DNS) domain names to be used in network transactions (email, web requests, etc.) instead of IP addresses. The root of the DNS distributed database is managed by 13 root nameservers. We passively measure the performance of one of them: F.root-servers.net.**

These measurements show an astounding number of bogus queries: from 60-85% of observed queries were repeated from the same host within the measurement interval. Over 14% of a root server's query load is due to queries that violate the DNS specification. Denial of service attacks using root servers are common and occurred throughout our measurement period (7-24 Jan 2001). Though not targeted at the root servers, DOS attacks often use root servers as reflectors toward a victim network. We contrast our observations with those found in an earlier study of DNS root server performance by Danzig et. al. [1].

Keywords—**DNS, Root Server**

I. Introduction

DNS, the Domain Name System, translates from domain names used by people to the corresponding IP addresses required by all network software. Data is stored in a distributed database where each nameserver is responsible (authoritative) for its own piece of the naming tree. Delegation of authority occurs via NS (nameserver) records and must be consistent between parent nodes and children in the naming tree. The robustness and redundancy of the DNS protocols hide many configuration errors at the local site, so that such errors only appear further up the tree or in the logfiles of servers trying to contact the misconfigured site. This distributed control and configuration is a double-edged sword: it allows the system to scale to Internet sizes, but it also allows for incredible misconfiguration.

Incorrect nameserver implementations put additional query load on the servers, particularly at the root of the tree. BIND, the Berkeley Internet Name Domain system (Internet Software Consortium) [2] is the most widely used implementation. For years BIND was the basis for all vendor's implementations, but recent independent implementations by Microsoft and others have introduced interoperability issues that are largely invisible until we examine query/response behavior at a root server.

The DNS protocol [3] uses UDP for queries and responses. The client side query process typically starts with an application program on the end user's workstation which via a resolver library contacts a local nameserver. That client side nameserver queries the root servers for the name in question and gets back a referral to a nameserver who should know the answer. The client's nameserver will recursively follow referrals re-asking the query until it gets an answer or is told there is none. Caching of that answer should happen at all nameservers except those at the root or top level domains (.com for example). Recent versions of BIND also include negative caching in which 'no' answers are also cached. Reliability is achieved by the DNS client repeating unanswered queries up to 12 times each separated by an exponential backoff timeout interval.

TABLE I
Root Nameserver Data Collection Regime.

Size	Queries	Distinct Q's (%)	Date/Time
3.6 GB	10.3 M	2.7 M (26.2%)	Jan 7, 11 am
5.9 GB	18.0 M	4.8 M (26.7%)	Jan 9, 3 pm
10.4 GB	29.1 M	4.5 M (15.5%)	Jan 8, 1 pm
338 MB	1 M	380 K (37.9%)	Jan 10, hourly
690 MB	2 M	622 K (31.2%)	Jan 12,17–19,24

In 1992 Danzig et. al. [1] examined the DNS system from the viewpoint of the ISI (Information Sciences Institute) root server. Their error analysis identified several bugs that are still with us today: the recursion bug, the zero answer bug, the server failure detection bug and faulty retransmission timers. Today, we also see malformed queries, impossible questions and a serious intermixing of internal Microsoft naming with DNS naming.

II. Measurement Methodology

Our measurements are passive; we observe DNS traffic flowing to and from of the F root nameserver with the UNIX utility *tcpdump* set to capture the entire DNS packet.

F.root-servers.net is located at PAIX, the Palo Alto Internet Exchange and run by ISC, the Internet Software Consortium. F is actually two DEC Alpha machines located behind a Cisco router using CEF (Cisco Express Forwarding) to load balance queries between the two machines. The F root servers run BIND 8.2.3. BIND 8 is not multithreaded and uses only one of the four processors in each Alpha; our computations and data capture use the others and do not interfere with F's name serving functions.

Data was collected in 1 hour, 2 hour, 2 million packet or 4 million packet chunks during the second and third weeks of January, 2001. We merged data from each server during processing. Table I details our data collection.

The smaller data sets adequately represent the data seen in the larger traces, see Table II, except when repeated queries

Support for this work is provided by DARPA NGI Contract N66001-98-2-8922, NSF Award NCR-9711092 'CAIDA: Cooperative Association for Internet Data Analysis,' and The University of Auckland.

or denial of service attacks occur and distort the data.

We also had access to a full set of error logs. They showed two main types of error: (1) denied attempts to dynamically update the root server, and (2) dropped queries that were received with source port 0.

We gathered traces on January 24, 2001 to look at the Microsoft DNS problems that day and to see how a highly visible site's problems can impact the root server system.

III. Experimental Results

Our measurements and log files on the F root servers allow us to identify many types of broken nameserver/resolver implementations, several types of misconfiguration of DNS zone data, failures to follow the DNS protocol, and denial of service attacks. We show examples of each type of anomaly found and also quantify the amount of traffic showing each error.

A. Query rates

Since essentially all of the traffic to and from the F root server is DNS request/response packets, we use the *netstat* command to measure the raw query rate. We collected data from 11pm January 6 until 11am January 16, 2001 and again from January 25 to 31. We gathered data at 5 second intervals, but have aggregated it to 10 minute intervals in Figure 1. The typical traffic pattern of the U.S. work week is visible with query load peaking at about 5000/sec. The very high spikes coincide with times when we copied the *tcpdump* output files from one F root to the other for processing.

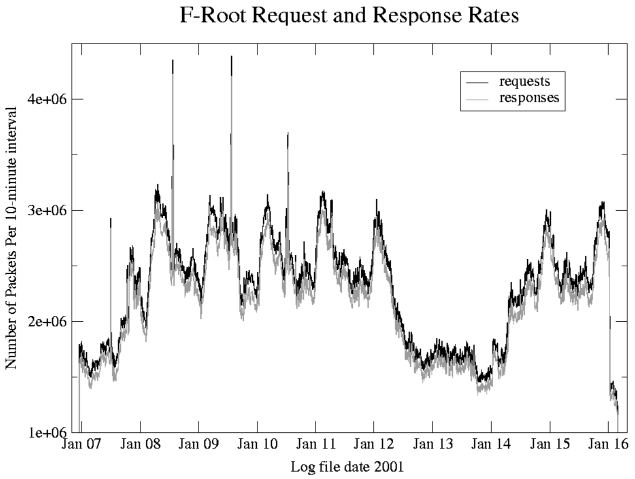

Fig. 1. Combined query load, F root servers, Jan 6-16 2001, 10 minute bins.

The F root nameserver responds to about 93% of the input packets immediately. The remaining 7% represent queries that the F root server cannot answer. For example, queries from private address space [4] cannot be answered because there is no route back to the querying host. In order to accurately categorize all the traffic seen at the F root servers, we removed the router filters that eliminate some unanswerable queries. *netstat* was later run for a shorter period after these filters were reinstalled; the data shows that over 97% of the input packets were immediately answered with this filtering in place.

Malformed queries go in the unanswerable pile as well. DNS queries should contain just one question. A 16 bit integer field in the header specifies how many queries follow. We have found several instances of this count being 256, but the packet only contains one query. We suspect a big endian, little endian byte order problem in the nameserver code on some NT4/Win95/Win98 machines. In the January 7, 2001 data trace (1 hour) there were 78,000 queries from 1400 distinct nameservers with this bug.

B. Error Taxonomy

One would hope that in the almost 10 years since Danzig's original paper [1] analyzing the DNS system, the implementation bugs known in 1992 would be mostly gone. Sadly, this is not the case and many of the old bugs continue to strain the root server system. The worst today include three problems identified in 1992: broken timeouts with respect to repeated queries; not understanding no (*NXDomain*) for an answer; and not understanding referrals.

Surprisingly few of the queries arriving at the F root server are really valid. We found anomalies that looked like broken nameservers, others that were clearly misconfiguration, and some malicious attacks. A local nameserver querying a root server should not expect the root server to use recursion and return a final answer. A typical root answer is a referral to an authoritative server at the next level down in the naming tree or the answer *NXDomain*, (non-existent domain).

B.1 Repeated Queries

Our first aggregation tool, dns-trace-histo.pl [5] counts packets into bins indexed by the source of the query and the question asked. Sorting the output by the count field identifies the most egregious broken implementations. For example, in the one hour trace file taken on January 7, 2001, the top two entries in the output file:

```
564028  IP1.mil PTR 65.224.102.166.in-addr.arpa.
374679  IP2.mil PTR 65.224.102.166.in-addr.arpa.
```

account for 9.1% of the queries. These hosts are both on the same .mil subnet and are asking the same question, namely what is the hostname associated with the IP address 166.102.224.65. The trace file shows F root answering with a referral to two other nameservers (almost a million times!). Looking up this address by hand results in a response that says the local nameserver failed to contact the appropriate server (query returned SERVFAIL), presumably because that remote server is down.

These two nameservers do not understand referrals or SERVFAIL responses and just ask the question over again, on average 154 times per second. In the January 9 trace a single host repeated a query 2,112,962 times in an hour, each time receiving a referral to a server that returns SERVFAIL. That's 587 times per second. We used *nmap* to determine the operating system making these repeated queries. The results for the top 37 repeat offenders were:

```
12  down at the time of the nmap run
 5  could not be identified or unknown
 1  AIX v4.2
 1  Cobalt Linux 4.0 (Fargo) Kernel 2.0.34C52_SK on MIPS
    or TEAMInternet Series 100 WebSense, Linux 2.0.35-38
 5  Solaris 2.6 - 2.7, Solaris 7
13  Windows NT4 / Win95 / Win98 / Win2k
```

Repeated queries may be the result of a broken nameserver or a broken client. In the worst cases there is no time for the answer to reach the querying nameserver before it re-issues the same query. In our sample traces, the longer the trace period, the higher the percentage of repeated queries (same source host, same question):

```
trace sizes   #queries   %repeated queries
-----------   --------   -----------------
4-minute          1M          62%
8-minute          2M          69%
1 hour        10-18M          74%
2 hours          29M          85%
```

B.2 Private Address Space

RFC 1918 defines several networks that can be used internally by any site but that cannot be routed on the Internet. These addresses should never reach the root servers, but should stay within local intranets. We see both packets from these addresses and packets querying about these addresses. F root's operator, Paul Vixie, was willing to temporarily remove his router filters in order to determine how many RFC1918 source addresses would naturally try to reach F root. Without those filters, between 2-3% of the queries arriving at F root have the source IP address in RFC 1918 space. A smaller percentage have the autoconfigure address 169.254/16, which should never leave the local network segment. BIND cannot answer these queries.

About 7% of the queries are asking for the hostname associated with an RFC 1918 address. These are also unanswerable. Note that these two sets are not mutually exclusive, in fact, approximately 7% of queries from an RFC 1918 address ask about such an address.

These queries are examples of misconfigured nameservers at the local site. In BIND (Berkeley Internet Name Domain system software) terminology the site is using split DNS and data is leaking from the internal network out to the Internet.

B.3 Top Level Domains (TLDs)

Browsing the query logs shows a myriad of strange, invalid top level domain names. We analyzed several traces to see how extensive the queries about non-existent top level domains were. In the 1 hour trace of January 7, 2001, queries for 143,783 distinct invalid top level domains were made. Here are some examples of the invalid TLDs found:

```
.local, .localhost, .loghost, .localdomain
.workgroup, .msft, .home, .domain, .office, .ntdomain
_ldap._tcp.Default-First-Site-Name._sites.gc._msdcs.SHFEX02
.http
.HP_NETWORK_PRINTERS
www.bcs.WSCOOPER.WSCOOPER.WSCOOPER.WSCOOPER.WSCOOPER ...
```

This last example is typical of a bug in the DNS data files at the local site. If a name that should end in a dot and therefore be absolute has no dot, the local nameserver will add the local domain to complete the name. A bug in at least one nameserver implementation adds the local domain recursively until it is 255 characters long (the limit), as in the WSCOOPER example.

This particular trace had about 10 million queries from 127,000 distinct nameservers of which 20% were for invalid TLDs. 83.5% of the servers asked at least one query with a valid TLD, implying that 16.5% of the servers asked only invalid queries. 37.1% of the servers asked at least one query with an invalid TLD. Spelling errors on a user's part can account for a few invalid TLD queries, but not the quantity that we see.

B.4 Bogus A Queries

The mapping from a hostname to an IP address, as is needed to send an email or contact a web site, involves a DNS query for A records in the DNS database. A properly formed A query has the hostname as a target. Between 12 and 18% of the queries arriving at the F root during our measurements were A queries with an IP address as a target. This violates the DNS specification. These queries were not just from a few confused nameservers, 12% of the servers contacting the F root during our trace period sent at least one bogus A query. Such queries go to the root because an IP address, for example, 192.168.1.33 has the same form as a hostname (dot separated strings) but is interpreted as being in the top level domain '33'. A root server would answer 'no such domain' (*NXDomain*).

We explored several possible causes for this error: misconfigured MX records; UNIX programs ported to Windows where the resolver library was not quite the same; Windows programs to process web logs; firewall products with shaky networking stacks, etc. Even combined these could not account for 14% of the queries being of this form. We then installed and deployed at three sites a special version of BIND that would log just those queries. This version identified the next host back in the query chain, from which we identify several problem end user applications.

To date, we have determined three causes of the bogus A queries, all on Windows systems: the Win2k resolver library [6], the *snow white* virus, also called W32.HybrisF [7], that infects the Winsock dynamically linked library and the *wininit* virus, called W32.HLLW.Bymer [8], that trolls for disk shares to infect via port 137. The OpenBSD resolver and some DSL modem boxes seem to be guilty too. The extremely high query rate suggests either a bug in the Microsoft networking libraries or a phenomenally large number of infected Windows boxes (or both). We are skeptical that the Win2k resolver library bug is the primary culprit; the fix is in service pack 2. Continued monitoring of the percentage of bogus A queries may let us measure both the frequency of the Win2k resolver bug and the degree of service pack 2 deployment.

B.5 Source Port Zero

Port 0 is reserved and not valid in either UDP or TCP packets, yet a few nameservers use 0 as the source port of their queries. The source port is normally a high numbered random port, but recent BIND distributions allow it to be configured. We summarized the log files for Jan 6-7 and found 27 sites using source port 0; 20 were in the uswest.net domain. BIND runs on all the root servers and never answers these queries. The robust design of the DNS system is covering for this system administration mistake.

B.6 Dynamic Updates

Recent versions of DNS have supported a feature called dynamic update. In the local environment, the DHCP (Dynamic Host Configuration Protocol) server when dynamically assigning an IP address to a host, can also tell the nameserver about the address assignment. The local nameserver can be configured to accept these updates from the DHCP server and the DNS database is automatically kept up to date. Though proper behavior locally, there is no reason a local machine should try to update the root servers. When Win2k was first released it flooded the root servers with requests to update the root zone. Figure 2 shows the frequency of update requests and the number of queries from source port 0 for the period Dec 1999–Feb 2001.

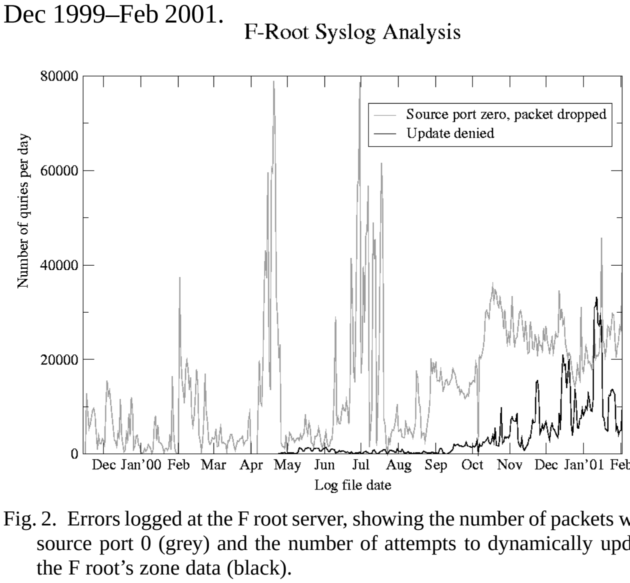

Fig. 2. Errors logged at the F root server, showing the number of packets with source port 0 (grey) and the number of attempts to dynamically update the F root's zone data (black).

The update requests are increasing. We examined one update spike in detail and found that a single host tried to update F root 15,000 times in one day.

C. Attacks

We found two types of denial of service attacks in our measurement data, each with a different signature. One attack involved spoofing source IP addresses and was targeted at 209.67.50/24, a register.com customer. The attack began January 4, 2001; we observed it during our entire measurement period. It was not targeted at the root server but rather used the root as a reflector, flooding the attack target with answers to questions it did not ask.

We discovered a second attack when we looked at the number of queries made by each individual server and found servers with hundreds of thousands of queries but few or no repeats. This appears to be someone scanning the IP space but not understanding that you should reverse the IP address bytes when querying for an associated hostname. For example:

```
6    199.170.0.2.1024 PTR 54.11.193.155.in-addr.arpa.
5    199.170.0.2.1024 PTR 54.8.235.158.in-addr.arpa.
5    199.170.0.2.1024 PTR 54.3.188.143.in-addr.arpa.
```

The '54' is a parameter to the attack script; we found other instances with different values for the last byte of the IP address. This attack skews the typical query mix.

D. Microsoft's DNS Woes

On January 24, 2001, a router misconfiguration at Microsoft left their DNS servers unable to communicate with the outside world [9]. The DNS address record for the microsoft.com domain had a TTL of 2 hours. This caused nameservers around the world to expire their cached records of Microsoft's IP address after 2 hours. The Microsoft nameservers remained unreachable (they were all on the same subnet behind the misconfigured router) and the load on the root servers increased as more and more queries for microsoft.com and related names (expedia.com, passport.com, msn.com, msnbc.com, etc.) arrived at the roots. Windows nameservers do not cache negative answers, so queries were repeated again and again. We took data at 5 times during the 24th and saw the query load for Microsoft names go from normal rates of 6000/2-million queries (0%) to over 25% of the total query load. A problem in the local DNS for a popular site can have a significant disruptive effect on the root servers. Of course, Microsoft violated rule #1 for robust DNS deployment and put all of their externally visible nameservers on the same subnet. They have since outsourced their DNS provisioning.

E. Summary and Quantification of the Errors Seen

Table II summarizes the types of bad behavior seen at the F root server. Each line represents a sample taken and is labeled with the date and time. The first three samples taken on January 7-9 were 1 hour, 2 hours, and 1 hour respectively. All samples on January 10 were 2,000,000 packets (about 4 minutes); samples after January 10 were all 4,000,000 packets (about 8 minutes).

TABLE II

TAXONOMY OF BOGUS QUERIES, F ROOT NAMESERVER, JAN. 2001.

trace	rfc1918	rfc1918?	A+IP	TLD	windows	top10	top100	>1/min
jan7.11a	2.5	2.0	12.0	19.6	1.79	15.7	23.7	51.7
jan8.1p	2.6	6.1	14.9	23.1	1.36	5.3	14.1	44.0
jan9.3p	3.0	6.2	12.2	20.0	1.38	18.0	23.9	50.0
jan10.10a	3.4	7.7	12.6	22.3	1.58	5.9	14.9	24.8
jan10.11a	3.3	8.1	13.3	23.5	1.56	6.6	13.8	23.6
jan10.12p	3.6	7.4	13.4	23.7	1.91	7.9	15.4	25.1
jan10.1p	3.1	6.9	14.0	24.6	1.42	7.9	16.9	26.6
jan10.2p	3.5	7.2	14.5	25.2	1.48	6.0	13.9	24.0
jan10.3p	3.4	6.8	14.5	25.6	1.55	6.3	15.0	25.9
jan10.4p	3.1	9.7	14.6	26.1	1.55	6.2	13.1	22.5
jan10.5p	3.3	10.0	15.5	25.8	1.59	6.9	13.6	22.0
jan10.6p	3.1	9.2	17.9	28.0	1.80	5.7	12.5	22.9
jan10.7p	3.4	5.7	18.5	29.1	1.70	5.2	12.6	23.9
jan10.8p	3.5	6.7	18.7	29.4	1.65	5.7	13.3	24.9
jan10.9p	3.3	8.2	18.7	29.7	1.74	5.6	13.3	24.8
jan12.9a	2.8	8.4	13.5	23.4	1.45	3.1	9.4	26.1
jan12.5p	2.5	7.0	16.3	25.7	1.64	6.8	15.6	30.3
jan17.11	3.1	7.9	13.9	22.5	1.44	3.9	10.7	24.9
jan17.4p	3.4	11.5	16.6	25.7	1.73	2.9	8.5	23.5
jan18.6p	3.0	8.1	13.6	20.5	1.28	12.1	21.5	38.4
jan18.10p	0.03	6.3	14.1	21.4	1.34	12.4	22.3	40.1
jan19.2p	0.0	4.3	13.0	20.3	1.36	11.9	20.2	36.2
jan24.10a	0.0	4.2	14.3	25.2	1.87	3.0	8.3	22.7
jan24.2p	0.0	4.8	12.4	20.6	1.46	3.9	8.2	25.3
jan24.5p	0.0	10.7	14.5	22.9	1.60	3.6	10.3	27.7
jan24.9p	0.0	5.5	14.6	23.1	1.85	4.4	13.0	34.2

where:

rfc1918   queries from RFC 1918 private address space
rfc1918?   queries for the hostname of an RFC 1918 address
A+IP   A queries with IP address target not a hostname
TLD   queries for an invalid top level domain
windows   queries about Microsoft document system (msdcs)
top10   top 10 src/query pairs, repeated query bugs
top100   top 100 src/query pairs, repeated query bugs
>1/min   queries repeated more than once a minute

We argue that the smaller 4 minute or 8 minute samples are representative of the longer samples and of the total distribution with respect to the taxonomy of bogus queries. The numbers are percent of queries that fit into a certain category. The categories in Table II are not mutually exclusive, many queries have multiple sins.

Error percentages shown in Table II for RFC 1918 queries, bogus A records, invalid TLDs, and Windows categories consistently and tightly cluster around their average values, suggesting legitimacy in the representativeness of relatively small traces. For A records and invalid TLDs, we computed the number of distinct nameservers making those errors at least once to be sure we were not seeing a few badly broken servers. About 13% of the servers communicating with the F root issued bogus A queries and about 35% queried about bogus TLDs. We have identified several sources of the bogus A queries: the Win2k resolver, the OpenBSD resolver, and various Win95/98 viruses; we suspect the Win95/98/NT resolver as well. 35% of the nameservers are leaking information from their internal intranet out to the Internet either from Microsoft naming protocols or split DNS configuration errors. Some networking boxes (e.g. DSL network devices) may also be guilty.

The values for the top10, top100, and >1min columns in Table II are quite variable. This may indicate the need for longer sampling periods to arrive at good estimates of these values. On bad days, e.g. Jan 7, 9, 18, 19, a few broken nameservers relentlessly pummeled the roots with repeated queries. The worst offender in our data samples was a single UUnet customer who asked for a non-existent SOA record over 2 million times in an hour. Negative caching at this site would have reduced the 2 million to less than 10. On January 24, the day the Microsoft DNS servers were unreachable, the percentage of repeated queries shifted downward strongly due to the extraordinary increase in legitimate queries for microsoft.com related names.

The number of invalid top level domains queried in a relatively short period of time was surprising. We suspect Microsoft's internal naming mechanisms leaking out to the Internet, perhaps due to the documentation and defaults used in some of their products that deal with Microsoft 'domains'. Top level domain names like .local, .domain, .workgroup might indicate confusion on the part of the naive Windows user trying to configure a Microsoft product but not quite understanding the jargon.

IV. CONCLUSIONS AND FUTURE WORK

We gathered and analyzed mountains of data at the F root server. The numbers of bogus queries and broken nameservers consuming root server resources were particularly surprising. Especially disturbing were the bogus A queries that do not follow the DNS protocol specification and are prevalent enough (14% of the query load on average) to significantly impact global system load. Invalid TLDs and repeated queries also contributed significantly to bogus query counts. Negative caching, if widely deployed, would help with many of the repeated query bugs. Microsoft nameservers do not yet support negative caching as BIND does.

Further analysis of the root server measurements will enable a longer term view of systemic performance issues. In particular we need to pinpoint the origins and causes of the bogus A queries. We suspect that the win2k resolver and viruses are not the complete answer. As root nameserver operators struggle to keep up with increasing query loads, we must work to diagnose and repair egregious implementation errors and deploy negative caching to limit the impact of configuration errors to the local network.

ACKNOWLEDGMENTS

We would like to thank Paul Vixie for accommodating our measurements on the F root, and Brad Huffaker for help with the graphs.

REFERENCES

[1] P. B. Danzig, K.Obraczka, A. Kumar, *A*n Analysis of Wide-Area Name Server Traffic, ACM SIGCOMM, 1992.

[2] BIND website. http://www.isc.org/products/BIND/

[3] P. Mockapetris, *D*omain Names - Concepts and Facilities, Internet Standard 0013 (RFCs 1034, 1035), November 1987.

[4] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot and E. Lear, *A*ddress Allocation for Private Internets, RFC 1918, February 1996

[5] dns-trace-histo.pl, Paul Vixie, private communication.

[6] James Gilroy, private communication.

[7] http://www.sarc.com/avcenter/cgi-bin/virauto.cgi?vid=29038

[8] http://service1.symantec.com/
sarc/sarc.nsf/html/W32.HLLW.Bymer.html

[9] http://www.washingtonpost.com/wp-dyn/articles/A43208-
2001Jan24.html