Domain Name System (DNS) Fundamentals

Mike Jager
Network Startup Resource Center
mike.jager@synack.co.nz

Slide Title (Arial 44)

- Initial bullet point (Arial 32)
  - Secondary bullet (Arial 28)
    - Tertiary bullet (Arial 24)

Standard text (Arial 32)

Why Use Domain Name System?

Computers Use IP Addresses
Why Do We Need Names?

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

HOSTS.TXT

The old way: 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

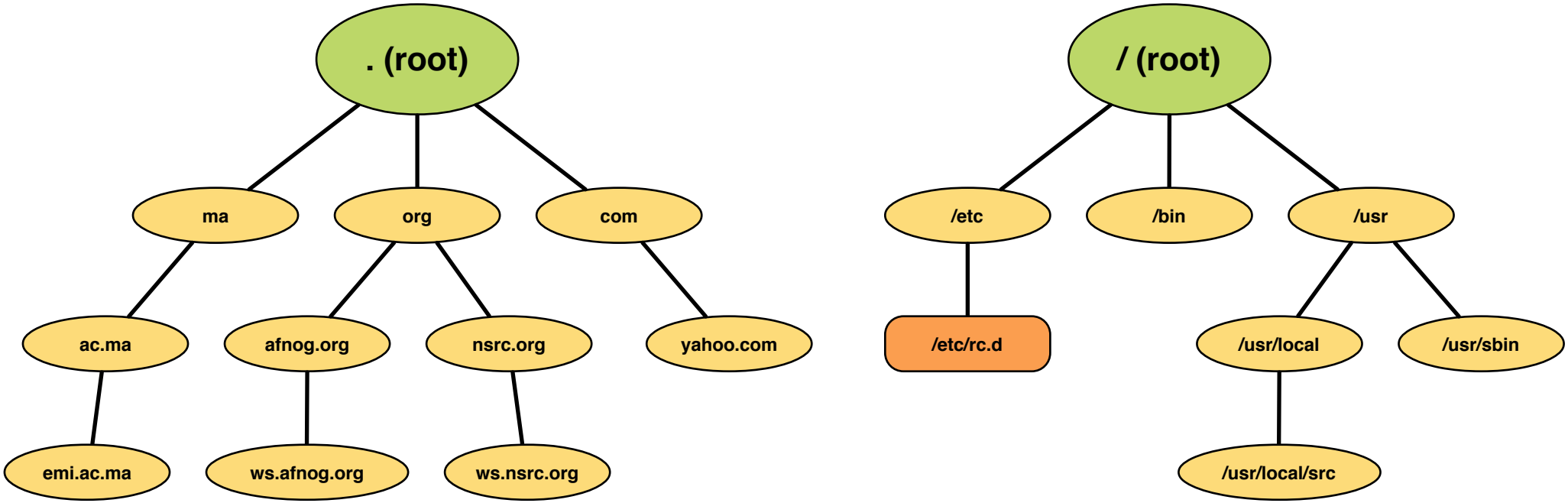
hosts.txt does not scale

- Huge file (traffic and load)
- Name collisions (name uniqueness)
- Consistency
- Always out of date
- Single point of Administration
- Did not scale well

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 & improved performance through
  - replication
  - and caching
- Employs a client-server architecture
- A critical piece of the Internet's infrastructure

DNS is Hierarchical



DNS Database **Unix Filesystem**

It forms a tree structure

DNS is Hierarchical (continued)

- Globally unique names
- Administered in zones (parts of the tree)
- You can give away ("delegate") control of part of the tree underneath you
- Example:
  - nsrc.org on one set of nameservers
  - ws.nsrc.org on a different set
  - noc.ws.nsrc.org on another set

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 (_)

Using the DNS

- A Domain Name (like www.ws.afnog.org) 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

Commonly Seen Resource Records (RRs)

- A (address): map hostname to IPv4 address
- AAAA (quad A): map a hostname to IPv6 address
- PTR (pointer): map IP address to hostname
- 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

A Simple Example

- Query: **nsrc.org.**
- Query type: **A**
- Result:

```
nsrc.org.   83855   IN   A    128.223.157.19
```

- **In this case a single RR is found, but in general, multiple RRs may be returned.**
  - (IN is the "class" for INTERNET use of the DNS)

Possible Results From A Query

- POSITIVE
  - one or more RRs found
- NEGATIVE
  - definitely no RRs match the query
- SERVER FAIL
  - cannot find the answer
- REFUSED
  - not allowed to query the server

Reverse Lookups

- Look up the name for an IP address
- 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 128.223.157.19

Domain name: 19.157.223.128.in-addr.arpa.

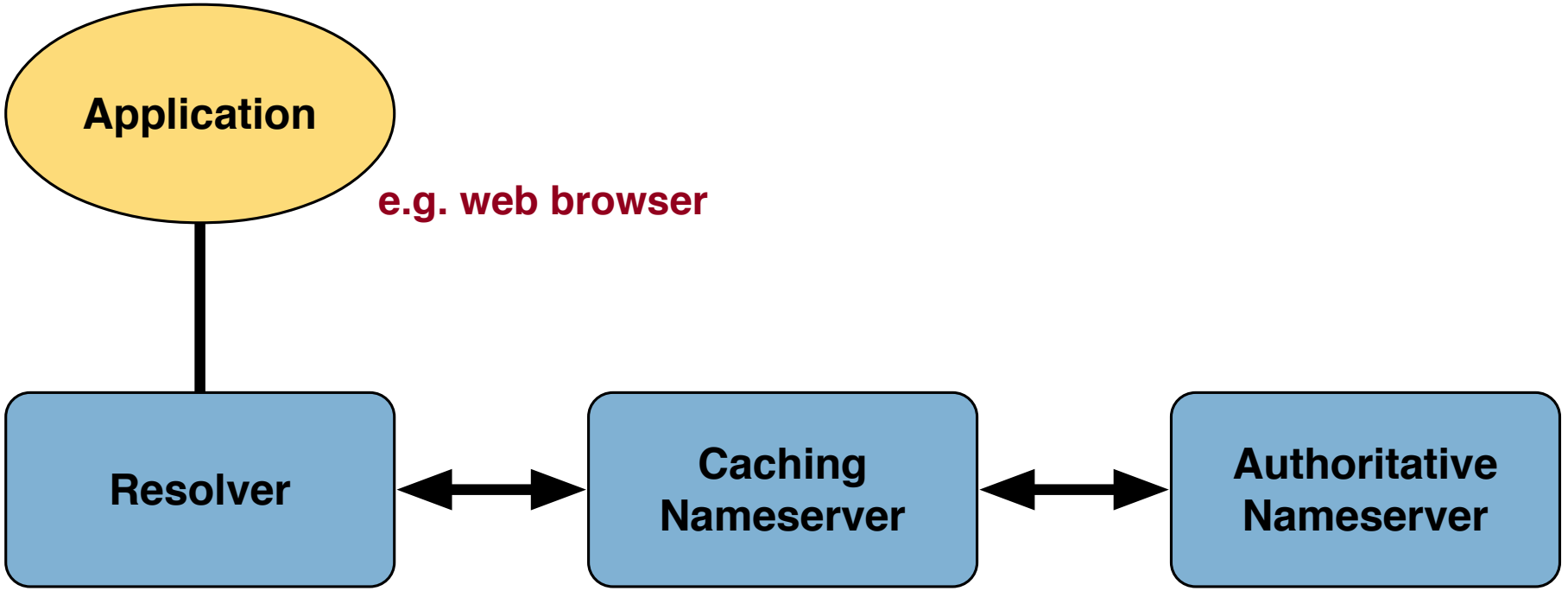
Query Type: PTR

Result: nsrc.org.

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 large requests (larger than 512-bytes) e.g. zone transfer from master to slave or IPv6 AAAA (quad A) record.

The Three Roles of DNS



The Three Roles of DNS

- RESOLVER
  - Takes app request, creates a UDP packet, sends to cache
- CACHING NAMESERVER
  - Returns the answer if already known
  - Or searches for an authoritative server with information
  - Caches the result for future queries
  - Also known as RECURSIVE nameserver
- AUTHORITATIVE NAMESERVER
  - Contains information put into the DNS by domain owner

The Three Roles of DNS

- The SAME protocol is used for
  - resolver ↔ cache
  - cache ↔ auth NS communication
- One name server can be caching & authoritative
- It still performs only one role for each incoming query
- It's NOT RECOMMENDED to use one server for both
  - we will see why later

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

How does the name server find a caching resolver?

- 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

Which Cache Should You Use?

- 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?

Resolvers Can Have Default Domains

- 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

Example: Unix Resolver Configuration

/etc/resolv.conf

```
nameserver 10.10.0.254
domain ws.nsrc.org
search ws.nsrc.org
```

That's all you need to configure a resolver

Testing DNS

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

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 nsrc.org.
  -- defaults to query type "A"
dig nsrc.org. mx
  -- specified query type
dig @128.223.157.19 nsrc.org. mx
  -- send to particular cache (overrides
     /etc/resolv.conf)
```

The Trailing Dot

```
# dig nsrc.org.
```

- Prevents any default domain being appended
- Always use it when testing DNS
  - only on domain names, not IP addresses or e-mail addresses

Anatomy of a DNS Query

```
[field@term /usr/home/field]$ dig @zoe.dns.gh. downloads.dns.gh. a
; <<>> DiG 9.7.0-P1 <<>> @zoe.dns.gh. downloads.dns.gh. a
; (1 server found)
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 34963
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 3, ADDITIONAL: 0

;; QUESTION SECTION:
;downloads.dns.gh.              IN      A

;; ANSWER SECTION:
downloads.dns.gh.       3600    IN      CNAME   zoe.dns.gh.
zoe.dns.gh.             3600    IN      A       147.28.0.23

;; AUTHORITY SECTION:
dns.gh.                 3600    IN      NS      zoe.dns.gh.
dns.gh.                 3600    IN      NS      mantse.gh.com.
dns.gh.                 3600    IN      NS      snshq902.ghanatel.com.gh.

;; ADDITIONAL SECTION:
zoe.dns.gh.       3600 IN   AAAA 2001:418:1::23

;; Query time: 287 msec
;; SERVER: 147.28.0.23#53(147.28.0.23)
;; WHEN: Tue Apr 17 08:04:58 2012
;; MSG SIZE  rcvd: 173
```

Understanding Output from dig

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

Understanding Output from dig

- 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 addresses for authoritative nameservers)
  - AAAA ("quad A") record or the IPv6 address
- Total query time
- Check which server gave the response!
  - If you make a typing error, the query may go to a default server

Practical Exercise

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