CS 43: Computer Networks

Naming and DNS

September 24, 2019
Last class

• DNS: Domain Name System
  – Core Internet Functionality
  – Application Layer Protocol – E2E design!
  – Client-Server Architecture
  – Hierarchical, Distributed
Today

• Domain Name System
  – Query sequences: Record types
  – Caching: Load Balancing
  – Security
DNS: domain name system

Input: www.google.com  (hostname)
Output: 8.8.8.8  (IP address)
Recall: TCP/IP Protocol Stack

Human-readable strings: www.example.com

(Not much addressing here, ports to ID socket)

IP addresses (32-bit IPv4, 128-bit IPv6)

(Network dependent) Ethernet: 48-bit MAC address
DNS protocol, messages

- **query and reply messages, both with same message format!**

**Binary Protocol!**
- Delimiters: pre-defined lengths/field
- Names: `<len><name>`

**Sent via UDP (User Datagram Protocol)**
- No connection established
- Not reliable

<table>
<thead>
<tr>
<th>identification</th>
<th>flags</th>
</tr>
</thead>
<tbody>
<tr>
<td># questions</td>
<td># answer RRs</td>
</tr>
<tr>
<td># authority RRs</td>
<td># additional RRs</td>
</tr>
</tbody>
</table>

- questions (variable # of questions)
- answers (variable # of RRs)
- authority (variable # of RRs)
- additional info (variable # of RRs)
DNS protocol, messages

- **query** and **reply** messages, both with same **message format**

**Message header**
- **identification**: 16 bit id for query, reply to query uses same id.
- **flags**: recursion, query/reply
- # Resource Records to follow

<table>
<thead>
<tr>
<th>identification</th>
<th>flags</th>
</tr>
</thead>
<tbody>
<tr>
<td># questions</td>
<td># answer RRs</td>
</tr>
<tr>
<td># authority RRs</td>
<td># additional RRs</td>
</tr>
</tbody>
</table>

- questions (variable # of questions)
- answers (variable # of RRs)
- authority (variable # of RRs)
- additional info (variable # of RRs)
## DNS Services

**DNS**: distributed DB storing resource records (RR)

| RR format: (name, value, type, ttl) |

- **type=A**
  - name is hostname
  - value is IP address

- **type=NS**
  - name is domain (e.g., foo.com)
  - value is hostname of authoritative name server for this domain

- **type=CNAME**
  - name is alias name for some “canonical” (the real) name
  - www.ibm.com is really servereast.backup2.ibm.com
  - value is canonical name

- **type=MX**
  - value is name of mailserver associated with name
DNS Types

RR format: (name, value, type, ttl)

- Type = A / AAAA
  - Name = domain name
  - Value = IP address
  - A is IPv4, AAAA is IPv6

- Type = NS
  - Name = partial domain
  - Value = name of DNS server for this domain
  - “Go send your query to this other server”

Query
Name: cs.swarthmore.edu
Type: A

Resp.
Name: cs.swarthmore.edu
Value: 130.58.68.9

Query
Name: cs.swarthmore.edu
Type: NS

Resp.
Name: cs.swarthmore.edu
Value: 130.58.68.9
DNS Types, Continued

RR format: (name, value, type, ttl)

• Type = CNAME
  – Name = hostname
  – Value = canonical hostname
  – Useful for aliasing
  – CDNs use this

• Type = MX
  – Name = domain in email address
  – Value = canonical name of mail server
DNS Directory Design: which would you choose?

A. Flood the query to all end-hosts, the end-host with the name responds.
B. Centralized server: all data and queries handled by one machine.
C. Push data to all end-hosts (/etc/hosts): each end-host stores the full listing.
D. Something else
Domain Name System (DNS)

• Distributed administrative control
  – Hierarchical name space divided into zones
  – Distributed over a collection of DNS servers
Domain Name System (DNS)

- Hierarchy of DNS servers
  - Root servers
  - Top-level domain (TLD) servers
  - Authoritative DNS servers
Domain Name System (DNS)

• Performing the translations
  – Local DNS servers
  – Resolver software
DNS: a distributed, hierarchical database

Root DNS Servers

- com DNS servers
  - yahoo.com DNS servers
  - amazon.com DNS servers

- org DNS servers
  - pbs.org DNS servers

- edu DNS servers
  - swarthmore.edu DNS servers
  - umass.edu DNS servers
DNS: a distributed, hierarchical database

Root DNS Servers

- com DNS servers
  - yahoo.com DNS servers
  - amazon.com DNS servers

- org DNS servers
  - pbs.org DNS servers

- edu DNS servers
  - swarthmore.edu DNS servers
  - umass.edu DNS servers
  - cs.swarthmore.edu DNS servers
DNS: a distributed, hierarchical database

Root DNS Servers

com DNS servers
- yahoo.com DNS servers
- amazon.com DNS servers

org DNS servers
- pbs.org DNS servers

edu DNS servers
- swarthmore.edu DNS servers
- umass.edu DNS servers
- cs.swarthmore.edu DNS servers

Max Tree Depth = 128

allspice.cs.swarthmore.edu.

Host (other cs hosts)
DNS: a distributed, hierarchical database

- Root DNS Servers
  - com DNS servers
    - yahoo.com DNS servers
  - org DNS servers
    - amazon.com DNS servers
    - pbs.org DNS servers
  - edu DNS servers
    - swarthmore.edu DNS servers
      - cs.swarthmore.edu DNS servers
        - allspice.cs.swarthmore.edu Host

- Nameless root, Usually implied.
- allspice.cs.swarthmore.edu.
Why do we structure DNS like this? Which of these helps the most? Drawbacks?

A. It divides up responsibility among parties.

B. It improves performance of the system.

C. It reduces the size of the state that a server needs to store.

D. Some other reason.
DNS: a distributed, hierarchical database

- Root DNS Servers
  - com DNS servers
    - yahoo.com DNS servers
  - org DNS servers
    - amazon.com DNS servers
    - pbs.org DNS servers
  - edu DNS servers
    - swarthmore.edu DNS servers
      - cs.swarthmore.edu DNS servers
        - allspice.cs.swarthmore.edu (other cs hosts)
        - Host
    - umass.edu DNS servers

ICANN
DNS: Root Name Servers

- **Know how to find top-level domains (.com, .edu, .gov, etc.)**
- approx. 400 total root servers
- Significant amount of traffic is not legitimate

13 root name “servers” worldwide

- a. Verisign, Los Angeles CA (5 other sites)
- b. USC-ISI Marina del Rey, CA
- c. Cogent, Herndon, VA (5 other sites)
- d. U Maryland College Park, MD
- e. NASA Mt View, CA
- f. Internet Software C. Palo Alto, CA (and 48 other sites)
- g. US DoD Columbus, OH (5 other sites)
- h. ARL Aberdeen, MD
- i. Netnod, Stockholm (37 other sites)
- j. Verisign, Dulles VA (69 other sites)
- k. RIPE London (17 other sites)
- l. ICANN Los Angeles, CA (41 other sites)
- m. WIDE Tokyo (5 other sites)
Root Name Servers

- Responsible for the Root Zone File

com. 172800 IN NS a.gtld-servers.net.
com. 172800 IN NS b.gtld-servers.net.
com. 172800 IN NS c.gtld-servers.net.

- In practice, most systems cache this information
- Lists the TLDs and who controls them
- ~272KB in size
Top Level Domain (TLD) servers

Verisign

Root DNS Servers

com DNS servers
- yahoo.com DNS servers
- amazon.com DNS servers

org DNS servers
- pbs.org DNS servers

edu DNS servers
- swarthmore.edu DNS servers
- umass.edu DNS servers
- cs.swarthmore.edu DNS servers
  - allspice.cs.swarthmore.edu (other cs hosts)
  - Host

Slide 27
Top Level Domain (TLD) servers

• who maintains the servers?:
  – Verisign: .com, .net
  – Educause: .edu (Verisign backend)
  – local governments or companies

• Responsible for:
  – com, org, net, edu, gov, aero, jobs, museums,
  – all top-level country domains, e.g.: uk, fr, de, ca, jp, etc
Authoritative Servers

Root DNS Servers

...  ...

com DNS servers  org DNS servers  edu DNS servers

yahoo.com DNS servers  amazon.com DNS servers  pbs.org DNS servers

swarthmore.edu DNS servers  umass.edu DNS servers

cs.swarthmore.edu DNS servers

allspice.cs.swarthmore.edu (other cs hosts) Host
Authoritative Servers

• Organization’s own DNS server(s),
  – for organization’s named hosts
  – authoritative hostname - IP mappings

• maintained by:
  – organization or service provider, easily changing entries
  – Often, but not always, acts as organization’s local name server (for responding to look-ups)
Resolution Process: As an end host if you want to look up a hostname (umass.edu) who do you contact?

A. Contact the swarthmore DNS servers
B. Contact edu DNS servers
C. Contact the Root DNS servers
D. Someone else should do this job.
Local DNS Name Server

• Each ISP
  – (residential ISP, company, university) ...
  – has (at least) one

• also called “default name server”
• Local DNS server
  – acts as proxy, forwards query into hierarchy
  – has local cache of recent name-to-address translation pairs (but may be out of date!)
allspice wants IP address for gaia.cs.umass.edu

iterative query:
- contacted server replies with name of server to contact
  - “I don’t know this name, but ask this server”
recursive query:

- each server asks the next one, in a chain

DNS name resolution example #2

requesting host: allspice.cs.swarthmore.edu

local DNS server: dns.cs.swarthmore.edu

authoritative DNS server: dns.cs.umass.edu

TLD DNS server: gaia.cs.umass.edu

root DNS server: root DNS server

Slide 36
Which would you use? Why?

A. Iterative

1. Requesting host: allspice.cs.swarthmore.edu
2. Local DNS server: dns.cs.swarthmore.edu
3. TLD DNS server: dns.cs.umass.edu
4. Authoritative DNS server: dns.cs.umass.edu
5. gaia.cs.umass.edu

B. Recursive

1. Requesting host: allspice.cs.swarthmore.edu
2. Local DNS server: dns.cs.swarthmore.edu
3. TLD DNS server: dns.cs.umass.edu
4. Authoritative DNS server: dns.cs.umass.edu
5. gaia.cs.umass.edu
Example: iterative query using dig()

dig . ns

dig +norec demo.cs.swarthmore.edu @a.root-servers.net

dig +norec demo.cs.swarthmore.edu @a.edu-servers.net

dig +norec demo.cs.swarthmore.edu @ibext.its.swarthmore.edu
demo.cs.swarthmore.edu. 259200 IN A 130.58.68.26
DNS Caching

• Why cache?
  – approx. 1 sec latency before starting a download
  – Popular sites visited often

• Where to cache?
  – Local DNS server
  – Browser
DNS Caching

• When to cache?
  – learn a mapping? cache!
  – any name server can cache

• For how long?
  – until Time To Live (expires)

• What to cache?
  – TLD servers cached – almost never change
  – Root name servers usually, not visited legitimately
The TTL value should be…

A. Short, to make sure that changes are accurately reflected

B. Long, to avoid re-queries of higher-level DNS servers

C. Something else
Caching

• (+) Subsequent requests need not burden DNS

• (-) Cached entries may be out-of-date (best effort!)
  – If host’s name or IP address changes, it may not be known Internet-wide until all TTLs expire
Inserting (or changing) records

Example: new startup “Network Utopia”
Inserting (or changing) records

Example: new startup “Network Utopia”

- Step 1: Register networkuptopia.com at DNS registrar
  - provide names, IP addresses of authoritative name server
    (primary and secondary)
Inserting (or changing) records

Example: new startup “Network Utopia”

- Step 2: Registrar inserts two RRs into .com TLD server
  - (networkutopia.com, dns1.networkutopia.com, NS)
  - (dns1.networkutopia.com, 212.212.212.1, A)
Inserting (or changing) records

Example: new startup “Network Utopia”

• Step 3: Set up authoritative server at that name/address
  – Create records for the services:
Example: new startup “Network Utopia”

• Step 3: Set up authoritative server at that name/address
  – Create records for the services:
    • type A record for www.networkuptopia.com
    • type MX record for @networkuptopia.com email
The DNS system can be attacked because

A. can’t tell of reply comes from correct source
B. can’t tell if correct source tells the truth
C. malicious source can insert extra (mis)information
D. malicious bystander can spoof (mis)information
E. All of the above
Attacking DNS

**DDoS attacks**
- Bombard root servers with traffic
  - Not successful to date
  - Traffic Filtering
  - Local DNS servers cache IPs of TLD servers, bypassing root
- Bombard TLD servers
  - Potentially more dangerous

**Redirect attacks**
- Man-in-middle
  - Intercept queries
- DNS poisoning
  - Send bogus replies to DNS server that caches

**Exploit DNS for DDoS**
- Send queries with spoofed source address: target IP
- Requires amplification
Other DNS Uses

• Use of DNS for geo-replicated content
  – Customized responses to queries
  – Inferring the user’s location
  – Knowing the user’s IP address

• Policy issues
  – Use of DNS to block access to Web sites
  – Collateral damage of DNS injection
  – Redirecting DNS for ads and profit (e.g., Paxfire)
Summary

• DNS maps human readable names to IP addresses

• DNS arranged into a hierarchy
  – Scalability / distributed responsibility
  – Autonomous control of local name servers

• Caching crucial for performance

• DNS has no authentication