第IV部

DNSの拡張および運用環境

第4部
DNSの拡張および運用環境

第1章 DNSワーキンググループ

DNSワーキンググループは、DNSの運用環境およびDNSの拡張に関する議論を行う。現在、DNS WGではルートネームサーバの運用技術や、DNSのセキュリティ拡張に関する議論、DNSの計測に関する研究などを主として行っている。

現在の主なトピックは以下の通り。

- DNSの計測
  DNS運用技術の研究のため、種々のDNS全般にわたっての計測について研究している。主なターゲットは以下の通り。
  1. ルートネームサーバ
  2. jp secondary サーバ
  3. ccTLD ネームサーバ
- DNSの健全化
  DNSは堅牢なシステムであり、多少誤った構成であっても動作する。そのため、構成の異常さに気が付いていない誤った構成のネームサーバが多く存在し、上位サーバやネットワークに大きな負担をかけている。
  そこで、誤った構成のネームサーバを健全化するための試みがJPNICと共同で行われている。これはまだ手法を模索している段階ではあり、チェックすべき内容とその範囲、および正常化への誘導の方法などが議論されている。

第2章 ルートネームサーバ

ルートネームサーバは世界に13個存在し、日本ではm.root-servers.netを運用している。このルートネームサーバはdns-wgのメンバーで運用されており、運用技術を研究するための計測が行われている。

2.1 ルートネームサーバの構成

ルートネームサーバは現状でIPv4とIPv6の両方のサービスを行えるシステムになっている。

ルートネームサーバのIPv6化に関しては、IPv4用サーバとIPv6用サーバを分離して構成する方法と、dual stackサーバを用いて、両方のプロトコルを同一サーバで提供する方法がある。2002年3月現在、IPv6アドレスのDNSでの表記方法の標準はまだ決まっておらず、引続き議論が行われていることから、2002年度中の正式サービスを提供することはできなかった。そのため、標準の動向に対して柔軟に対応できる構成で設計することにした。

ルートネームサーバは冗長構成になっており、東京と大阪に分離して配置されている。東京では5台のPCでIPv4のサービスを提供し、1台のPCでIPv6のサービスを提供している。大阪は東京のシステムのバックアップとして設計され、2台のPCがIPv4のバックアップサーバ、1台のPCがIPv6のバックアップサーバとなる。これらのバックアップサーバは災害時等、東京のサーバが稼働できなくなったときに稼働できるように設計されている。

2.2 ルートネームサーバの計測

上記のルートネームサーバが配置されているそれぞれのネットワーク上にトラフィックデータ測定用のサーバを配置し、ルートネームサーバに寄せられるトラフィックの測定を行っている。

第3章 DNSの計測

DNSワーキンググループでは以下のような内容でDNSの計測を行っている。

3.1 各ルートネームサーバの応答時間/応答喪失率の測定

多地点で、13個ある各ネームサーバに対して問い

合わせを送り、その応答時間と応答の喪失率を測定する。現在は測定用のツールが開発され、実際に多地点での測定が行われつつあるという状態である。

また、同様の測定を ccTLD のネームサーバに対しても行っている。

3.2 jp ドメインセカンダリサーバの測定

WIDE project では、jp ドメインのセカンダリサーバを引き受けているが、そのトラフィックの測定を行い、一定時間内に受けた問い合わせの数および回答の数を測定している。

第 4 章　Operation of a Root DNS Server

4.1 summary

The authors have been in charge of the operation of one of the root DNS servers for more than three years. In this report, the overview of our system to provide high availability is introduced. In the following sections, a traffic analysis system to analyze the characteristics of the DNS queries and the brief summary which may help future DNS system deployment is described.

4.2 Introduction

Domain Name System[101] is a hierarchically defined name space to specify resources in the Internet. Each node in the tree is associated with a label which is defined in a scope of its parent node. The root node has no label, however, it is sometimes referred to as "`.`". DNS is implemented by a large number of DNS servers distributed over the Internet. In general, each DNS server takes part in authority of one or more zones which represent contiguous part of the tree. A zone includes RRs (Resource Record) and each RR describes an IP address or other information corresponding to a node.

When a user specifies a resource on the Internet by domain name, an application program submits a DNS query to a local DNS server. If the server has the requested information (e.g. an IP address) corresponding to the specified name, the server responds with the information. If the server does not have required information, it will submit a query to another server which is expected to provide better information. If, unfortunately, the server has no knowledge regarding the domain name in question, then the server sends a query to one of the Root DNS servers, which are authoritative for domain "`.`" as a last resort. In order to make this possible, all DNS servers are assumed to have a list of the Root DNS servers, which is referred to as `root.cache` file. Each DNS server is also equipped with cache in order to reduce the number of queries to other DNS servers including the Root DNS servers.

Note that for performance reason, DNS queries should be sent in UDP rather than in TCP. As fragmentation of UDP packets on the way should be avoided as much as possible, the DNS specification requires that DNS packet length (excluding transport header) is 512 bytes or less because default minimum IP MTU has been defined in 576 bytes. While most of the Internet links in these days are capable to transmit a packet whose length is up to 1500 bytes, the limitation is still effective; most of the DNS implementations use 512 bytes buffers.

The DNS system highly depends on the Root DNS servers which are the last resort for unknown queries. We usually specify the resources in the Internet by their names rather than by their IP addresses. So the stable operation of the Root DNS Servers are extremely important in the Internet. Its operational requirements are defined in RFC 2870[24] and the servers are currently governed by ICANN RSSAC (Root Server System Advisory Committee).

4.3 Background

DNS has been developed to replace the old ARPANET host table, which had been maintained in a centralized manner and distributed to all nodes. 4.3BSD UNIX was released with an im-

plementation of DNS server called `bind` in 1986 and this successors have become the most popular implementations of DNS. A `root.cache` file generated in November 1987 included 7 Root DNS servers with 10 IP addresses.

By December 1991, there was a reconfiguration of the servers and the first Root DNS server outside of U.S. became operational in Sweden. In September 1995, all of the servers were renamed to the current convention of "x`.root-servers.net`" in order to improve encoding efficiency. There were 9 servers, each with single IP address and then enhanced to include 13 servers by May 1997. IEPG decided relocation of two Root DNS servers to out of U.S. to cope with the growth of the Internet and the selected locations were London and Tokyo. The authors have been working on development and operation of the Root DNS server in Tokyo, sometimes referred to as "`M`."

4.4 Configuration

While most of the Root DNS servers have served some top level domains like `.edu` and `.com` as well as the root zone, `M` has served the root zone only from its epoch. (Note that after reconfiguration in 2000, top level domains have moved to other servers.)

As the Root DNS servers are critical for the Internet, it is requested that each server provides high availability even through there are 13 servers in the world. `M` has been operational since August 1997 and the configuration basically remains unchanged. The system consists of two servers and two routers. Each router is connected to a major Internet exchange point as shown in Figure 4.1. One of the servers works as a primary server and usually it serves everything. The other server is used as a backup, however, the server process is always operational. The routers send queries to the primary server when it is operational. Otherwise they send queries to the backup server.

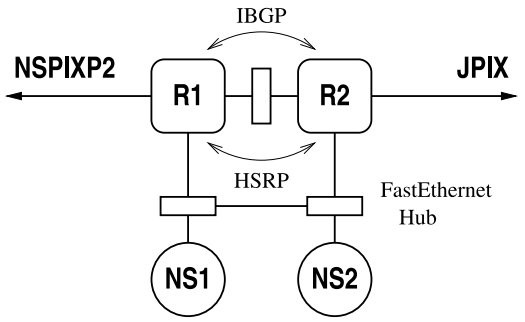

Fig. 4.1. System Configuration

Server selection is performed by RIP[63]. A RIP process resides on a server periodically checks if the server process is running and reports the service address reachability with a configured metric. When the RIP process finds the server process is dead, it sends a RIP update with metric 16 immediately so that the routers can reroute query packets to the backup. Usually, convergence of RIP is not fast. We modified timing parameters of RIP to send updates in every 10 seconds and the routers give up a route if no RIP update is received in 40 seconds. This makes it possible to switch to the working server within 1 minute in the worst case.

To check activeness of the DNS server process by the RIP process is useful when the server process crashed but the machine is operational. Even in this situation (happened about 10 times in 3.5 years), the router will send queries to the backup server within 20 seconds.

The routers run Cisco's HSRP so that the DNS servers are able to send all responses to a logical address defined by HSRP. Therefore, the servers don't have to receive RIP updates. As each router is connected to a different exchange point, there is no single point of failure in the system other than the power supply.

The growth of one day average traffic in query per second is illustrated in Figure 4.2. Glitches are mainly due to the server hardware/software upgrade. Note that thanks to the dual server system, while statistics was not correctly recorded, there was no service interruption according to the upgrade. Only exception was interface upgrade of a router and the service was unavailable for about 15 minutes. Other glitches were due to DoS type security incidents and contiguous queries by a con-

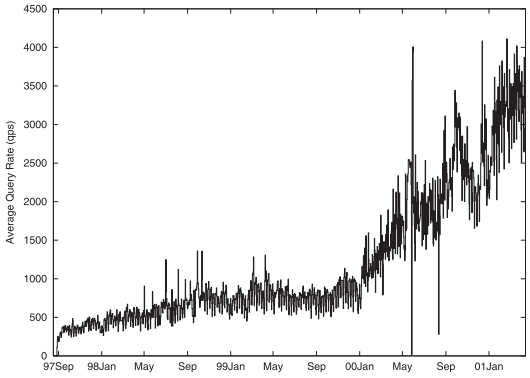

Fig. 4.2. Traffic Growth of `M` DNS Server

figuration trouble of another DNS server.

4.5 Traffic Analysis System

There are a large number of servers running in the Internet. Among them, DNS servers are unique in a sense that most of the Internet users are depending on DNS system implicitly through regular web access or sending emails. By analyzing traffic at a Root DNS server, it is expected that a generic way of summary of the Internet activities can be known.

According to the DNS servers version analysis by Yuji Sekiya in late 1999, more than 96% of the DNS servers are any versions of `bind`. At startup, `bind` checks the up-to-date list of the Root DNS servers by sending a query to one of the Root DNS servers known by `root.cache` file.

When several name servers are available according to a particular zone, the `bind` at boot up assigns estimated RTT (Round Trip Time) randomly and chooses a server with smallest RTT value. Then it updates the RTT value based on the actual measurement. `bind` also reduces the RTT value for other name servers slightly so that their values will eventually become smallest. This technique enables all of the name servers to be used while one with the smallest RTT value is frequently used to minimize the average RTT. This fact indicates that every DNS server has possibility to send queries to all Root DNS servers. It may be possible to estimate the global DNS traffic by analyzing the traffic at a Root DNS server.

The traffic analysis system is illustrated in Figure 4.3. It is a totally passive system which does not send any packets other than its control traffic. In order to consider the relationship between the routing system and the captured traffic, there is a BGP[121] module in the system so that all routing information is introduced by having a EBGP session with a router of the Root DNS system. In order to sort the information based on the prefix, there is a routing table like tree structure and Radish tree is used in this system. It is based on TRIE and provides convenient and fast lookup as well as node add/delete operations.

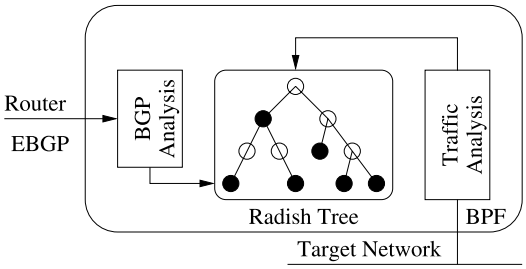

Fig. 4.3. Traffic Analysis System

The DNS traffic is captured via BPF and all but DNS query packets are filtered out at BPF or in the program. If the traffic analysis at host granularity as well as BGP prefix granularity is required, host information is put into the radish tree. This method is not very efficient in the memory utilization because the number of the nodes in the tree is about twice of the number of prefixes and hosts. However, the maximum number of lookups is proportional to the length of IP address and the number of discarded packets can be minimized.

Two sets of counters are associated to each prefix or each host. One of them are updated by the captured traffic and the other is used for output to a file. A control command or a signal swaps the two sets of counters and starts writing to a file. Other information including origin AS number and AS hop count (AS path length) is associated with each prefix.

When a DNS query packet is captured via a BPF interface, the program looks up the Radish table with the source IP address of the packet. If no exact match is found, an entry corresponding to the source host is created. Then the counters

of the entry is incremented accordingly. The program also follows the Radish tree upward to find an active prefix learned from BGP, and increments the counters there. If no active prefix is found, then it increments the counters of the root node.

All of the counters have to be maintained until the statistics are written to a file even if BGP cease messages are received. In a regular routing table, corresponding nodes should be removed immediately. In this program only a flag indicating its activeness is cleared. This yields the program's memory usage grows up monotonously. Our experience shows that most of those prefixes will become active again eventually, and this is not a serious problem. But the number of the source hosts can be large, periodical cleanup is performed only for host nodes. If all of the current and past counters are zeros, then the node is removed from the tree.

The traffic analysis system is built on a PentiumIII/850 with 512MB memory running BSD/OS 4.1. Its load is light enough except when the program writes the statistics to a file or when the program performs cleanups. In the current environment with about 100,000 prefixes learned from BGP, the program occupies about 40-60MB memory at startup.

4.6 Traffic Analysis

The traffic analysis in this chapter is based on a 72 hours traffic monitoring performed in April 2001. There was about 785 million inbound packets (3028 packets per second). Outbound packets were not taken into account. Almost all of the packets were in UDP and the packets in TCP was only 0.093%. Figure 4.4 illustrates 1-hour average traffic in packet per second. As the Root DNS servers are used by DNS servers all over the world, fluctuation of the traffic change in time is not large. About 0.13% of the packets were from unknown source addresses, i.e., the private address space or other prefixes which were not available in BGP. While not all of the inbound packets were DNS queries, ratio of other types of message

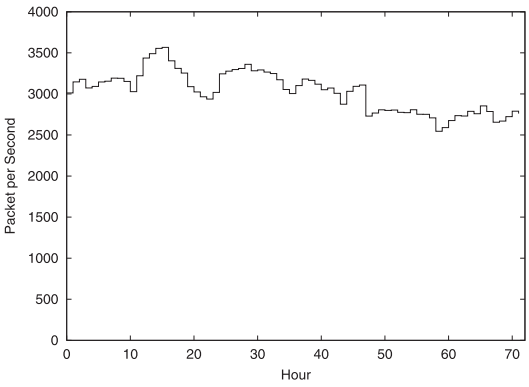

Fig. 4.4. 1 Hour Average Traffic

including notification and update was negligibly small.

In each hour, the number of hosts observed was from 85 thousands to 138 thousands with 108 thousands average. During the particular 72 hours, 781 thousands of hosts sent their packets to M.

While we had about 103 thousands routes in the default free zone of the Internet, the number of hosts which sent packets to M was only 22 thousands in 1 hour average, or 46 thousands during the 72 hours. Cumulative growth of the number of the active prefix is shown in Figure 4.5. This indicates that nearly half of the current Internet prefixes are either 1) not actively used (just prefixes were announced), or 2) no DNS servers inside.

The number of the origin AS of those 781 thousands of hosts observed in the 72 hours was 8755. The average AS path length to those 8755 ASes was 3.83 (this number didn't include the AS of the M server) or 3.20 (weighted by the number of packets) which is illustrated in Figure 4.6. Be-

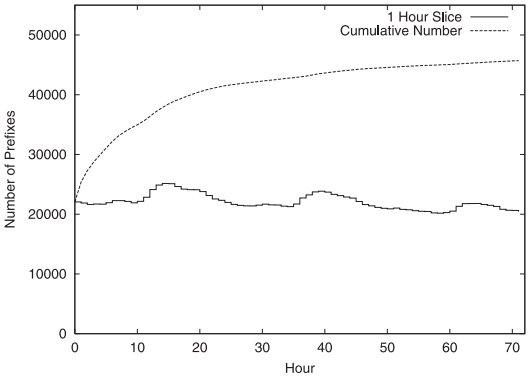

Fig. 4.5. Cumulative Number of Active Prefix

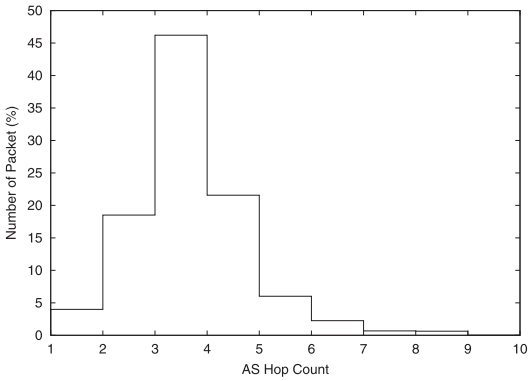

Fig. 4.6. AS Hop Count Distribution

cause most of modern routers prefer prefixes with shortest AS path length, 90% of the world was reachable within 4 AS hops.

The TTL field in the IP header indicates the number of hops from the source and the M server. The distribution of the TTL values for 785 million queries is shown in Figure 4.7. There are 4 peaks in the figure, indicating the TTL value at the origin. In 4.3BSD UNIX BNR2 released in 1988 it was 30 whereas RFC1060[124] published in 1990 defined suggested TTL value was 32. As it is not possible to distinguish those two, 30 was assumed in this paper. Other possible values are 64 based on RFC1340[122] released in 1992, 128, and 255. With those numbers, it is possible to estimate the number of the hops from the source to the destination, M server, as shown in Figure 4.8. The average of hop count was 15.2. Note that about 2% hosts are considered using TTL of 30 or 32. This suggests that very old implementations are still working in the Internet.

Based on the origin AS numbers, it is possible

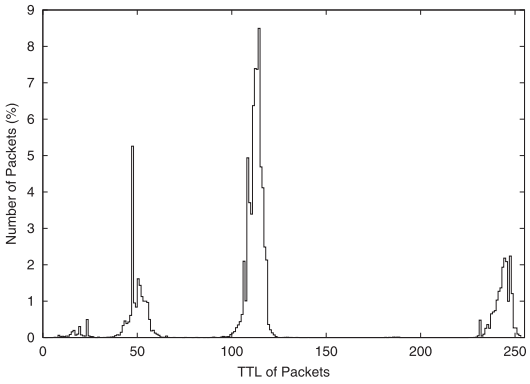

Fig. 4.7. TTL distribution

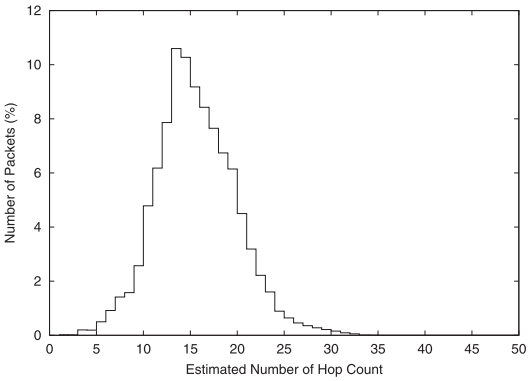

Fig. 4.8. Estimated Hop Count Distribution

to estimate the origin of the countries to know the world-wide distribution. For those ASes who contributed more than 0.01% of the entire traffic out of 8755 ASes, we examined the country of the ASes with the registration record of each Regional Internet Registry. 96.5% of entire traffic was classified. We assumed that if an AS was registered in U.S., all of the packets whose origin AS was that one were all from U.S.

One of the authors performed similar analysis in June 1999 based on a different system[4]. It captured IP headers of all packets and analyzed later based on a BGP routing table. It showed the similar trends where the majority of the packets were from U.S. However, the order of other countries was KR, JP, TW, MX, UK, CA, DE, AU, CN, BR, and so on. A notable difference is that the packets from CN has been increased compared with other countries.

The analysis system described in this paper records only summarized information currently taken every hour. The growth of output file is about 100MB per day and it is easy to handle in the storage capacity available these days. While the system does not record each packet, it is easy to modify the system so that not only the IP header but also the fields of the DNS packet are

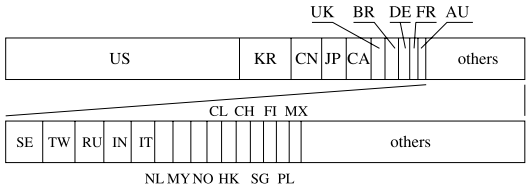

Fig. 4.9. Per Country Traffic

taken into account.

The current system also logs the number of "root query", which is a query from DNS servers to check the integrity of the Root DNS servers. `M` server receives such queries once per 1.75 second in average. This fact suggests that a possible way to introduce new RRs to describe the IPv6 addresses of the Root DNS Servers. When a DNS response does not fit into a single UDP packet, the server set a "truncation occurred" flag. In this case, the client may fall back to use TCP to query again to the server. This may result in a delay of DNS transaction as well as an extra overhead to the server. Provided the rate of such queries is small, using TCP may be a solution. At a Root Server, the router may redirect the TCP queries to other servers for performance improvement.

There is a DNS implementation based on EDNS0[140], which makes it possible to extend the message size limitation of 512 bytes. It sends a query with OPT RRs indicating the maximum payload size that it is able to receive. Unfortunately, most popular implementation of DNS, `bind` version 8, does not handle this extension and returns an error. The servers based on EDNS0 then send queries without OPT RRs to those servers. Due to stability and performance requirements, `M` runs `bind8` as of this writing and is unable to respond queries with OPT RRs. The cumulative number of hosts which sent queries with OPT RRs is illustrated in Figure 4.10. While the observation was not enough to determine the number of hosts equipped with the latest version of DNS implementations, it may be possible to conclude that the number was less than 10 thousands.

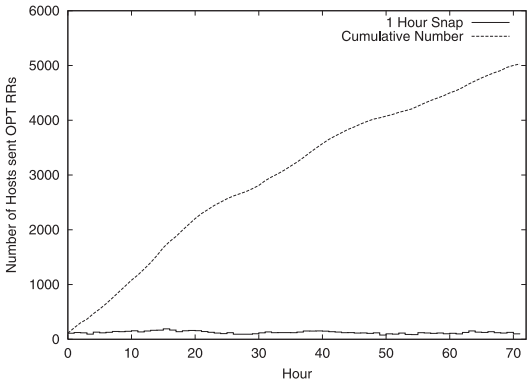

Fig. 4.10. Number of Hosts sending queries with OPT RRs

4.7 Summary

In this paper, the current configuration of the `M` Root DNS server is introduced. The dual server system with RIP and HSRP has been working well for 3.5 years. As the system has redundant configuration, it is easy to upgrade the server software or server hardware to enhance the service or to fix security holes without causing service interruption.

According to the deployment of IPv6 and security demands, support of new features including IPv6 specific RRs and secure DNS is required while the packet size limitation is still effective. In the transition phase of those new functionalities, redundant configuration is useful. By introducing new features only in the primary server, the backup server may take over within a minute when the primary server crashes.