

DNS Session 1: Fundamentals

Brian Candler
NSRC

Computers use IP addresses. Why do we need names?

- Easier for people to remember
- Computers may be moved between networks, in which case their IP address will change

1

Old solution: hosts.txt

- A centrally-maintained file, distributed to all hosts on the Internet

```
SPARKY           128.4.13.9
UCB-MAILGATE     4.98.133.7
FTPHOST          200.10.194.33
... etc
```

This feature still exists:
/etc/hosts [Unix]
c:\windows\hosts [Windows]

3

hosts.txt doesn't scale

- ✗ Huge file
- ✗ Needs frequent copying to ALL hosts
- ✗ Consistency
- ✗ Always out-of-date
- ✗ Name uniqueness
- ✗ Single point of administration

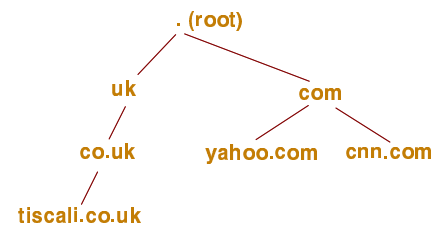
4

The Domain Name System was born

- DNS is a Distributed Database for holding name to IP address (and other) information
- Distributed:
 - Shares the administration
 - Shares the load
- Robustness and performance through:
 - Replication
 - Caching
- A *critical* piece of Internet infrastructure

5

DNS is Hierarchical



Forms a tree structure

6

DNS is Hierarchical (2)

- Gives globally unique names
- Administered in zones (parts of the tree)
- You can give away ("delegate") control of part of the tree underneath you
- Example:
 - isoc.org on one set of nameservers
 - isocws.isoc.org on a different set
 - t1.isocws.isoc.org on another set

7

Domain Names are (almost) unlimited

- Max 255 characters total length
- Max 63 characters in each part
 - RFC 1034, RFC 1035
- If a domain name is being used as a host name, you should abide by some restrictions
 - RFC 952 (old!)
 - a-z 0-9 and minus (-) only
 - No underscores (_)

8

Using the DNS

- A Domain Name (like www.tiscali.co.uk) is the KEY to look up information
- The result is one or more RESOURCE RECORDS (RRs)
- There are different RRs for different types of information
- You can ask for the specific type you want, or ask for "any" RRs associated with the domain name

9

Commonly seen RRs

- A (address): map hostname to IP address
- PTR (pointer): map IP address to name
- MX (mail exchanger): where to deliver mail for user@domain
- CNAME (canonical name): map alternative hostname to real hostname
- TXT (text): any descriptive text
- NS (name server), SOA (start of authority): used for delegation and management of the DNS itself

1

Simple example

- Query: www.tiscali.co.uk
- Query type: A
- Result:

```
www.tiscali.co.uk.  IN  A  212.74.101.10
```

In this case just a single RR is found,
but in general, multiple RRs may be returned

(IN is the "class" for INTERNET use of the DNS)

11

Possible results

- Positive (one or more RRs found)
- Negative (definitely no RRs match the query)
- Server fail (cannot find the answer)

1

How do you use an IP address as the key for a DNS query?

- Convert the IP address to dotted-quad
- Reverse the four parts
- Add ".in-addr.arpa." to the end; special domain reserved for this purpose
e.g. to find name for 212.74.101.10

10.101.74.212.in-addr.arpa.
→ PTR **www.tiscali.co.uk.**

Known as a "reverse DNS lookup"
(because we are looking up the name for an IP address,
rather than the IP address for a name)



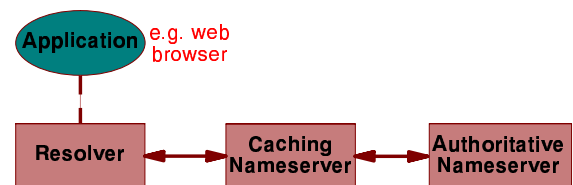
13

1

DNS is a Client-Server application

- (Of course - it runs across a network)
- Requests and responses are normally sent in UDP packets, port 53
- Occasionally uses TCP, port 53
→ for very large requests, e.g. zone transfer from master to slave

There are three roles involved in DNS



15

1

Three roles in DNS

- **RESOLVER**
→ Takes request from application, formats it into UDP packet, sends to cache
- **CACHING NAMESERVER**
→ Returns the answer if already known
→ Otherwise searches for an authoritative server which has the information
→ Caches the result for future queries
→ Also known as **RECURSIVE** nameserver
- **AUTHORITATIVE NAMESERVER**
→ Contains the actual information put into the DNS by the domain owner

17

Three roles in DNS

- The **SAME** protocol is used for resolver→cache and cache→auth NS communication
- It is possible to configure a single name server as both caching and authoritative
- But it still performs only one role for each incoming query
- Common but **NOT RECOMMENDED** to configure in this way (see later)

1

ROLE 1: THE RESOLVER

- A piece of software which formats a DNS request into a UDP packet, sends it to a cache, and decodes the answer
- Usually a shared library (e.g. libresolv.so under Unix) because so many applications need it
- EVERY host needs a resolver - e.g. every Windows workstation has one

19

How does the resolver find a caching nameserver?

- It has to be explicitly configured (statically, or via DHCP etc)
- Must be configured with the IP ADDRESS of a cache (why not name?)
- Good idea to configure more than one cache, in case the first one fails

2

How do you choose which cache(s) to configure?

- Must have PERMISSION to use it
 - e.g. cache at your ISP, or your own
- Prefer a nearby cache
 - Minimises round-trip time and packet loss
 - Can reduce traffic on your external link, since often the cache can answer without contacting other servers
- Prefer a reliable cache
 - Perhaps your own?

21

Resolver can be configured with default domain(s)

- If "foo.bar" fails, then retry query as "foo.bar.mydomain.com"
- Can save typing but adds confusion
- May generate extra unnecessary traffic
- Usually best avoided

2

Example: Unix resolver configuration

/etc/resolv.conf

```
search tiscali.co.uk
nameserver 212.74.112.66
nameserver 212.74.112.67
```

That's all you need to configure a resolver

23

Testing DNS

- Just put "www.yahoo.com" in a web browser?
- Why is this not a good test?

2

Testing DNS with "dig"

- "dig" is a program which just makes DNS queries and displays the results
- Better than "nslookup", "host" because it shows the raw information in full

```
dig tiscali.co.uk.  
-- defaults to query type "A"  
  
dig tiscali.co.uk. mx  
-- specified query type  
  
dig @212.74.112.66 tiscali.co.uk. mx  
-- send to particular cache (overrides  
  /etc/resolv.conf)
```

25

The trailing dot

dig tiscali.co.uk.

- Prevents any default domain being appended
- Get into the habit of using it always when testing DNS
 - only on domain names, not IP addresses or E-mail addresses

2

```
# dig www.gouv.bj. a  
; <<> DiG 9.3.0 <<> www.gouv.bj a  
; global options: printcmd  
; Got answer:  
; ->HEADER<- opcode: QUERY, status: NOERROR, id: 2462  
; flags: qr aa rd ra QUERY: 1, ANSWER: 2, AUTHORITY: 4, ADD'L: 4  
; QUESTION SECTION:  
;www.gouv.bj. IN A  
  
; ANSWER SECTION:  
www.gouv.bj. 86400 IN CNAME waib.gouv.bj.  
waib.gouv.bj. 86400 IN A 81.91.232.2  
  
; AUTHORITY SECTION:  
gouv.bj. 86400 IN NS rip.psg.com.  
gouv.bj. 86400 IN NS ben02.gouv.bj.  
gouv.bj. 86400 IN NS nakayo.leland.bj.  
gouv.bj. 86400 IN NS ns1.intnet.bj.  
  
; ADDITIONAL SECTION:  
ben02.gouv.bj. 86400 IN A 81.91.232.1  
nakayo.leland.bj. 18205 IN A 81.91.225.1  
ns1.intnet.bj. 18205 IN A 81.91.225.18  
rip.psg.com. 160785 IN A 147.28.0.39  
  
; Query time: 200 msec  
; SERVER: 212.74.112.67#53(212.74.112.67)  
; WHEN: Tue Dec 28 19:50:01 2004  
; MSG SIZE rcvd: 237
```

27

Interpreting the results: header

- STATUS
 - NOERROR: 0 or more RRs returned
 - NXDOMAIN: non-existent domain
 - SERVFAIL: cache could not locate answer
- FLAGS
 - AA: Authoritative answer (not from cache)
 - You can ignore the others
 - QR: Query/Response (1 = Response)
 - RD: Recursion Desired
 - RA: Recursion Available

2

Interpreting the results

- Answer section (RRs requested)
 - Each record has a Time To Live (TTL)
 - Says how long the cache will keep it
- Authority section
 - Which nameservers are authoritative for this domain
- Additional section
 - More RRs (typically IP addresses for the authoritative nameservers)
- Total query time
- Check which server gave the response!
 - If you make a typing error, the query may go to a default server

29

Practical Exercise

- Configure Unix resolver
- Issue DNS queries using 'dig'
- Use tcpdump to show queries being sent to cache

3