CSE 127: Intro to Computer Security

WI24

Lecture 10 - Intro to Networking

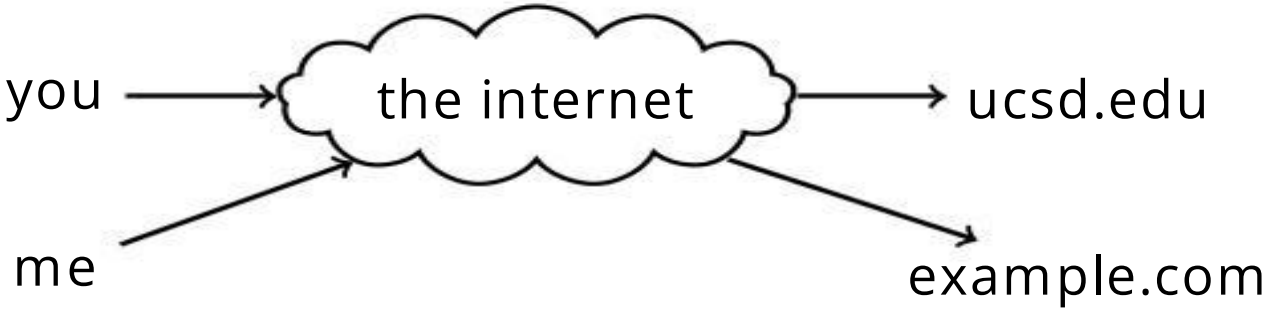
Slide Credit: Deian Stefan, Zakir Durumeric, David Wagner, Nadia Heninger, George Obaido, Earlence Fernandes, Imani Munyaka

Announcements

Grades Posted
Review regrade request policy

HW/PA3 - available, start now

The Internet



Original Idea:

- Network is dumb
- Simple, robust service
- Shift complexity to endpoints
- Acts like postal system (packet-based) rather than traditional phone system (circuit-based)

Need protocol to actually communicate

A protocol is an agreement on how to communicate.

Includes syntax and semantics.

- **Syntax:** How communication is specified and structured.
    - Format, order messages are sent and received.

Need protocol to actually communicate

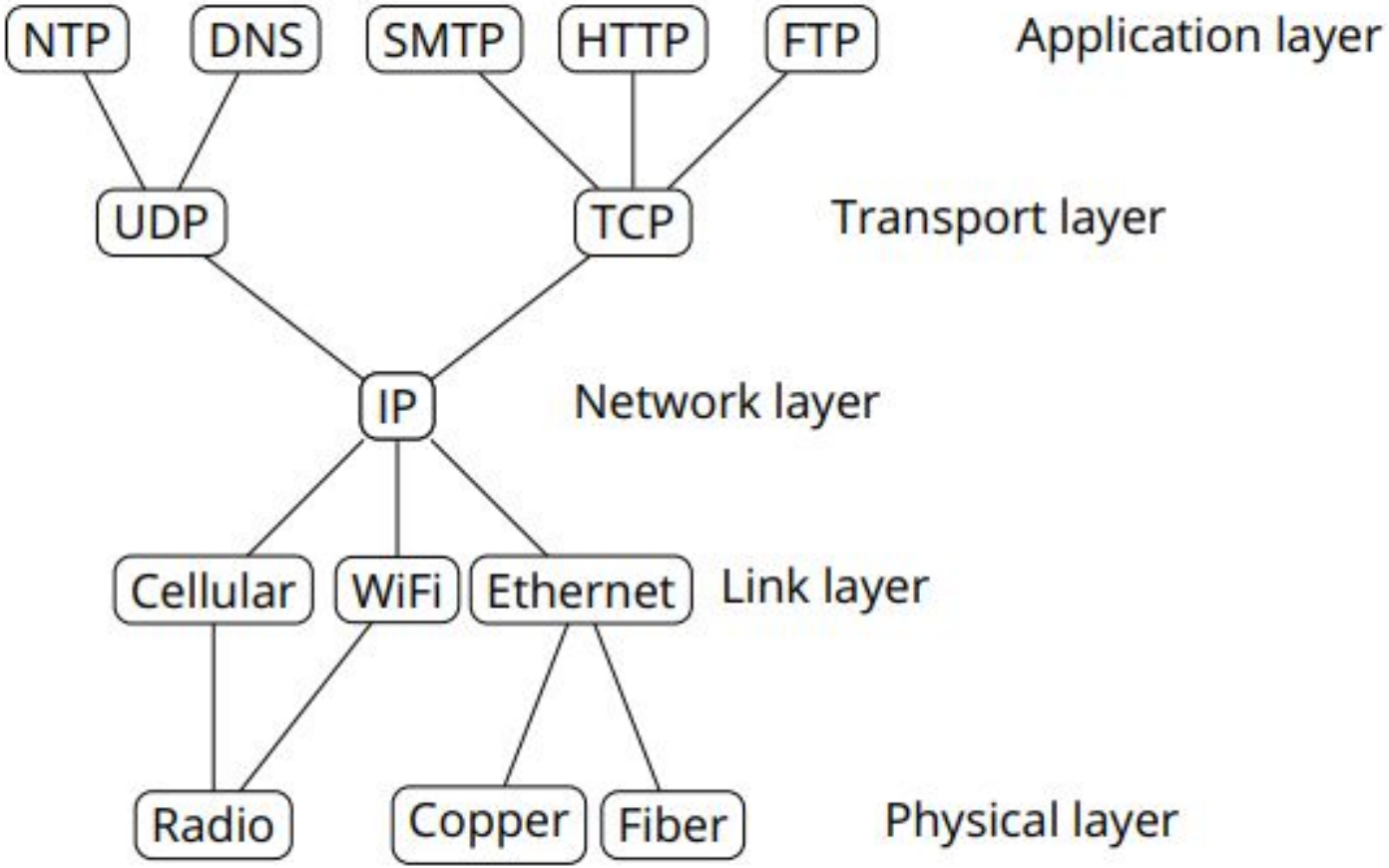
A protocol is an agreement on how to communicate.
Includes syntax and semantics.

- **Syntax:** How communication is specified and structured.
  - Format, order messages are sent and received.
- **Semantics:** What a communication means
  - Actions taken when transmitting, receiving, or timer expires.
- **Example:** RFC 2616 (HTTP/1.1)
  - Section 5: Syntax of HTTP Requests
  - Section 9.3: Semantics of GET Requests

Protocols are layered

- Networks use a stack of layers
- Lower layers provide services to layers above
  - Don’t care what higher layers do
- Higher layers use services of layers below
  - Don’t care how lower layers implement services
- Layers define abstraction boundaries
  - At a given layer, all layers above and below are opaque

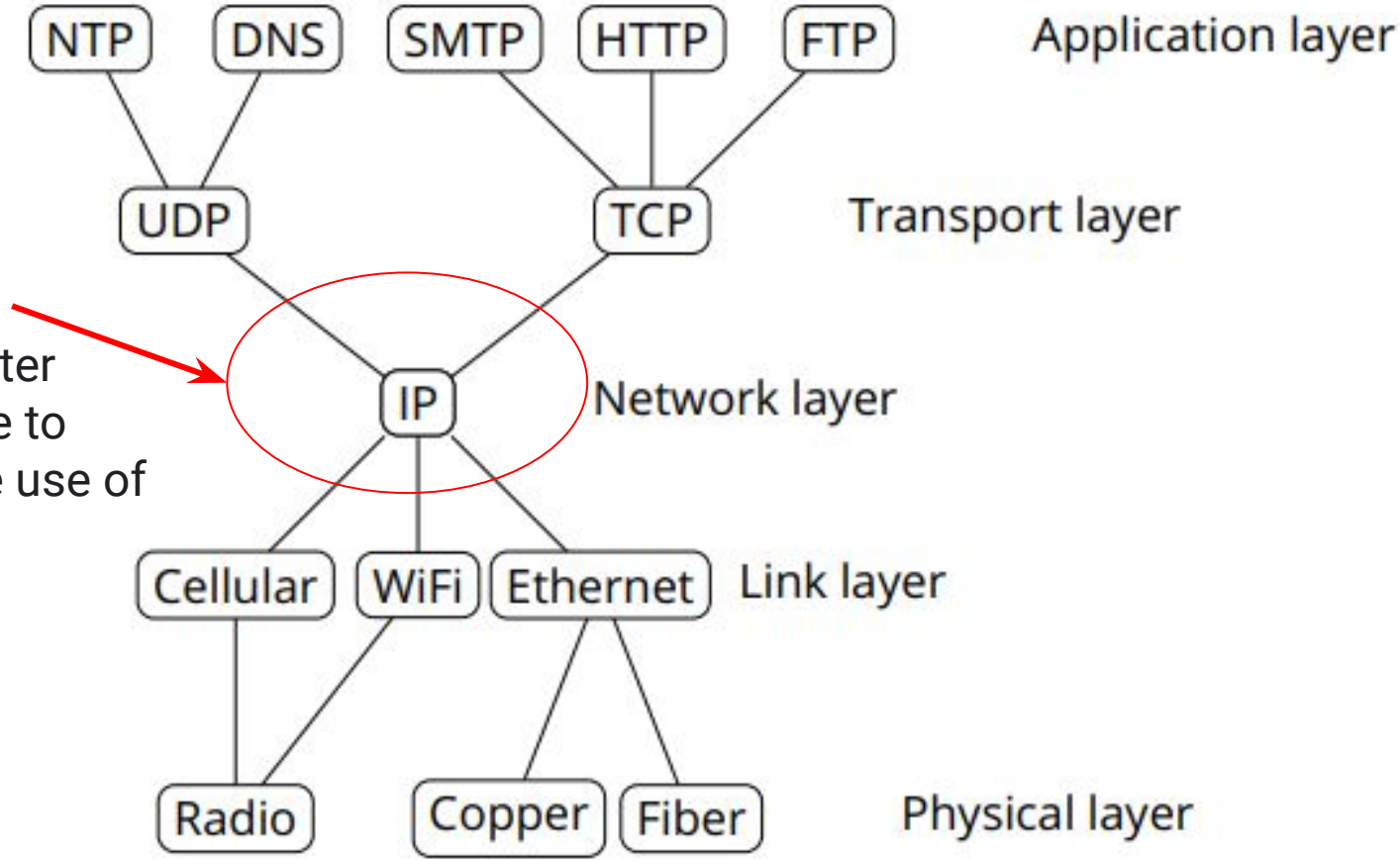
Basic Internet Architecture “Hourglass”



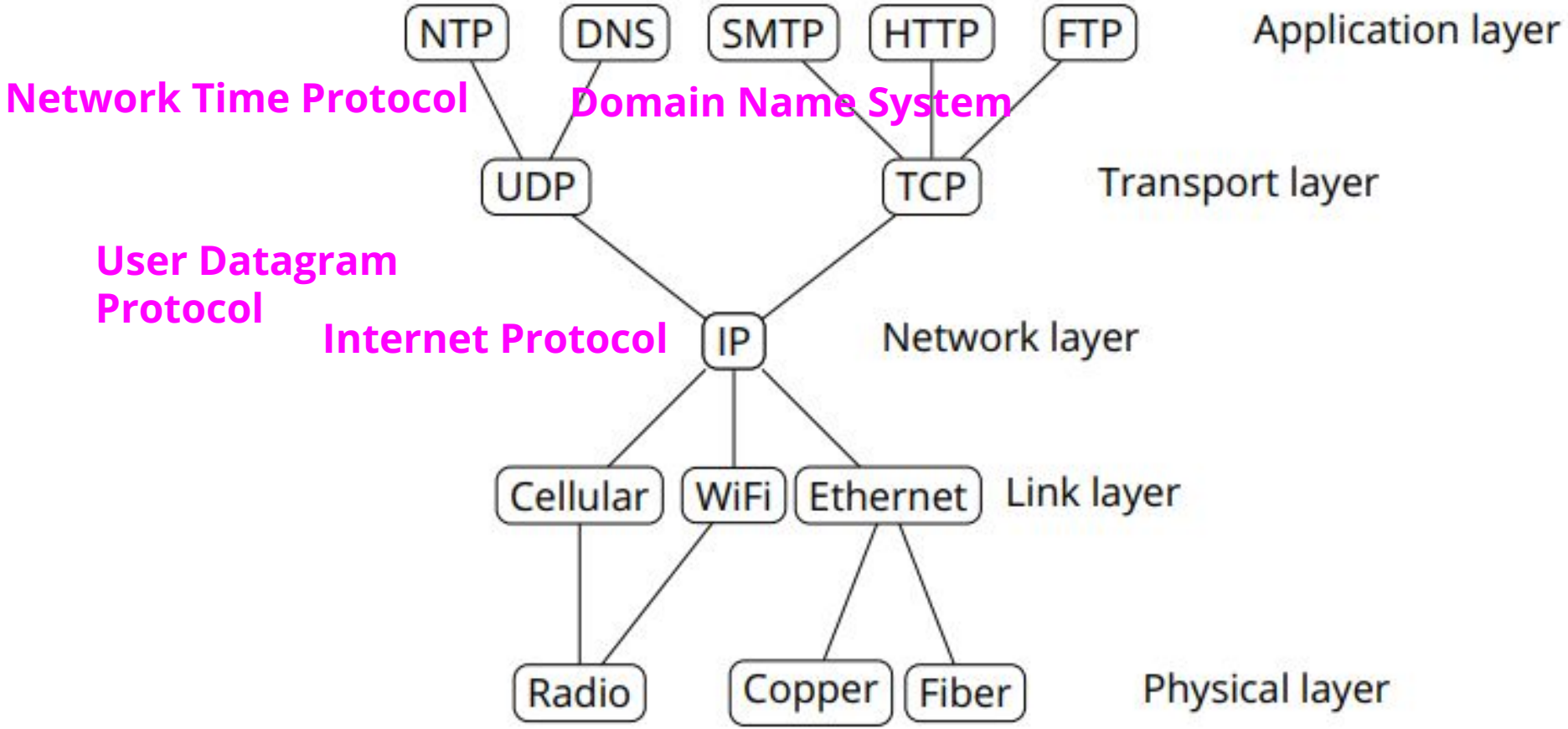
Basic Internet Architecture “Hourglass”

interoperability =

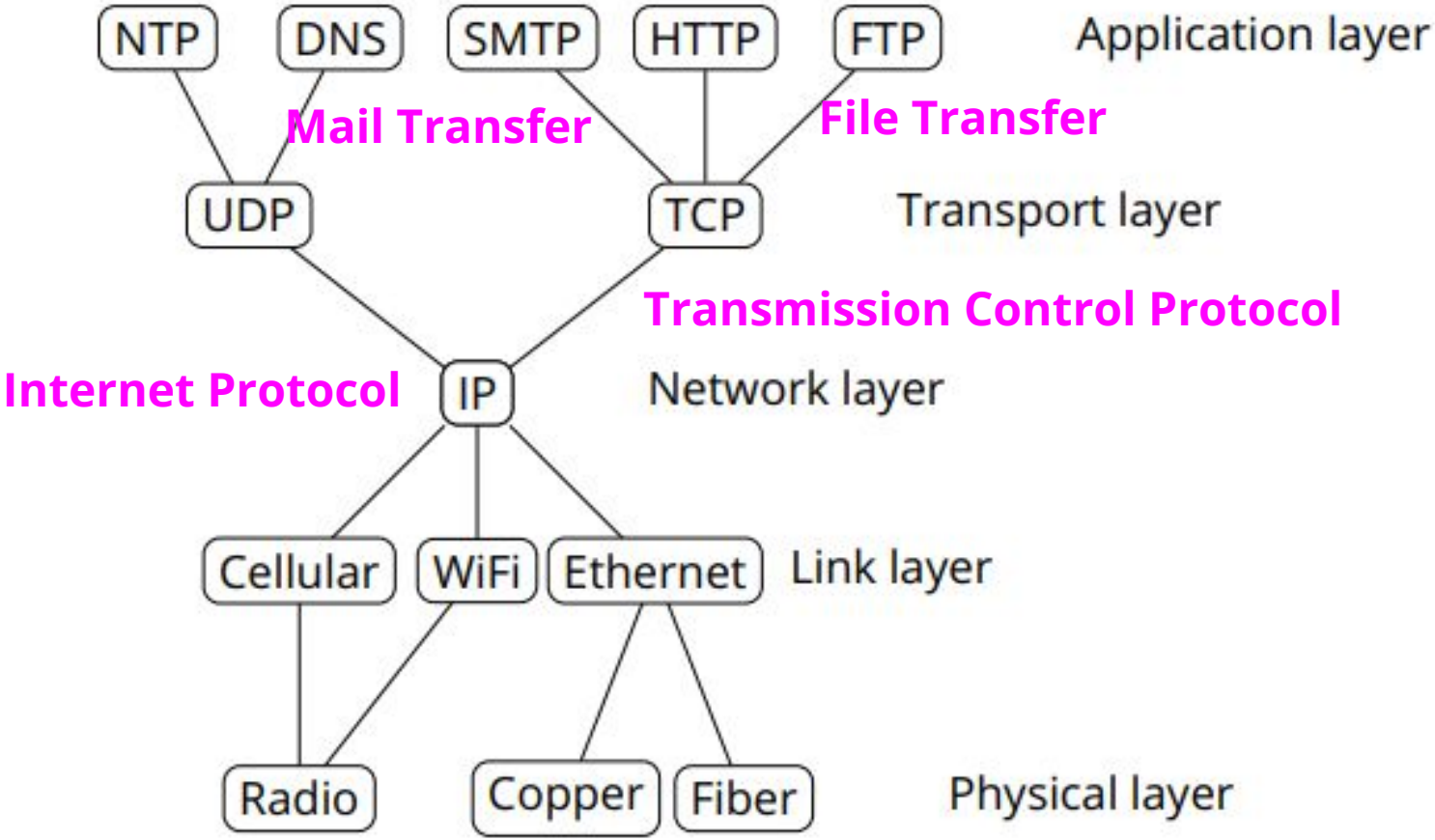
the ability of computer systems or software to exchange and make use of information.



Basic Internet Architecture “Hourglass”

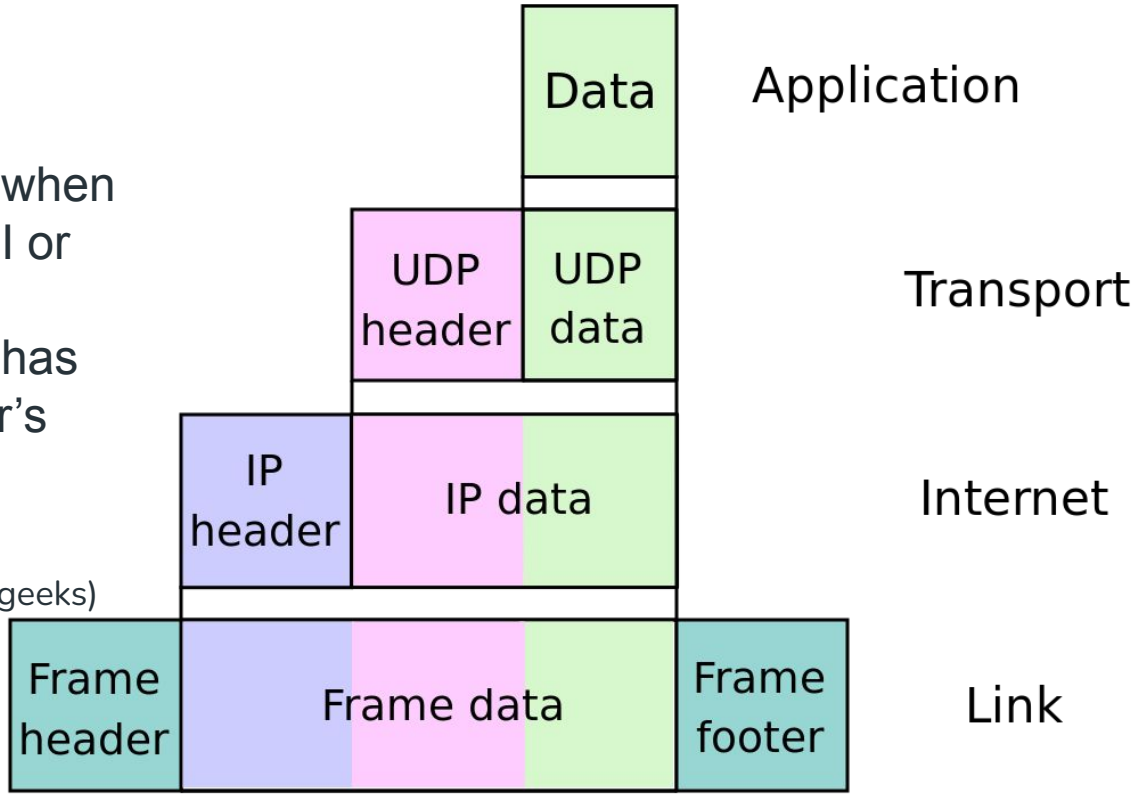


Basic Internet Architecture “Hourglass”



Packet encapsulation at each layer

Encapsulation is the process of adding additional information when data is traveling in OSI or TCP/IP model. The additional information has been added on sender's side, starting from Application layer to Physical layer. (Geeksforgeeks)

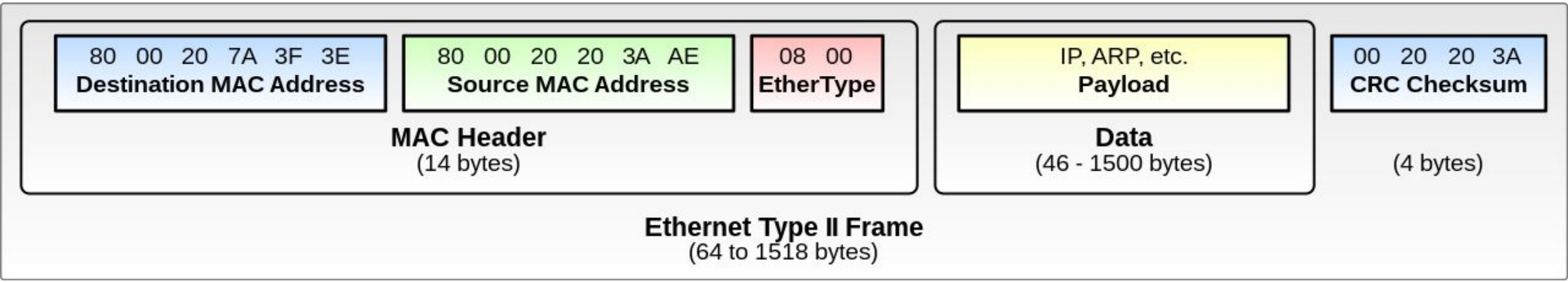


Source: Wikipedia

Link layer: Connecting hosts to local network

Most common link layer protocol: **Ethernet**

CRC = cyclic redundancy check



- Messages organized into frames
- Every node has a globally unique 6-byte MAC address
  Source: Wikipedia

Link Layer: Connecting host to local network

- Originally a broadcast protocol: every node on network received every packet
- Now switched: switch learns the physical port for each MAC address and sends packets to correct port if known
- WiFi similar to Ethernet, but nodes can move

IP: Internet Protocol

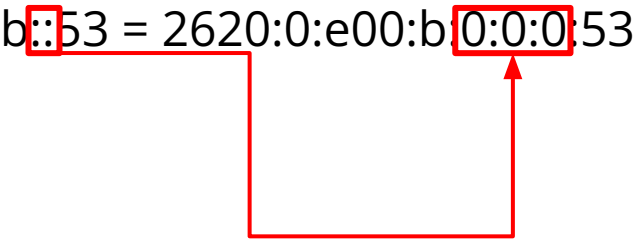
- Connectionless delivery model
- “Best effort” = no guarantees about delivery
- No attempt to recover from failure
- Packets might be lost, delivered out of order, delivered multiple times
- Packets might be fragmented
- Provides hierarchical addressing scheme

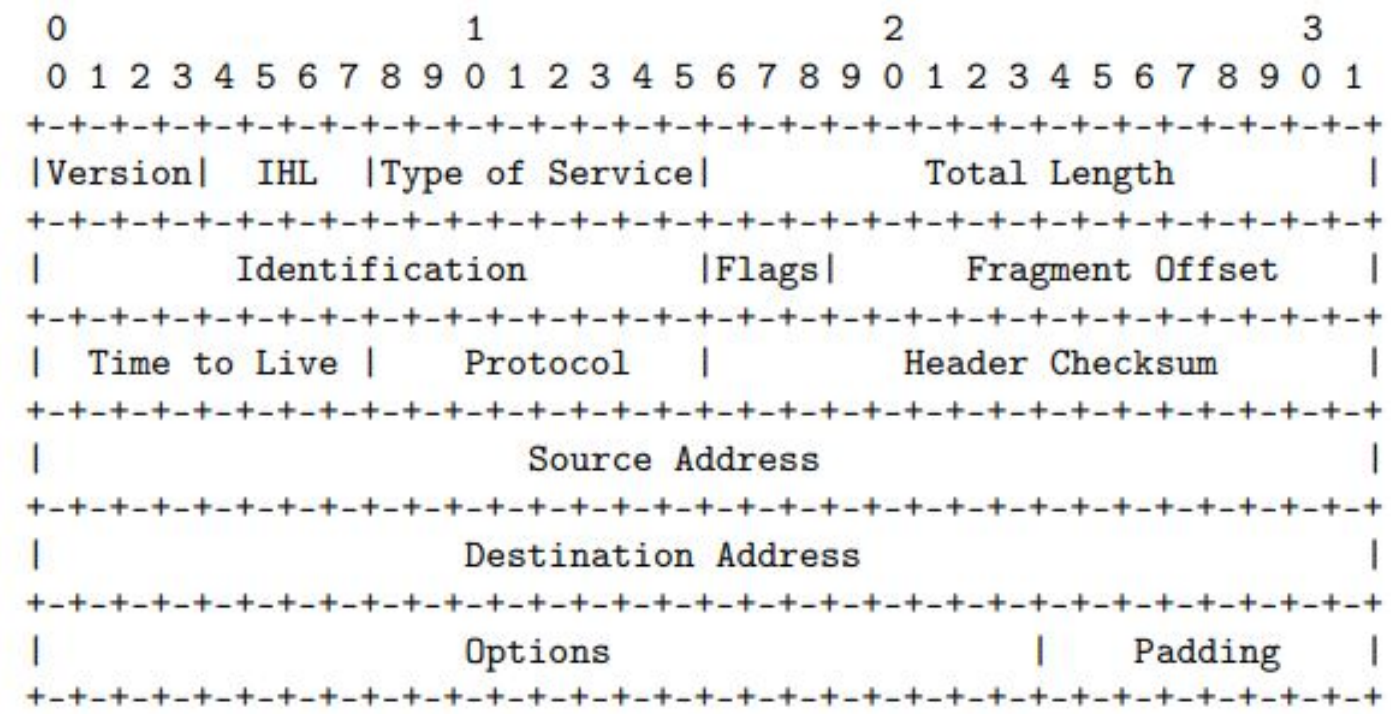
IP: Internet Protocol

- IPv4
  - 32-bit host addresses
  - Written as 4 bytes in decimal,
  - e.g. 192.168.1.1
- IPv6
  - 128-bit host addresses
  - Written as 16 bytes in hex
  - :: implies zero bytes
  - e.g. 2620:0:e00:b::53 = 2620:0:e00:b:0:0:0:53

IP: Internet Protocol

- IPv4
  - 32-bit host addresses
  - Written as 4 bytes in decimal,
  - e.g. 192.168.1.1
- IPv6
  - 128-bit host addresses
  - Written as 16 bytes in hex
  - :: implies zero bytes
  - e.g. 2620:0:e00:b::53 = 2620:0:e00:b:0:0:0:53





Example Internet Datagrarm Header

Note that each tick mark represents one bit position.

ARP: Address Resolution Protocol

- Problem: How does a host learn what MAC addresses to send packets to?
- ARP lets hosts build table mapping IP addresses to MAC addresses.

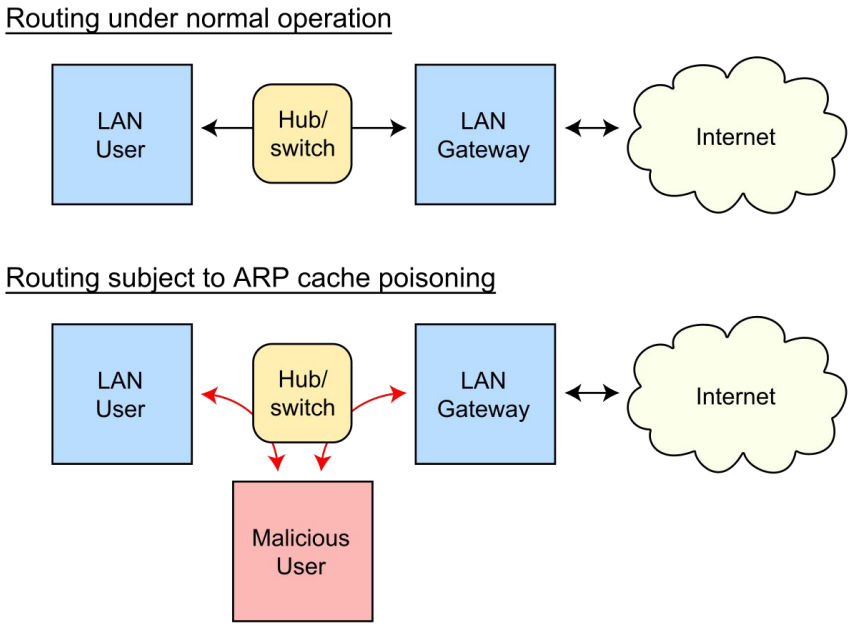
ARP: Address Resolution Protocol

•Problem: How does a host learn what MAC addresses to send packets to?

• ARP lets hosts build table mapping IP addresses to MAC Addresses.

• ARP request: source MAC, dest MAC, “Who has IP address N?”

• ARP reply: source MAC, dest MAC, “IP address N is at MAC address M.”



translation between IP addresses and MAC layer addresses

Routing: BGP (Border Gateway Protocol)

• Internet organized into ASes (Autonomous Systems) with peer, provider, or customer relationships between them

• Rough tree shape, with a small number of backbone ASes in a clique at the root

• BGP allows routers to exchange information about their routing tables

• Routers maintain global table of routes

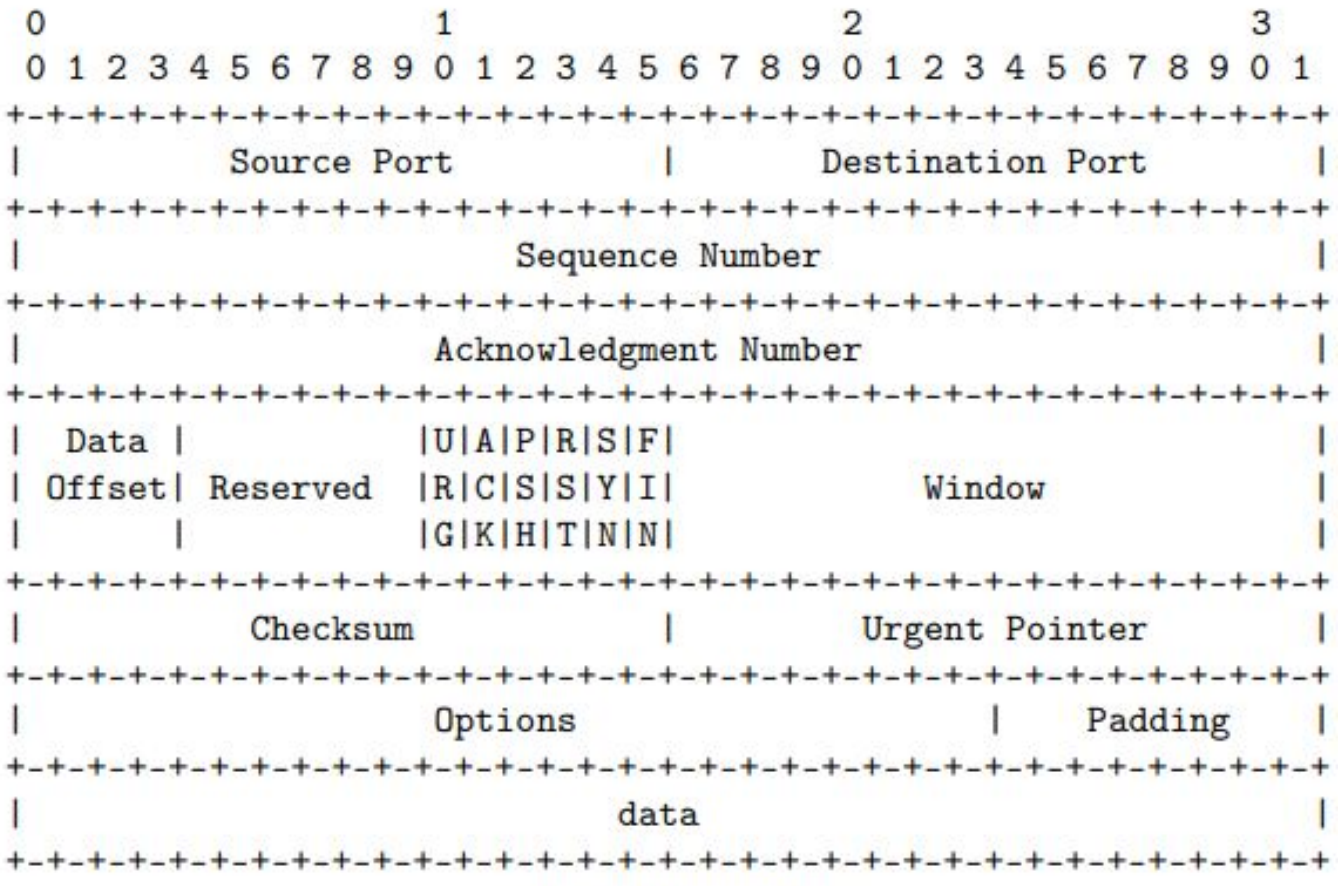
• Each router announces what it can route to its neighbors

• Routes propagate through network

TCP (Transmission Control Protocol)

- Want abstraction of a stream of bytes delivered reliably and in-order between applications on different hosts

- TCP provides:
  - Reliable in-order byte stream
  - Connection-oriented protocol
  - Explicit setup/teardown
  - End hosts (processes) have multiple concurrent long-lived dialogs
  - Congestion control: adapt to network path capacity, receiver's ability to receive packets



TCP Header Format

Ports

- Each application is identified by a port number
- TCP connection established between port A on host address M to port B on host address N. Ports are 16 bits, 1–65535
- Some destination ports are used for particular applications by convention

  80 HTTP (web)
  443 HTTPS (web)
  25 SMTP (mail)
  67 DHCP (host configuration)
  22 SSH (secure shell)
  23 telnet

TCP Sequence Numbers

- Bytes in application data stream numbered with 32-bit sequence number

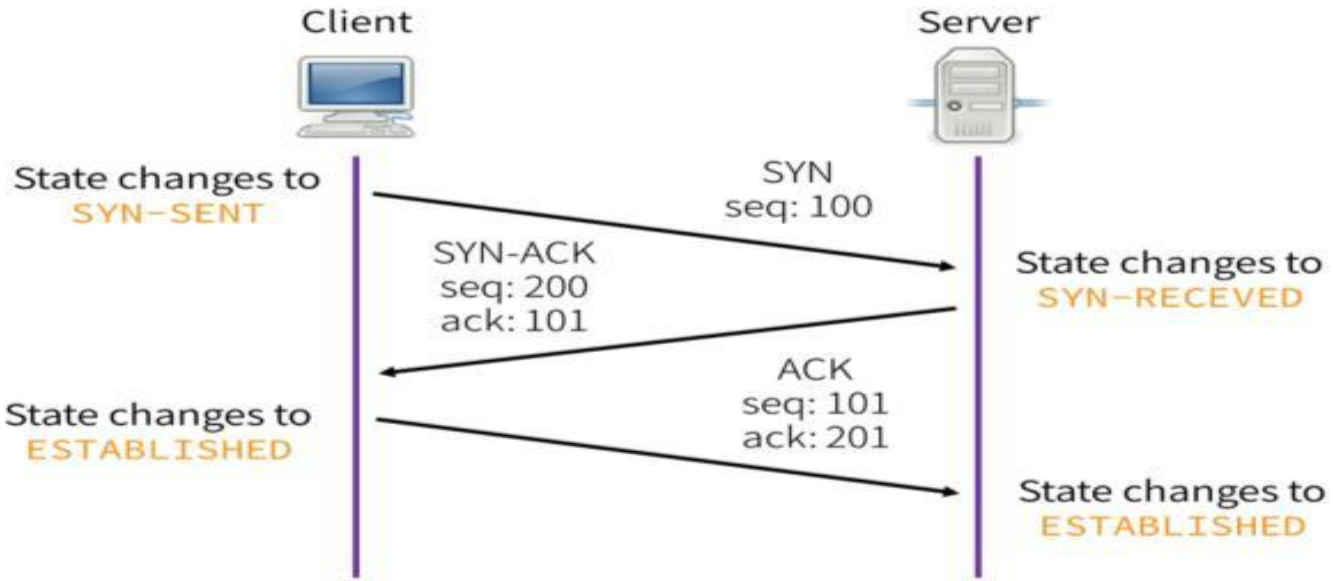
- Data sent in segments: sequences of contiguous bytes sent in a single IP datagram
- Sequence number indicates where data belongs in byte sequence
- Sequence number in packet header is the sequence number of the first byte in the payload

TCP Sequence Numbers and Acknowledgement

- Two logical data streams in a TCP connection: one in each direction
- Receiver acknowledges received data: acknowledgement number is sequence number of next expected byte of stream in opposite direction
- ACK flag set to acknowledge data
- Sender retransmits lost data
- Congestion control: sender adapts retransmission according to timeouts

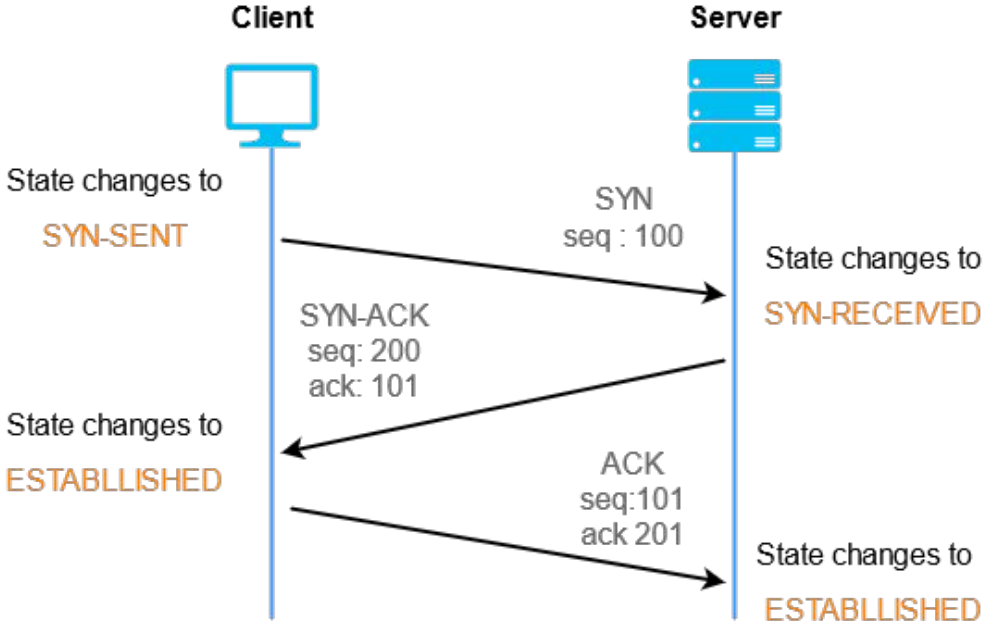
TCP 3-Way Handshake

Starting a TCP connection



TCP 3-Way Handshake

Starting a TCP connection



1. Client sends a SYN (open; “synchronize sequence numbers”) to Server
2. Server returns a SYN acknowledgment (SYN+ACK)
3. Client sends an ACK to acknowledge the SYN+ACK

FIN/RST: Closing TCP connections

- FIN initiates a clean close of a TCP connection, waits for ACK from receiver

FIN/RST: Closing TCP connections

- FIN initiates a clean close of a TCP connection, waits for ACK from receiver
- If a host receives a TCP packet with RST flag, it tears down the connection
- Designed to handle spurious TCP packets from previous connections

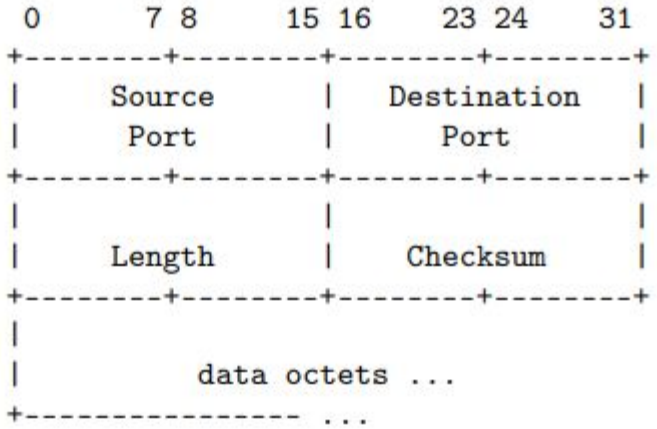
FIN/RST: Closing TCP connections

- UDP offers no service quality guarantee
- Essentially a transport layer protocol that is a wrapper around IP
- Adds ports to let applications demultiplex traffic
- Useful for applications that only need best-effort guarantee
- e.g. DNS, NTP

RFC 768

J. Postel
ISI
28 August 1980

User Datagram Protocol



User Datagram Header Format

Using the internet: A worked example

You connect your laptop to a cafe wifi network and type ucsd.edu into your browser's URL bar. What happens?

Using the internet: A worked example

1. Your laptop uses DHCP (Dynamic Host Configuration Protocol) to bootstrap itself on the local network.
    - New host has no IP address, doesn't know who to ask
    - Broadcasts DHCPDISCOVER to 255.255.255.255 with its MAC address
    - DHCP server responds with config: lease on host IP address, gateway IP address, DNS server information

Using the internet: A worked example

2. Your laptop makes an ARP request to learn the MAC address of the local router.

- Every connection outside the local network will be encapsulated in a link-layer frame with the local router's MAC address as the destination.
- Your laptop encapsulates each IP packet in a WiFi Ethernet frame addressed to the local router.
- The local router decapsulates these Ethernet frames and re-encodes them to forward them on its fiber connection to its upstream ISP, or to another part of the network.
- Each hop re-encodes the link layer for its own network

Using the internet: A worked example

3. Your laptop does a DNS lookup on ucsd.edu.
    - It learned the IP address of a local DNS server from DHCP, or had a server (like 9.9.9.9) already hard-coded.
    - Each request is a DNS query encapsulated in one or more UDP packets encapsulated in one or more IP packets.
    - Each response tells the laptop what authority to query, until it learns the final IP address (75.2.44.127) for ucsd.edu
    - This address is cached, along with the authorities for the hierarchy in the hostname

Using the internet: A worked example

4. Your laptop opens a TCP connection to 75.2.44.127.
    - Each packet of the TCP triple handshake is encoded in an IP packet that is encoded as Ethernet frames that are decoded and re-encoded as they pass through the network.
    - The local router has a routing table that contains IP prefixes that it matches against the IP address that tells it what address to forward the packets to.
    - The packet passes through a series of Autonomous Systems (ASes).
    - From cafe network (ATT), go through sbcglobal.net → att.net → level3.net → cenic.net → ucsd.edu.

Using the internet: A worked example

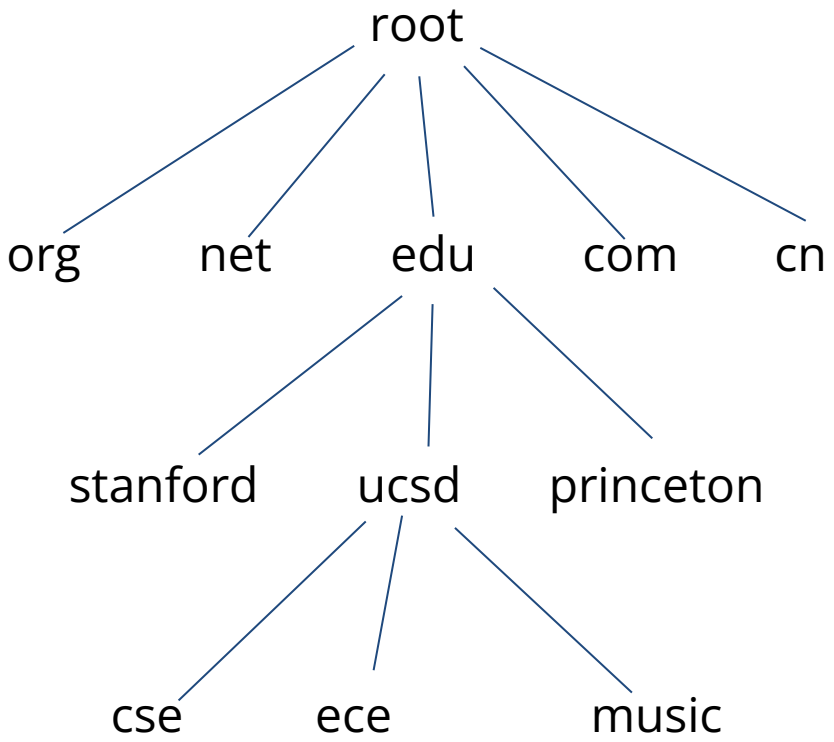
5. Your laptop sends a HTTP GET request inside the TCP Connection.

6. Based on the HTTP response, the laptop performs a new DNS lookup, TCP handshake, and HTTP GET requests for every resource in the HTML as it renders

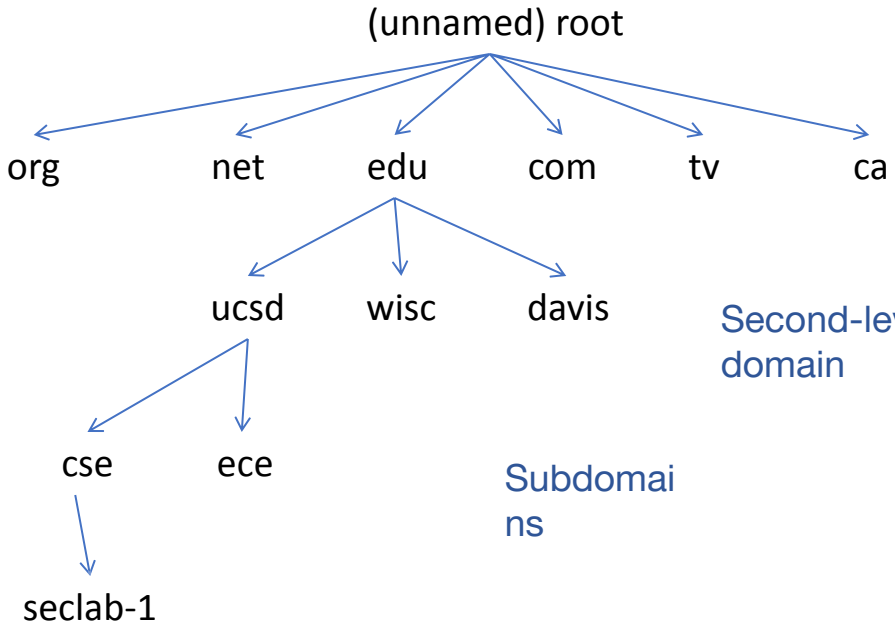
DNS (Domain Name Service)

- Handle mapping between host names (e.g. ucsd.edu) and IP addresses (e.g. 132.239.180.101)

- DNS is a delegatable, hierarchical name space



Hierarchical domain namespace



Top Level domains (TLD)

Second-level domain

Subdomains

Separated by '.'

FQDN: Fully qualified domain name

`seclab-1.cse.ucsd.edu`

Hostname Subdomain Domain TLD

`./edu/ucsd/cse/seclab-1`

max 63 characters

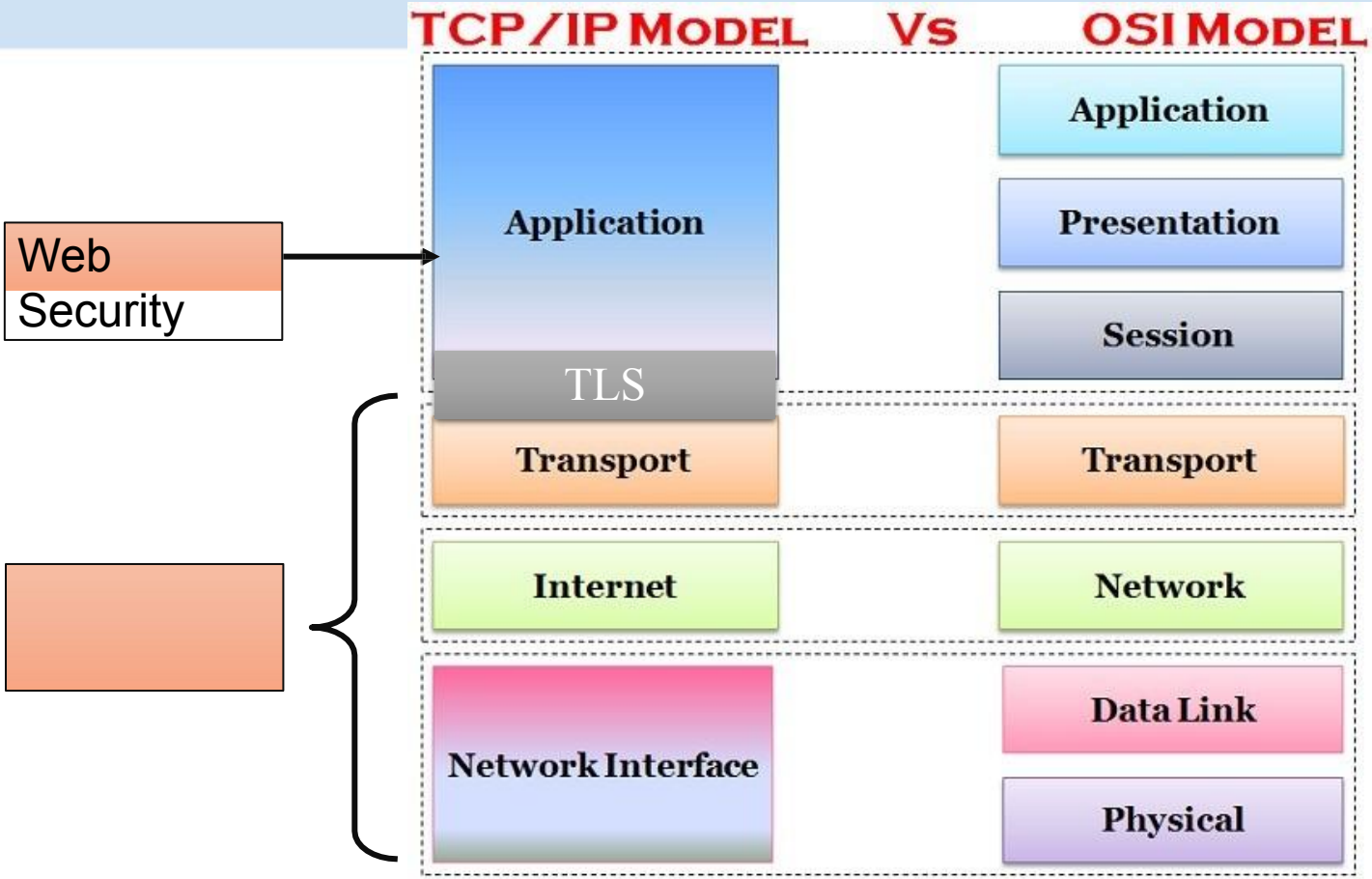
DNS Details

- 13 main DNS root servers
- DNS responses are cached for quicker responses
- DNS authorities queried progressively according to domain name hierarchy

```
$dig cseweb.ucsd.edu +trace

; <<>> DiG 9.10.6 <<>> cseweb.ucsd.edu +trace
;; global options: +cmd
. 105604 IN NS d.root-servers.net.
. 105604 IN NS h.root-servers.net.
. 105604 IN NS c.root-servers.net.
. 105604 IN NS j.root-servers.net.
...
. 105604 IN NS l.root-servers.net.
. 105604 IN NS i.root-servers.net.
. 105604 IN RRSIG NS 8 0 518400 20191115050000 20191102040000 22545 . Z14B+vD/MKz0X1UBwu04kzwQNajhg1AfIK7j5Jvd9NZac1HZ/M9xdSGN
F85s/5ITxEiWWeiBhRghy9PkdOmN3ZzhzS5E8ZeIibm0DdIse+qlPNas sfmNZEsIRbXEOER98eQ+Ieb0hjOlu7Y5l6Mo3dnuyE203IxXZTmtD9QH
zMRbX8gOrBnee1XYe7kxw+S2AN6BBeRHNFPHuT5nBCwWQlDVFao2ICrV 0oU97YJE7fwDNzyBgb89G++GjVKhQoM/0Bmr4D2vUAqCz7Nt9Gb28TOt
A+FpA6Ax9MpjZSCH8dOvz1nytjWfRMYyF5LVGEN6oPW6BKX2fWrfhIC4 TWIfWA==
;; Received 525 bytes from 192.168.1.254#53(192.168.1.254) in 44 ms
edu. 172800 IN NS b.edu-servers.net.
edu. 172800 IN NS f.edu-servers.net.
edu. 172800 IN NS i.edu-servers.net.
...
edu. 172800 IN NS c.edu-servers.net.
edu. 172800 IN NS e.edu-servers.net.
edu. 172800 IN NS d.edu-servers.net.
edu. 86400 IN DS 28065 8 2 4172496CDE85534E51129040355BD04B1FCFEBAE996DFDDE652006F6 F8B2CE76
edu. 86400 IN RRSIG DS 8 1 86400 20191116170000 20191103160000 22545 . BsoO9Wl4UphacN5rL0B4f3bCzVPptbmTCKHwcMgb6edhjhEbeH4YDzDd
HFdr0hQQLSCPdLZ6TyOITD53FRf8y/drtaJqsdsmuySOwC+woN3pDuUj aTm/wpohn8TP3eIYg0V8y+wTlPf7RpHP1K4tX4ug3SO905Cw6n1pkedL
3Il5FShKovBMMWlsnK+fh20IvErYJQ4L98CrGrt1k4Ch7EsxPsTrUcFy bxhTw63LbLGnClNFJNvM+GhS6x4jHMFVGnZdwCJinD/UgV5VjTNbzPzC
45JW2xP/B7bI1zmOZsyYEeRXnM6nK0KKCH5tAsDlJNVfJhPFKZ+3Iqm3 nlfU2A==
;; Received 1174 bytes from 192.58.128.30#53(j.root-servers.net) in 20 ms
ucsd.edu. 172800 IN NS ns-auth2.ucsd.edu.
ucsd.edu. 172800 IN NS ns-auth3.ucsd.edu.
9DHS4EP5G85PF9NUFK06HEK0O48QGK77.edu. 86400 IN NSEC3 1 1 0 - 9V5L4LUB1VNJ9EQQLIHEQCBREACL25O0 NS SOA RRSIG DNSKEY NSEC3PARAM
9DHS4EP5G85PF9NUFK06HEK0O48QGK77.edu. 86400 IN RRSIG NSEC3 8 2 86400 20191111043435 20191104032435 47252 edu.
M5VYkUSvz94kzGxoiSTurXi0HcguXZ9mBTgYa/LcYh/UwZazqyFFPQja yBDpiwbKLmVHkB/OOw0oEBjyfPJ05nJ6uS80/xw+RYncNMVLgUM3EZgR
16h4X0Sjfc/vgOZYrqqxiN7KZmnpSOmb1eCue7dItqDc78DmE2Xrs3wM 8+xlar+xcKR45TzUPlz8eRes0bs47F2Ern3/FTnnlJOAkw==
3FTB9RSLROQJUOPDNLJJE2I31U25M4MG.edu. 86400 IN NSEC3 1 1 0 - 4586U2HHMPSEAQHJD6R9INNA38POF8KL NS DS RRSIG
3FTB9RSLROQJUOPDNLJJE2I31U25M4MG.edu. 86400 IN RRSIG NSEC3 8 2 86400 20191111041950 20191104030950 47252 edu.
BKveV5lagKfQxbNb2hd96O89QU+/Z8PE0FCsRbnJvaAPIucvPCOgUrxJ iimvslQ4a4bkS3dEWBbxfB3t4a7EKRP8n3ZohVK8xm/ehFbYdeSNweEK
lwLcr2wp2ddWRY+mX0H6uhNrRoeFSbLiHqiO9qzquyVc6OC+l49VcjLR lj9FCupdY7WvPc30DYpOdgf/C43/aKW7ZgNSMi2NCrAi1A==
;; Received 671 bytes from 192.41.162.30#53(l.edu-servers.net) in 9 ms
cseweb.ucsd.edu. 3600 IN CNAME roweb.eng.ucsd.edu.
roweb.eng.ucsd.edu. 3600 IN A 132.239.8.30
```

DNS and BGP


TCP/IP Model
Vs
OSI Model
Application
Application
Presentation
Session
TLS
Transport
Transport
Internet
Network
Network Interface
Data Link
Physical
Web
Security

128.105.37.141

We don’t want to remember IP addresses

```
earlence@earlence-surface3:/mnt/c/Users/earle$ nslookup www.earlence.com
Server:         128.104.254.254
Address:        128.104.254.254#53

Non-authoritative answer:
www.earlence.com        canonical name = earlence-uwm.github.io.
Name:   earlence-uwm.github.io
Address: 185.199.109.153
```

128.105.37.141

We don’t want to have to remember IP addresses

```
user@box:~$ cat /etc/hosts
127.0.0.1 localhost
127.0.1.1 box.localdomain
          box
127.0.0.1 zoobar.org
127.0.0.1 www.zoobar.org
127.0.0.1 zoomail.org
# The following lines are desirable for IPv6 capable
hosts
::1 localhost ip6-localhost
ip6-loopback  fe00::0 ip6-localnet
ff00::0
ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Early days of ARPANET: manually managed

Internet-wide namespace

- ICANN (Internet Corporation for Assigned Names and Numbers)

- DNS Servers
  - DNS resolver
  - root nameservers – 13 of them worldwide A through M
  - authoritative nameservers – authorized to provide IP for a (sub)domain / hostname

- **Zone**: a contiguous portion of domain namespace
  - A subtree

```
A.ROOT-SERVERS.NET. IN A 198.41.0.4
B.ROOT-SERVERS.NET. IN A
192.228.79.201  C.ROOT-SERVERS.NET.
IN A 192.33.4.12
...
M.ROOT-SERVERS.NET. IN A 202.12.27.33
```

.NET referrals

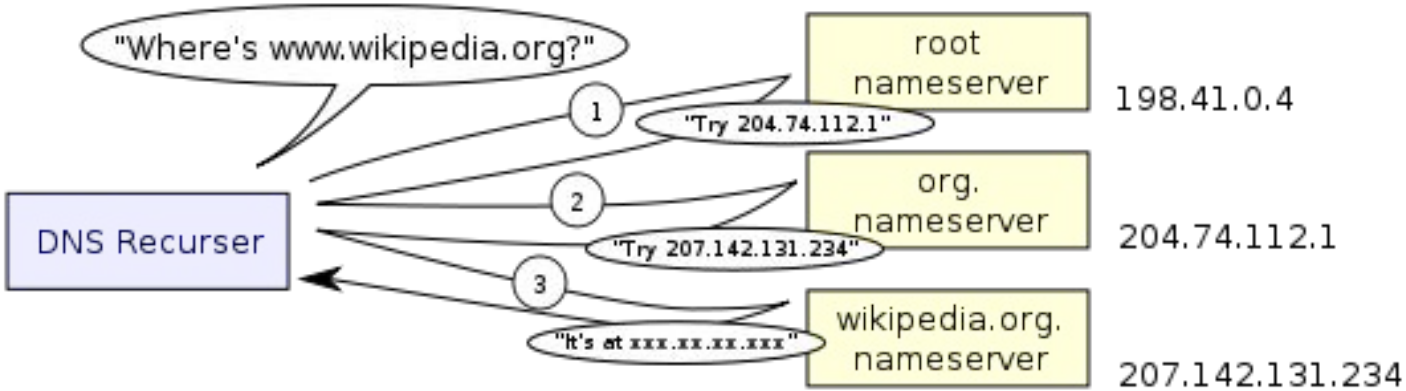
```
/* Authority section */
NET.                    IN  NS A.GTLD-SERVERS.NET.
                        IN  NS B.GTLD-SERVERS.NET.
                        IN  NS C.GTLD-SERVERS.NET.
                        ...
                        IN  NS M.GTLD-SERVERS.NET.

/* Additional section - "glue" records */
A.GTLD-SERVERS.net.     IN  A   192.5.6.30
B.GTLD-SERVERS.net.     IN  A   192.33.14.30
C.GTLD-SERVERS.net.     IN  A   192.26.92.30
...
M.GTLD-SERVERS.net.     IN  A   192.55.83.30
```

https://www.iana.org/domains/root/servers

HOSTNAME	IP ADDRESSES	OPERATOR
a.root-servers.net	198.41.0.4, 2001:503:ba3e::2:30	Verisign, Inc.
b.root-servers.net	199.9.14.201, 2001:500:200::b	University of Southern California, Information Sciences Institute
c.root-servers.net	192.33.4.12, 2001:500:2::c	Cogent Communications
d.root-servers.net	199.7.91.13, 2001:500:2d::d	University of Maryland
e.root-servers.net	192.203.230.10, 2001:500:a8::e	NASA (Ames Research Center)
f.root-servers.net	192.5.5.241, 2001:500:2f::f	Internet Systems Consortium, Inc.
g.root-servers.net	192.112.36.4, 2001:500:12::d0d	US Department of Defense (NIC)
h.root-servers.net	198.97.190.53, 2001:500:1::53	US Army (Research Lab)
i.root-servers.net	192.36.148.17, 2001:7fe::53	Netnod
j.root-servers.net	192.58.128.30, 2001:503:c27::2:30	Verisign, Inc.
k.root-servers.net	193.0.14.129, 2001:7fd::1	RIPE NCC
l.root-servers.net	199.7.83.42, 2001:500:9f::42	ICANN
m.root-servers.net	202.12.27.33, 2001:dc3::35	WIDE Project

Resolving names



From:
http://en.wikipedia.org/wiki/File:An_example_of_theoretical_DNS_recursion.svg

Example DNS record (and query) types

A	Address mapping record (get me an IPv4 address)
AAAA	Same for IPv6 address
NS	name server, the DNS zone
TXT	machine readable text data, has been used for many things, including encryption mechanisms, policy
MX	mail exchange (SMTP mail server for the domain)
CNAME	Canonical name, alias of a domain

DNS Records

```
$ dig cseweb.ucsd.edu

; <<>> DiG 9.10.6 <<>> cseweb.ucsd.edu
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 3727
;; flags: qr rd ra ; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

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

;; ANSWER SECTION:
cseweb.ucsd.edu. 3140 IN CNAME roweb.eng.ucsd.edu.
roweb.eng.ucsd.edu. 2855 IN A 132.239.8.30

;; Query time: 57 msec
;; SERVER: 192.168.1.254#53(192.168.1.254) ;
; WHEN: Sun Nov 03 20:49:08 PST 2019
;; MSG SIZE rcvd: 84
```

Caching

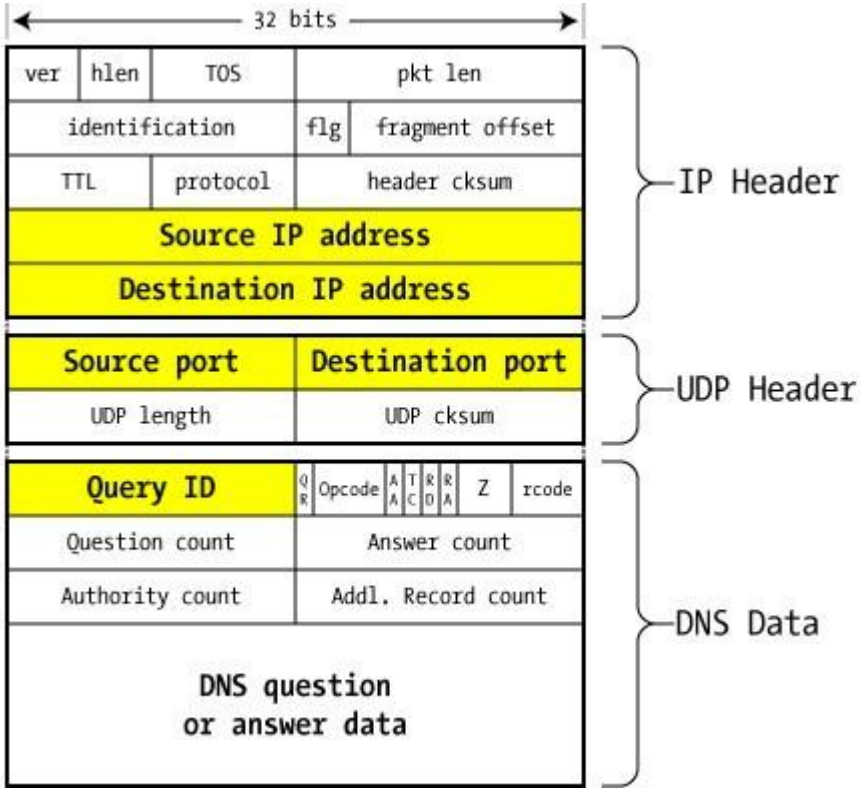
- DNS servers will cache responses
  - Both negative and positive responses
  - Speeds up queries
  - periodically times out. TTL set by data owner

DNS packet on wire

We'll walk through the example from Friedl's document (on Website)

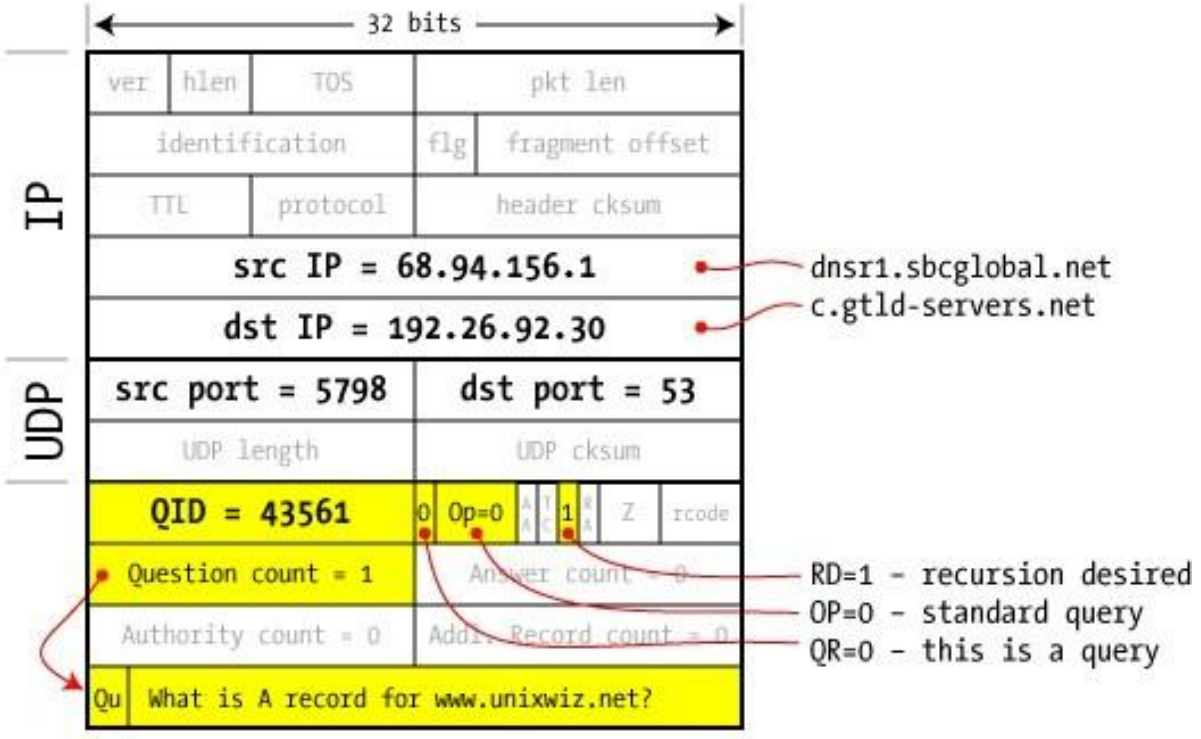
www.unixwiz.net

Query ID is 16-bit value

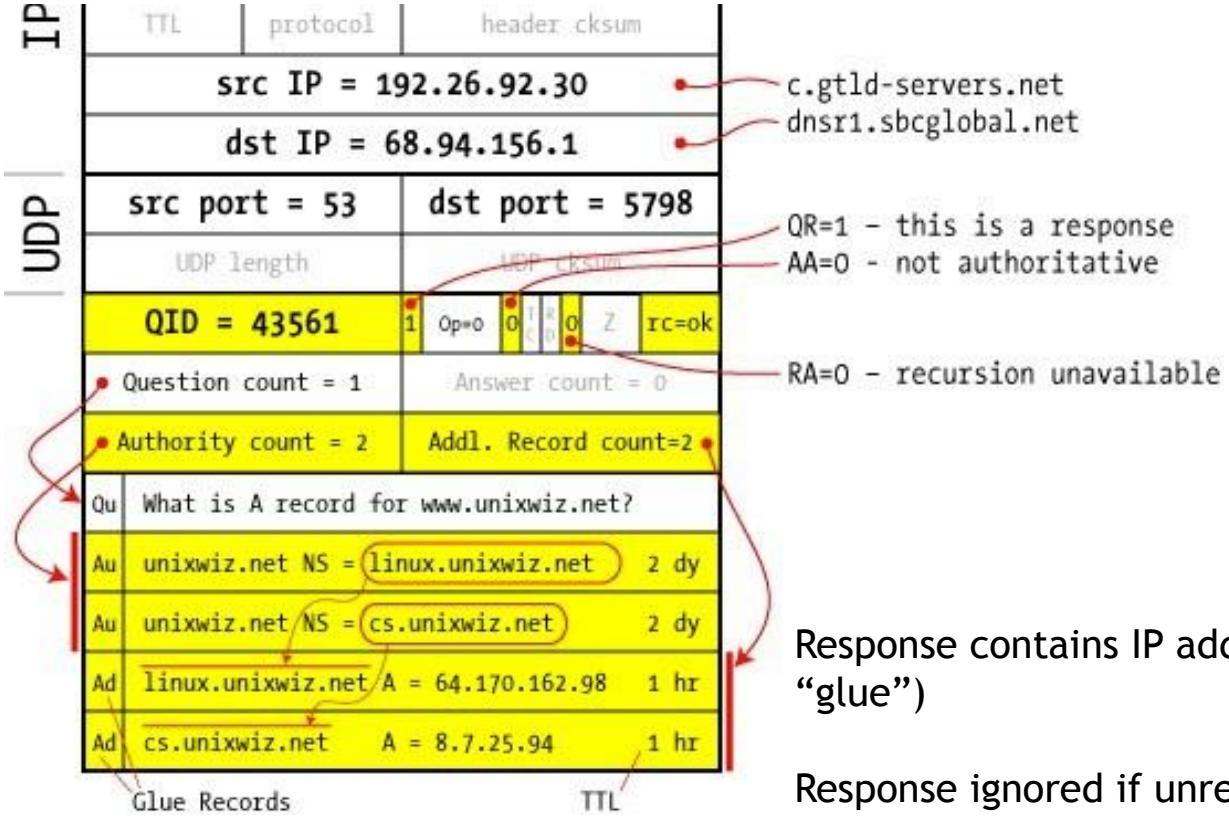


From Friedl explanation of DNS cache poisoning, as are following diagrams

Query from resolver to NS



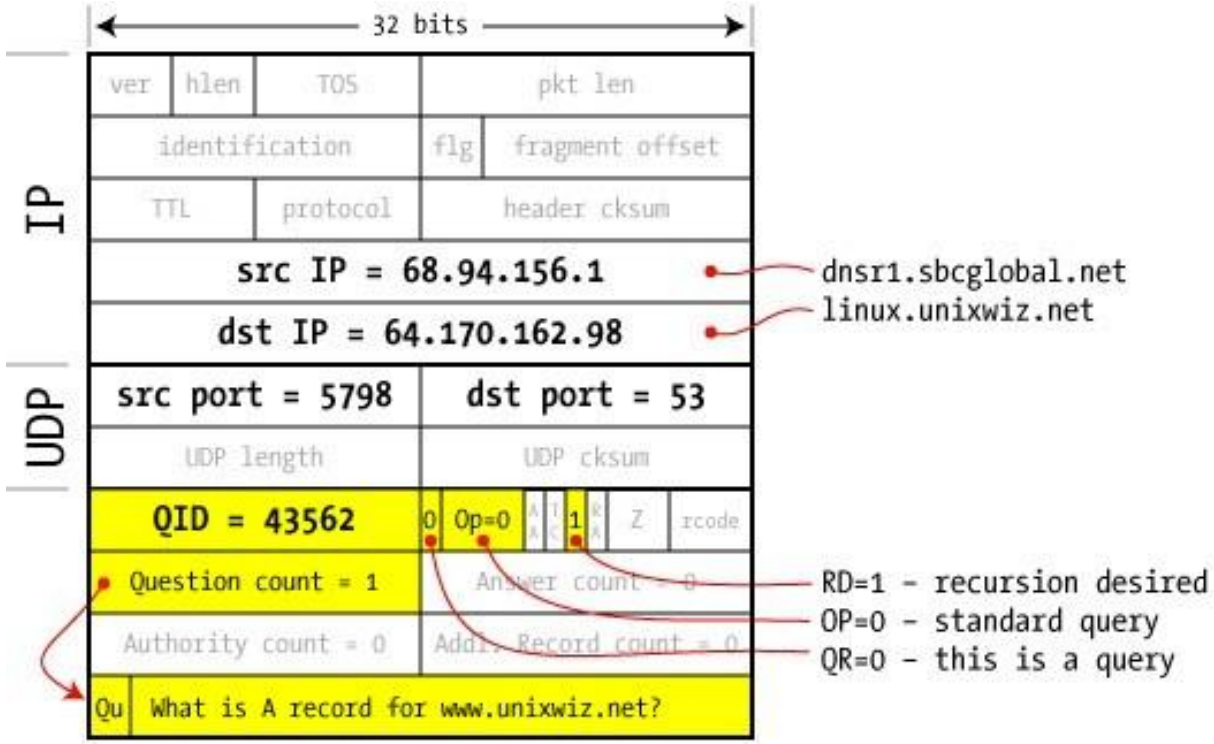
Reply from NS to Resolver



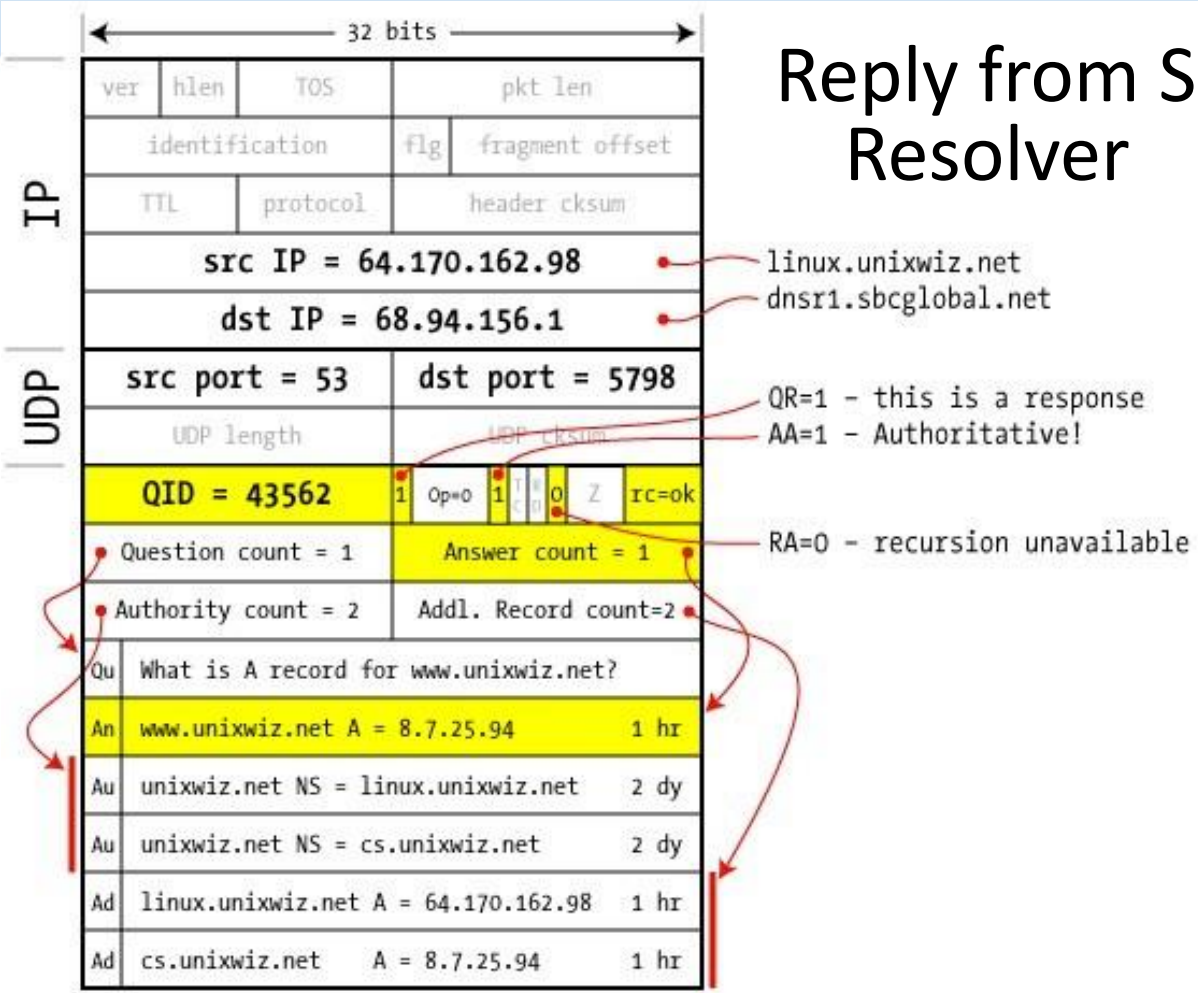
Response contains IP addr of next NS server (called "glue")

Response ignored if unrecognized QueryID

Query to Second NS



Reply from Second NS to Resolver



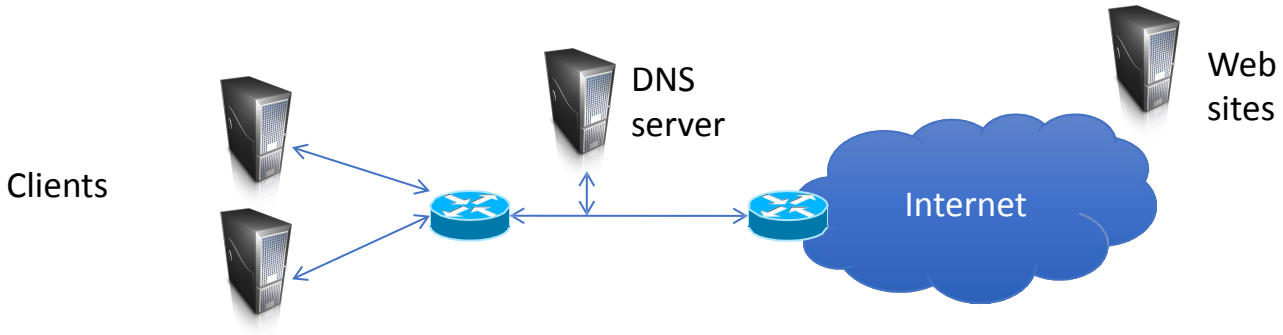
Caching is the key

- DNS servers are queried trillions of times, though they seem fast, doing it again and again could
  - burden the network
  - slowdown everything

- Therefore, authoritative responses are cached for limited amount of time
  - Both **NS** and **A** records are cached
  - TTL – how long to keep the DNS record in cache

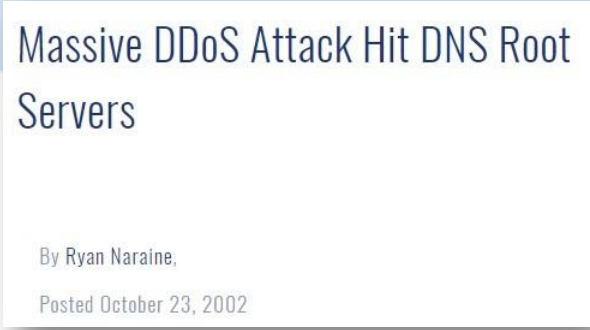
- bailiwick checking response is cached if it is within the same domain of query
  - i.e. `ns.unixwhiz.net cannot reply with fraudulent info about google.com`

Attacks against DNS?



- Corrupted nameservers
- Intercept & manipulate requests
- DDoS
- Cache poisoning
- Phishing / typo squatting / piggy-backing

DDoS against DNS


Massive DDoS Attack Hit DNS Root Servers

By Ryan Naraine,

Posted October 23, 2002


Data Centre ▸ Networks

Internet's root servers take hit in DDoS attack

Who's testing the limits of the DNS system?

By Kieren McCarthy in San Francisco 8 Dec 2015 at 23:10 27 SHARE ▼

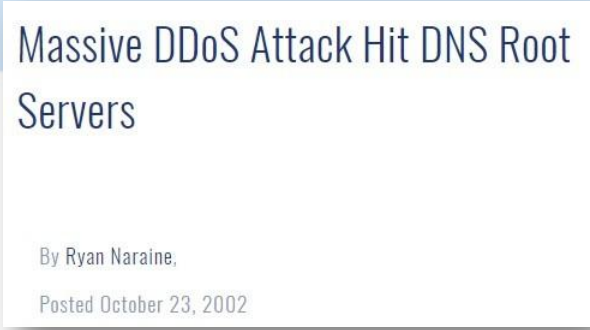
- Denial of Service
  - attacker leverages the functionality of open DNS resolvers in order to overwhelm a target server or network with an amplified amount of traffic, rendering the server and its surrounding infrastructure inaccessible (cloudflare)
  - exploit a disparity in bandwidth consumption (cloudflare)
- DoD purportedly has interesting response:

"In the event of a massive cyberattack against the country that was perceived as originating from a foreign source, the United States would consider launching a ***counterattack or bombing the source of the cyberattack,*** *Hall said. But he noted the preferred route would be warning the source to shut down the attack before a military response."*

http://www.computerworld.com/s/article/9010921/RSA_U.S._cyber_counterattack_Bomb_one_way_or_the_other

DDoS against DNS

- Denial of Service
  - take down DNS server, clients can't use Internet
  - Attack against root servers:


Massive DDoS Attack Hit DNS Root Servers

By Ryan Naraine,

Posted October 23, 2002


Data Centre ▸ Networks

Internet's root servers take hit in DDoS attack

Who's testing the limits of the DNS system?

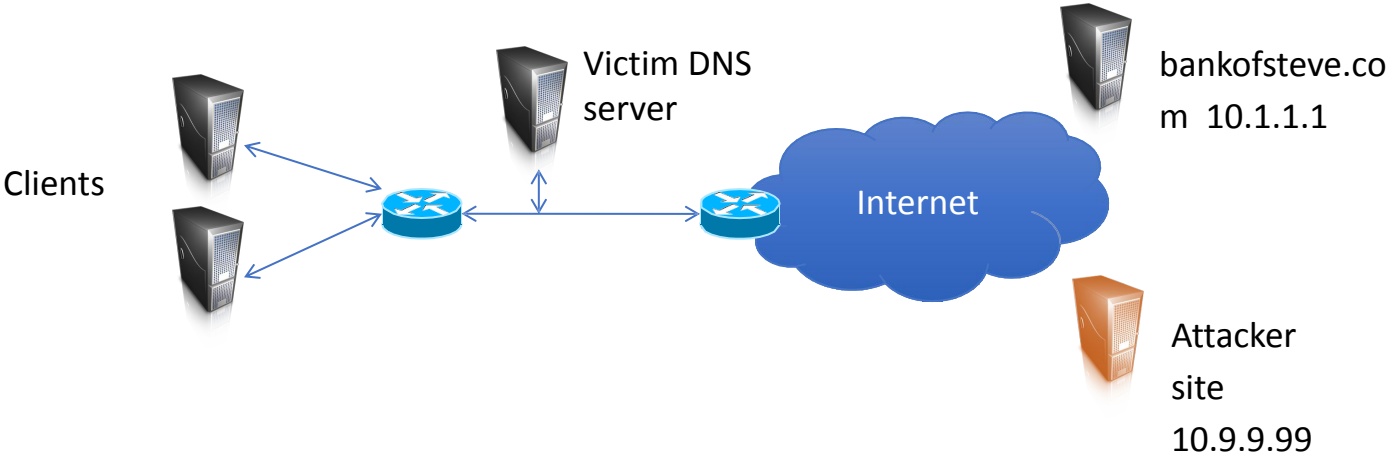
By Kieren McCarthy in San Francisco 8 Dec 2015 at 23:10 27 SHARE ▼

- DoD purportedly has interesting response:

"In the event of a massive cyberattack against the country that was perceived as originating from a foreign source, the United States would consider launching a ***counterattack or bombing the source of the cyberattack****, Hall said. But he noted the preferred route would be warning the source to shut down the attack before a military response."*

http://www.computerworld.com/s/article/9010921/RSA_U.S._cyber_counterattack_Bomb_one_way_or_the_other

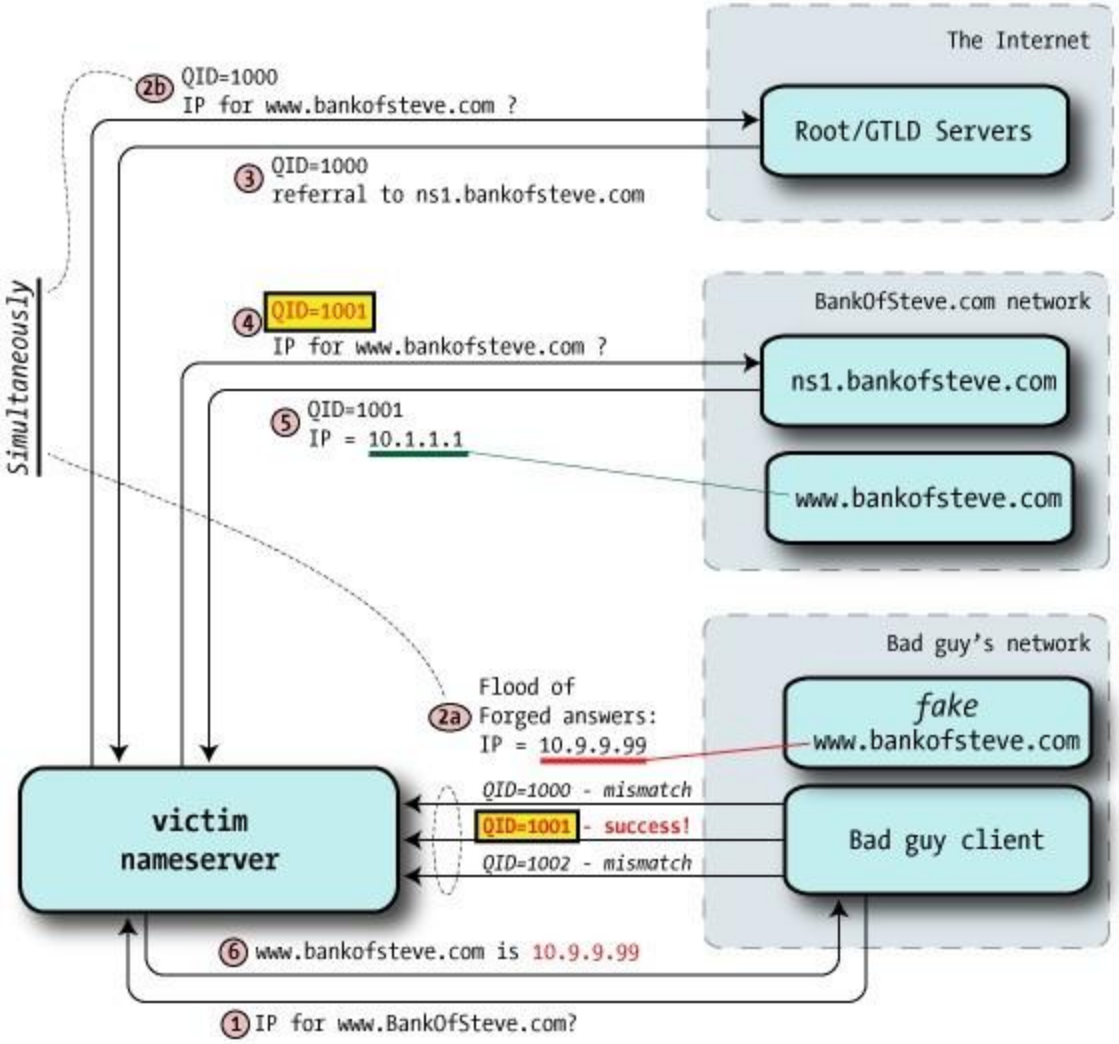
DNS cache poisoning



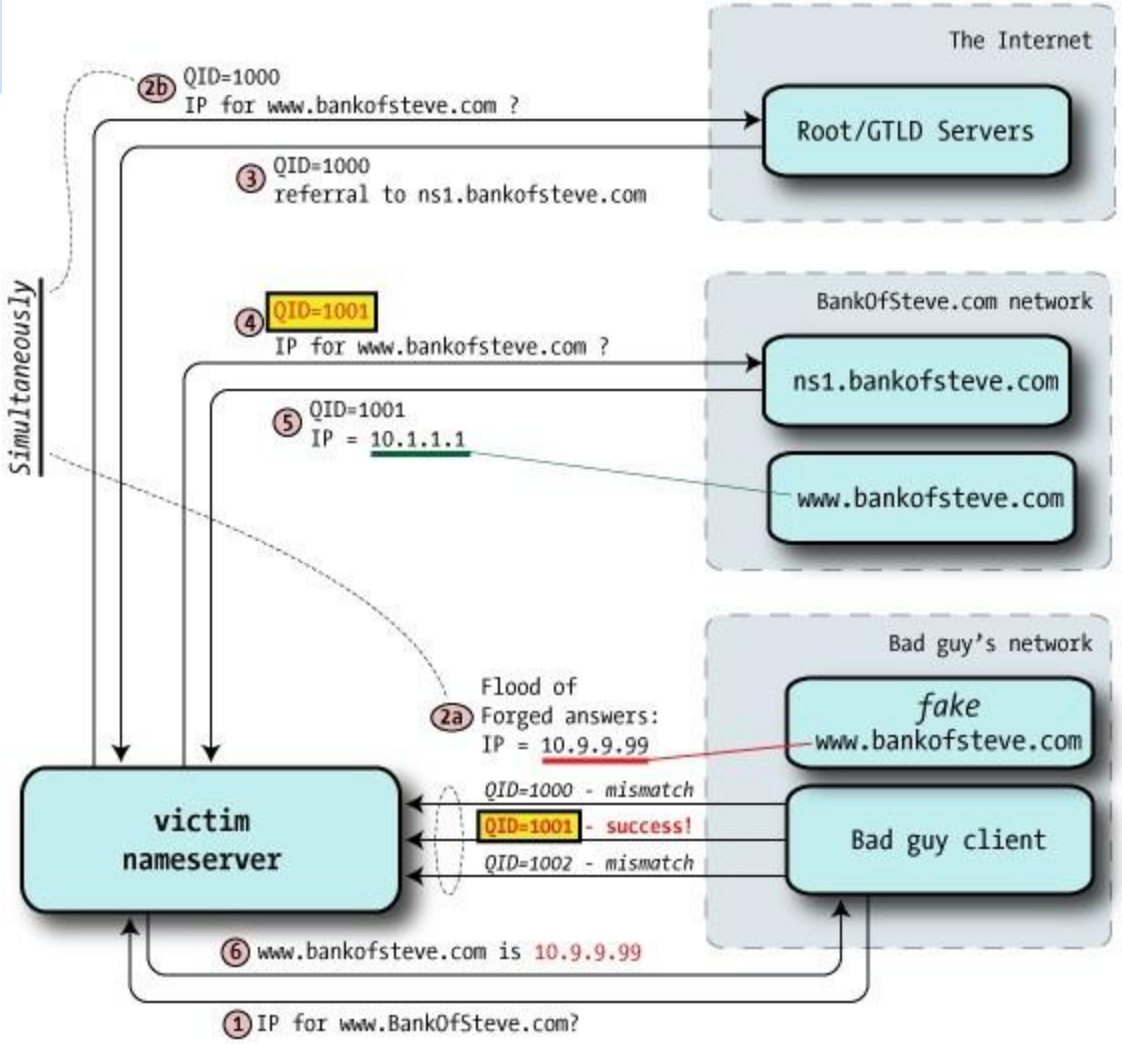
How might an attacker do this?

Assume DNS server uses predictable UDP port

Race with the real NS



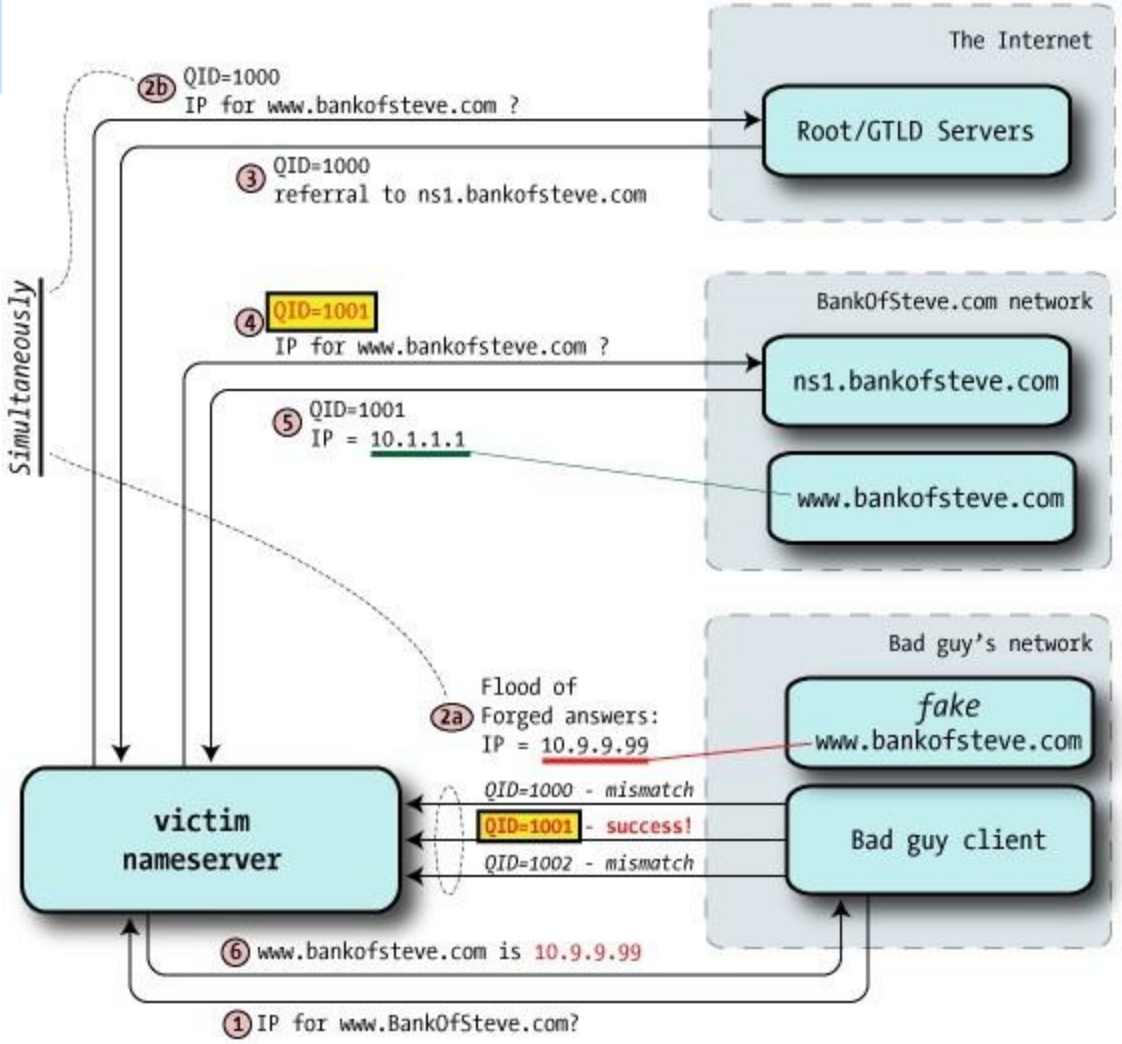
Race with the real NS



Step 1 — bad guy client requests a random name within the target domain (www12345678.bankofsteve.com), something unlikely to be in cache even if other lookups for this domain have been done recently.

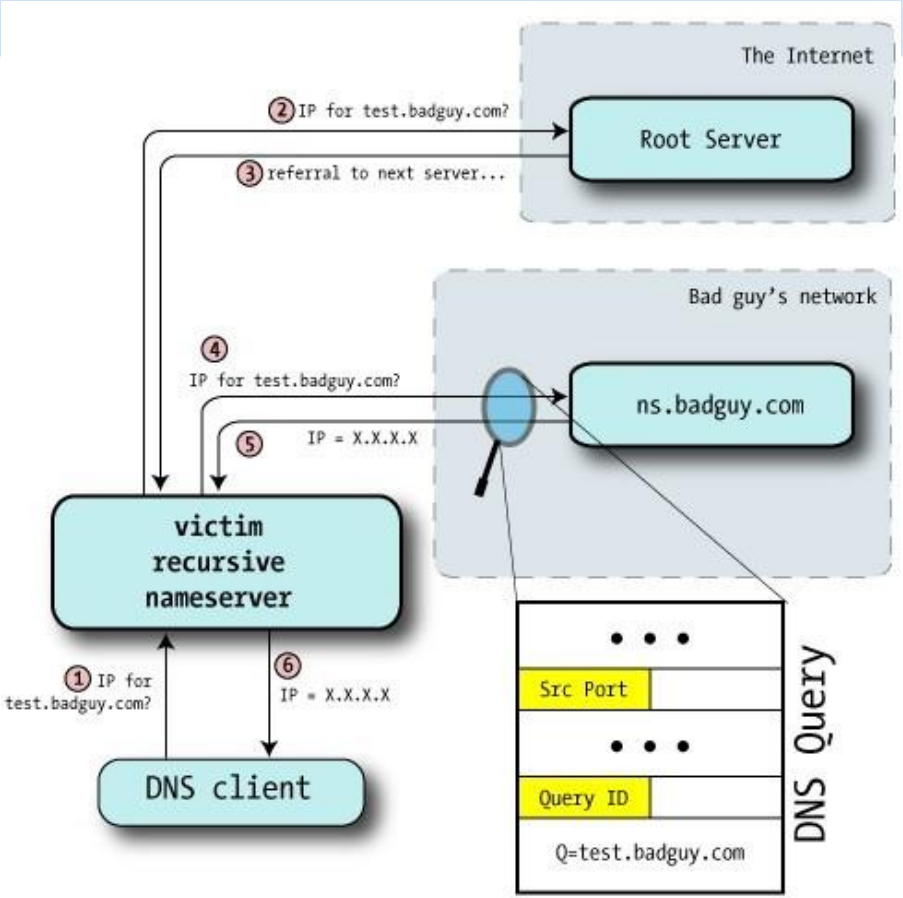
Step 2a — As before, the bad guy sends a stream of forged packets to the victim, but instead of A records as part of an Answer, it instead delegates to another nameserver via Authority records. "I don't know the answer, but you can ask over there".

Race with the real NS



Step 1 — bad guy client requests a random name within the target domain (www12345678.bankofsteve.com), something unlikely to be in cache even if other lookups for this domain have been done recently.

Step 2a — As before, the bad guy sends a stream of forged packets to the victim, but instead of A records as part of an Answer, it instead delegates to another nameserver via Authority records. "I don't know the answer, but you can ask over there".



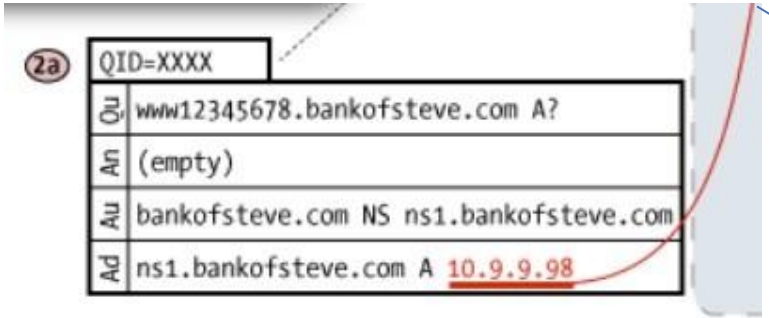
How to predict the query ID and source port?

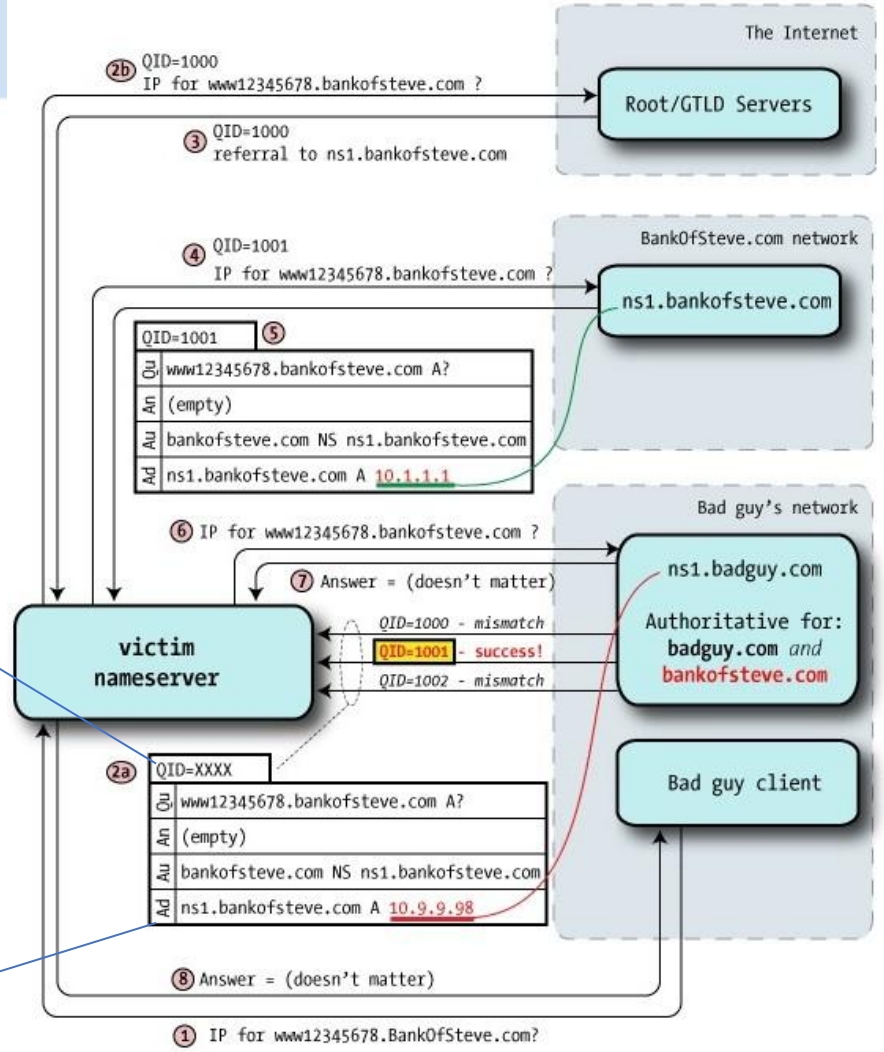
Another idea (Dan Kaminsky's attack):

- Poison cache for NS record instead
- Now can take over all of second level domain

How many tries does this require?

- 16 bit query id field
- If choosing randomly: 256 (birthday)
- If predictable, choose in range



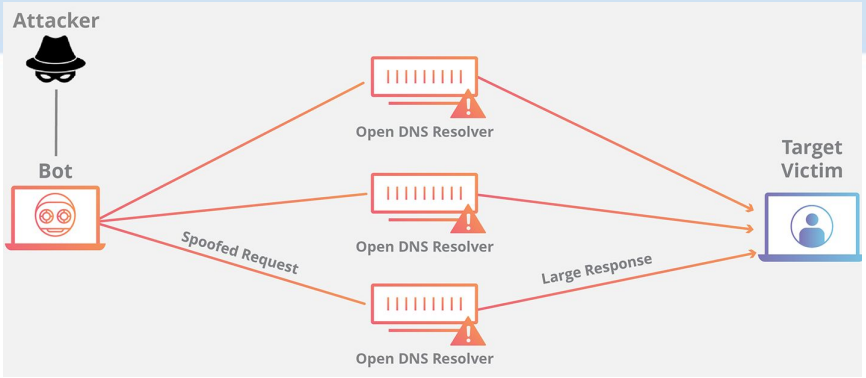


Defenses (and attacks)

- Query ID size is fixed at 16 bits
- Repeat each query with fresh Query ID
  - (randomize)
- Randomize UDP ports: not enough randomness in query ID only
- DNSsec
  - Cryptographically sign DNS responses, verify via chain of trust from roots on down

… but DNSSec vulnerable to DDoS

- Create large amount traffic from the DNS resolvers to the victim computer/server



Help Net Security
September 18, 2019

Share

DNSSEC fueling new wave of DNS amplification attacks

DNS amplification attacks swelled in the second quarter of this year, with the amplified attacks spiking more than 1,000% compared with Q2 2018, according to Nexusguard.

Does happen in the wild

HD Moore pwned with his own DNS exploit, vulnerable AT&T DNS servers to blame

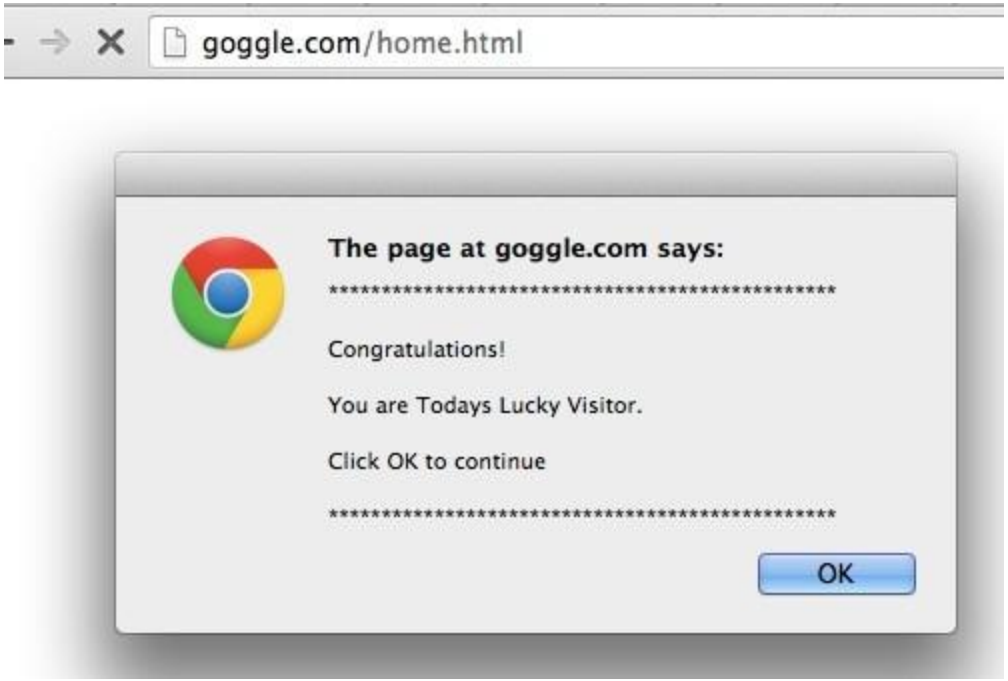
By Dancho Danchev | July 30, 2008, 8:08am PDT

Summary: *A week after)ruid and HD Moore release part 2 of DNS exploit, HD Moore's company BreakingPoint has suffered a traffic redirection to a rogue Google site, thanks to the already poisoned cache at AT&T servers to which his company was forwarding DNS traffic : "It happened on Tuesday morning, when Moore's company, BreakingPoint had some [...]*

http://www.zdnet.com/blog/security/hd-moore-pwned-with-his-own-dns-exploit- vulnerable-at-t-dns-servers-to-blame/1608?tag=content;siu-container

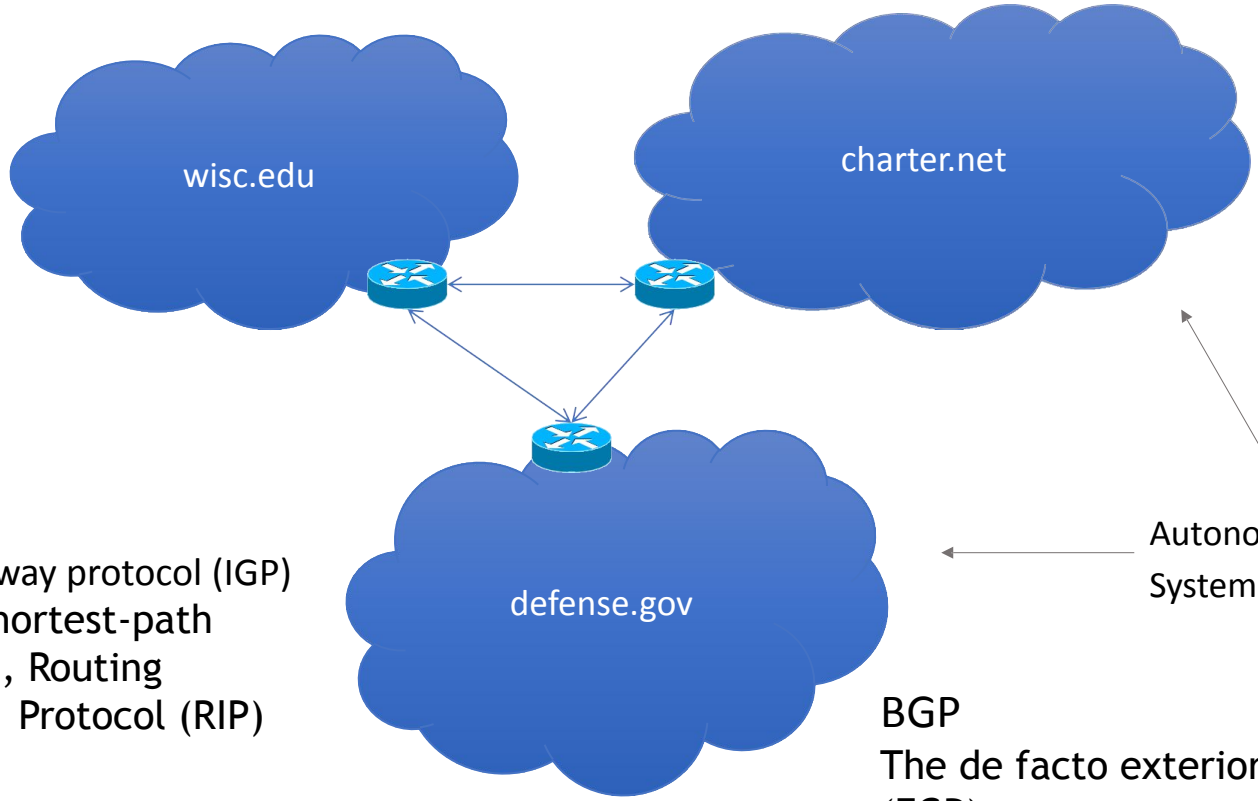
Phishing is common problem

- Typo squatting:
  - www.qpple.com
  - www.goggle.com
  - www.nytmes.com
- Other shenanigans:
  - www.badguy.com/(256 characters of filler)/www.google.com
- Phishing attacks
  - These just trick users into thinking a malicious domain name is the real one


goggle.com/home.html
The page at goggle.com says:
****
Congratulations!
You are Todays Lucky Visitor.
Click OK to continue
****
OK

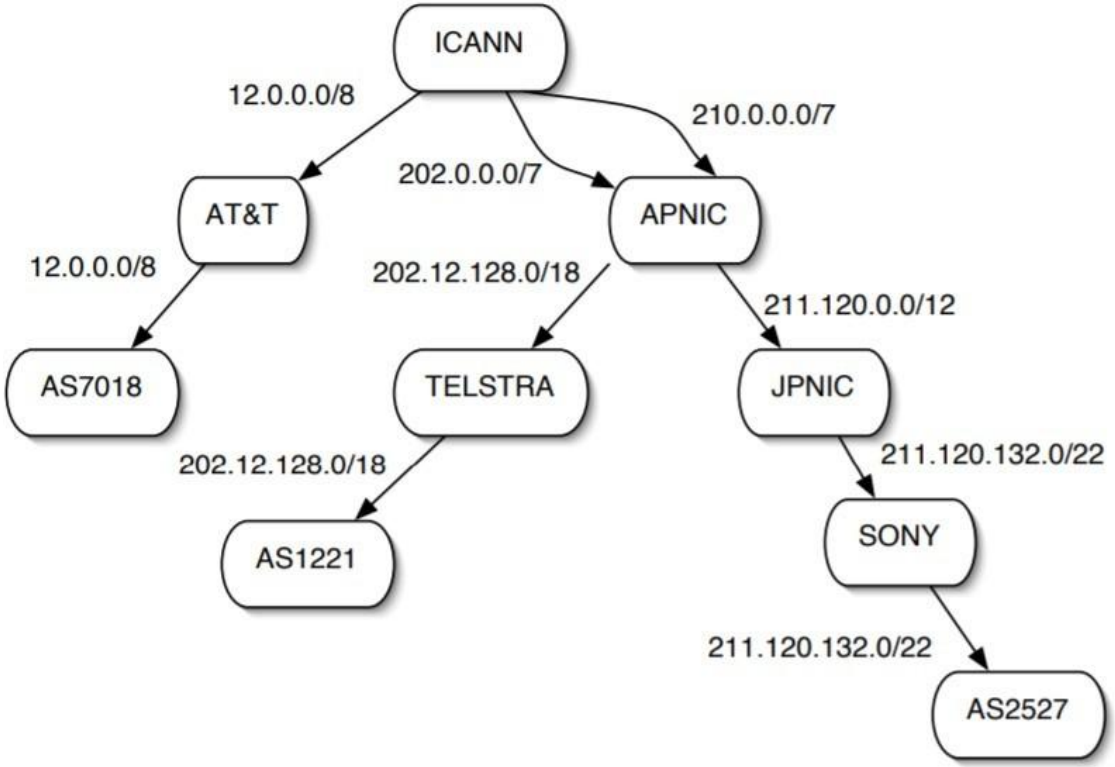

WARNING!
YOUR COMPUTER MAY BE INFECTED:
System detected (2) Potentially Malicious Viruses.
The data on your computer is NOT SAFE!
Your Personal & Financial Information IS NOT SAFE
To Remove Viruses, Call Tech Support Now:
855-521-0242
(24/7 - Toll free- High Priority Virus & Spyware Removal Call Line for Your IP Address: 128.105.35.160)
Clean Now!

BGP and routing



Interior Gateway protocol (IGP)
E.g, Open shortest-path first (OSPF), Routing Information Protocol (RIP)

BGP
The de facto exterior gateway protocol (EGP)



Source:
http://patrickmcdaniel.org/pubs/td-5ugj33.pdf

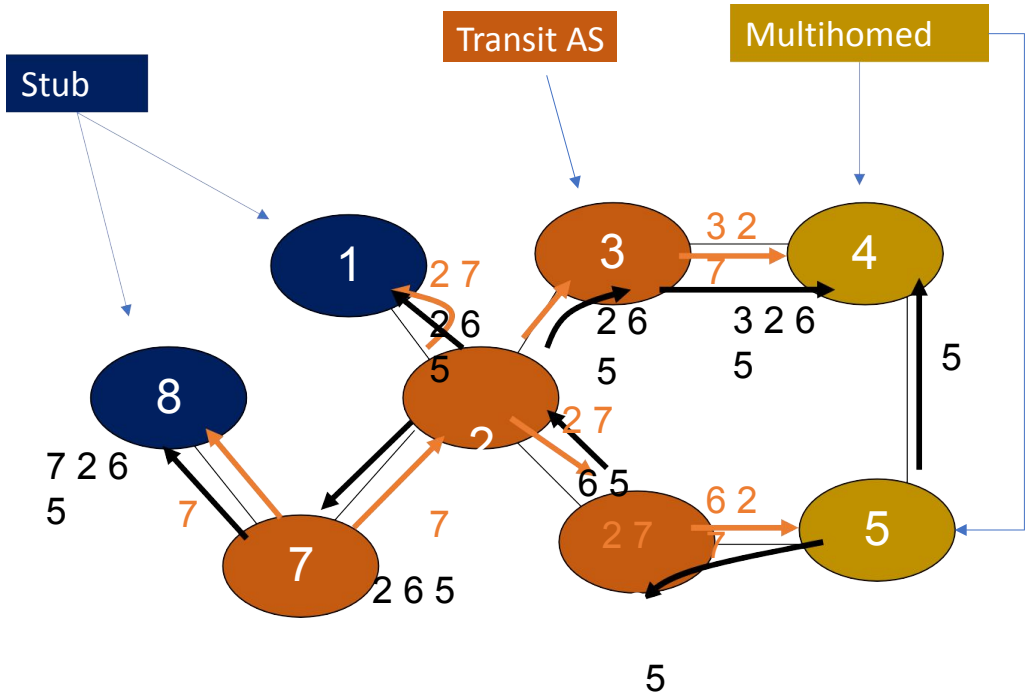
BGP

- Policy-based routing
  - AS can set policy about how to route
    - economic, security, political considerations
- BGP routers use TCP connections to transmit routing information
- Iterative announcement of routes

BGP example

[D. Wetherall]

- Algorithm seems to work OK in practice
  - BGP does not respond well to frequent node outages



IP hijacking

- BGP is unauthenticated
  - Anyone can advertise any routes
  - False routes will be propagated
- This allows IP hijacking
  - AS announces it originates a prefix it shouldn’t
  - AS announces it has shorter path to a prefix
  - AS announces more specific prefix

Malicious or misconfigurations?

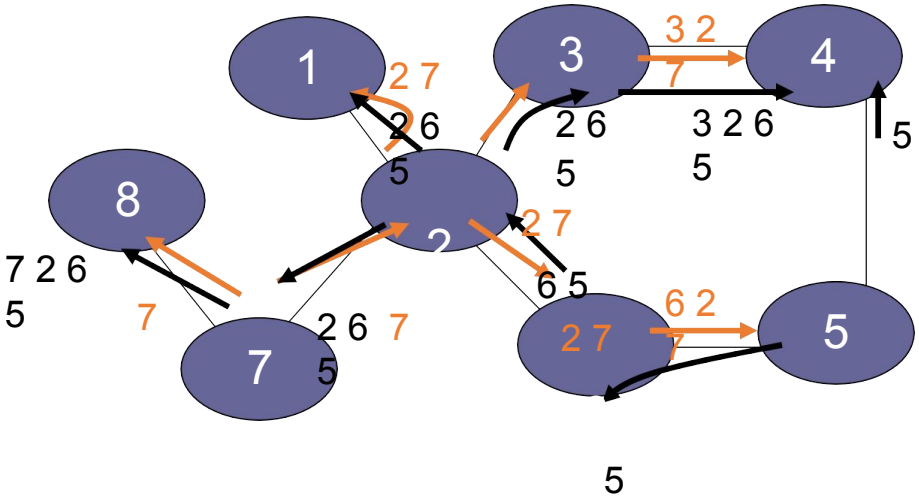
- AS 7007 incident in 1997
  - “Okay, so panic ensued, and we unplugged *everything* at 12:15PM almost to the second.” [sic]
  - http://www.merit.edu/mail.archives/nanog/1997-04/msg00444.html
- China Telecom hijacks large chunks of Internet in 2010
  - http://bgpmon.net/blog/?p=282

https://www.bgpmon.net

YouTube incident (2008)

- Pakistan attempts to block Youtube
  - youtube is 208.65.152.0/22
  - youtube.com = 208.65.153.238
- Pakistan ISP advertises 208.65.153.0/24
  - more specific, prefix hijacking
- Internet thinks youtube.com is in Pakistan!
- Outage resolved in 2 hours…

BGPsec



Deploy360 16 October 2017

- Route announcements must be cryptographically signed
  - AS can only advertise as itself
  - AS cannot advertise for IP prefixes it does not own
- Requires a public-key infrastructure (PKI)

BGPSec – A reality now

RFC 8205

Need to wait for ASes to catch up!

Summary: Internet Security

- Recurring themes:
  - Built without any authenticity mechanisms in mind
  - Functionality mechanisms (sequence #’s) become implicit security mechanisms
  - New attempts at (somewhat) backwards-compatible security mechanisms
    - IP -> IPsec
    - DNS -> DNSsec
    - BGP -> BGPsec