Applications & Application-Layer Protocols: The Domain Name System

Jasleen Kaur

February 7, 2019

Application-Layer Protocols Outline

- The architecture of distributed systems
  - Client/Server computing
- Example client/server systems and their application-level protocols
  - The World-Wide Web (HTTP)
  - Reliable file transfer (FTP)
  - E-mail (SMTP & POP)
  - Internet Domain Name System (DNS)
- The programming model used in constructing distributed systems
  - Socket programming
Application-Layer Protocols

The Domain Name System (DNS)

- Computers (hosts, routers) connected to the Internet have two forms of names:
  - IP address — a 32-bit identifier used for addressing hosts and routing data to them
  - Hostname — an ASCII string used by applications

- The DNS is an Internet-wide service that provides mappings between IP addresses and hostnames
  - The DNS is a distributed database implemented in a hierarchy of name servers
  - The DNS is also an application-layer protocol

- Hosts and routers use name servers to resolve names (address/name translation)
  - Name resolution is an essential Internet function implemented as an application-layer protocol

The Domain Name System

Web browsing (HTTP) example

- The DNS is mainly used by applications, not end-users
  » And virtually all applications use the DNS for every request they generate

- Web browsing: User enters URL www.someSchool.edu
  » In order to create the socket to www.someSchool.edu, the OS (TCP) must resolve the hostname to an IP address
  » The OS contacts a DNS name server to learn the web server’s IP address
  » The IP address is then used by TCP to create the socket to the server
  » All this happens transparently to the user and the browser!
The Domain Name System
Name Hierarchy in DNS

- hostname = “dot” separated concatenation of domain names along path toward the root
  » unc.edu  » cs.unc.edu  » classroom.cs.unc.edu
- There are name servers associated with every domain

Name Hierarchy in the DNS
Top level domains

- Generic domains:
  » (1980) com, org, net, edu, gov, mil, int
  » (2000) biz, info, name, pro
- Special sponsored names
  » (2000) aero, coop, museum
  » (2003) asia, cat, jobs, mobi, tel, travel
- Country code domains
  » au, de, jp, us, … (250 more!)
Names Are Valuable
And prices are “more” rational

<table>
<thead>
<tr>
<th>Name</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>smoking.com</td>
<td>$500,000</td>
</tr>
<tr>
<td>beef.com</td>
<td>$250,000</td>
</tr>
<tr>
<td>sample.com</td>
<td>$90,000</td>
</tr>
<tr>
<td>upscale.com</td>
<td>$80,000</td>
</tr>
<tr>
<td>clerical.com</td>
<td>$75,000</td>
</tr>
<tr>
<td>snake.com</td>
<td>$50,000</td>
</tr>
<tr>
<td>barbecues.com</td>
<td>$30,000</td>
</tr>
<tr>
<td>geeky.com</td>
<td>$25,000</td>
</tr>
<tr>
<td>mime.com</td>
<td>SOLD!</td>
</tr>
<tr>
<td>dinner.com</td>
<td>$20,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>career.net</td>
<td>$150,000</td>
</tr>
<tr>
<td>invest.net</td>
<td>$75,000</td>
</tr>
<tr>
<td>dunk.net</td>
<td>$50,000</td>
</tr>
<tr>
<td>pornos.net</td>
<td>$35,000</td>
</tr>
<tr>
<td>wholesale.net</td>
<td>$30,000</td>
</tr>
<tr>
<td>exploring.net</td>
<td>$20,000</td>
</tr>
<tr>
<td>bonded.net</td>
<td>$15,000</td>
</tr>
<tr>
<td>worked.net</td>
<td>$10,000</td>
</tr>
<tr>
<td>wealthy.net</td>
<td>$8,000</td>
</tr>
<tr>
<td>russians.net</td>
<td>$7,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>mr.com</td>
<td>$350,000</td>
</tr>
<tr>
<td>teeth.net</td>
<td>$55,000</td>
</tr>
<tr>
<td>science.tv</td>
<td>$100,000</td>
</tr>
<tr>
<td>british.tv</td>
<td>$25,000</td>
</tr>
<tr>
<td>dancing.tv</td>
<td>$20,000</td>
</tr>
<tr>
<td>cafes.tv</td>
<td>$10,000</td>
</tr>
<tr>
<td>performer.tv</td>
<td>$10,000</td>
</tr>
<tr>
<td>bowling.tv</td>
<td>$10,000</td>
</tr>
<tr>
<td>merger.tv</td>
<td>$10,000</td>
</tr>
<tr>
<td>article.tv</td>
<td>$5,000</td>
</tr>
<tr>
<td>grandma.tv</td>
<td>$5,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>new.tv</td>
<td>$150,000</td>
</tr>
<tr>
<td>now.tv</td>
<td>$150,000</td>
</tr>
<tr>
<td>mr.com</td>
<td>$350,000</td>
</tr>
<tr>
<td>teeth.net</td>
<td>$55,000</td>
</tr>
<tr>
<td>science.tv</td>
<td>$100,000</td>
</tr>
<tr>
<td>british.tv</td>
<td>$25,000</td>
</tr>
<tr>
<td>dancing.tv</td>
<td>$20,000</td>
</tr>
<tr>
<td>cafes.tv</td>
<td>$10,000</td>
</tr>
<tr>
<td>performer.tv</td>
<td>$10,000</td>
</tr>
<tr>
<td>bowling.tv</td>
<td>$10,000</td>
</tr>
<tr>
<td>merger.tv</td>
<td>$10,000</td>
</tr>
<tr>
<td>article.tv</td>
<td>$5,000</td>
</tr>
</tbody>
</table>
New Names Are Valuable And prices are "more" rational

Growth of DNS Registrations
Internet Domain Survey Host Count

Source: Internet Systems Consortium (http://www.isc.org)
The Domain Name System

Why not centralize the DNS
  » A server process on a big, well connected supercomputer?

Centralized systems do not scale!
  » Poor reliability: centralized = single point of failure
  » Poor performance: centralized = “remote access” for most users
  » Difficult to manage: centralized = all customer traffic goes to one location, a large staff has to be present to handle registrations

A centralized system is not politically feasible in an international network

Designing a Distributed Service

DNS Name Servers

No server has every hostname-to-IP address mapping

Authoritative name server:
  » Every host is registered with at least one authoritative server that stores that host’s IP address and name
  » The authoritative name server can perform name/address translation for that host’s name/address

Local authoritative name servers:
  » Each ISP, university, company, has a local (default) name server authoritative for its own hosts
  » Resolvers always query a name server local to it to resolve any host name

What if the name is not a local host (e.g., www.yahoo.com)?
A root name server is contacted when a local name server that can’t resolve a name.
- The root server either resolves the name or provides pointers to authoritative servers at lower level of name hierarchy.
- In 1998, there were a dozen root name servers worldwide.

In 2011 there were a few more servers...
13 independent sites

- .com, .org, .net server locations (separated from root servers)

DNS Name Servers
Generic TLD servers (Verisign Corp.)

DNS Name Servers
Using a server hierarchy for resolving names

- Example: Host swift.cs.unc.edu wants to know the IP address of www.yahoo.com
  - Swift contacts its local DNS server bristol.cs.unc.edu
  - Swift contacts the authoritative server dns.yahoo.com (if necessary)
  - Results propagate back to swift

To resolve a non-local name the local name server queries the root server (if necessary)

The root server contacts the authoritative server dns.yahoo.com (if necessary)

Results propagate back to swift
DNS Name Servers
Using a server hierarchy for resolving names

- It’s possible that the root name server may not know the authoritative name server for a domain
- The root server contacts an *intermediate* name server that knows the authoritative name server
- The intermediate name server contacts the authoritative name server
- Results propagate back to the requesting host

The DNS supports two forms of queries:
- Recursive queries
- Iterative queries

Recursive queries place the burden of name resolution (recursively) on the contacted server
In an iterated query the contacted server simply replies with the name of the server to contact
- “I don’t know; trying asking X”
*Swift* wants to know the IP address of **www.yahoo.com**

» *Swift* contacts its local DNS server *bristol.cs.unc.edu*

*If necessary, the local name server queries the root server*

» “What server is the authority for **www.yahoo.com**?”

*The root server returns the name and IP address of the server it knows is the closest match to the query*

» “Try **dns.yahoo.com**”

*The local DNS server sends the same query to the closest match server*

» “What server is the authority for **www.yahoo.com**?”

*The process can be iterated until the local authoritative name server is found and responds*

*(And iterated and recursive queries can be combined!)*
DNS Name Servers
Caching and updating DNS entries

- Every server caches all the mappings it learns
  - TLD servers typically cached in local name servers
  - (Thus root name servers not often queried)
- Cache entries are “soft state”
  - They timeout (are deleted) after some time period
  - Called the “time to live” (“TTL”)
- So cached entries can be out of date!
- DNS cache update/notify mechanisms under design by the IETF
  - See RFC 2136

DNS Name Servers
DNS resource records

- The DNS is a distributed database storing resource records (RRs)
- Type = A
  - name is a hostname
  - value is hostname’s IP address
- Type = NS
  - name is a domain
  - value is name of authoritative name server for this domain
- Type = CNAME
  - name is an alias name for some “canonical” (the real) name
  - value is canonical name
- Type = MX
  - name is name of mail server host associated with name

RR format: <name, value, type, time_to_live>
The Domain Name System
Inserting records into the DNS

◆ Example: New startup “Network Utopia”
◆ Register name networkuptopia.com at DNS registrar (e.g., Network Solutions)
  » You provide names & IP addresses of authoritative name server (primary and secondary)
  » The registrar inserts two RRs into .com TLD server:
    ◆ networkuptopia.com, dns1.networkuptopia.com, NS
    ◆ dns1.networkuptopia.com, 212.212.212.1, A
◆ You stand up dns1.networkuptopia.com running BIND and create:
  » Authoritative server type A record for www.networkuptopia.com
  » MX record for networkuptopia.com

The Domain Name System
The DNS protocol

◆ The DNS service is implemented by the DNS protocol
◆ A request/response protocol run on top of UDP
  » Uses port 53
◆ Why UDP?!
  » Doesn’t reliability matter?!
**DNS Protocol**

**DNS query and reply messages**

- DNS query and reply messages both have the same message format.
- Messages have a fixed length message header:
  - Identification — 16 bit query/reply identifier used to match replies to queries.
  - Flags:
    - Query/Reply bit
    - "Reply is authoritative" bit
    - "Recursion desired" bit
    - ....

- DNS query and reply messages both have the same message format.
- Messages have a fixed length message header:
  - Identification — 16 bit query/reply identifier used to match replies to queries.
  - Flags:
    - Query/Reply bit
    - "Reply is authoritative" bit
    - "Recursion desired" bit
    - ....

- Messages have a variable-length “question & answer” body.
- Questions:
  - The name and type fields (type A or MX) for a query — hotmail.com MX.
- Answers:
  - One RR for each IP address answering query.
- Authority:
  - Resource records of other authoritative servers.

---

<table>
<thead>
<tr>
<th>Identification</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of questions</td>
<td>Number of answer RRs</td>
</tr>
<tr>
<td>Number of authority RRs</td>
<td>Number of additional RRs</td>
</tr>
<tr>
<td>Questions (variable number)</td>
<td></td>
</tr>
<tr>
<td>Answers (variable number of records)</td>
<td></td>
</tr>
<tr>
<td>Authority (variable number of records)</td>
<td></td>
</tr>
<tr>
<td>Additional Information (variable number of records)</td>
<td></td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Identification</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of questions</td>
<td>Number of answer RRs</td>
</tr>
<tr>
<td>Number of authority RRs</td>
<td>Number of additional RRs</td>
</tr>
<tr>
<td>Questions (variable number)</td>
<td></td>
</tr>
<tr>
<td>Answers (variable number of records)</td>
<td></td>
</tr>
<tr>
<td>Authority (variable number of records)</td>
<td></td>
</tr>
<tr>
<td>Additional Information (variable number of records)</td>
<td></td>
</tr>
</tbody>
</table>
nslookup query/reply message example

Server:   bristol.cs.unc.edu
Address:  152.2.131.228

QUESTIONS:
  - www.yahoo.com, type = A, class = IN

ANSWERS:
- www.yahoo.com
canonical name = www.yahoo-ht3.akadns.net
  internet address = 69.147.114.210

AUTHORITY RECORDS:
- akadns.net
  nameserver = zc.akadns.org.
- akadns.net
  nameserver = zd.akadns.org.
- akadns.net
  nameserver = eurl.akadns.net.
- akadns.net
  nameserver = use3.akadns.net.
- akadns.net
  nameserver = use4.akadns.net.
- akadns.net
  nameserver = usw2.akadns.net.
- akadns.net
  nameserver = asia9.akadns.net.
- akadns.net
  nameserver = za.akadns.org.
- akadns.net
  nameserver = zb.akadns.org.
DNS Resource Records
nslookup query/reply message example

ADDITIONAL RECORDS:
- za.akadns.org
  internet address = 195.219.3.169
- zb.akadns.org
  internet address = 206.132.100.105
- zc.akadns.org
  internet address = 124.211.40.4
- zd.akadns.org
  internet address = 63.209.3.132
- eur1.akadns.net
  internet address = 213.254.204.197
- use3.akadns.net
  internet address = 204.2.178.133
- use4.akadns.net
  internet address = 208.44.108.137
- usw2.akadns.net
  internet address = 63.209.3.132
- asia9.akadns.net
  internet address = 220.73.220.4

Non-authoritative answer:
- Address: 69.147.114.210

DNS Example
DNS processing for an iterated query

◆ Resolve the hostname in http://www.cnn.com

- com.
  NS    A.GTLD-SERVERS.NET   (A) 198.41.0.1
- cnn.com
  NS    DNS.CNN.COM          (A) 128.2.232.1
- www.cnn.com
  A     207.25.71.28
**DNS Example**

DNS processing for an iterated query

- Resolve the hostname **www2.cnn.com** with a warm cache

<table>
<thead>
<tr>
<th>Domain</th>
<th>Type</th>
<th>TTL</th>
<th>IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>com</td>
<td>NS</td>
<td></td>
<td>A.GTLD-SERVERS.NET (A) 198.41.0.1</td>
</tr>
<tr>
<td>cnn.com</td>
<td>NS</td>
<td></td>
<td>DNS.CNN.COM (A) 128.2.232.1</td>
</tr>
<tr>
<td><a href="http://www.cnn.com">www.cnn.com</a></td>
<td>A</td>
<td></td>
<td>207.25.71.28</td>
</tr>
</tbody>
</table>

**The Domain Name System**

**Attacking the DNS**

- **DDoS attacks**: Bombard root servers with requests
  - Not successful to date(!)
  - Defeated by traffic filtering
  - Local DNS servers cache IPs of TLD servers, allowing root server bypass
  - Bombard TLD servers — Potentially more dangerous

- **Redirect attacks**
  - “Man-in-middle” (Intercept queries)
  - DNS poisoning: Send bogus replies to a DNS server, which will cache them & return to others

- **Exploit DNS for DDoS**
  - Send queries with spoofed source address!
  - (Requires amplification)
The Domain Name System
Summary

- F gets 270,000,000+ hits per day
  - Other servers have comparable load
- The Verisign TLD servers answer 5,000,000,000 queries per day
- Clearly the DNS would collapse without:
  - Hierarchy
  - Distributed processing
  - Caching

- If DNS fails, Internet services stop working!