The Domain Name System (DNS)

Brad Smith
brad@soe.ucsc.edu
CE150, 3/10/09
Domain Names

- Each node in the DNS hierarchy represents a DNS **domain name**
  - cruzmail.ucsc.edu
  - services.soe.ucsc.edu
- Each subtree of the DNS hierarchy is a DNS **domain**.
  - ucsc.edu
  - soe.ucsc.edu
- A DNS domain can contain hosts or other domains (subdomains)
  - E.g. “neon.cs.virginia.edu.”
Resource Records

• The DNS contains one or more resource records for each domain name.
• Each RR is a five-tuple.
  – Name.
  – TTL (in seconds)... vs. $TTL and SOA “MINIMUM” field
  – Class (usually “IN” for Internet info).
  – Type: type of RR.
  – Value.

```plaintext
$TTL 86400
mylab.com. IN SOA PC4.mylab.com. hostmaster@mylab.com. (1 ; serial 28800 ; refresh 7200 ; retry 604800 ; expire 86400 ; ttl )
;
;
localhost A 127.0.0.1
PC4.mylab.com. A 10.0.1.41
PC3.mylab.com. A 10.0.1.31
PC2.mylab.com. A 10.0.1.21
PC1.mylab.com. A 10.0.1.11
```
dig Unix utility

- Command line DNS query utility
- `dig [@server] [name] [type]`

```bash
dig services.cse.ucsc.edu
```
DNS Protocol

- DNS has two types of messages: query and response.
- Both have the same format
  - A query includes a header and a question section
  - A response includes
    - Header
    - Sections: question, answer, authoritative, and additional
- The `dig` utility returns this information.
DNS Protocol - Header Format

- Twelve bytes.

<table>
<thead>
<tr>
<th>Identification</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of question records</td>
<td>Number of answer records</td>
</tr>
<tr>
<td></td>
<td>(all 0s in query message)</td>
</tr>
<tr>
<td>Number of authoritative records (all 0s in query</td>
<td>Number of additional records</td>
</tr>
<tr>
<td>message)</td>
<td>(all 0s in query message)</td>
</tr>
</tbody>
</table>
Resource Record Types

- **A (or AAAA)**: address.
  - Contains 32-bit IP address.
  - Single name <-> several A RRs.
- **CNAME**: canonical name.
  - Alias.
- **HINFO**: host description.
  - Provides information about host, e.g., CPU type, OS, etc.
- **MX**: mail exchange.
  - Name of mail server for this domain.
- **PTR**: Reverse DNS lookups (IP address to domain name mapping)
- **TXT**: arbitrary string of characters.
  - Generic description of the domain, where it is located, etc.
- Following apply only to domains
  - **SOA**: start of authority.
    - Marks beginning of zone’s database.
    - Provides general info about the zone: e-mail address of admin, default TTL, etc.
  - **NS**: name server.
    - Name of name server for this domain.
dig Unix utility

dig www.soe.ucsc.edu

dig ucsc.edu soa

- 5 part value
  - Source host
  - Contact e-mail
  - Serial number
  - Refresh time – time (secs) for secondary to try zone xfer
  - Retry time – time (secs) for secondary to retry zone xfer
  - Expire time – time (secs) secondary will expire zone
  - Minimum TTL – min TTL for zone records (3600 default)

dig ucsc.edu ns
PTR Records and the *in-addr.arpa* Domain

- Used for **inverse** lookups:
  - IP address $\rightarrow$ domain name.
- **PTR** records
- Reversed IP address concatenated with “in-addr.arpa”
  
  \[
  10.48.114.128.in-addr.arpa
  \]
- Allows for meaningful delegation of domains.
- `dig -x ...`
**dig** *Unix utility*

- “dig –x ...” reverse (PTR) lookup

  ```
dig -x 128.114.48.10
  ```
**Addresses of root servers (2004)**

<table>
<thead>
<tr>
<th>Address</th>
<th>Location</th>
<th>IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.ROOT-SERVERS.NET.</td>
<td>(VeriSign, Dulles, VA)</td>
<td>198.41.0.4</td>
</tr>
<tr>
<td>B.ROOT-SERVERS.NET.</td>
<td>(ISI, Marina Del Rey CA)</td>
<td>192.228.79.201</td>
</tr>
<tr>
<td>C.ROOT-SERVERS.NET.</td>
<td>(Cogent Communications)</td>
<td>192.33.4.12</td>
</tr>
<tr>
<td>D.ROOT-SERVERS.NET.</td>
<td>(University of Maryland)</td>
<td>128.8.10.90</td>
</tr>
<tr>
<td>E.ROOT-SERVERS.NET.</td>
<td>(Nasa Ames Research Center)</td>
<td>192.203.230.10</td>
</tr>
<tr>
<td>F.ROOT-SERVERS.NET.</td>
<td>(Internet Systems Consortium)</td>
<td>192.5.5.241</td>
</tr>
<tr>
<td>G.ROOT-SERVERS.NET.</td>
<td>(US Department of Defense)</td>
<td>192.112.36.4</td>
</tr>
<tr>
<td>H.ROOT-SERVERS.NET.</td>
<td>(US Army Research Lab)</td>
<td>128.63.2.53</td>
</tr>
<tr>
<td>I.ROOT-SERVERS.NET.</td>
<td>(Autonomica/NORDUnet)</td>
<td>192.36.148.17</td>
</tr>
<tr>
<td>J.ROOT-SERVERS.NET.</td>
<td>(Verisign, multiple cities)</td>
<td>192.58.128.30</td>
</tr>
<tr>
<td>K.ROOT-SERVERS.NET.</td>
<td>(RIPE, Europe multiple cities)</td>
<td>193.0.14.129</td>
</tr>
<tr>
<td>L.ROOT-SERVERS.NET.</td>
<td>(IANA, Los Angeles)</td>
<td>198.32.64.12</td>
</tr>
<tr>
<td>M.ROOT-SERVERS.NET.</td>
<td>(WIDE, Tokyo, Seoul, Paris)</td>
<td>202.12.27.33</td>
</tr>
</tbody>
</table>
dig **Unix utility**

- How get list of root servers?

  ```
  dig . ns
  ```
Query Types

- There are two types of queries:
  - Recursive queries
  - Iterative queries

- The type of query is determined by a bit in the DNS message

- **Recursive query:** when a server can’t resolve a query
  - Issues queries (typically iterative) to find resource records
  - Returns requested resource records (or error)
  - Typically issued by local DNS server

- **Iterative queries:** when a server can’t resolve a query
  - Returns a referral to another, better (“closest known”) server
  - Typically issued by DNS client
Recursive Query

1st query: neon.cs.virginia.edu

Referral to edu name server

2nd query: neon.cs.virginia.edu

Referral to virginia.edu name server

3rd query: neon.cs.virginia.edu

Referral to cs.virginia.edu name server

4th query: neon.cs.virginia.edu

IP address of neon.cs.virginia.edu

root server

edu server

virginia.edu server

cs.virginia.edu server

Resolver

Name server
Iterative Query

“dig +trace neon.cs.virginia.edu”
Interesting use of DNS...

- How could you use DNS to
  - Provide web server load balancing
  - Improve web performance
dig **Unix utility**

- “Akamaized” URLs...
  
  ```
  dig x.myspace.com
  dig wwwimages.adobe.com
  dig cache.vzw.com
  ```
Akamai Example

• Content Delivery Network (CDN)
  – Charge for delivering content
• How?
  – Customer modifies URL for Akamai served content
  – Special “host name” gets resolved to closest Akamai cache
  – Two levels
    • Region in Internet
    • Cache within region
• Example
  – Pick Akamai customer
  – Look at web site source
  – “dig” the host for a .gif URL
• Value provided by Akamai: monitoring Internet performance for mapping
Advertisements

• CE151
• Network Management and Operations (NMO) Lab
  – Contact

  Tammy Tooley-Chelossi
  ttchelossi@soe.ucsc.edu
Thank you!

Brad Smith
brad@soe.ucsc.edu