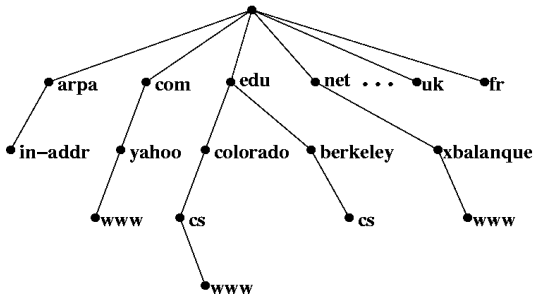
Lecture 18
DNS



Administration

- PA7 due tomorrow
- FINAL PROJECT writeups due Wednesday
- Midterm #2 a week from today (March 29)
  - You may bring one *double-sided* 8.5x11 set of handwritten notes
  - Content is anything in lecture since Feb 8
  - We'll have review in the 2nd part of lecture on Wednesday; bring questions!

2

The Web at Scale

- In the first third of this class we focused on point-to-point basics of the Web
  - HTML/HTTP basics
  - Sessions & server architecture
  - Crypto and XML
- Middle third: managing lots of Web activity, with emphasis on *semantics*
  - Web search and page management
  - Recommenders, auctions, data mining
- Next, building Web-scale *systems*

3

A Mini Syllabus

- Web is only interesting at massive scale
  - What's data mining without the data?
- Upcoming topics:
  - DNS (today)
  - Scaling and Distribution
  - Caching & Proxies
  - Google File System, MapReduce
  - Cloud Computing and the Datacenter

4

Locating Things Online

- **Addresses** describe a location
  - A map is OK
  - IP addresses, street addresses
- **Names** are mapped to addresses
  - Domain names; PO Boxes; email addresses
- **Content-based naming** uses the actual content to find the destination
  - Basis of publish/subscribe and peer-to-peer
- What about:
  - Phone #s?
  - Search terms
  - Twitter messages?
    - @username?
    - #topic?

5

Domain Name System

- The Internet's routers know only IP addresses
  - Fine for routers, not great for humans
- DNS translates domain names to IP addresses
  - `google.com => 74.125.95.99`
  - (or something similar)
  - The "phone book" of the internet
- When DNS goes down, the Web suffers a grievous, not quite deadly, blow

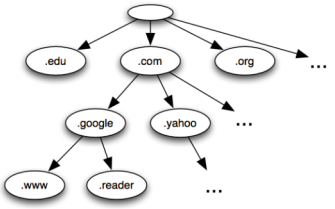
6

DNS Design

- DNS is a globally-accessible database of name/IP pairs. Some requirements:
  - Needs to be up all the time
  - Continuously updated by many parties
  - Must be accurate; errors prevent connections
  - Serves massive query load
  - Needs distributed administration
  - Source of political & commercial disputes
  - Potential terrorism target
- Overall: utterly shocking that it works

7

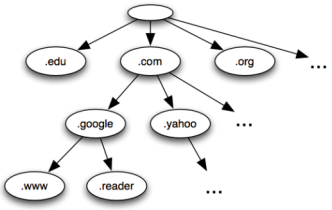
The Namespace



- Tree structure, 1-63 chars per node
- *Fully-qualified domain name* is leaf-to-root name for path
- Only *DNS root* has no parent

8

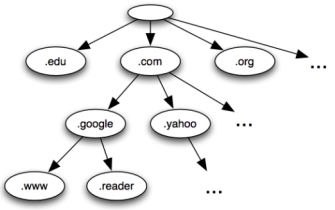
The Namespace



- Nodes grouped into administrative *zones*
  - E.g., root, com, google.com
- Each *zone* served by *authority servers (aka authoritative name servers)*
- AS can delegate subdomains to other ASes

9

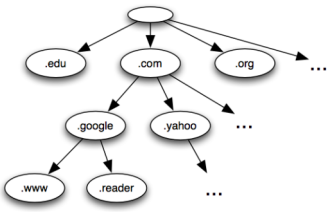
The Namespace



- Servers are primary or secondary
  - Primary ASes are given content by admins
  - Secondary ASes grab from primaries
- A domain registrar inserts your name into the primary AS for .com (or .net, .org, etc)

10

The Namespace



- Purchased domain names inserted into 1 primary, 1 secondary (in case of primary failure)

11

The Data

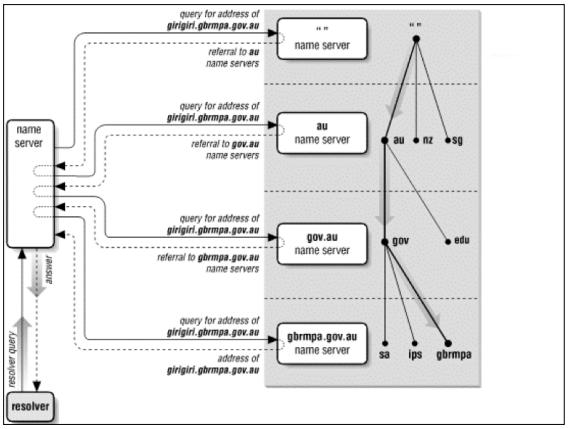
- Content of DNS consists of *resource records*
  - Host/IP and IP/Host are most popular type
  - Also: zone info, email servers, other info
- Most DNS activity is this:
  - DNS client sends request to AS; receives resource record in response
- DNS clients built into network libraries
  - Clients use UDP (not TCP) to grab data
  - If your connection is taking a long time to establish, possibly waiting for DNS

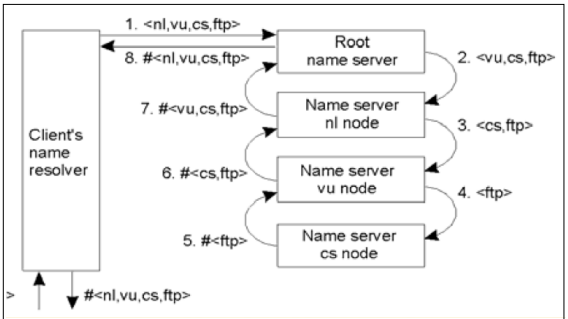
12

Resolution

- Single AS may not be enough for a FQDN

13





15

Caching

- There may be a chain of caching DNS servers between client and AS
  - There's one for CSE
  - Caches DNS requests; answers new requests from cache whenever possible
- Great! But when data changes?
  - Each RR has time-to-live (TTL) in seconds
  - Counts down from moment AS emits RR
  - Caches and clients must throw out RRs with expired TTLs

16

Root Nameservers

- Responsible for locating the TLD name servers (.com, .net, .org, …)
- Thirteen root name servers in world
  - Their locations are hard-coded in resolving DNS servers
  - Due to caching, involved in few queries

17

Administration!

- Until 1999, all US TLDs run by IANA
  - (At the time, IANA = Jon Postel)
- Now, non-profit ICANN administers set of for-profit registrars (under contract with gov't)
- For a time, looked like ICANN might be xfer'ed to United Nations. Probably won't be

18

Baffling Corner Cases

- DNS is not very clean; lots of weird holes you don't see in, say, TCP
  - ASes use RRs to describe zone children and parents
    - Parent & child often disagree about relationship, forcing client to do something… reasonable
  - Letter case is supposed to be preserved, but cache implementations mess this up
  - How many requests can a single client message make? In principle, many! In reality, one
- Yet totally world-beating

19

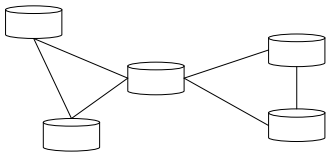
Akamai

- Stores images, videos, at many locations throughout the world
  - Large read-only data stored close to client
  - Reduce latency to faraway datacenter
  - Reduce bandwidth costs by sending data only a short distance
- The 1st CDN (content-delivery network)
  - You might think the image is from Yahoo, but Akamai is serving it
  - Yahoo pays Akamai to do so
- Built on top of giant DNS hack!

20

How it works: step 1

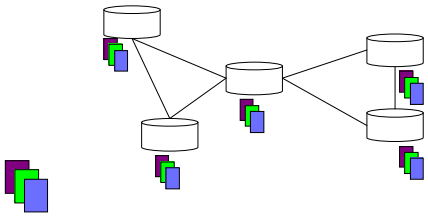
- Akamai places servers across country



21

How it works: step 2

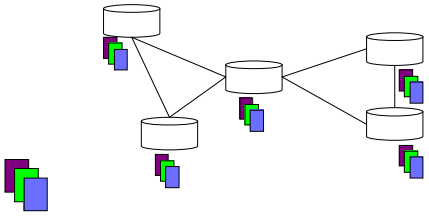
- Akamai gets client data (videos, imgs), copies to all nodes in network



22

How it works: step 3

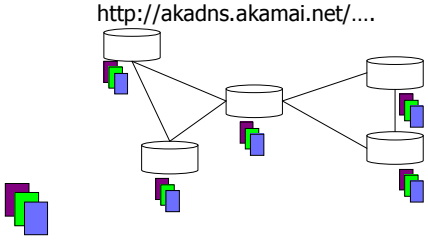
- How does client find nearby data?
  - Network conditions can change



23

How it works: step 3

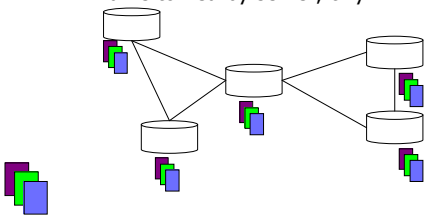
- Client rewrites URLs, uses new URLs
  - http://yahoo.com/… => http://akadns.akamai.net/….



24

How it works: step 3

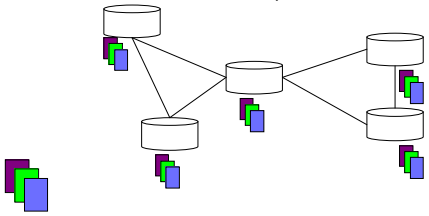
- URL contains Akamai-controlled name
  - Specialized Akamai DNS server resolves name to nearby server; tiny TTL



25

How it works: step 3

- Resolution strongly dependent on
  - Location of client; network conditions; load on Akamai servers; traffic estimation errors



26

Akamai Recap

- Uses custom DNS servers to dynamically direct clients to right caching servers
- How would you implement Akamai without DNS?

27

DNS Challenges

- Shocking scale
  - Every device, service, etc
  - Spam-filtering email requires 10+ DNS lookups per message
- Extreme security vulnerability
  - What if someone can remap `google.com`?
  - What about just denial-of-service?

28

DNS Cache Poisoning

- Bad info is inserted into a DNS server and cached
  - Both inadvertent and malicious
  - How could one attack CSE's DNS server?
- Remedies
  - DNSSEC requires that DNS entries be cryptographically signed
  - If you use HTTPS/SSL/TSL, problem mitigated

29