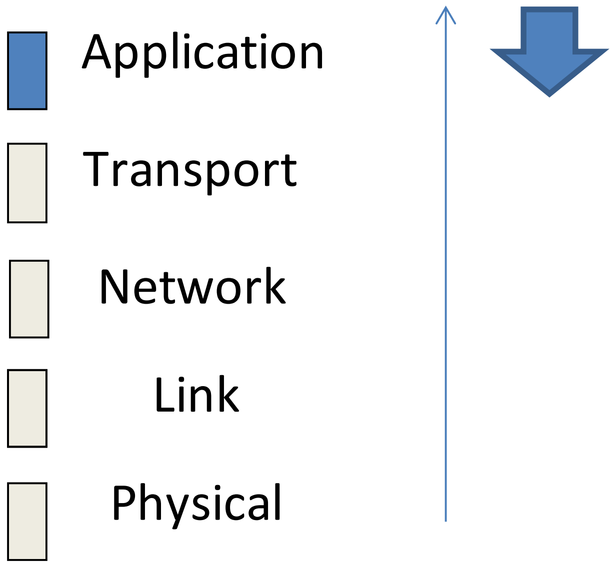
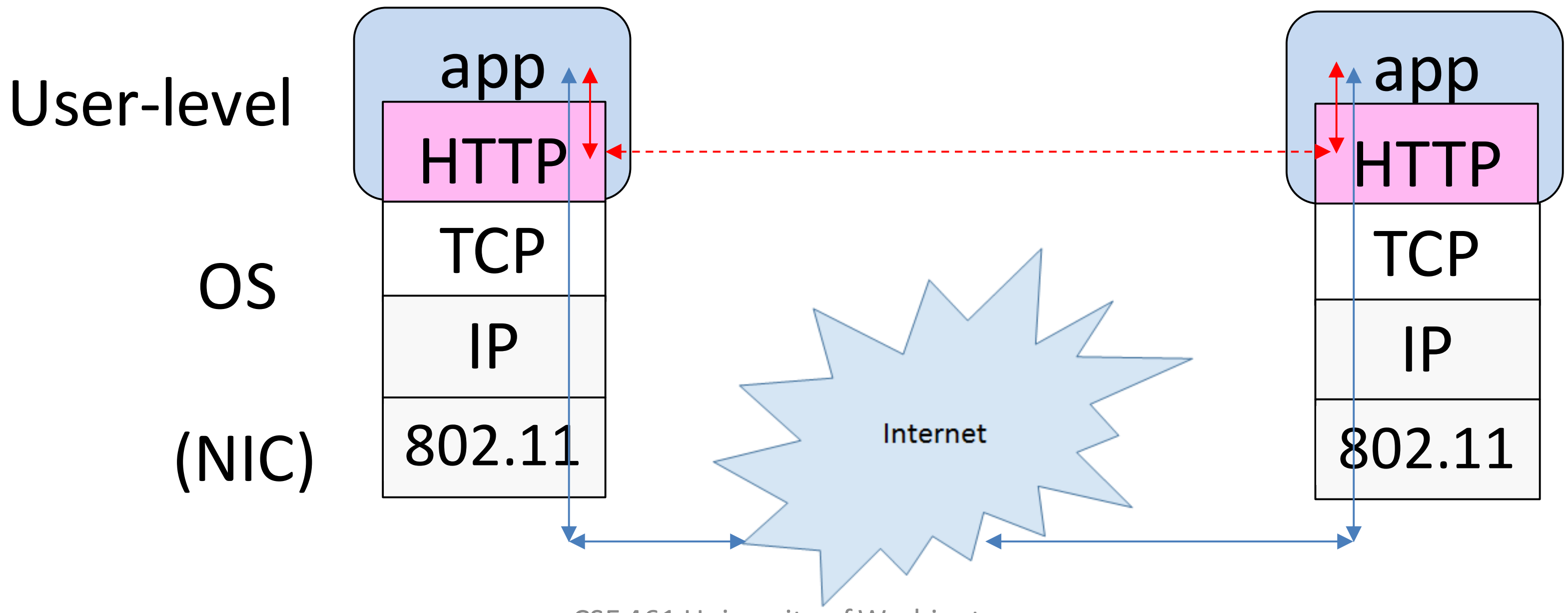
# 2- Application Level Protocols

# Where we are in the Course



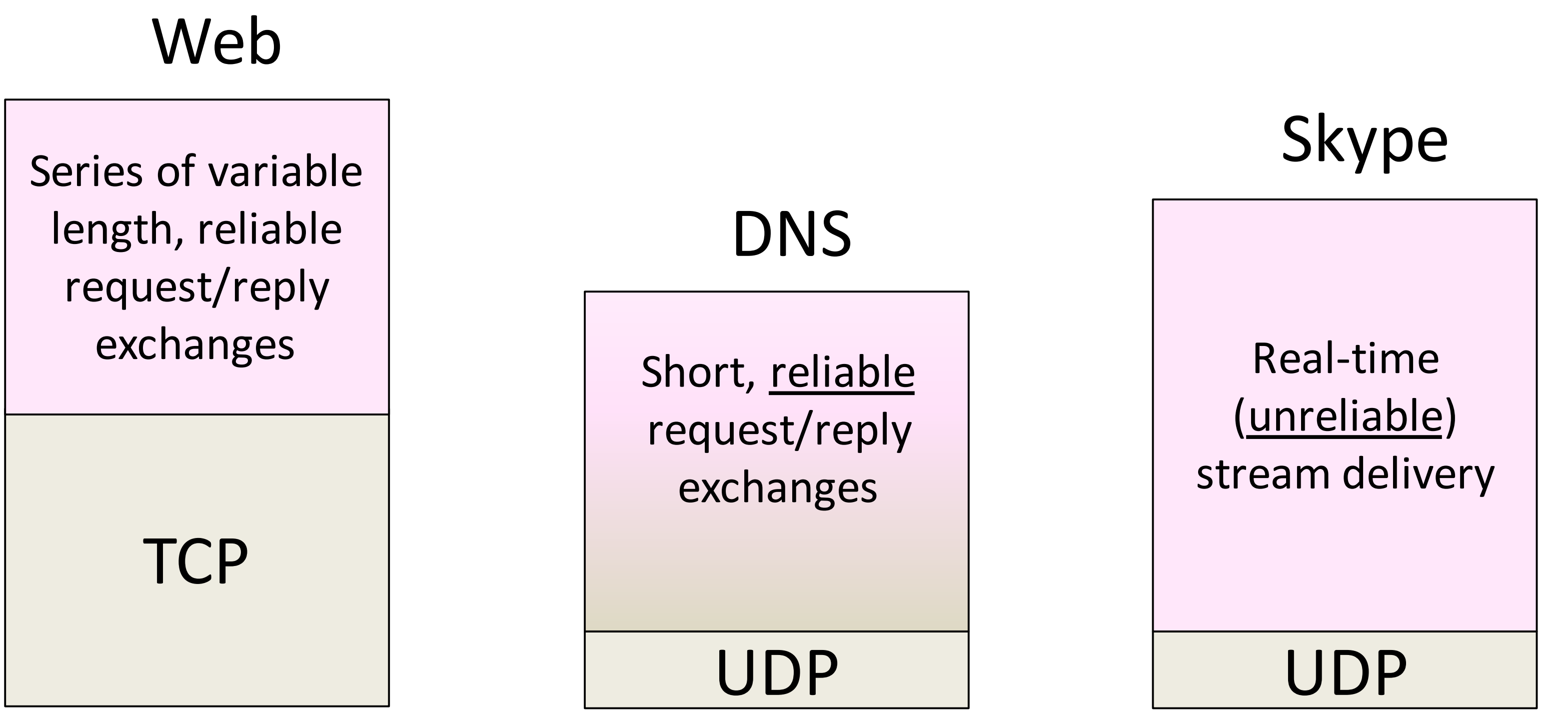
# Implementation

- Application layer protocols are often part of "the app"
  - Libraries running in untrusted space



# Application Communication Needs

- Vary widely; must build on Transport services



# OSI Session/Presentation Layers

- Two relevant concepts …

Considered part of the application, not strictly layered! (layers 6 and 5)

| | | |
|---|---|---|
| 7 | Application | – Provides functions needed by users |
| 6 | Presentation | – Converts different representations |
| 5 | Session | – Manages task dialogs |
| 4 | Transport | – Provides end-to-end delivery |
| 3 | Network | – Sends packets over multiple links |
| 2 | Data link | – Sends frames of information |
| 1 | Physical | – Sends bits as signals |

# Session Concept

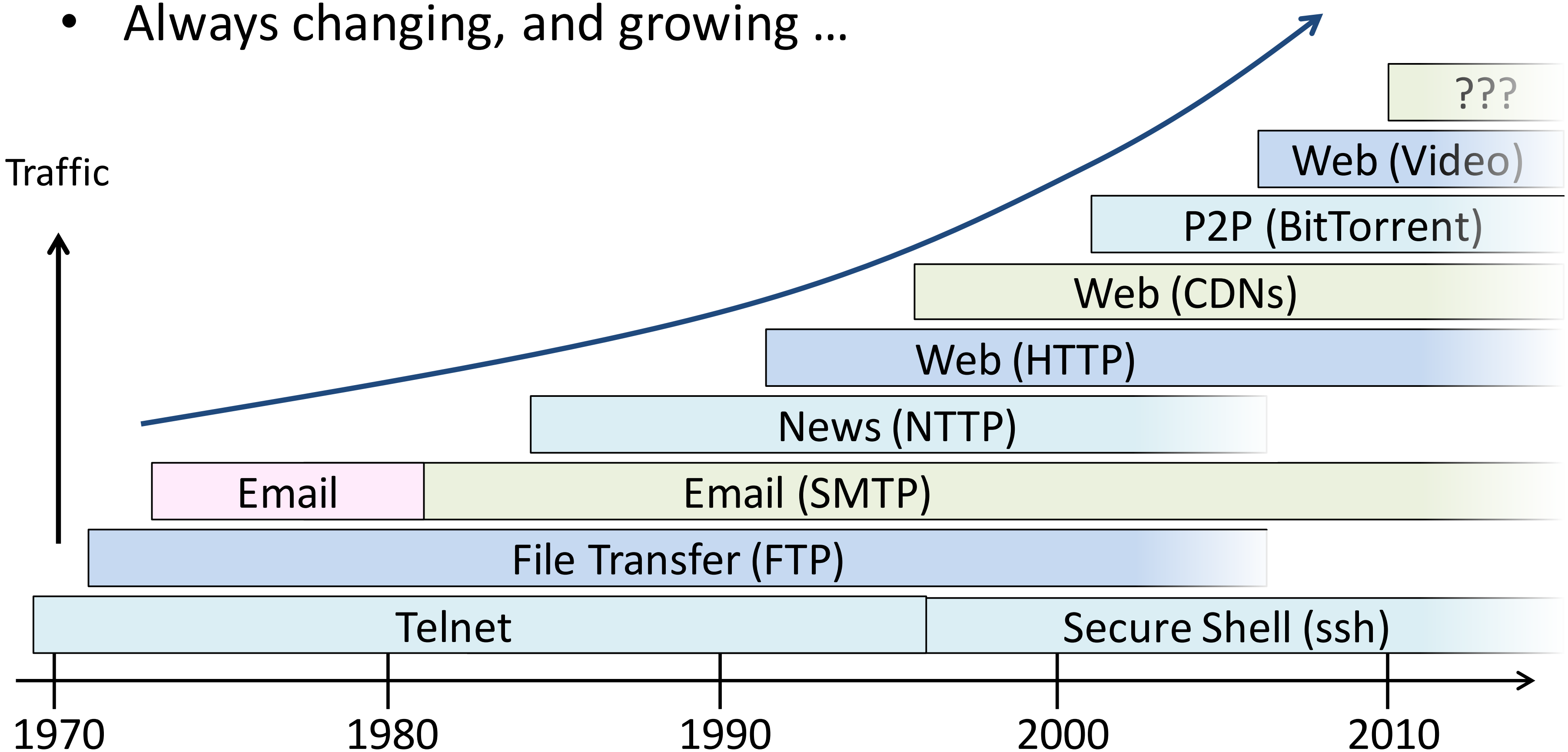
- A session is a series of related network interactions in support of an application task
  - Often informal, not explicit
  - Often related to an individual user
- Examples:
  - Web page fetches multiple resources
  - Skype call involves audio, video, chat

# Presentation Concept

- Apps need to identify the *type of content*, and *encode it for transfer*
  - These are Presentation functions
- Examples:
  - Media (MIME) types, e.g., image/jpeg, identify content type
  - Transfer encodings, e.g., gzip, identify the encoding of content
  - Application headers are simple/readable versus packed for efficiency

# Evolution of Internet Applications

- Always changing, and growing …

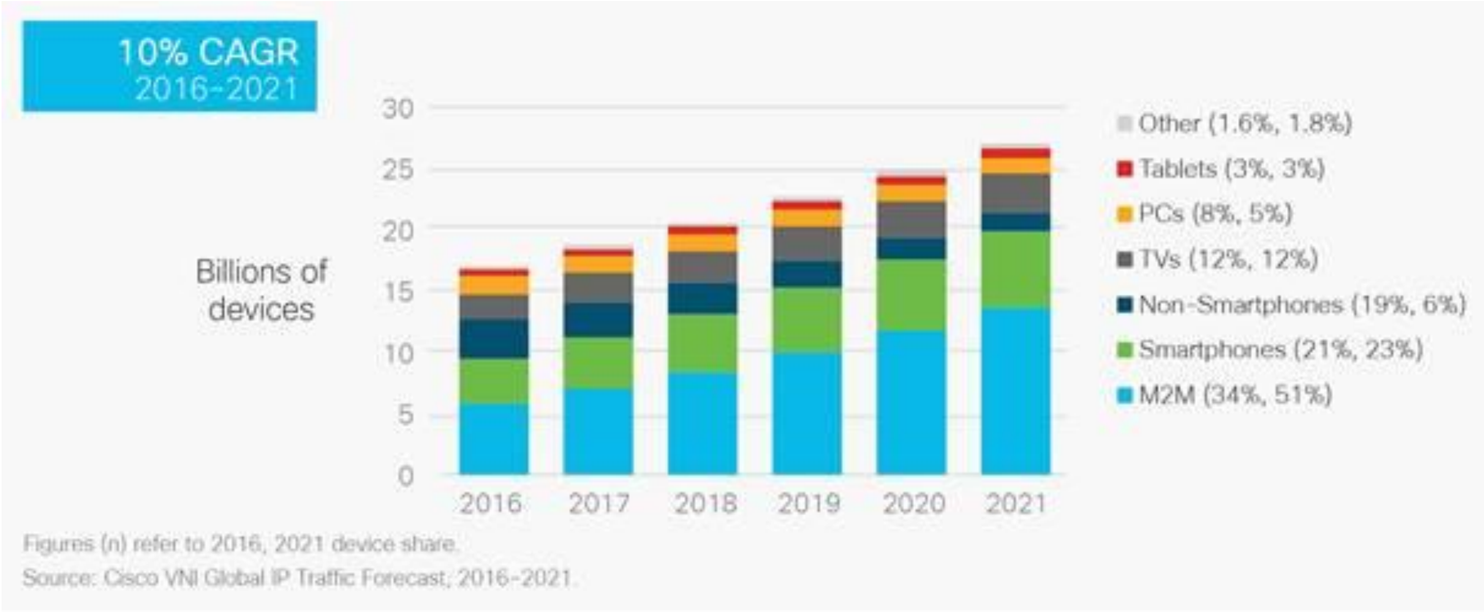


# n Apps Matter?

| Year | Global Internet Traffic |
|---|---|
| 1992 | 100 GB per day |
| 1997 | 100 GB per hour |
| 2002 | 100 GB per second |
| 2007 | 2,000 GB per second |
| 2016 | 26,6000 GB per second |
| 2021 | 105,800 GB per second |

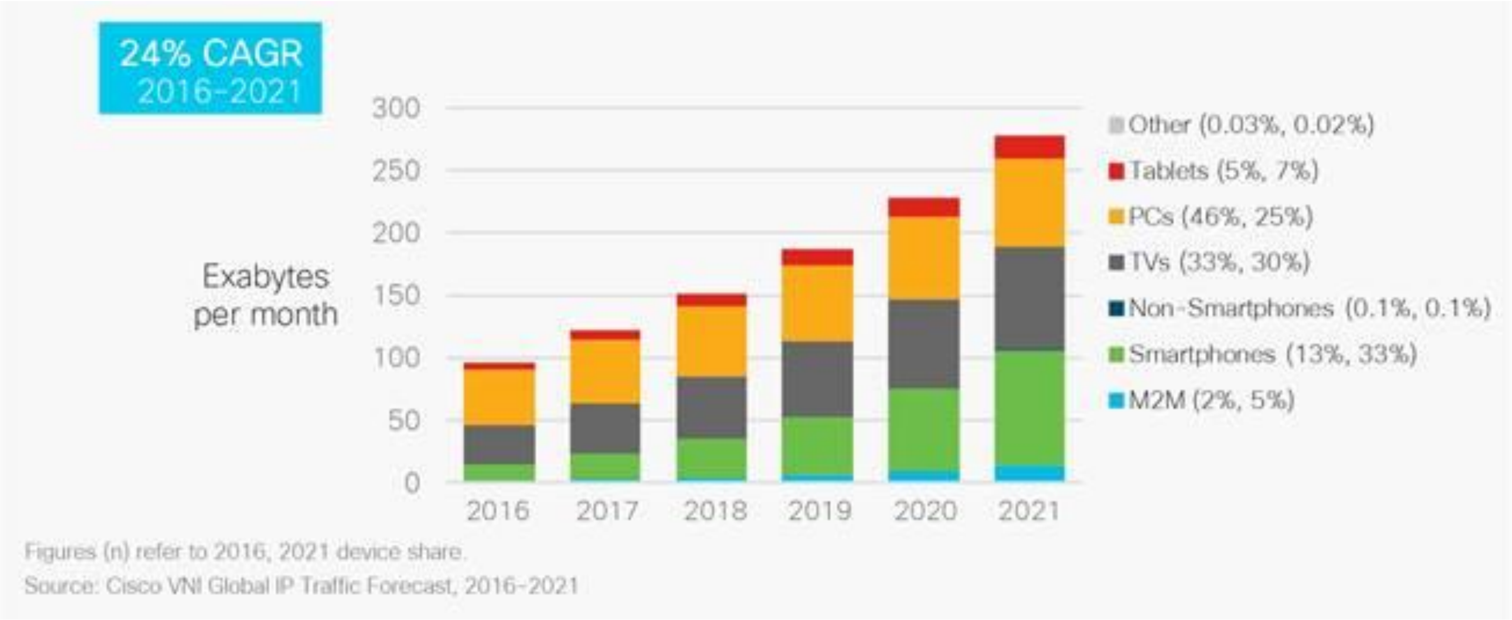
Much of the content here is from
https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-hyperconnectivity-wp.html

# Which Devices Matter?



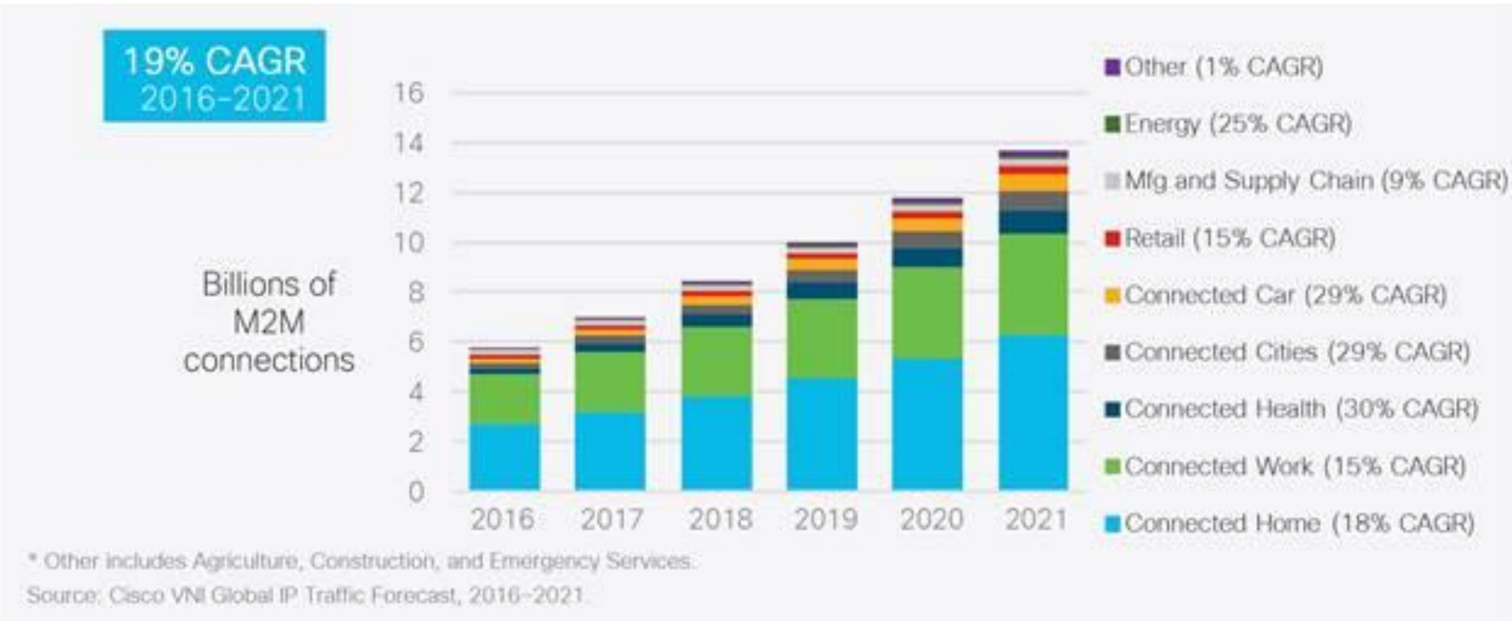

*Graph of connected devices*

# Which Devices Matter?




*Graph of traffic generated by devices*

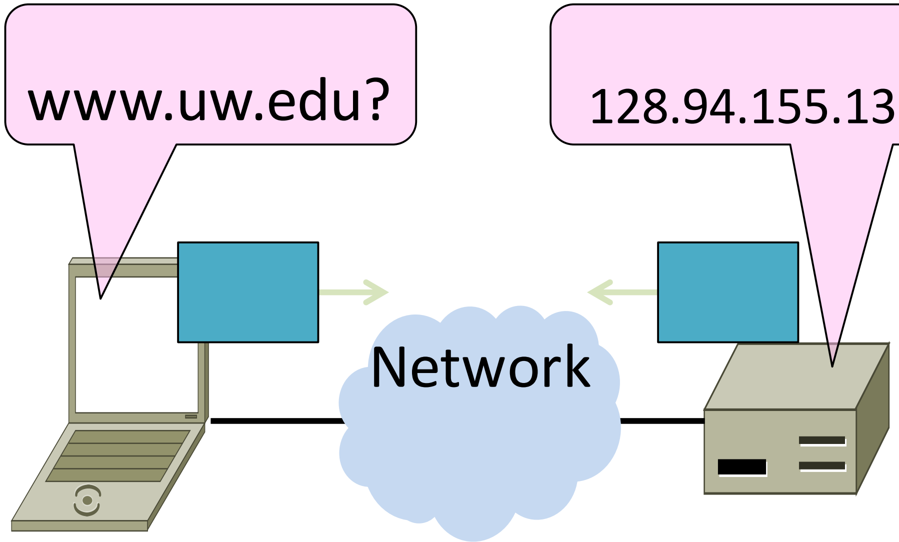
# M2M Connections By Industry



# DOMAIN NAME SYSTEM

# DNS

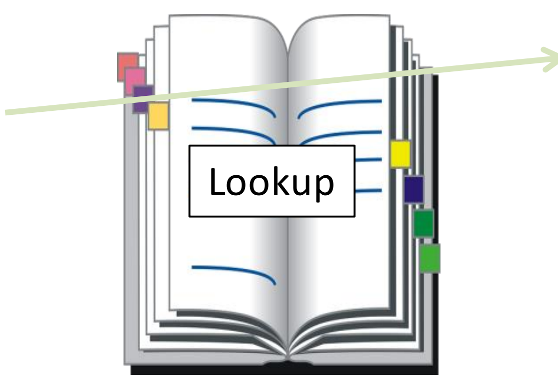
- Human-readable host names, and more



# Names and Addresses

- <u>Names</u> are higher-level identifiers for resources
- <u>Addresses</u> are lower-level locators for resources
  - Multiple levels, e.g. full name → email → IP address → Ethernet addr
- <u>Resolution</u> (or lookup) is mapping a name to an address

Name, e.g.
"Andy Tanenbaum,"
or "flits.cs.vu.nl"




Directory

Address, e.g.
"Vrijie Universiteit, Amsterdam"
or IPv4 "130.30.27.38"

# Before the DNS – HOSTS.TXT

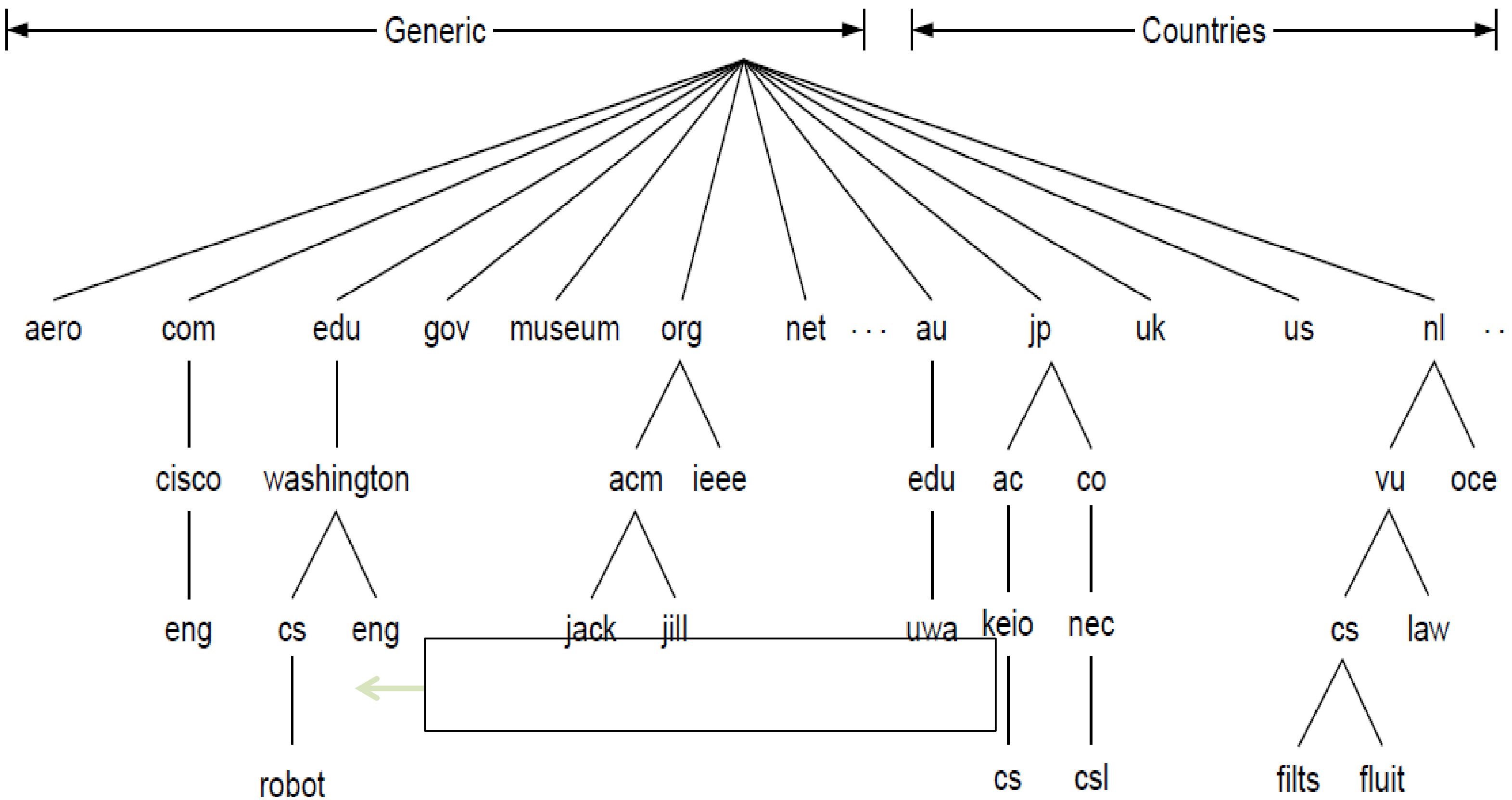
- Directory was a file HOSTS.TXT regularly retrieved for all hosts from a central machine at the NIC (Network Information Center)
- Names were initially flat, became hierarchical (e.g., lcs.mit.edu) ~1985
- Not manageable or efficient as the ARPANET grew …

# DNS

- A naming service to map between host names and their IP addresses (and more)
  - www.uwa.edu.au → 130.95.128.140
- Goals:
  - Easy to manage (esp. with multiple parties)
  - Efficient (good performance, few resources)
- Approach:
  - Distributed directory based on a hierarchical namespace
  - Automated protocol to tie pieces together

# DNS Namespace

- Hierarchical, starting from "." (dot, typically omitted)

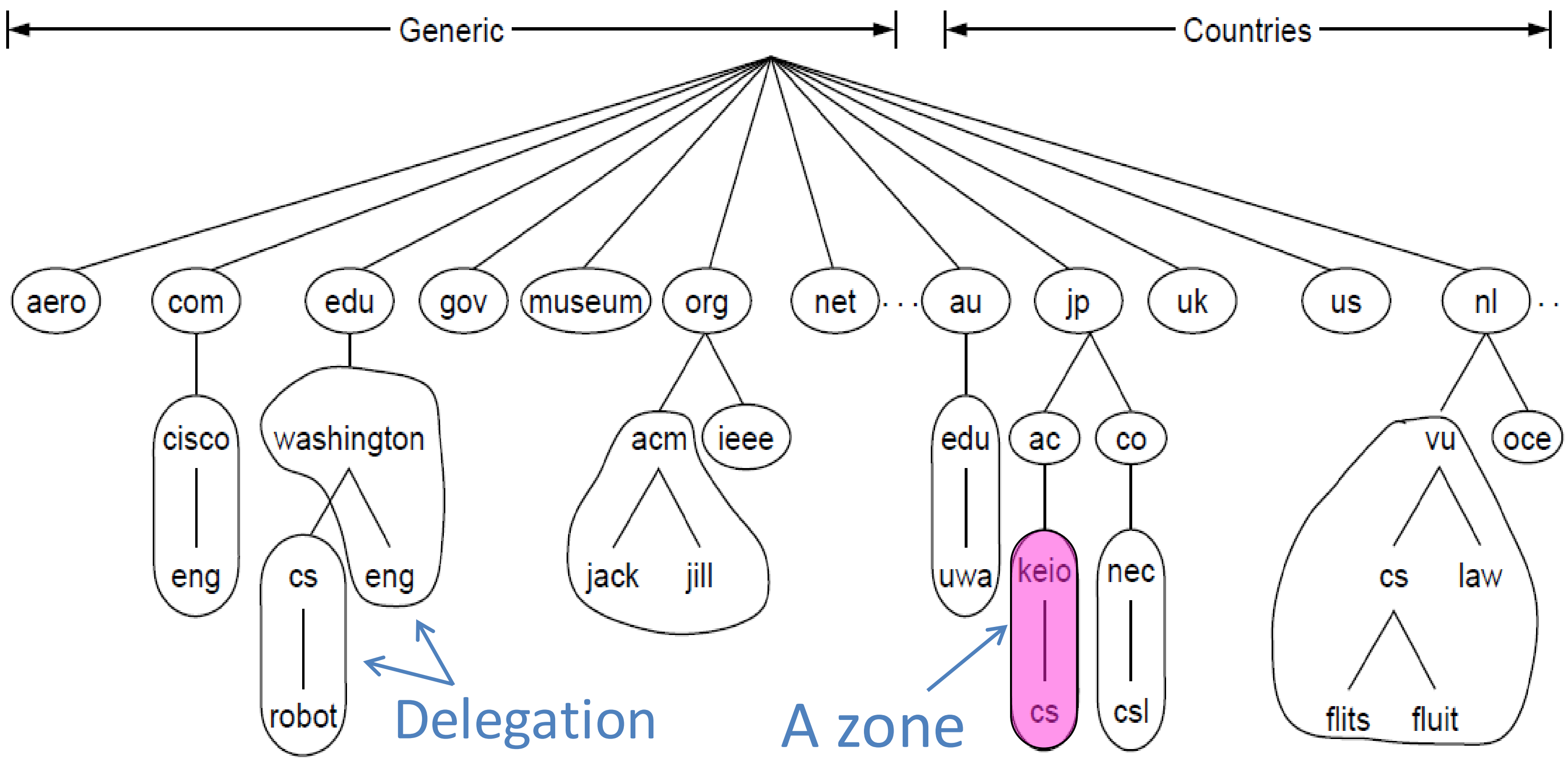


# TLDs (Top-Level Domains)

- Run by ICANN (Internet Corp. for Assigned Names and Numbers)
  - Starting in ‘98; naming is financial, political, and international ☺
- 700+ generic TLDs
  - Initially .com, .edu , .gov., .mil, .org, .net
  - Unrestricted (.com) vs Restricted (.edu)
  - Added regions (.asia, .kiwi), Brands (.apple), Sponsored (.aero) in 2012
- ~250 country code TLDs
  - Two letters, e.g., “.au”, plus international characters since 2010
  - Widely commercialized, e.g., .tv (Tuvalu)
  - Many domain hacks, e.g., instagr.am (Armenia), kurti.sh (St. Helena)

# DNS Zones

- A <u>zone</u> is a contiguous portion of the namespace



# DNS Zones (2)

- Zones are the basis for distribution
  - EDU Registrar administers .edu
  - UW administers washington.edu
  - CSE administers cs.washington.edu
- Each zone has a <u>nameserver</u> to contact for information about it
  - Zone must include contacts for delegations, e.g., .edu knows nameserver for washington.edu

# DNS Resource Records

- A zone is comprised of DNS resource records that give information for its domain names

| Type | Meaning |
|---|---|
| SOA | Start of authority, has key zone parameters |
| A | IPv4 address of a host |
| AAAA (“quad A”) | IPv6 address of a host |
| CNAME | Canonical name for an alias |
| MX | Mail exchanger for the domain |
| NS | Nameserver of domain or delegated subdomain |

# DNS Resource Records (2)

; Authoritative data for cs.vu.nl

| | | | | |
|---|---|---|---|---|
| cs.vu.nl. | 86400 | IN | SOA | star boss (9527,7200,7200,241920,86400) |
| cs.vu.nl. | 86400 | IN | MX | 1 zephyr |
| cs.vu.nl. | 86400 | IN | MX | 2 top |
| cs.vu.nl. | 86400 | IN | NS | star |
| | | | | |
| star | 86400 | IN | A | 130.37.56.205 |
| zephyr | 86400 | IN | A | 130.37.20.10 |
| top | 86400 | IN | A | 130.37.20.11 |
| www | 86400 | IN | CNAME | star.cs.vu.nl |
| ftp | 86400 | IN | CNAME | zephyr.cs.vu.nl |
| | | | | |
| flits | 86400 | IN | A | 130.37.16.112 |
| flits | 86400 | IN | A | 192.31.231.165 |
| flits | 86400 | IN | MX | 1 flits |
| flits | 86400 | IN | MX | 2 zephyr |
| flits | 86400 | IN | MX | 3 top |
| | | | | |
| rowboat | | IN | A | 130.37.56.201 |
| | | IN | MX | 1 rowboat |
| | | IN | MX | 2 zephyr |
| | | | | |
| little-sister | | IN | A | 130.37.62.23 |
| | | | | |
| laserjet | | IN | A | 192.31.231.216 |

← Name server

← IP addresses of computers

← Mail gateways

# dig

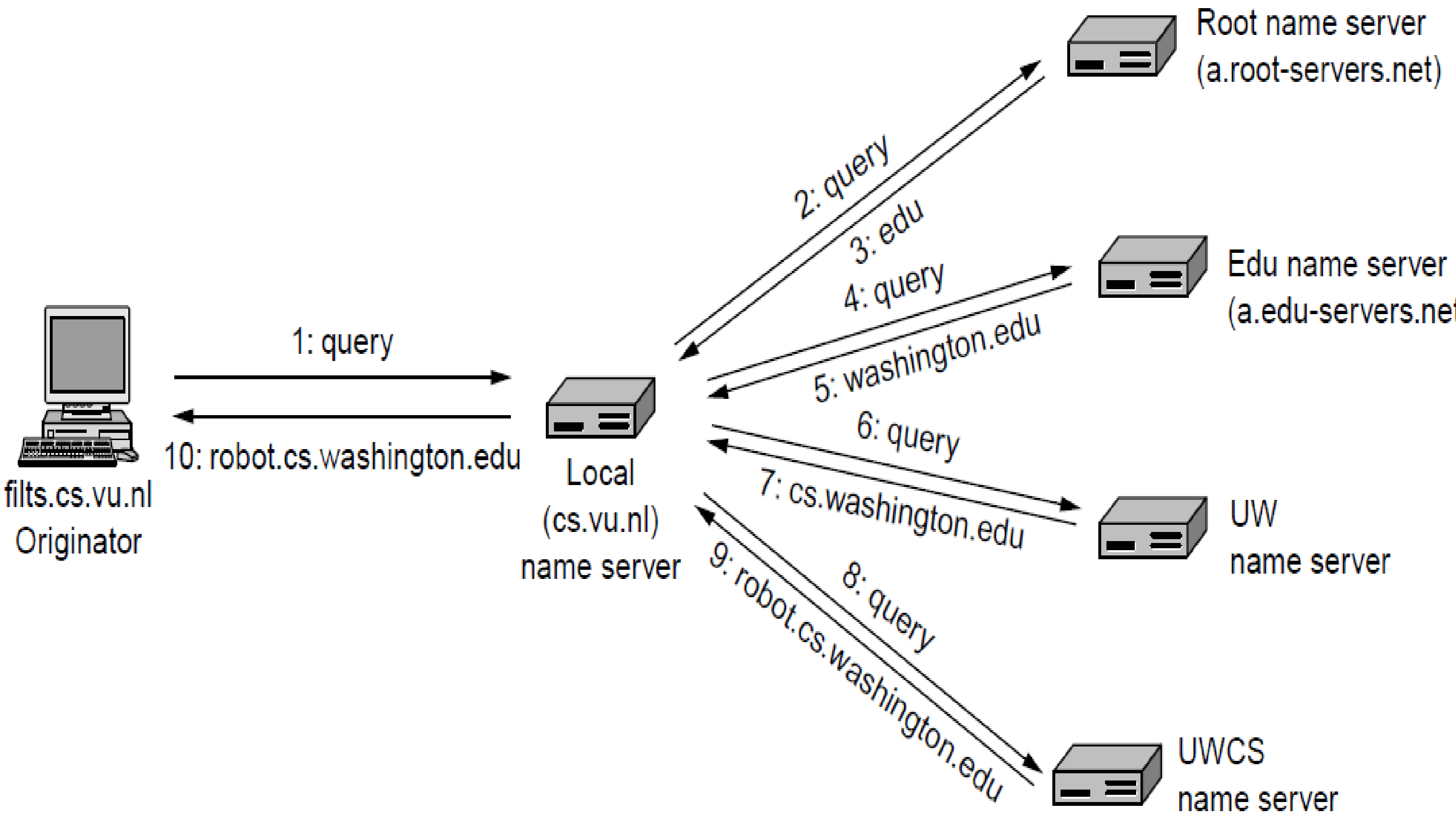
```
$ dig @june.cs.washington.edu attu.cs.washington.edu ANY

;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 34244
;; flags: qr aa rd; QUERY: 1, ANSWER: 4, AUTHORITY: 5, ADDITIONAL: 5
;; WARNING: recursion requested but not available

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;attu.cs.washington.edu.                IN             ANY

;; ANSWER SECTION:
attu.cs.washington.edu.    60           IN             A              128.208.1.139
attu.cs.washington.edu.    60           IN             A              128.208.1.138
attu.cs.washington.edu.    60           IN             A              128.208.1.137
attu.cs.washington.edu.    60           IN             A              128.208.1.140

;; AUTHORITY SECTION:
cs.washington.edu.         86400        IN             NS             lumpy.cs.washington.edu.
cs.washington.edu.         86400        IN             NS             marge.cac.washington.edu.
cs.washington.edu.         86400        IN             NS             hanna.cac.washington.edu.
cs.washington.edu.         86400        IN             NS             holly.s.uw.edu.
cs.washington.edu.         86400        IN             NS             june.cs.washington.edu.

;; ADDITIONAL SECTION:
june.cs.washington.edu.    1            IN             AAAA           2607:4000:200:17::104
lumpy.cs.washington.edu. 86400          IN             AAAA           2607:4000:200:17::102
june.cs.washington.edu.    86400        IN             A              128.95.1.4
lumpy.cs.washington.edu. 86400          IN             A              128.95.1.2
```

# DNS Resolution

- DNS protocol lets a host resolve any host name (domain) to IP address
- If unknown, can start with the root nameserver and work down zones
  - But can use cache to do as much resolution as possible
- Let's see an example first …

# DNS Resolution (2)

- flits.cs.vu.nl resolves robot.cs.washington.edu



# Iterative vs. Recursive Queries

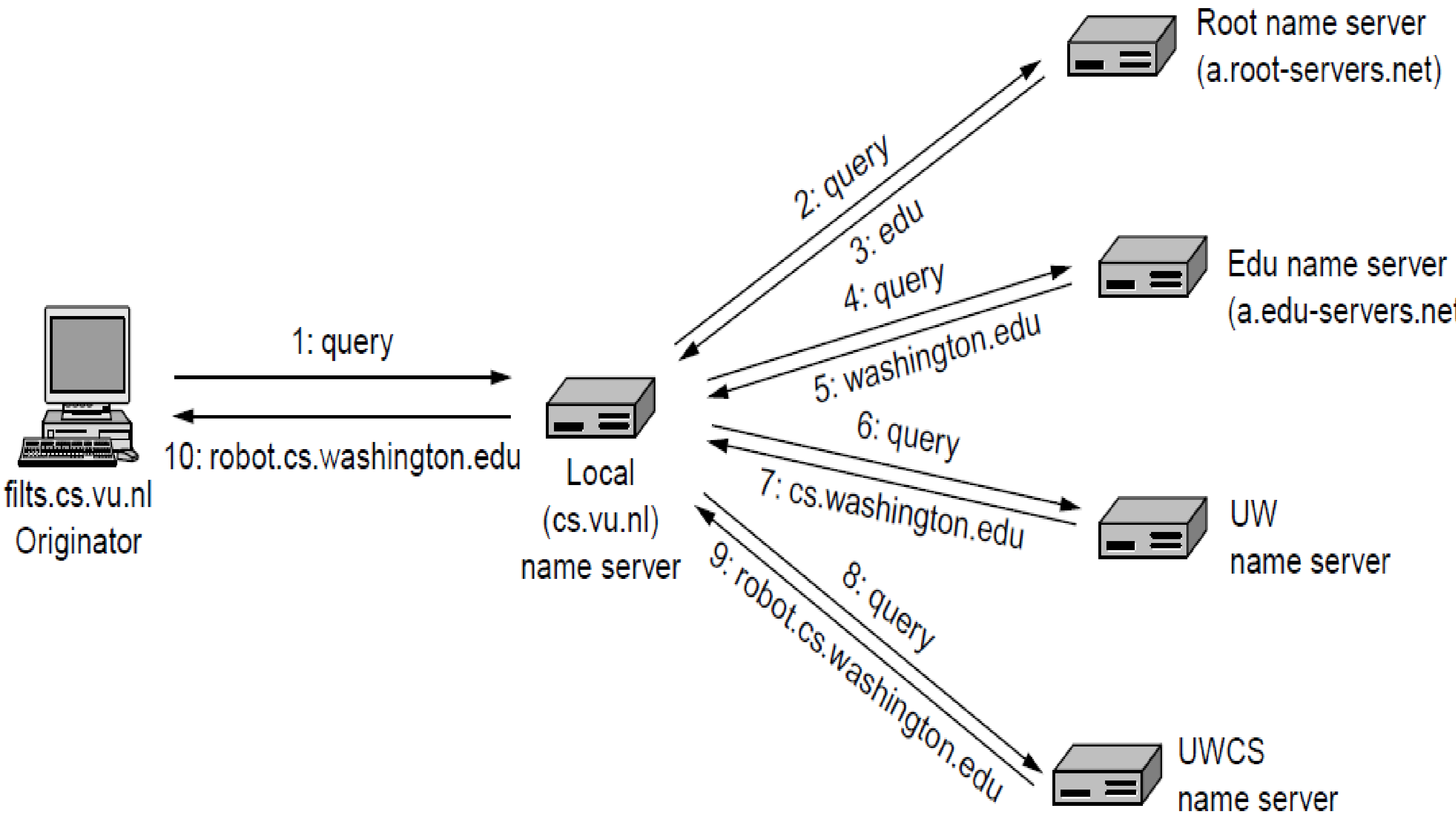
- Recursive query
  - Nameserver resolves and returns final answer
  - E.g., flits → local nameserver
- Iterative (Authoritative) query
  - Nameserver returns answer or who to contact for answer
  - E.g., local nameserver → all others

# Iterative vs. Recursive Queries (2)

- Recursive query
  - Lets server offload client burden (simple resolver) for manageability
  - Lets server cache over a pool of clients for better performance
- Iterative query
  - Lets server "file and forget"
  - Easy to build high load servers

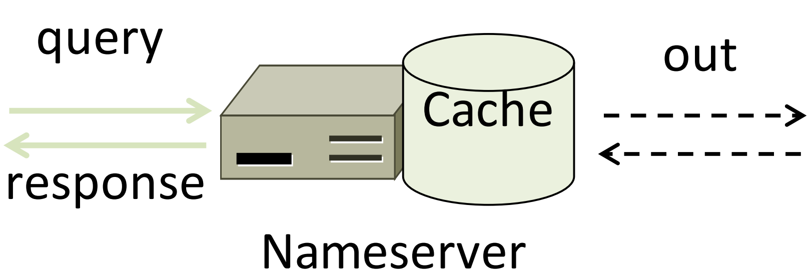
# DNS Resolution (2)

- flits.cs.vu.nl resolves robot.cs.washington.edu



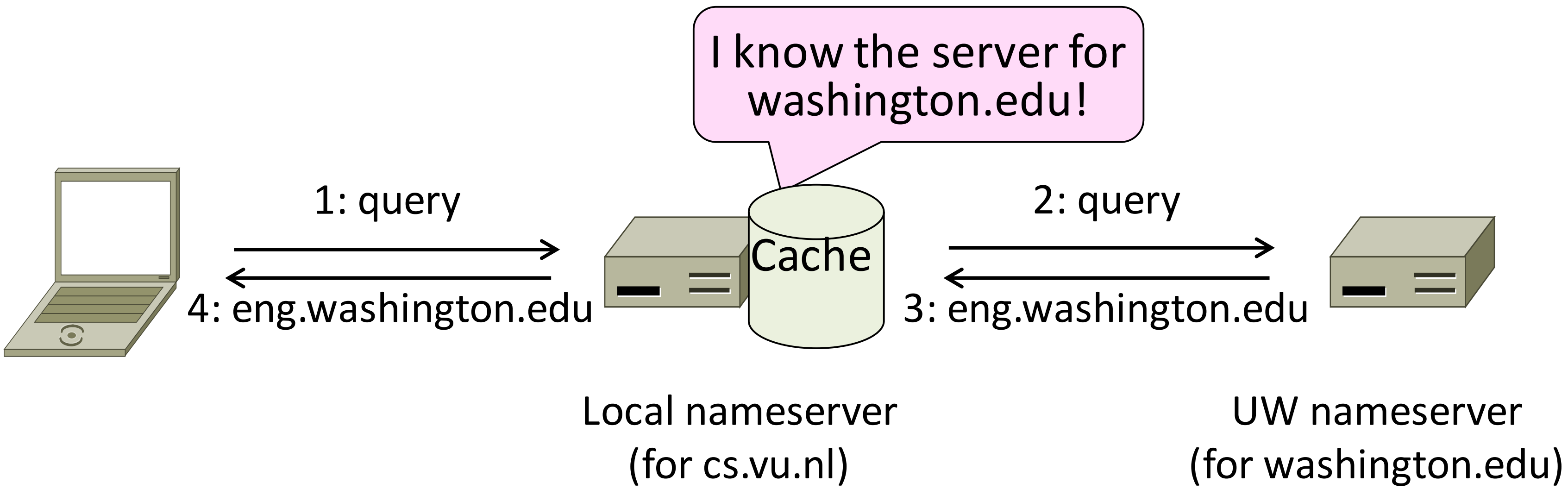
# Caching

- Resolution latency should be low
  - Adds delay to web browsing
- Cache query/responses to answer future queries immediately
  - Including partial (iterative) answers
  - Responses carry a TTL for caching



# Caching (2)

- flits.cs.vu.nl now resolves eng.washington.edu
  - And previous resolutions cut out most of the process



# Local Nameservers

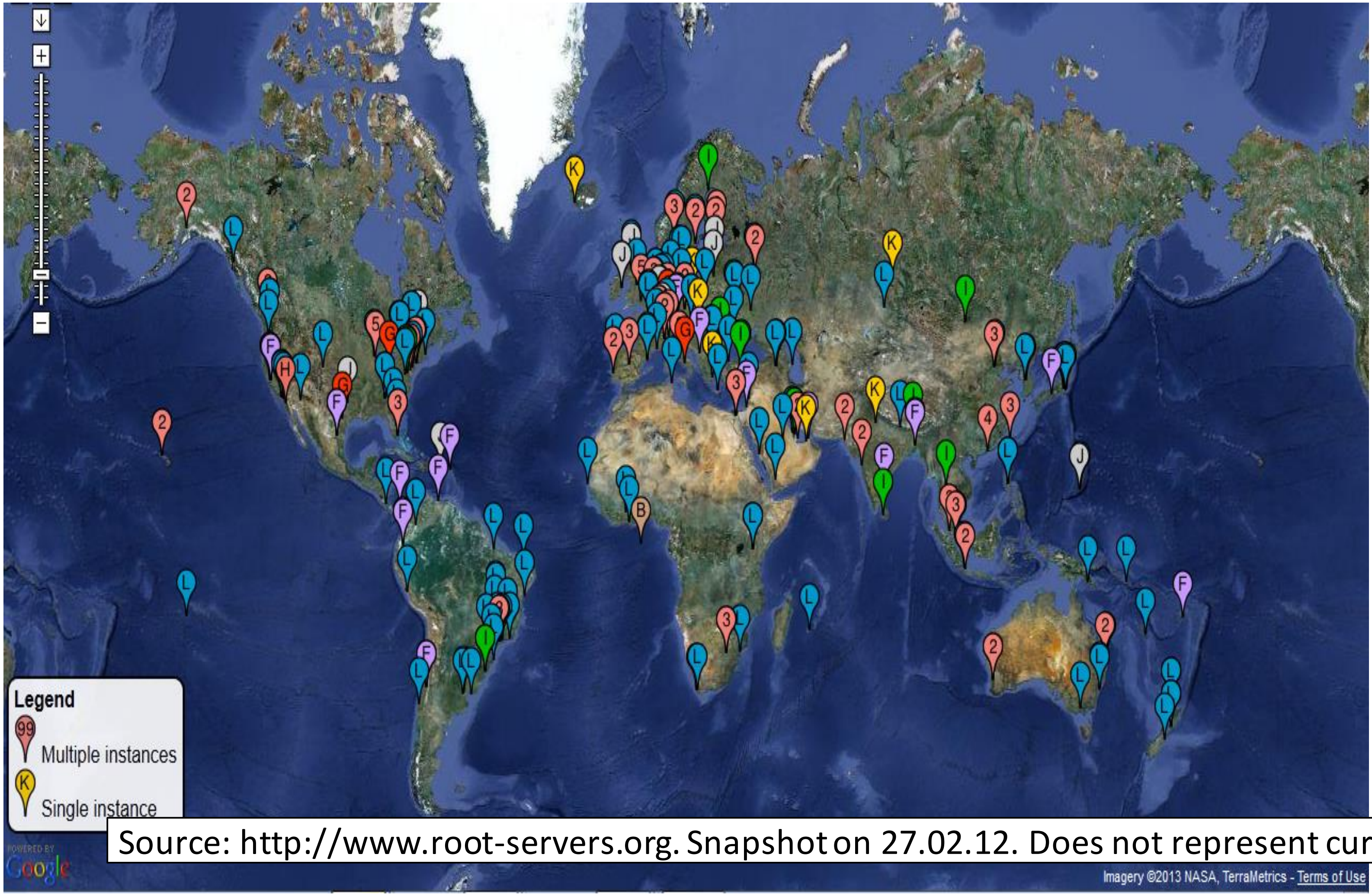
- Local nameservers often run by IT (enterprise, ISP)
  - But may be your host or AP
  - Or alternatives e.g., Google public DNS
- Clients need to be able to contact local nameservers
  - Typically configured via DHCP

# Root Nameservers

- Root (dot) is served by 13 server names
  - a.root-servers.net to m.root-servers.net
  - All nameservers need root IP addresses
  - Handled via configuration file (named.ca)

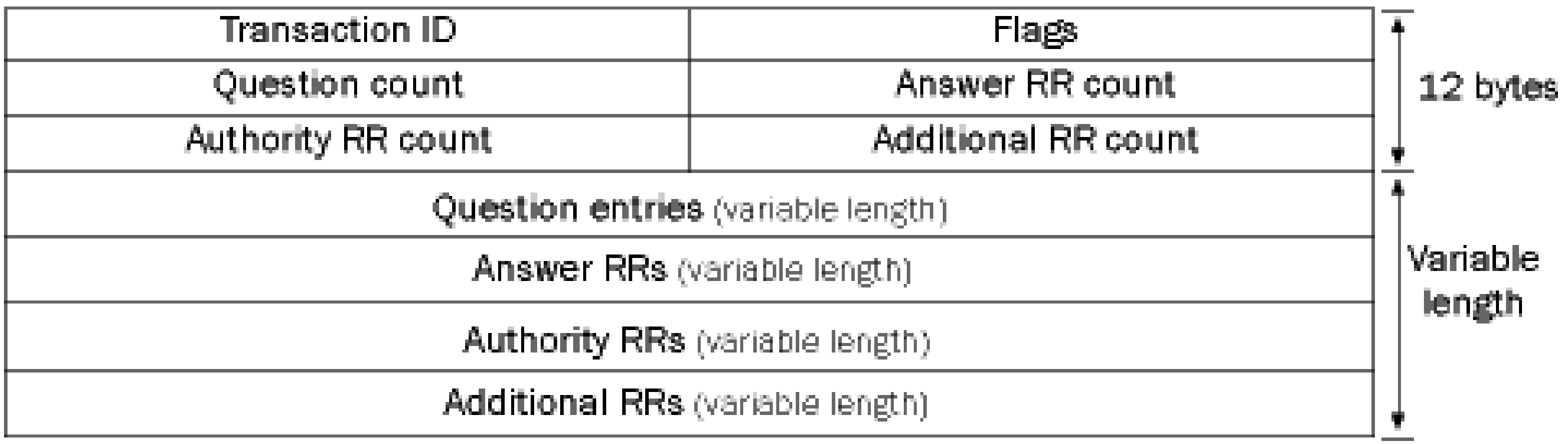
- There are >250 distributed server instances
  - Highly reachable, reliable service
  - Most servers are reached by IP anycast (Multiple locations advertise same IP! Routes take client to the closest one.)
  - Servers are IPv4 and IPv6 reachable

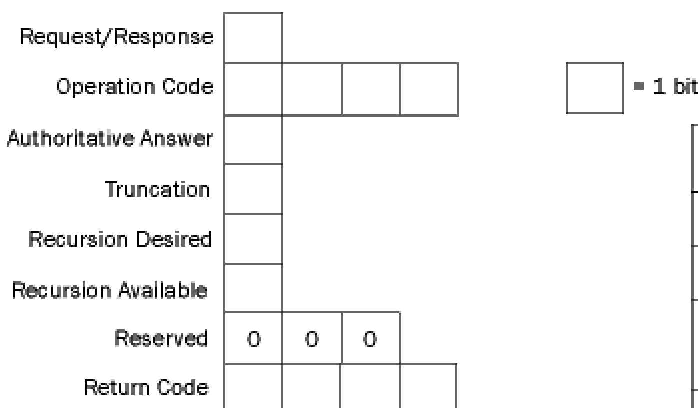
# Root Server Deployment

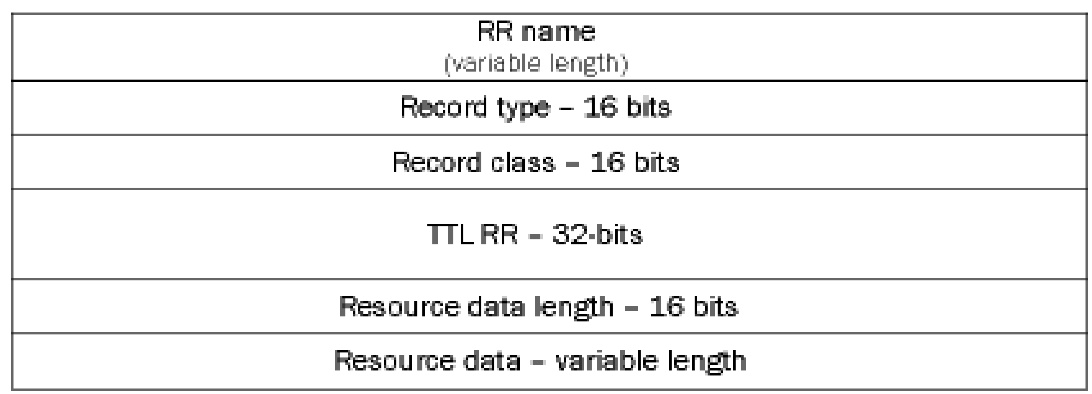



Source: http://www.root-servers.org. Snapshot on 27.02.12. Does not represent current deplo

# DNS Query Format

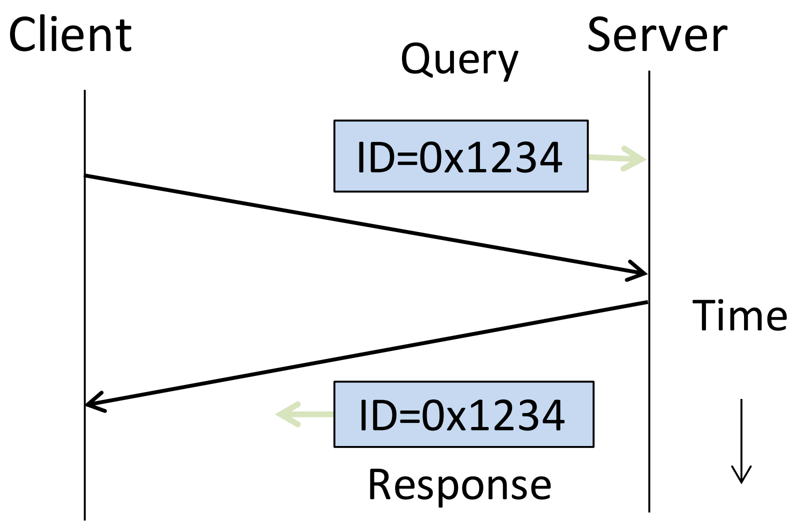


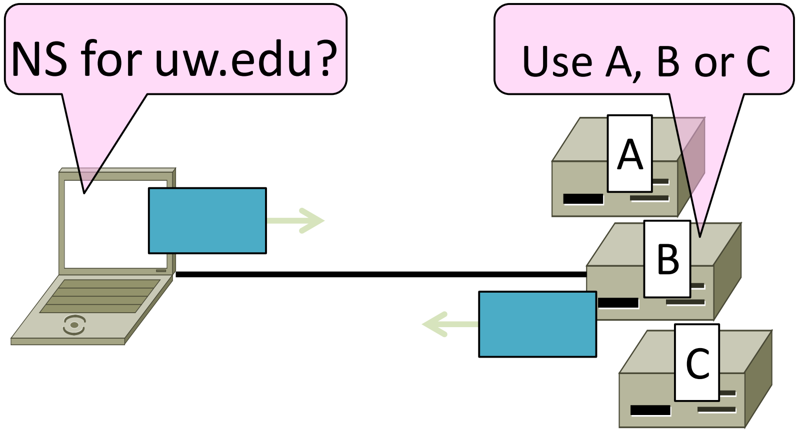
# DNS Protocol

- Query and response messages
  - Built on UDP messages, port 53
  - UDP is unreliable
    - time out and repeat request
    - server is stateless (and requests are idem potent)
  - Query/response linked by a 16-bit ID field



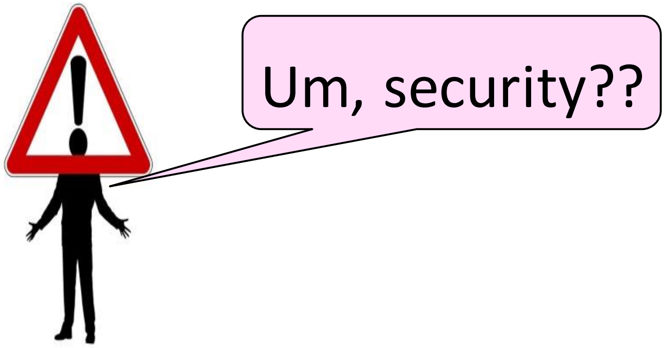
# DNS Protocol (2)

- Service reliability via replicas
  - Run multiple nameservers for domain
  - Return the list; clients use one answer
  - Helps distribute load too



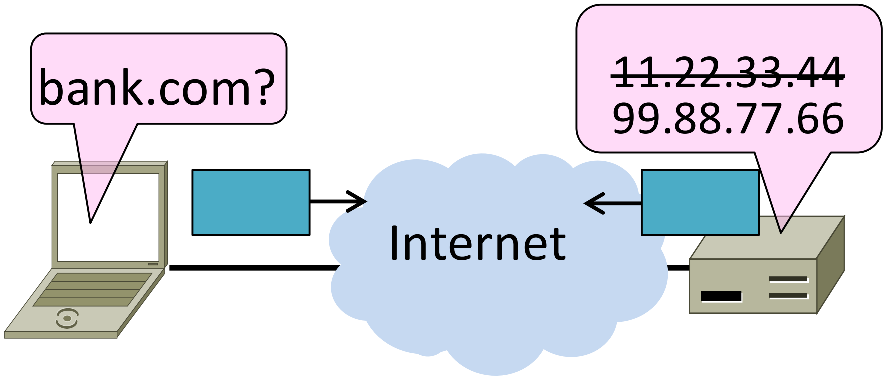
# DNS Protocol (3)

- Security is a major issue
  - Compromise redirects to wrong site!
  - Not part of initial protocols ..
- DNSSEC (DNS Security Extensions)
  - Mostly deployed



# Goal and Threat Model

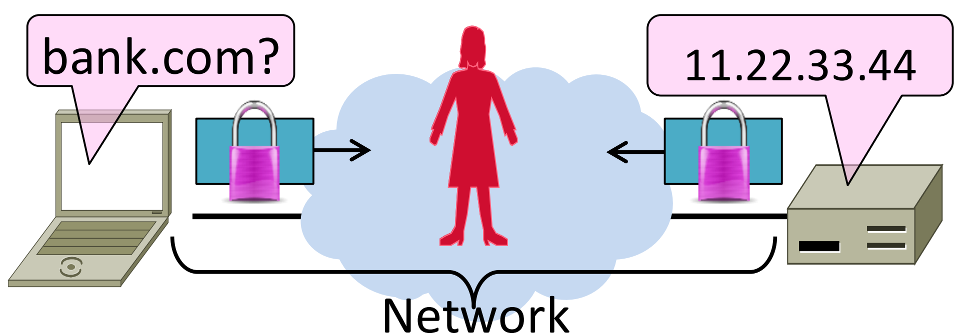
- Naming is a crucial Internet service
  - Binds host name to IP address
  - Wrong binding can be disastrous …



# Goal and Threat Model (2)

- Goal is to secure the DNS so that the returned binding is correct
  - Integrity/authenticity vs confidentiality

- Attacker can tamper with messages on the network

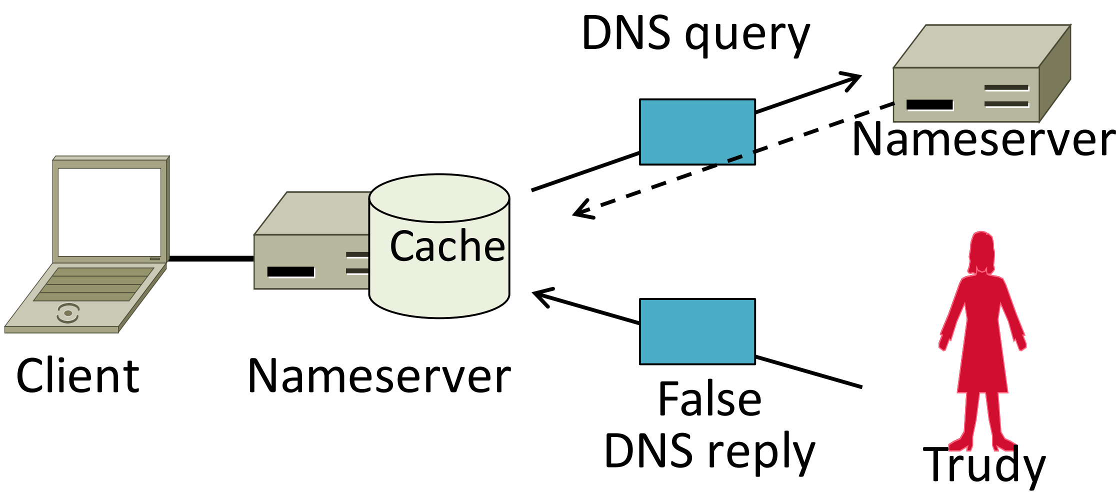


# DNS Spoofing

- How can attacker corrupt the DNS?
- Can trick nameserver into caching the wrong binding *by using the DNS protocol itself!*
  - This is called DNS spoofing

# DNS Spoofing (2)

- To spoof, Trudy returns a fake DNS response that appears to be true
  - Fake response contains bad binding



# DNS Spoofing (3)

- Lots of questions!
  1. How does Trudy know when the DNS query is sent and what it is for?
  2. How can Trudy supply a fake DNS reply that appears to be real?
  3. What happens when the real DNS reply shows up?
- There are solutions to each issue …

# DNS Spoofing (4)

1. How does Trudy know when the query is sent and what it is for?

- Trudy can make the query herself!
  - Nameserver works for many clients
  - Trudy is just another client

# DNS Spoofing (5)

2. How can Trudy supply a fake DNS reply that appears to be real?

- A bit more difficult. DNS checks:
  - Reply is from authoritative nameserver (e.g., .com)
  - Reply ID that matches the request
  - Reply is for outstanding query
- (Nothing about content though …)

# DNS Spoofing (6)

2. How can Trudy supply a fake DNS reply that appears to be real?

- Techniques:
  - Put IP of authoritative nameserver as the source IP address
  - ID is 16 bits (64K). Send many guesses! (Or if a counter, sample to predict.)
  - Send reply right after query
- Good chance of succeeding!

# DNS Spoofing (7)

3. What happens when real DNS reply shows up?

- Not likely a problem
  - There is no outstanding query after fake reply is accepted
  - So real reply will be discarded

# DNSSEC (DNS Security Extensions)

- Extends DNS with new record types
  - RRSIG for digital signatures of records
  - DNSKEY for public keys for validation
  - DS for public keys for delegation
  - First version in ‘97, revised by ’05
- Deployment requires software upgrade at both client and server
  - Root servers upgraded in 2010
  - Followed by uptick in deployment

# $ dig edu ANY

```
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32320
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 12, AUTHORITY: 6, ADDITIONAL: 8

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;edu.                    IN      ANY

;; ANSWER SECTION:
edu.    92995   IN   RRSIG   NS 8 1 172800 20171004051028 20170927040028 10478 edu.
fAO6AHlcNx4j7uE/Cu5KJWII7uLChR8bolrOFiCts/wEUVuaWaGd3vTQyl6XW+kJtLIVM74UIdqH77SJQolBecpSSPL5mmz/U2iEOSiHFl3/1rw2 7xAhrFHxVg0ZsyyZSKElJXbWLyWGpBlo3M8wQXiJfV055JmMOEQhZKzf /EA=
edu.    6595    IN   RRSIG   DNSKEY 8 1 86400 20171008150000 20170923145500 28065 edu.
LJ3AKlEw1mx6yHSFG4el9iBqKYZD3Z+aHb7fTg/Cvc9Nk02GpL2PxXOs y7tcZ76dV85KDPRy7Ev2jFZIlHODfPAG+FUa4b0fpaR2fk0f8tDfYdPi ON8cEIKzLbZSCMr6pYowHSp8JoM5YKkzUTiggZ3S3FHMlZM6ylUM/iO6
PH8eSBknniHzxR0We1jdwyLf8+FZMFzXNRPO3uYYmZ7etnjJ+02DWfYC psc9sA4Bjd4wLTWZIFWEYygS3OS+J8soO3VywcOkdCiLo57yPXcUPXCe wYs6u6+w3mEx8T/Fa24GEVuwPjRZDkcg/eB5Amq1LGa23bM7G9Ypet5h wle/OA==
edu.    6595    IN   DNSKEY  256 3 8
AQOurUqXhGxsfZjYo0s/l984A/wpP0WHqqmFsD9RHm/7nFhKilLG6VRD a7qTD/VDzNUL/jiaBEh/mM2QA4y3rUfsxqnJSVweosIPzw0scTahXGSF K9w2AvYqYj96dSQ0TJt9+AUkCF8Z+DTa0DvmXYUHyvMqhSLq2hfM7sAz 9mQ2AQ==
edu.    6595    IN   DNSKEY  257 3 8
AQOluvuVWACqSUX6EZr3d5YIIv13SDXBpDeGRCIET3T0l1wp0NprtJhf c1YVehIlNqucijLnEhSLN2i0Txnh3bU5aQJLRf/qm21etL35ozAIlnfS gY5D0nxLPoBIRq++8Mxvr+2YCSRaCNyrqaU4TnHaPiHLuKe7sGyHQ8K8
rXst7HxF5zgG+gIeytYIxpy4/bAaKCNVvGaYHxuNSBJBiYSrxjiZk268 EP6E3T+bSdi2HepB3ZuBWyOewsmHO8aoHmzIH049vBNI7iFZByRKY9yA J8TshANtiUH HwAF7JJtWLW9oO10BqygYG+zS4weKGtkcNpzOUDWs41FW M8aFs4L1
edu.    6595    IN   RRSIG   DS 8 1 86400 20171011170000 20170928160000 15768 .
Sb3xPG5UZWUx7nc9bERAOwTr1hnt+dE1Hv28/iw0ZSYwICJiEGPVZonU lPjujI/pyOjl6jSRX2MDX8vvm04GViF8GndeeQQjVq7Q/tqiw7RTXc17 rWW98LDENGE7AzyE/iAwZCrukYAwvck9Y88wRnwn6BVM89NRgfInPQ5G
hhF3ILevBIPae8cteADtX0A7tnmw5NWf0eQZzVGzlehgFD8BIj7l3n5j FhWCTCiAfCBrNgupY3OBcsLdCiBSQx5t1VGhGCrK3yinp/KlFzzAf2/TVKwZvmd6D+V OqHd/sbtpt4SOmQdgCmXEKPg2f8vfUg4gLMWYp0G06lsG J2H6Pg==
edu.    6595    IN   DS      28065 8 2
4172496CDE85534E51129040355BD04B1FCFEBAE996DFDDE652006F6 F8B2CE76
edu.    92995   IN   NS      a.edu-servers.net.
edu.    92995   IN   NS      l.edu-servers.net.
edu.    92995   IN   NS      g.edu-servers.net.
edu.    92995   IN   NS      c.edu-servers.net.
edu.    92995   IN   NS      d.edu-servers.net.
edu.    92995   IN   NS      f.edu-servers.net.

;; AUTHORITY SECTION:
edu.    92995   IN   NS      l.edu-servers.net.
edu.    92995   IN   NS      g.edu-servers.net.
edu.    92995   IN   NS      a.edu-servers.net.
edu.    92995   IN   NS      f.edu-servers.net.
edu.    92995   IN   NS      d.edu-servers.net.
edu.    92995   IN   NS      c.edu-servers.net.

;; ADDITIONAL SECTION:
a.edu-servers.net.   92995   IN   A      192.5.6.30
c.edu-servers.net.   92995   IN   A      192.26.92.30
d.edu-servers.net.   92995   IN   A      192.31.80.30
f.edu-servers.net.   92995   IN   A      192.35.51.30
g.edu-servers.net.   92995   IN   A      192.42.93.30
l.edu-servers.net.   92995   IN   A      192.41.162.30
g.edu-servers.net.   92995   IN   AAAA   2001:503:cc2c::2:36
```