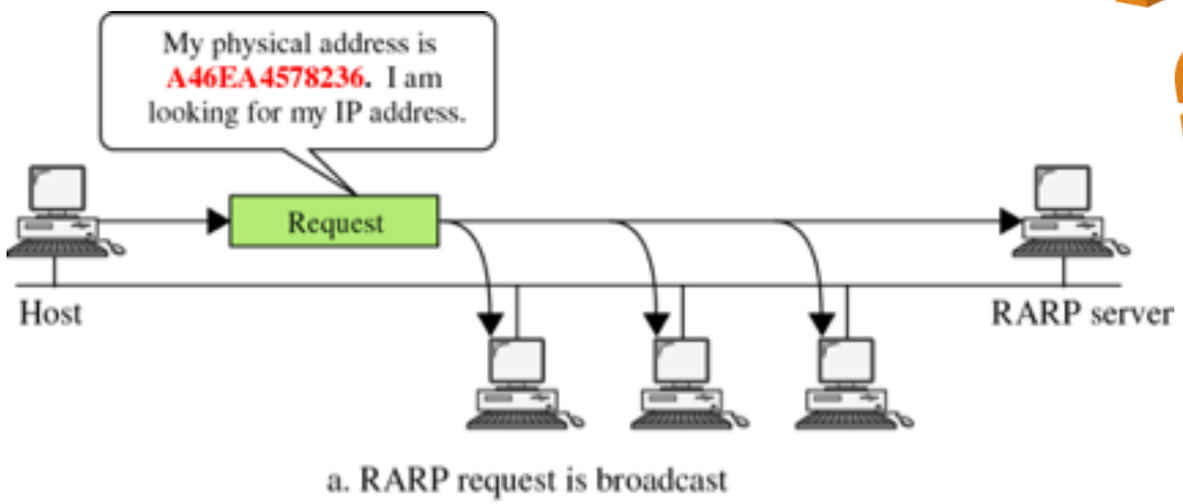
Network Security
(6CCS3NSE - 7CCSMNSE)

Diego Sempreboni
Department of Informatics
King's College London, UK

Second term 2019/20
Lecture 4

Previously on Network Security

- What is the point of sending the request to every node in the network, since only RARP Server has the information?



a. RARP request is broadcast

- Since in Networks, the client does not need prior knowledge of the network topology or the identities of servers capable of fulfilling requests, the protocol specifies that the message is broadcasted to everyone in the network. However, just a specific machine, who is actually waiting for RARP requests and often called with the name of RARP server, will respond to the requests made by the client.

Previously on Network Security

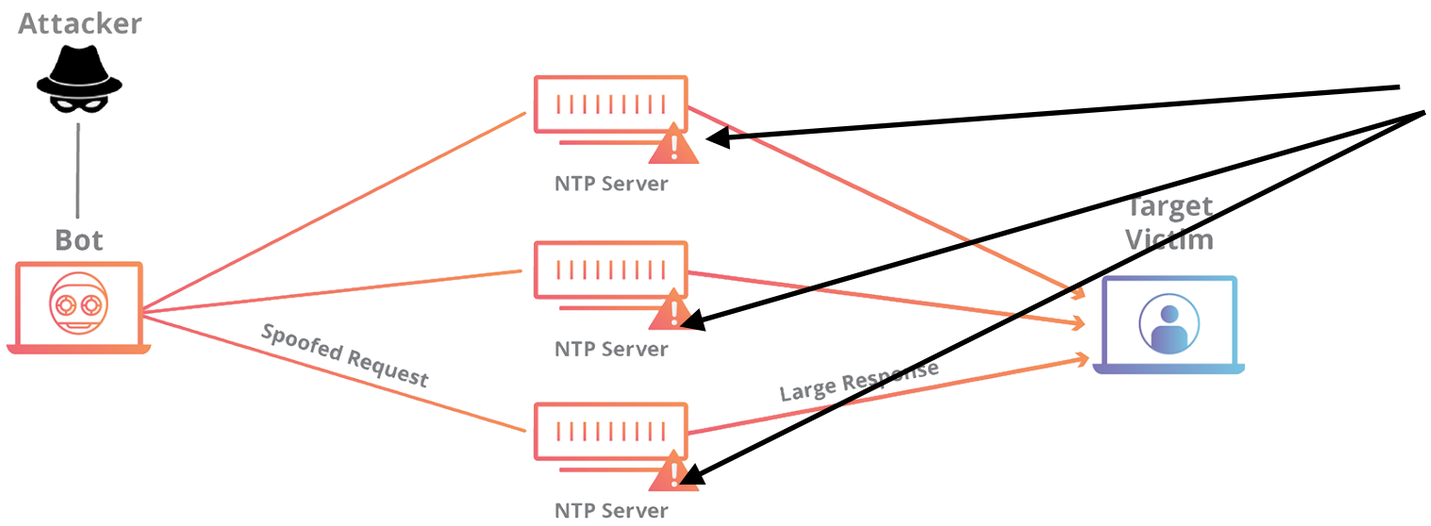
- NTP Amplification DoS attack:

  1. Find the IP address of the victim

  2. Forge packets with the spoofed IP address asking for monlist of a NTP server

  3. NTP server responds to the spoofed IP address (victim) with a maximum of 600 results.

  4. (1 -> 600) ... (2 -> 1200) ... (3 -> 1800) ... (4 -> 2400) ....



- An NTP server with the 'monlist' feature enabled will return a list of clients that recently queried the NTP server:

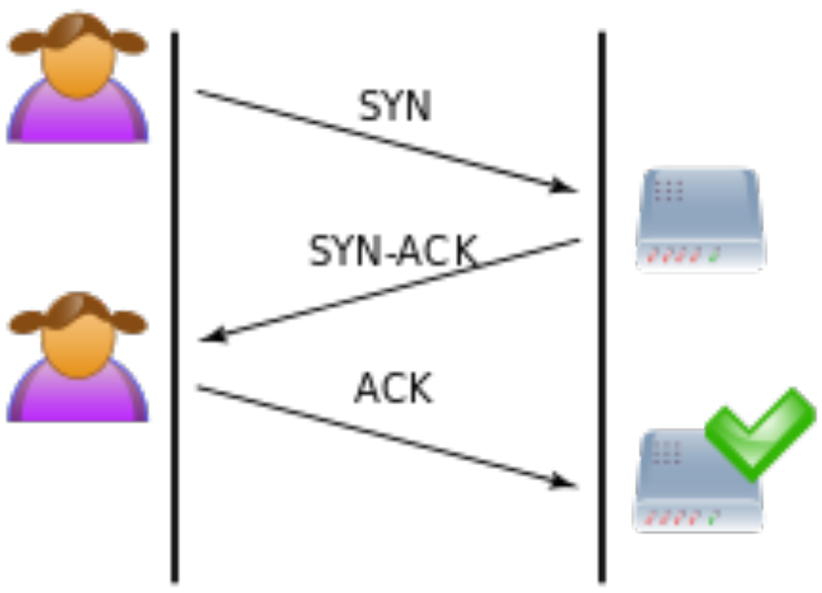
```
ntpdc –n –c monlist x.x.x.x
```

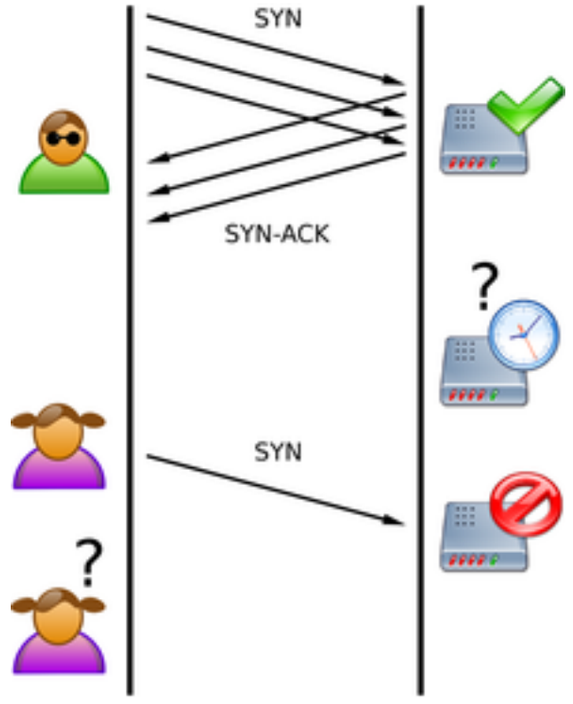
Previously on Network Security

- SYN Flooding:

  PHASE1: The attacker (Mallory) sends several packets but does not send the "ACK" back to the server. The connections are hence half-opened and consuming server resources.

  PHASE2: Alice, a legitimate user, tries to connect but the server refuses to open a connection resulting in a **denial of service.**





Objectives and learning outcomes

- To understand how network identity is stolen
  - 1st method: Spoofing
  - 2nd method: Hijacking
  - 3rd method: Poisoning

Objectives and learning outcomes

- List of attacks:
  - BGP route hijacking
  - TCP session hijacking
  - DNS poisoning

- Basic approach [HOW TO]:

  1. Learn basics of BGP, TCP, DNS
  2. Think of ways to subvert them


SCHULZ
SOMETIMES WHEN I DELETE AN EMAIL, I LIKE TO PRETEND THAT I'M KILLING THE PERSON WHO SENT IT
PEANUTWEETER.COM
@DIXIELOCKE

Principles

Spoofing vs hijacking vs poisoning

- **Spoofing**: imitating someone else
  - Typically used in a humorous way, not here!

Spoofing vs hijacking vs poisoning

- **Hijacking**: taking over what belongs to someone else, especially at “run-time”

Spoofing vs hijacking vs poisoning

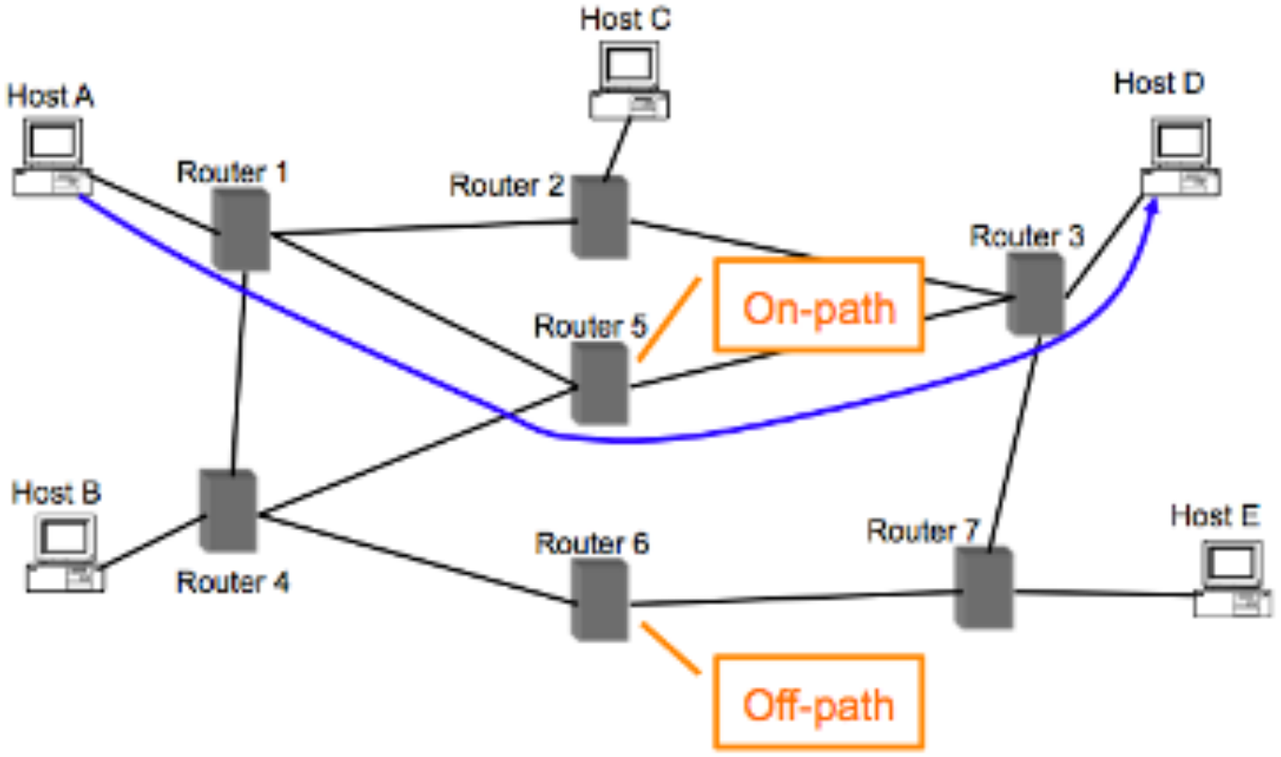
- **Poisoning**: contaminating some source of information (that is usually trusted to be good)

How to hijack connections

- When is it appropriate for an adversary to use the hijacking technique?
  - To disrupt, or jam communications
  - To insert false or malicious data
- Notice similarity to spoofing!
- Difference is that, in general, hijacking takes over an **existing** connection, spoofing initiates new unwanted connection (more on difference at end of lecture)

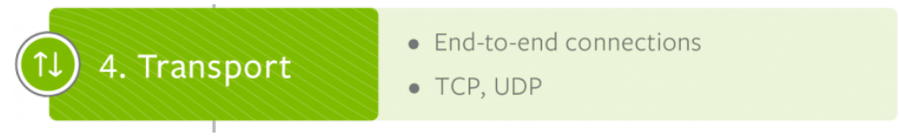
Off-path vs on-path adversaries

- On-path adversaries are more powerful.
- Inserting false data is very difficult if off-path
- On-path adversary can use connection state
  - Difficult to do at scale!

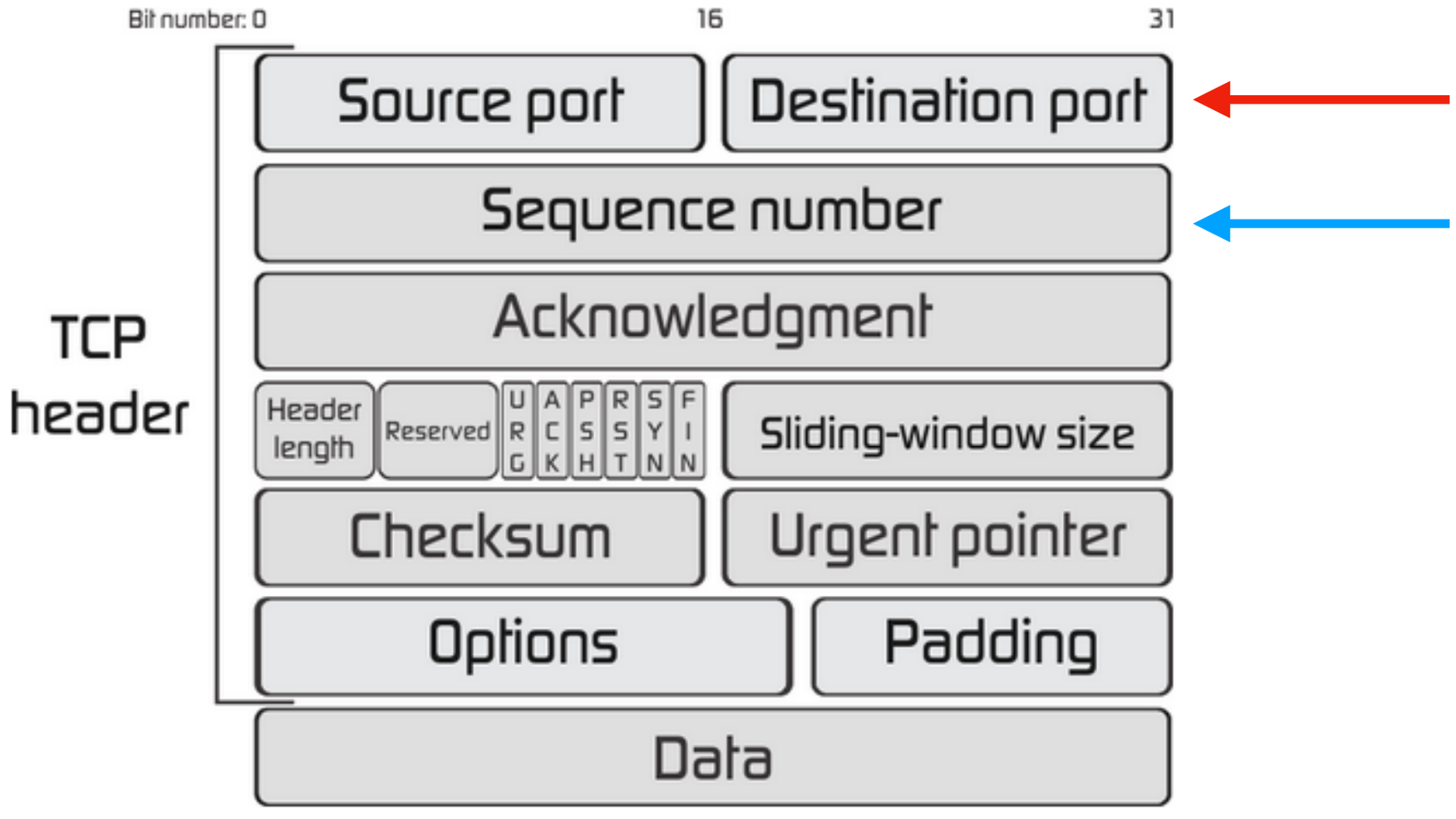


TCP Session hijacking

TCP: Transmission Control Protocol

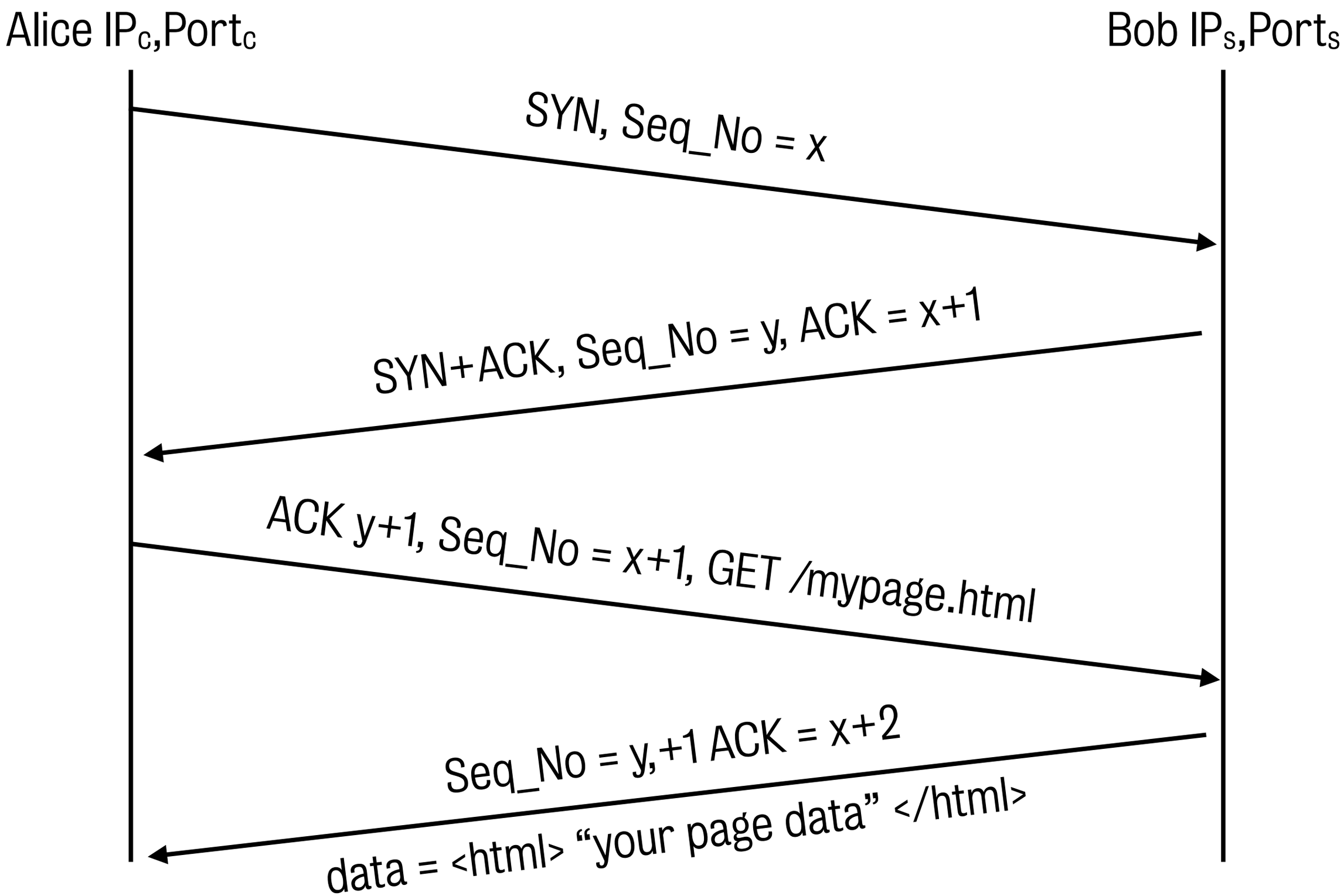


- Defines a connection: ordered sequence of bytes, over unordered IP network
  - IP address + port number defines connection
  - Sequence number defines order of packet payload within the larger stream/connection

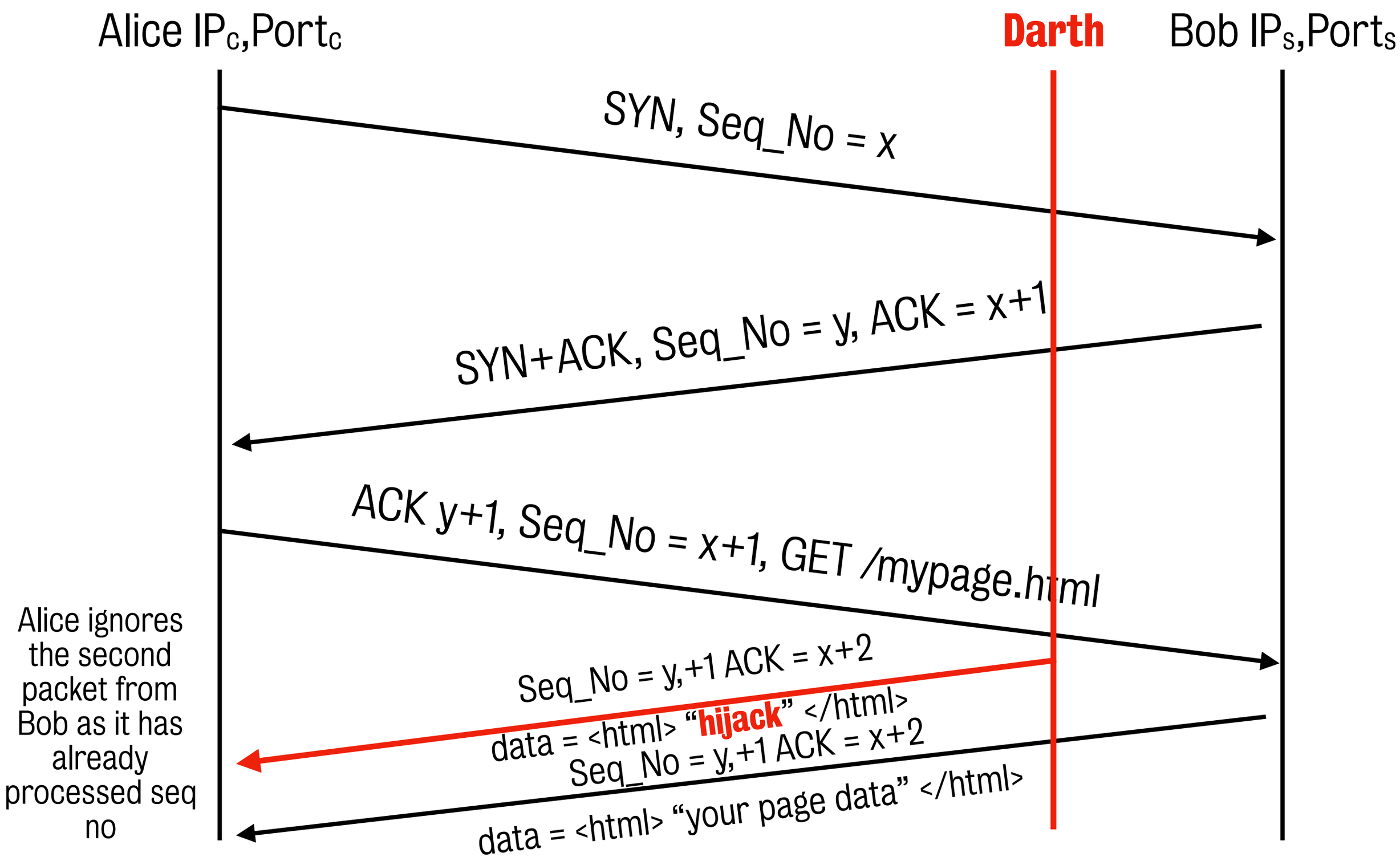


TCP connection set up: Three Way Handshake

TCP connection set up: Three Way Handshake



TCP connection hijacking



On-path TCP connection hijacking

- Man in the middle (Darth) can alter TCP stream without receiver realising the alteration.
  - BUT: Darth needs to know the sequence number (not so difficult if Darth is on-path)

- Steps to on-path TCP connection hijacking:

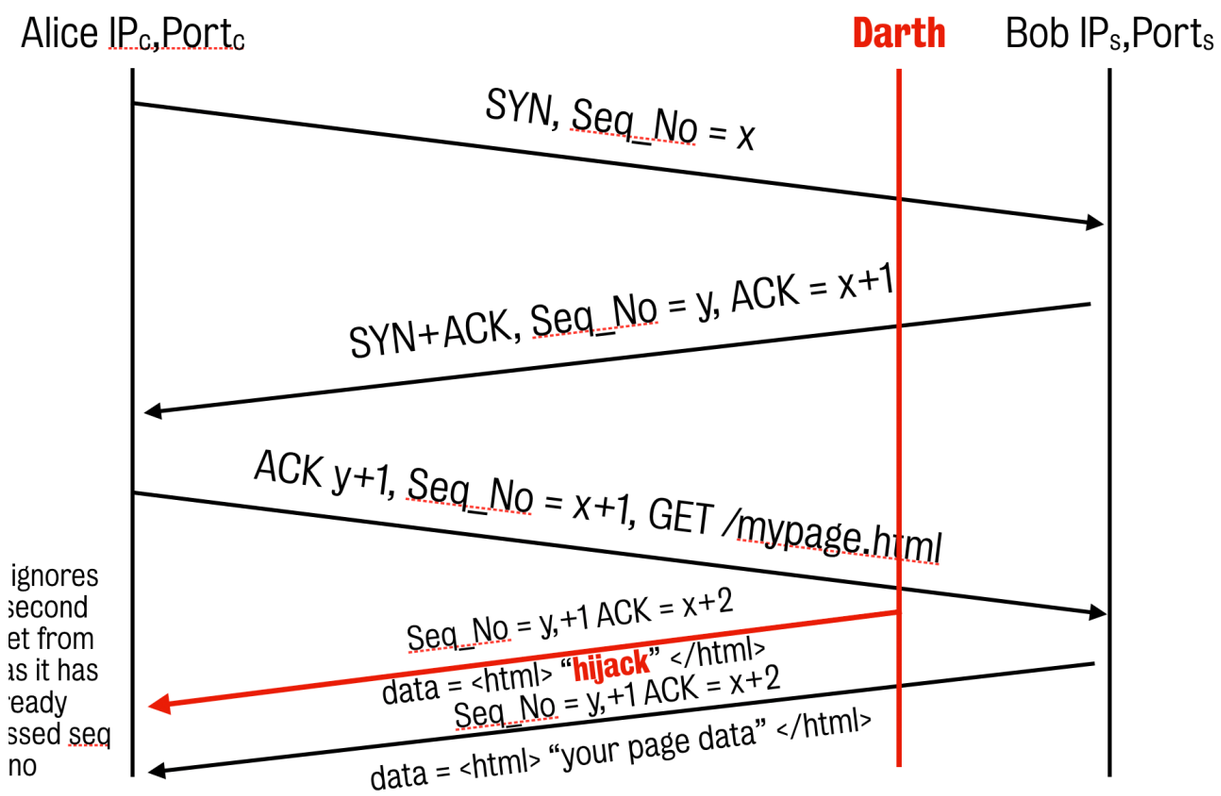
  1. Sniff packets
  2. Predict sequence number (client->server / server->client)
  3. Inject data

How can Darth use hijacked stream?

- **To spoof client:** authentication may happen at beginning of connection. By hijacking connection after authentication, Darth can leave a record of bytes as an “authenticated” user:

  e.g., HTTP POST / join the dark side!

- **To spoof server:** insert false data from server to client (attack as shown in timing diagram here)



What can Darth do if off-path?

- Remember spoofing?

- What if Darth blindly spoofs Alice, and sends a packet to Bob, pretending to be Alice?

- For this to work, Darth's packets have to be in right part of the stream, otherwise Bob will ignore packets

Initial sequence number attack

Darth pretending to be **Alice**
IP_c,$Port_c$

Bob IP_s,$Port_s$

SYN, Seq_No = x

SYN+ACK, Seq_No = y, ACK = x+1

ACK y+1 Seq_No = x+1, GET /mypage.html

Seq_No = y,+1 ACK = x+2

data = <html> “your page data” </html>

Some observation:
this gets sent to the real Alice.
(Do you see why?)
1. Alice may send RST and Bob will terminate the connection
2. Darth does not get to see y and **must guess y+1**

How to guess initial sequence number?

- RFC 793 (TCP Standard) says use your clock:

  When new connections are created, an initial sequence number (**ISN**) generator is employed which selects a new 32 bit ISN. The generator is bound to a (possibly fictitious) 32 bit clock whose low order bit is incremented roughly every 4 microseconds. Thus, the ISN cycles approximately every 4.55 hours. Since we assume that segments will stay in the network no more than the Maximum Segment Lifetime (MSL) and that the MSL is less than 4.55 hours we can reasonably assume that ISN's will be unique. [i.e., sequence numbers should not repeat for at least 4.55 hours]

- Why? To ensure undelivered packets from previous connection that arrive late do not overlap with current connection

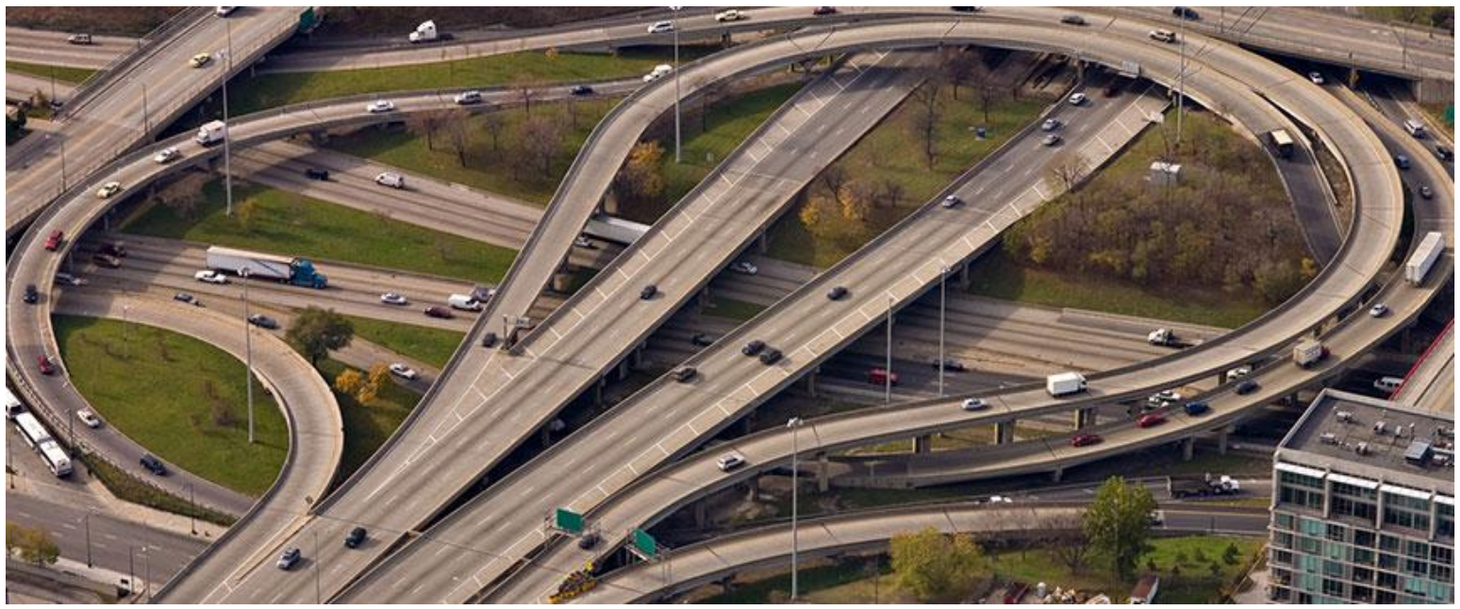
  Implementations differ: BSD increments ISN by 128,000 every second and 64,000 for every new TCP connection

- Thus ISN is easily guessed: e.g., what if Darth created a legitimate connection with Bob before spoofing Alice?

BGP Route hijacking

How data gets from here to there

- Millions of packets are trying to get through from X to Y



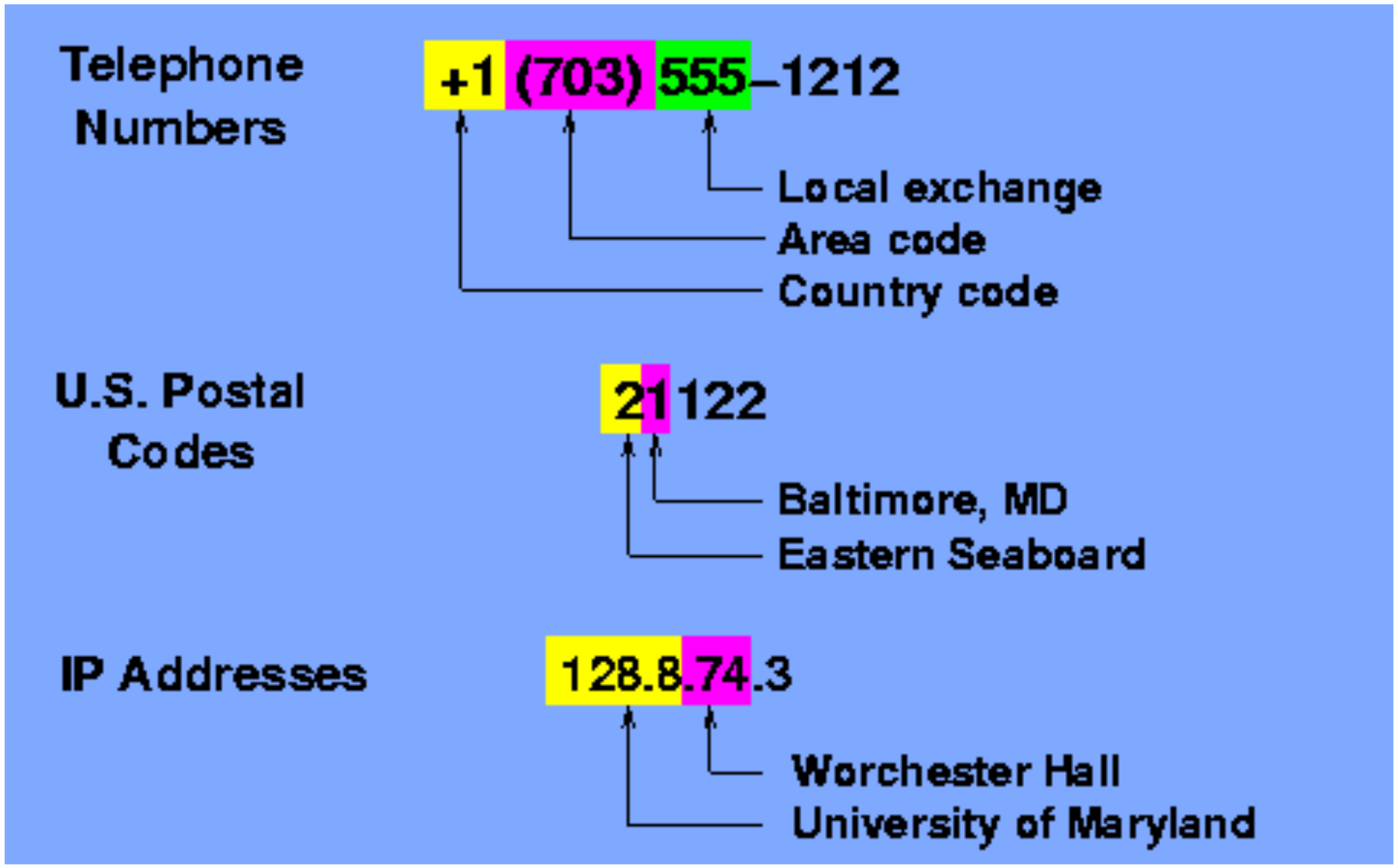
- The higher-lever picture can be quite complicated

- Routes need to be created

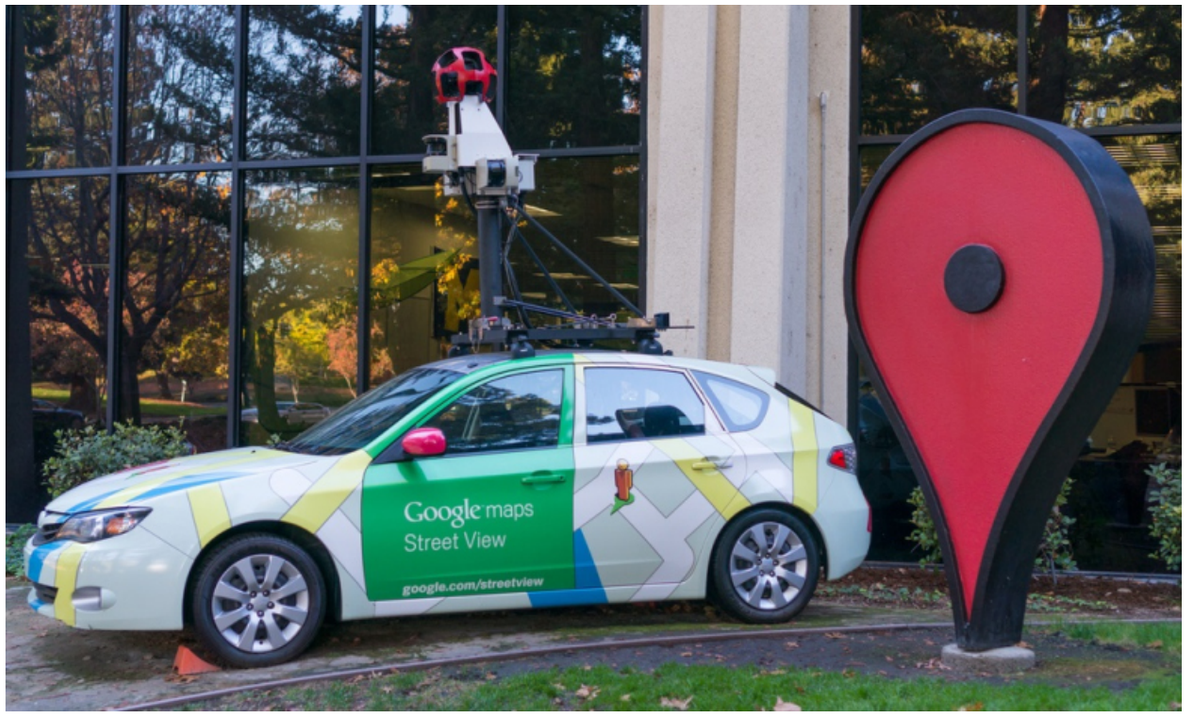
  Problem: static signs such as these do not cover all possible destinations (important destinations like services are clearly marked, but how to get to your home?)

Hierarchical addressing

- Key idea: route at different granularity levels



Additionally, need to learn the best routes



BGP Route hijacking

- Maliciously altering the **path** that a packet takes through the network
  - Changing the destination
  - Changing the route

- Let's see BGP first!

Simplified intro to BGP-1

- The **Border Gateway Protocol** controls the routes packets take through Autonomous System (ASes)

- Autonomous System: On the Internet, an autonomous system (AS) is the unit of router policy, either a single network or a group of networks that is controlled by a common network administrator (or group of administrators) on behalf of a single administrative entity (such as a university, a business enterprise, or a business division). An autonomous system is also sometimes referred to as a routing domain. An autonomous system is assigned a globally unique number, sometimes called an Autonomous System Number (ASN).

Simplified intro to BGP-1

- The **Border Gateway Protocol** controls the routes packets take through Autonomous System (ASes)
  - Inter domain routing
- ASes advertise prefixes that they can serve
  - e.g., 137.73.0.0/16

Simplified intro to BGP-1

- The **Border Gateway Protocol** controls the routes packets take through Autonomous System (ASes)
  - Inter domain routing
- ASes advertise prefixes that they can serve
  - e.g., 137.73.0.0/16

    prefix 16 bits

Simplified intro to BGP-2

- The **Border Gateway Protocol** controls the routes packets take through Autonomous System (ASes)
  - Inter domain routing
- ASes advertise prefixes that they can serve
  - e.g., 137.73.0.0/16
    Host address

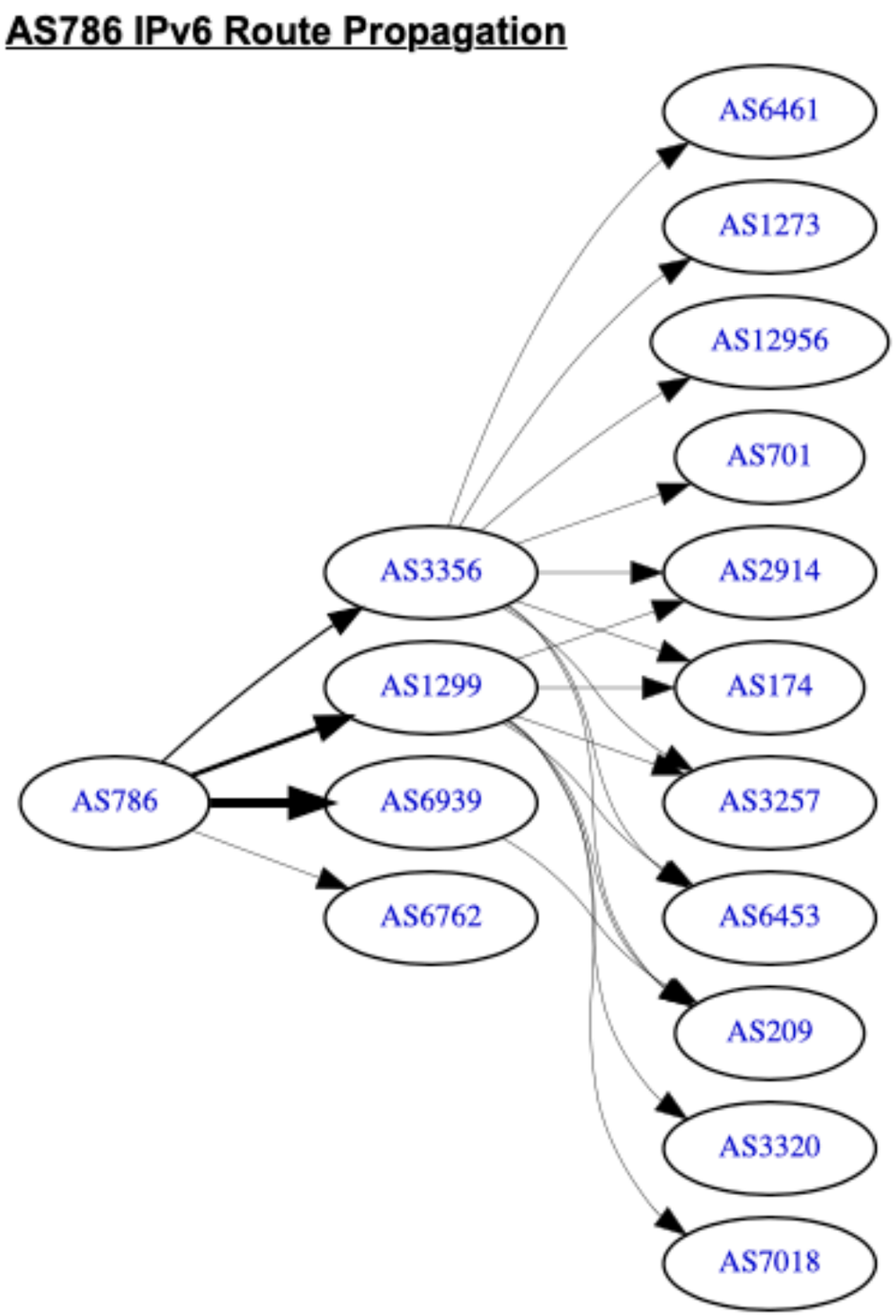
Simplified intro to BGP-3

- The **Border Gateway Protocol** controls the routes packets take through Autonomous System (ASes)
    - Inter domain routing
- ASes advertise prefixes that they can serve
    - e.g., 137.73.0.0/16

    ASes perform longest prefix matching to select which neighbours to route though

KCL Routes

- AS786 is KCL… and several other education institutions. Served through ja.net



137.73.0.0/16 is one of the routes announced by ja.net (AS786)

HURRICANE ELECTRIC
INTERNET SERVICES

Search

137.73.0.0/16

Quick Links

- BGP Toolkit Home
- BGP Prefix Report
- BGP Peer Report
- Exchange Report
- Bogon Routes
- World Report
- Multi Origin Routes
- DNS Report
- Top Host Report
- Internet Statistics
- Looking Glass
- Network Tools App
- Free IPv6 Tunnel
- IPv6 Certification
- IPv6 Progress
- Going Native
- Contact Us

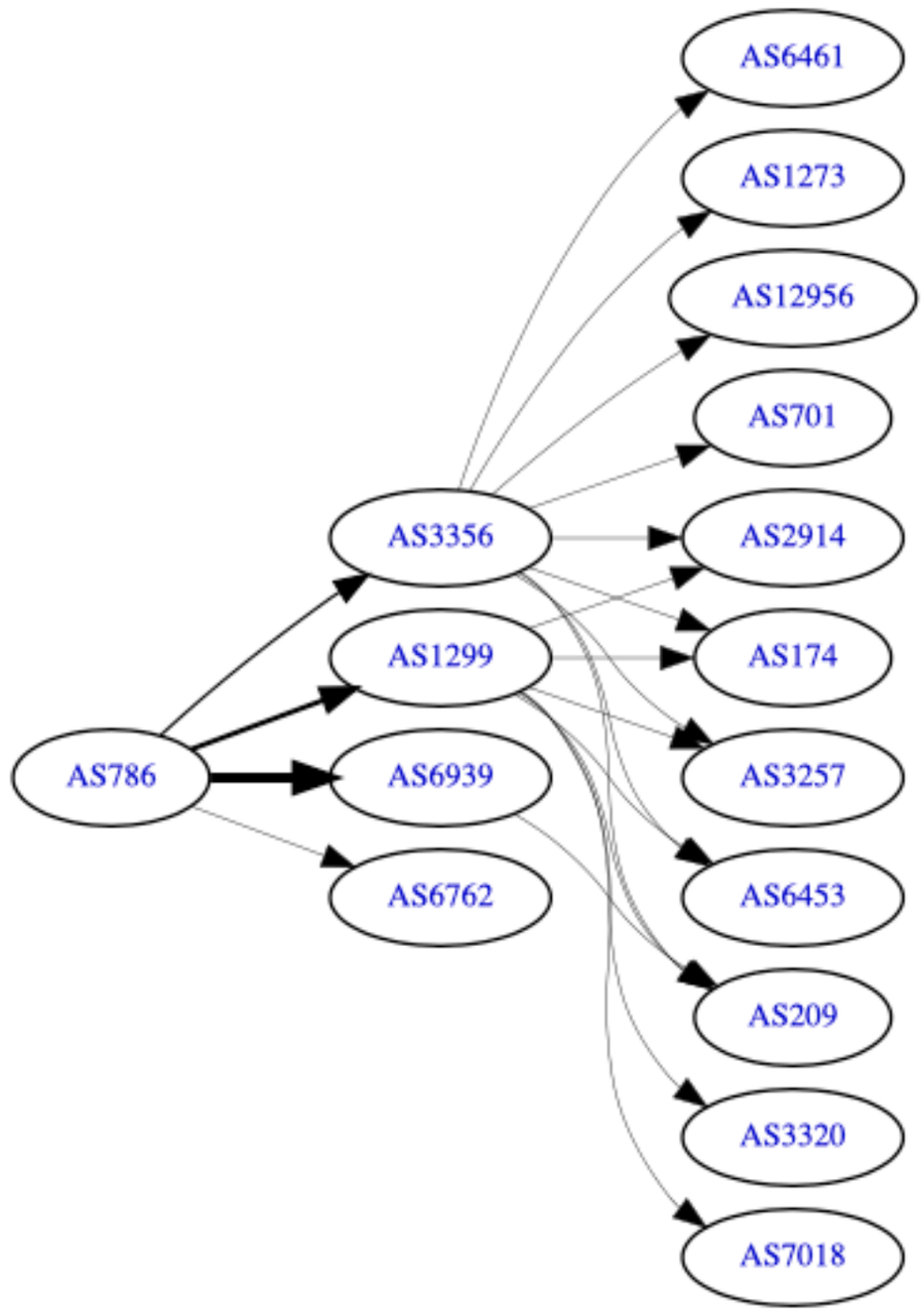
Network Info | Whois | DNS | IRR

Announced By		
Origin AS	**Announcement**	**Description**
AS786	137.73.0.0/16 ✓	King's College London

Updated 07 Dec 2019 00:08 PST © 2019 Hurricane Electric

AS786 advertises routes to KCL

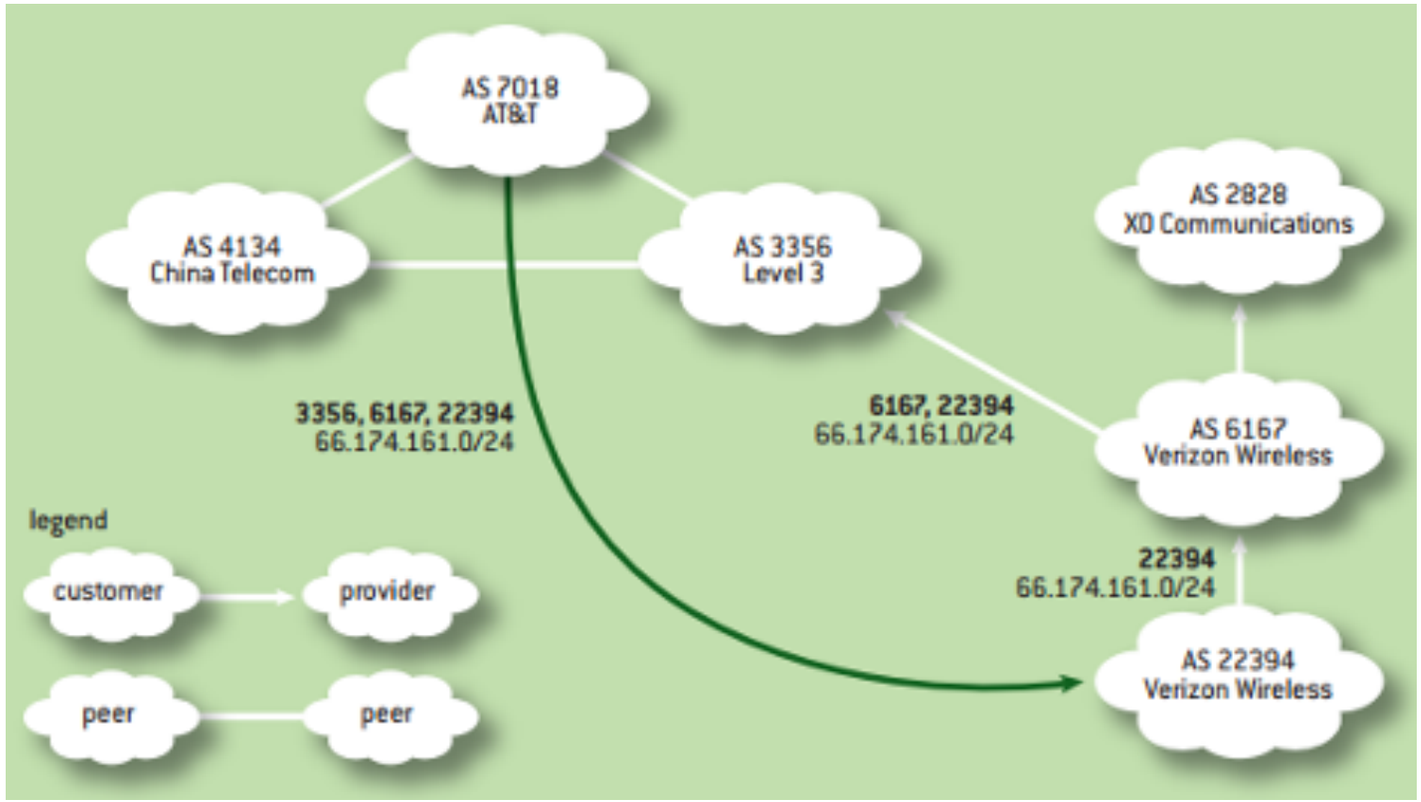
I can route to 137.73.0.0/16*



***graph is IPv6. For IPv4 see** http://bgp.he.net/AS786#_graph4

BGP Route advertisements

- AS7018 advertises 66.174.161.0/24 and it gives the sequence of other AS that lead to the advertised network (3356, 6167, 22394)
- AS22394 is the “origin”
- Note the green line represents AS7018 advertising a network originally advertised by AS22349 but they are not directly connect!



What happens if an AS lies?

- Prefix hijacks
  - Imagine UCL advertised routes to 137.73.0.0/16

- BGP prefers **longest** prefix match

- Sub prefix hijacks
  - Imagine UCL advertised routes to 137.73.0.0/24

How to create a Routing Blackhole

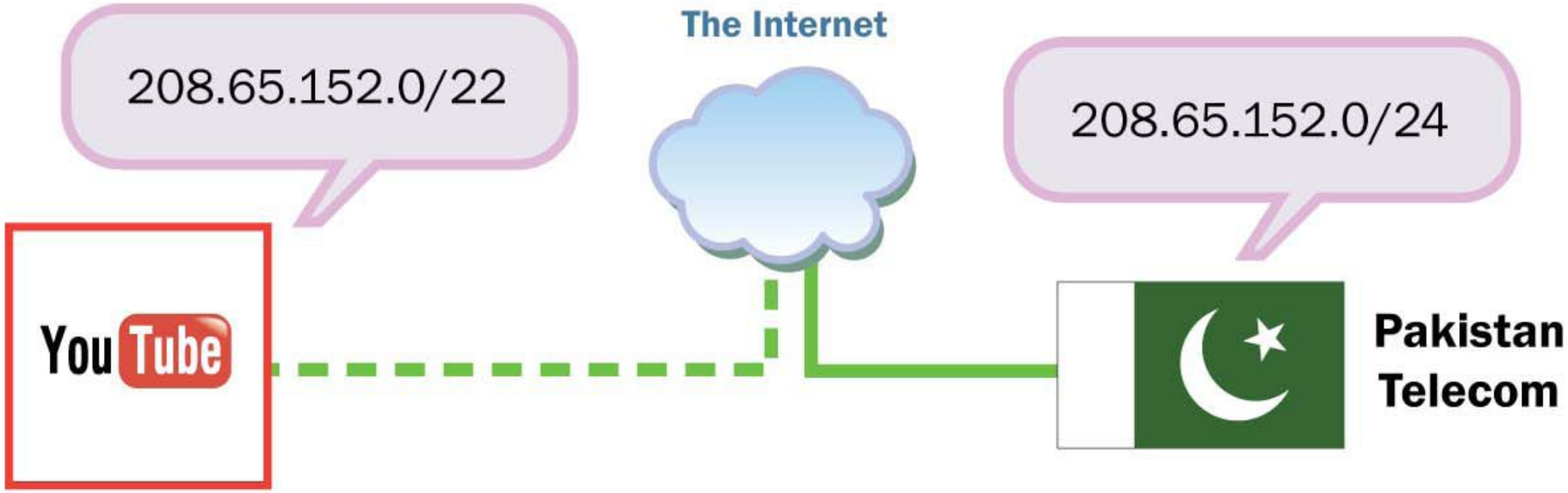
- ASes advertises routes it can't actually offer
  - Result: packets go into a network "black hole"

- April 25, 1997: "The day the Internet died"
  - AS7007 (Florida Internet Exchange) de-aggredated the BGP route table and re-advertised all prefixes as if it originated paths to them
  - Huge network instability as incorrect routing data

For full story of AS7007 incident from the network operator see:
http://lists.ucc.gu.uwa.edu.au/pipermail/lore/2006-August/000040.html

The day YouTube migrated to Pakistan

- Pakistan Telecom took down YouTube in an attempt to perform censorship (in Pakistan)



- For a concise yet interesting account of what happened, see: http://web.mit.edu/6.02/www/s2012/handouts/youtube-pt.pdf

DNS Cache Poisoning

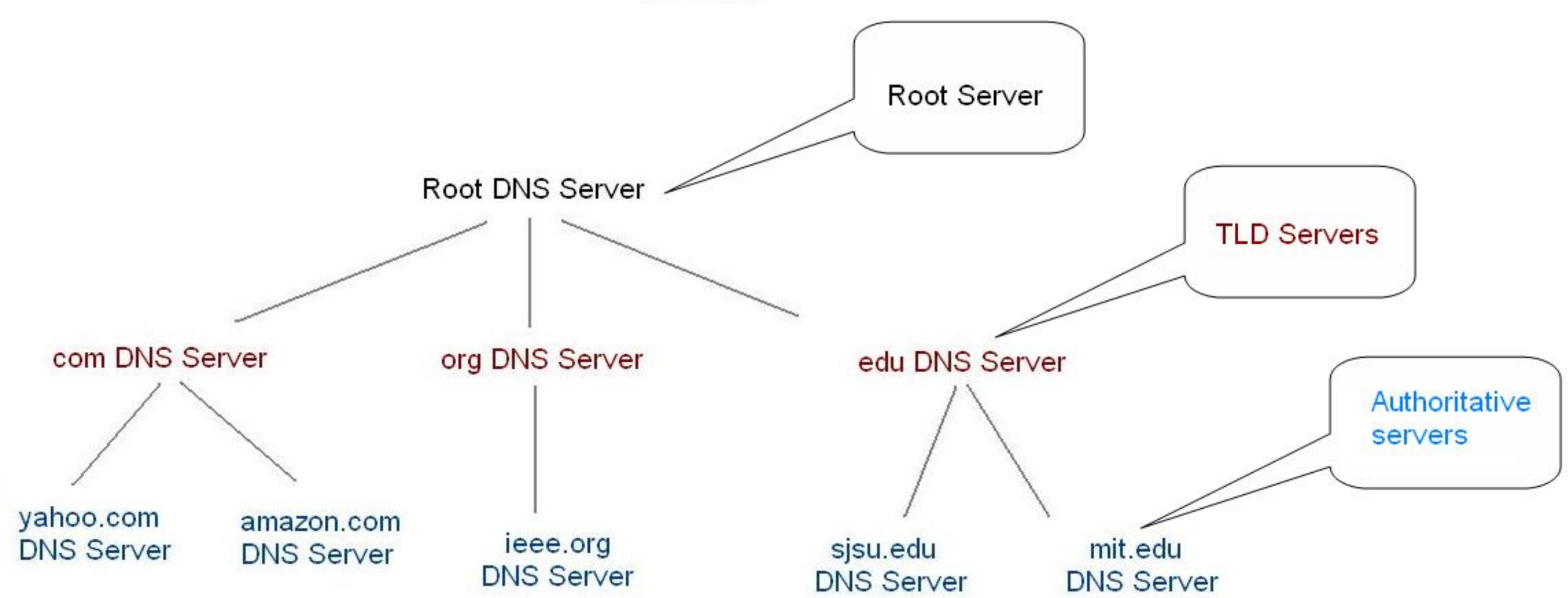
Historical motivation for DNS

- Humans find it easier to remember names like www.kcl.ac.uk, than 137.73.118.10

- Historically, mapping was maintained in a /etc/hosts file. This does not scale!

- Started off quite simple. Now a suite of RFCs describe DNS

DNS has a hierarchical structure



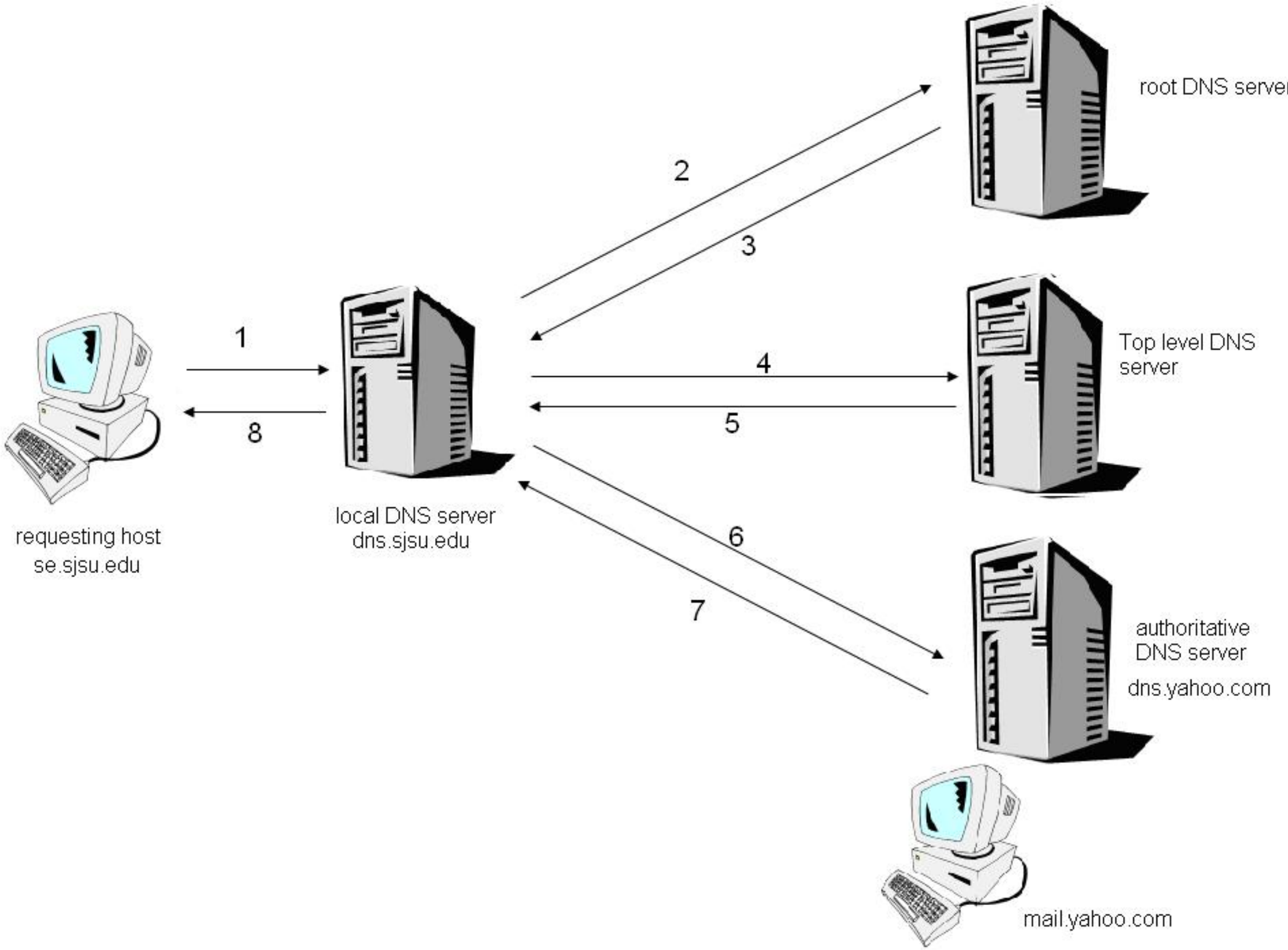
images from http://en.wikibooks.org/wiki/Communication_Networks/DNS

DNS name resolution

- It is the process used to map
  - names to IP addresses (POSIX: gethostbyname)
  - IP address to names (POSIX: gethostbyaddr)

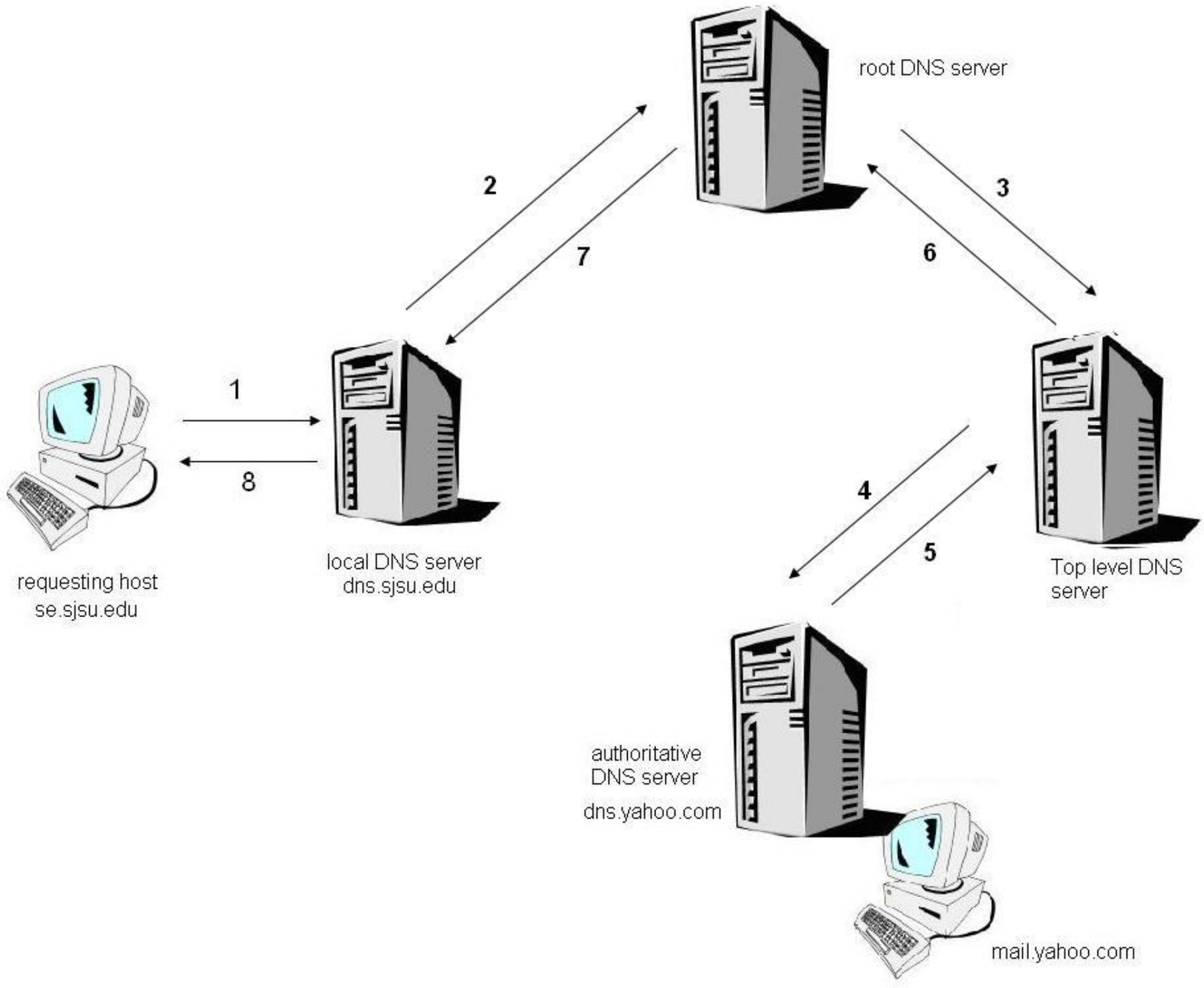
- A DNS **Resolver** performs lookups to do mapping, by contacting **nameserver**(s)

Iterative name resolution



images from http://en.wikibooks.org/wiki/Communication_Networks/DNS

Recursive name resolution



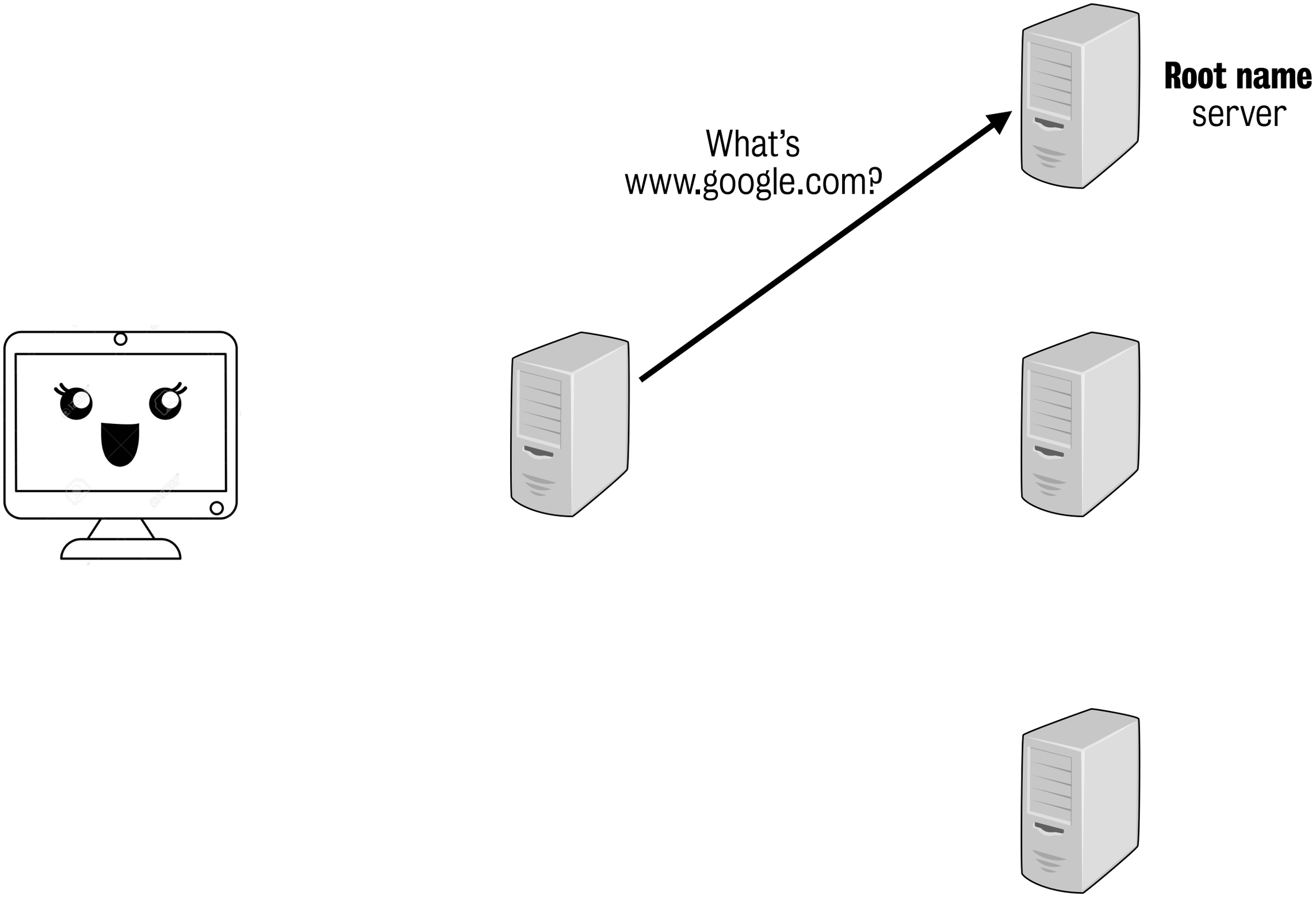
images from http://en.wikibooks.org/wiki/Communication_Networks/DNS

The need for DNS caching

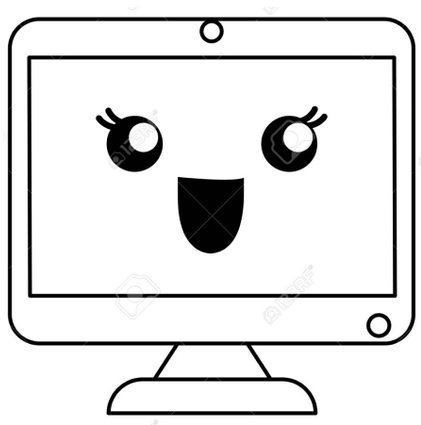
- Both recursive and iterative lookups require many steps… which are repeated many times across many lookups
  - DNS performance is now critical for WWW:
    - Most links are human-friendly DNS names, not IP!
    - Extra layer of indirection enables load-balancing
- Need to cache DNS queries so that second and subsequent lookups are cheap and easy

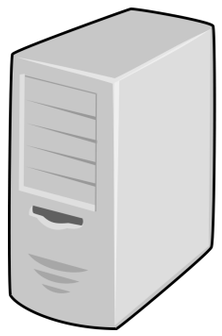
Iterative DNS query with caching

Iterative DNS query with caching

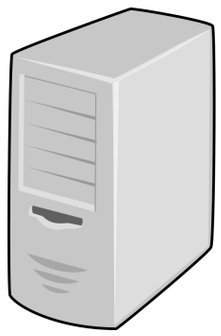


Iterative DNS query with caching



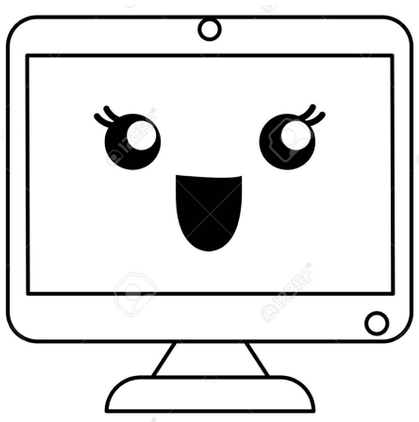


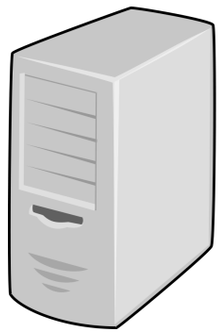
What's
www.google.com?

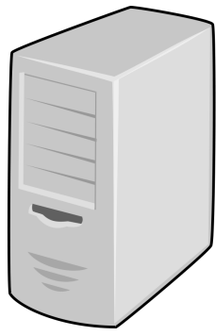


.com name
server

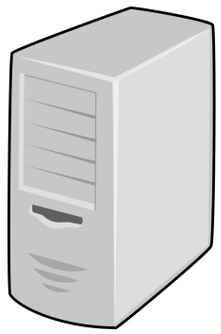
Iterative DNS query with caching







Root name server

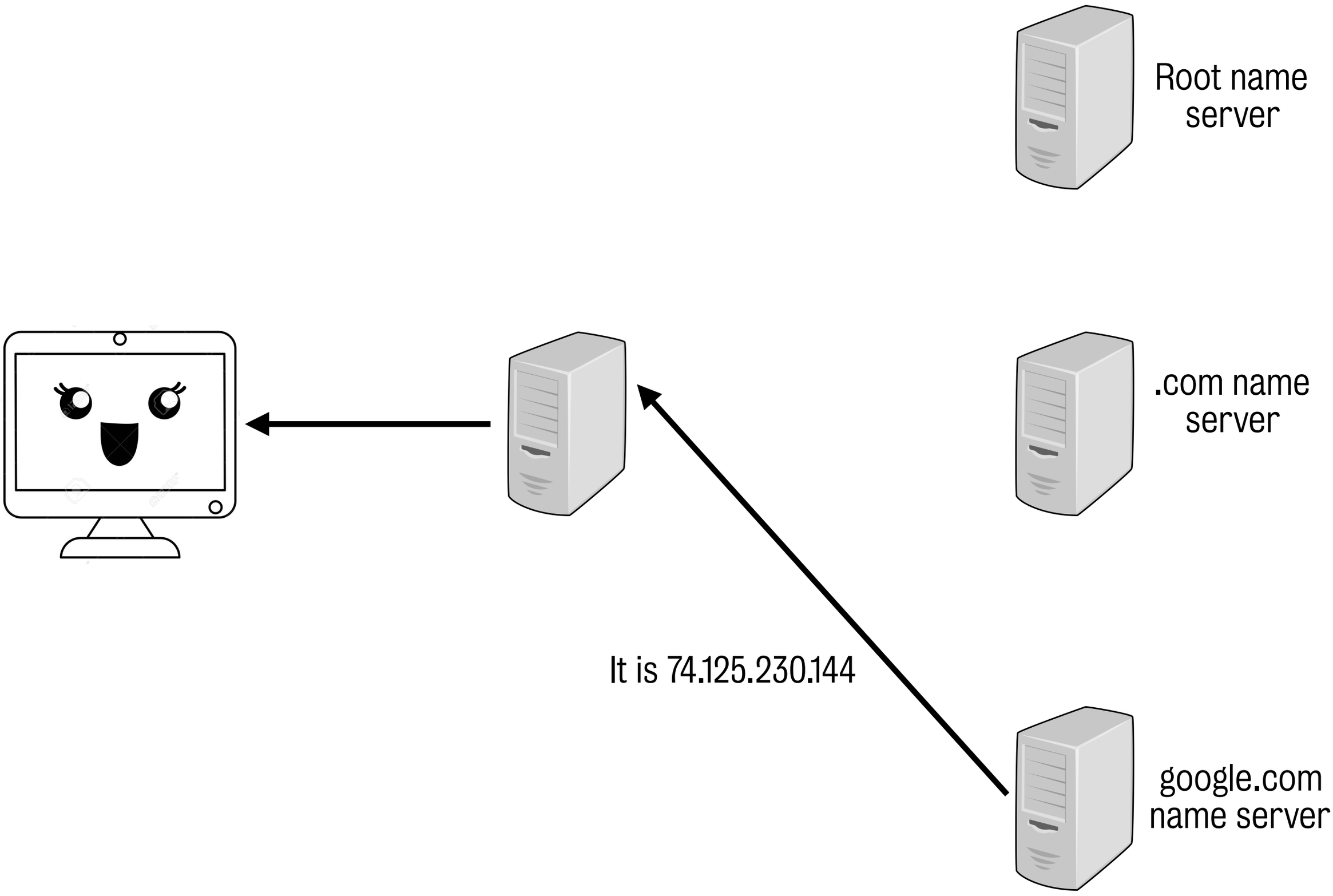


.com name server

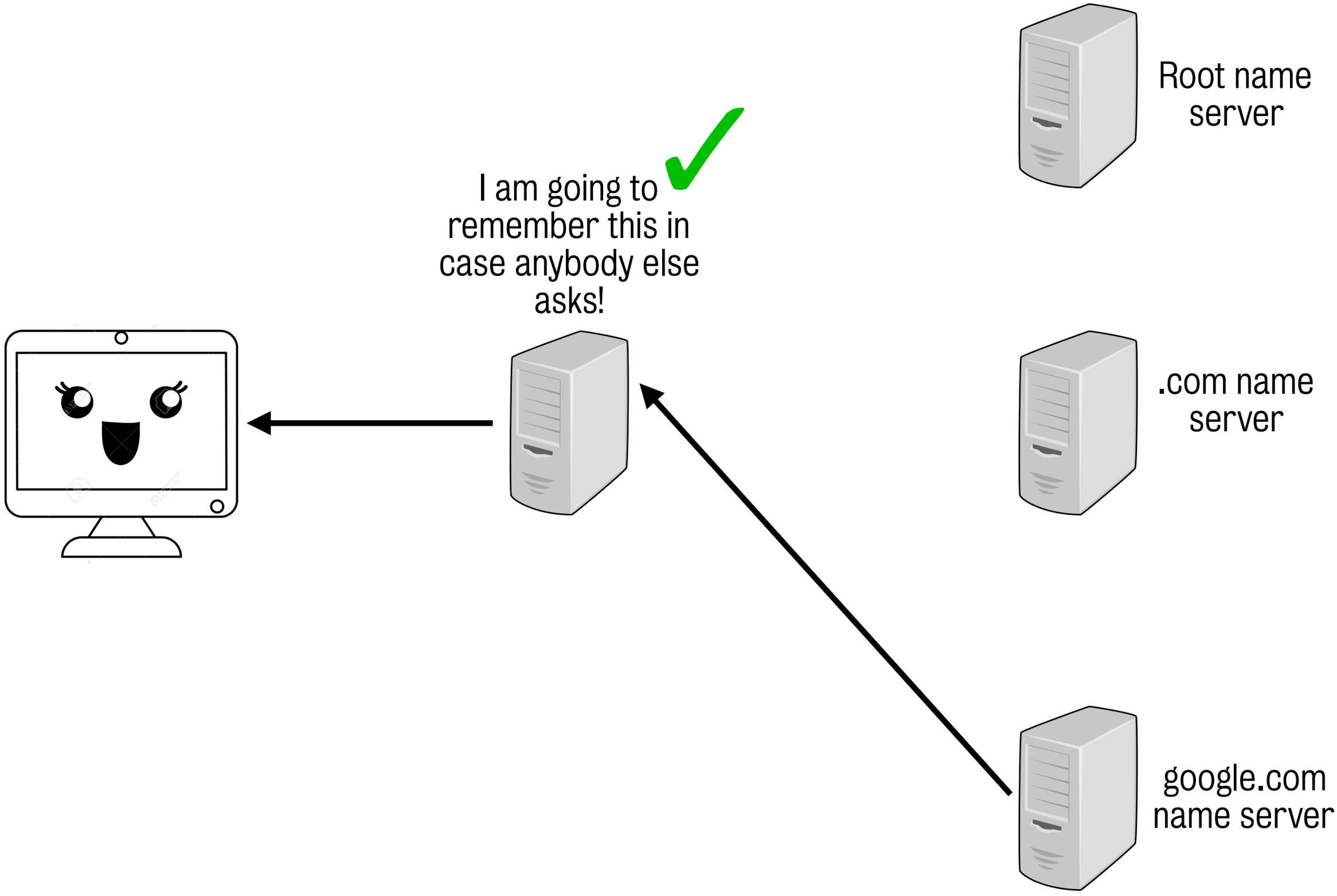
What's www.google.com?

google.com name server

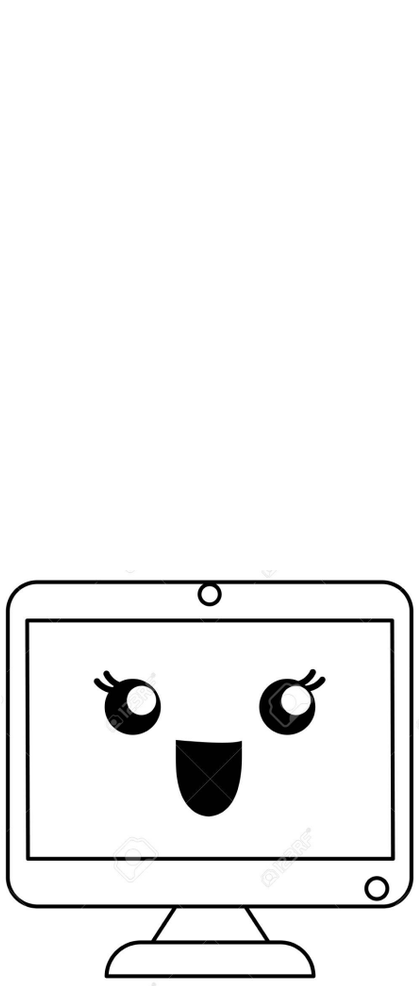
Iterative DNS query with caching

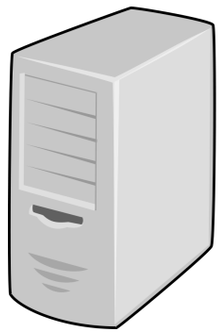


Iterative DNS query with caching

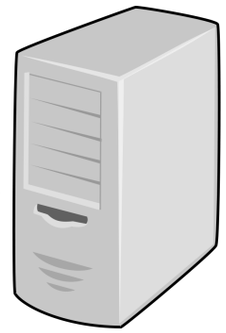


DNS attack: poisoning

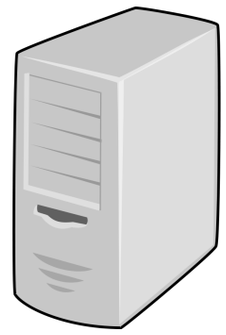


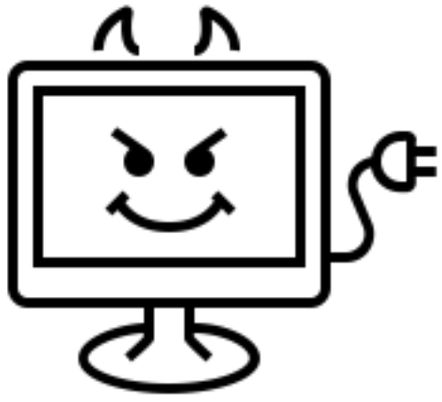


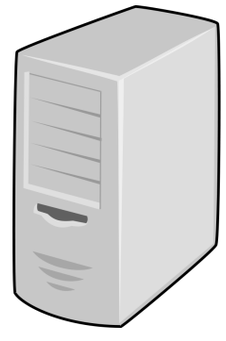
What's
www.google.com?



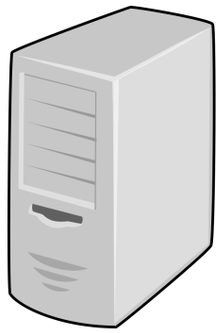
Root name
server



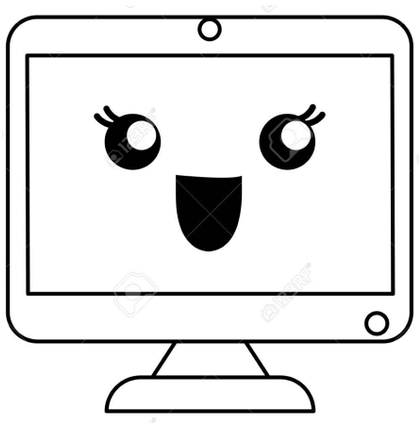


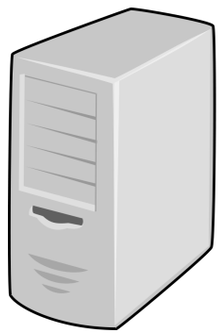


DNS attack: poisoning

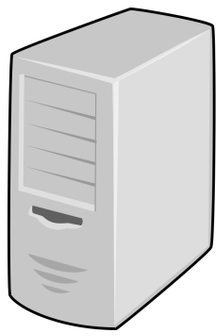


Root name server

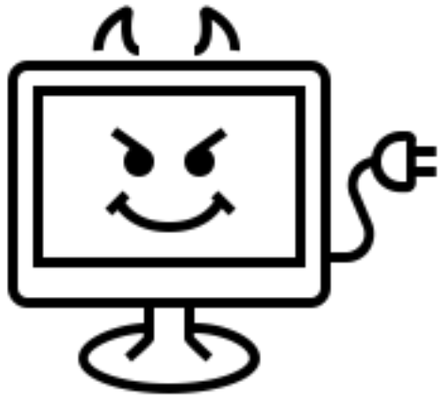




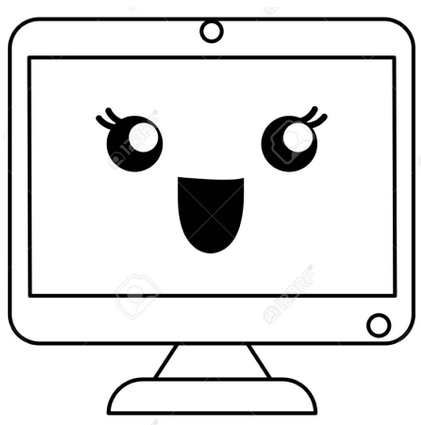
What's www.google.com?

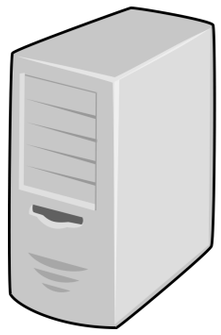


.com name server

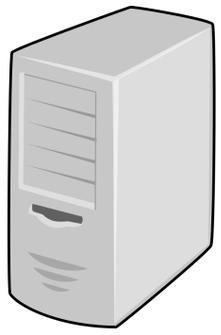


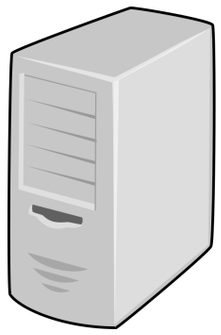
DNS attack: poisoning





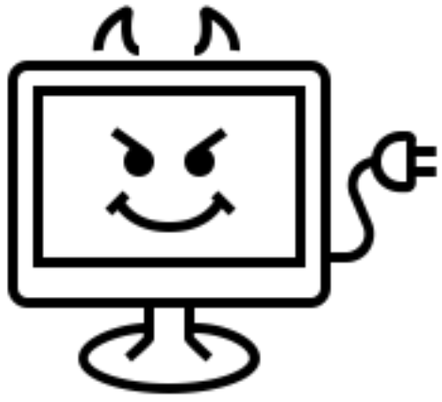
Root name server

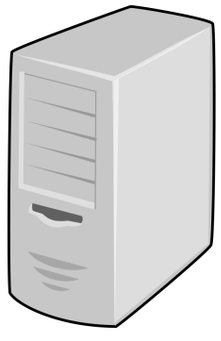




.com name server

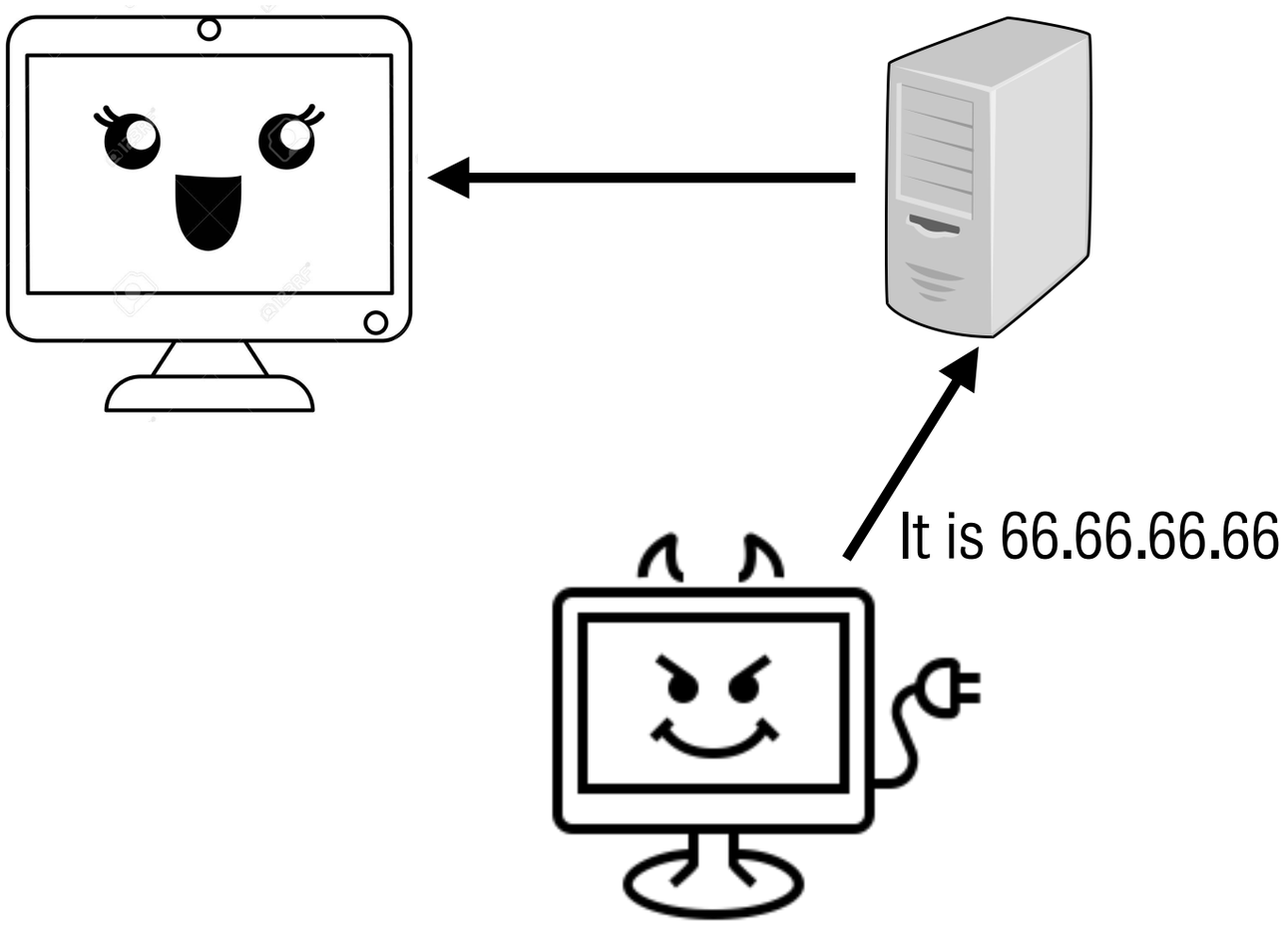
What's www.google.com?



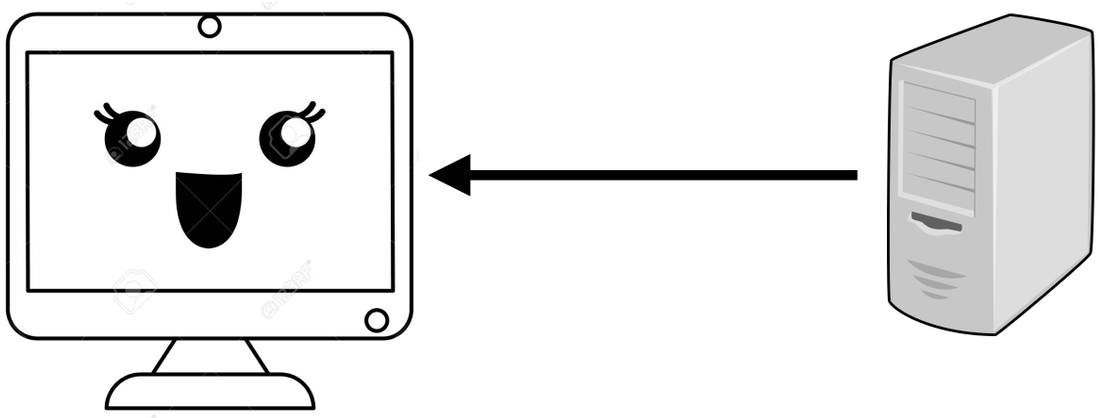


google.com name server

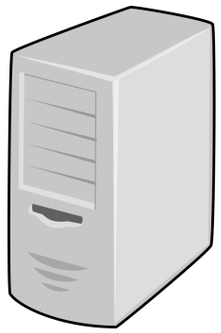
DNS attack: poisoning



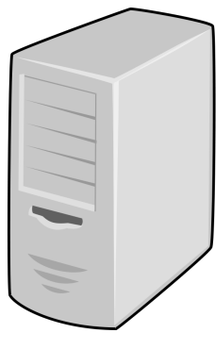
DNS attack: poisoning



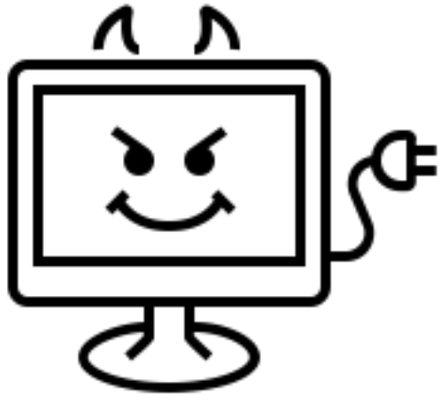
I am going to remember this in case anybody else asks!



Root name server

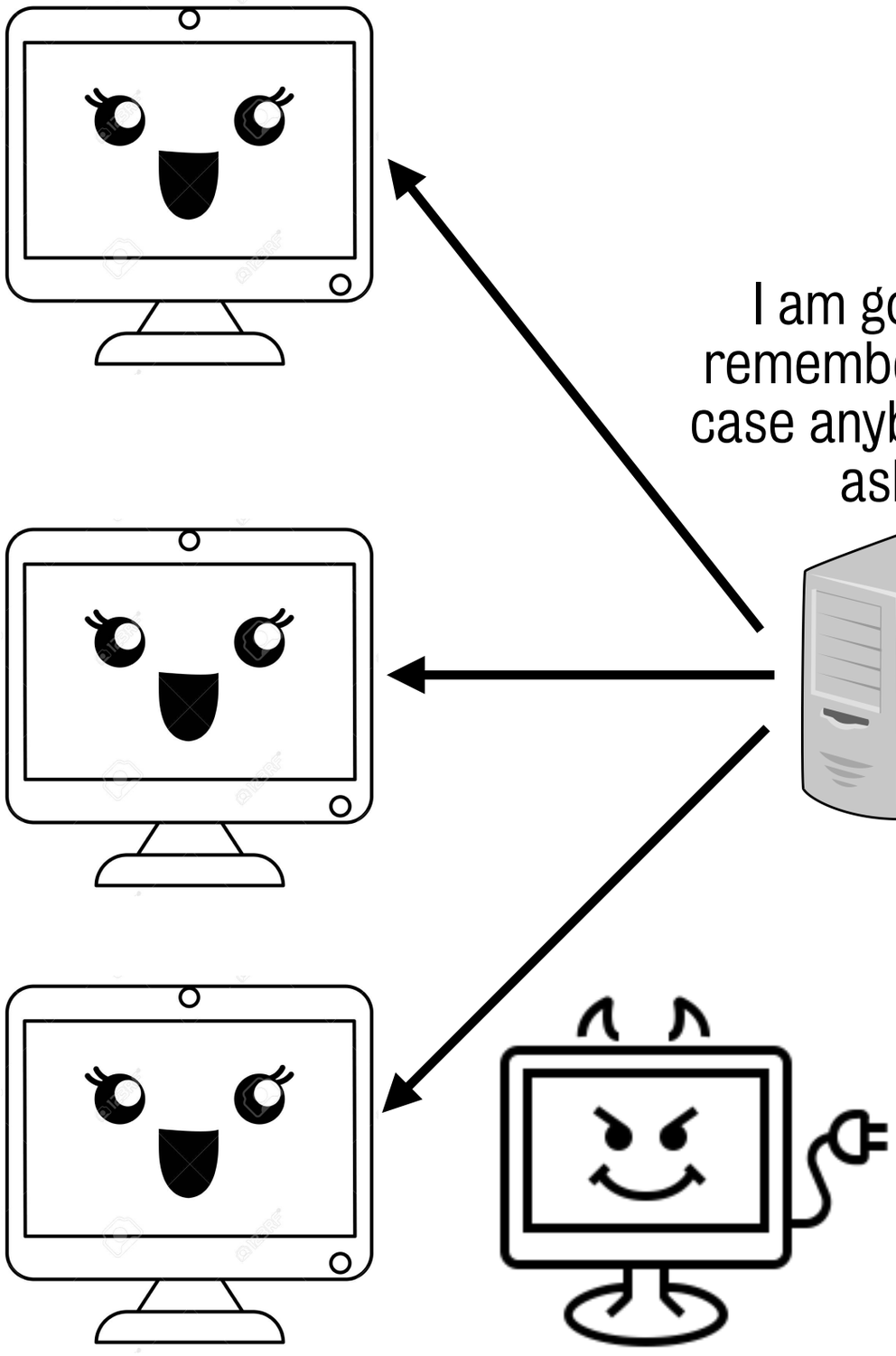


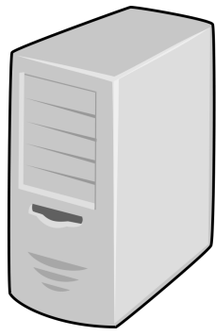
.com name server



google.com name server

DNS attack: poisoning



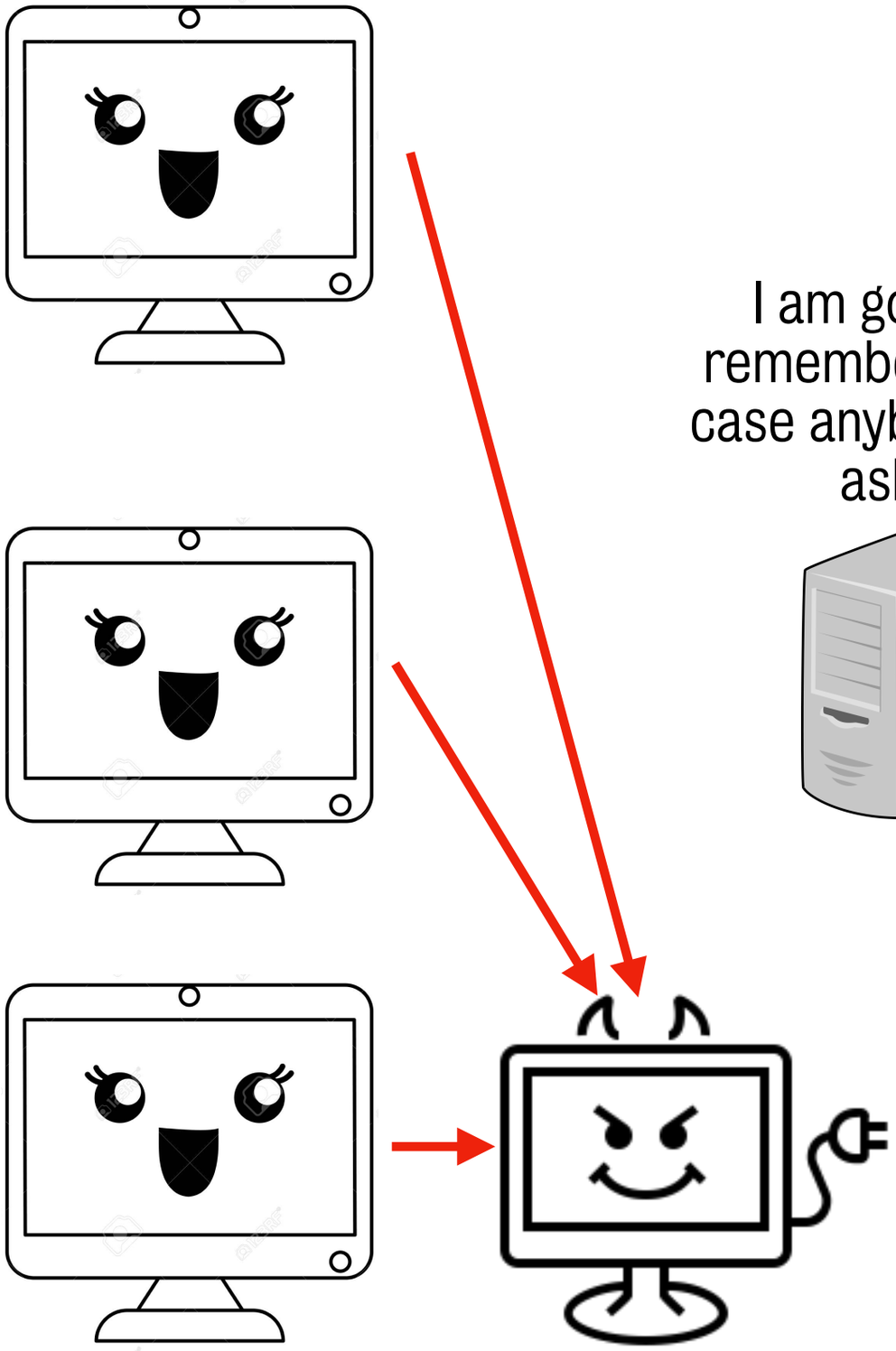


Root name server

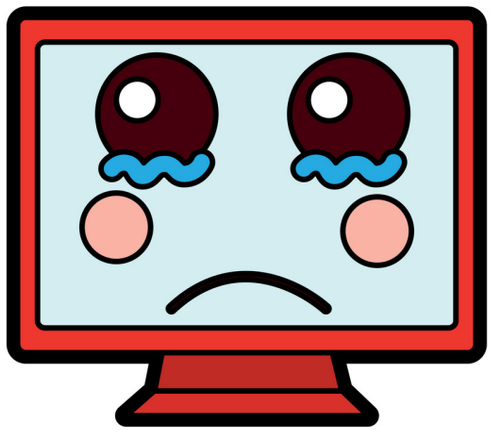
.com name server

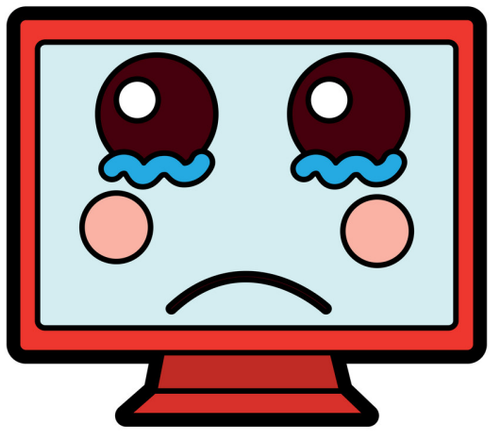
google.com name server

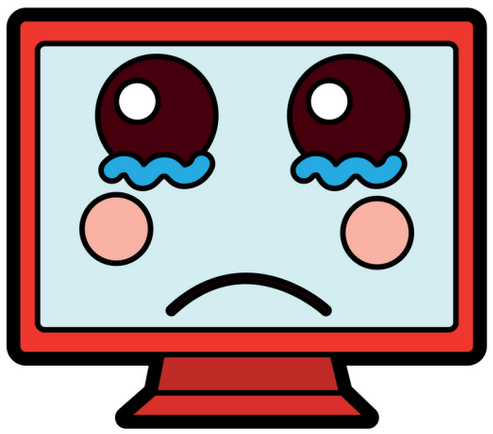
DNS attack: poisoning

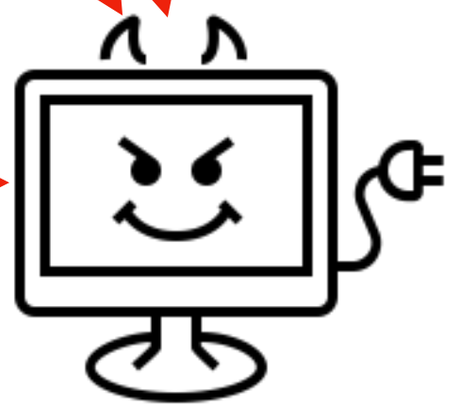


DNS attack: poisoning

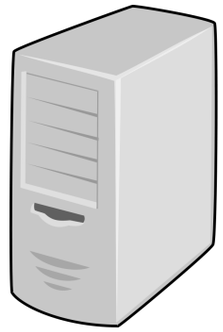




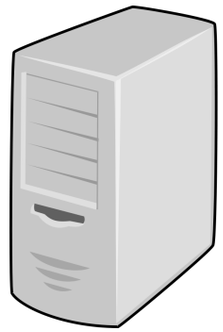




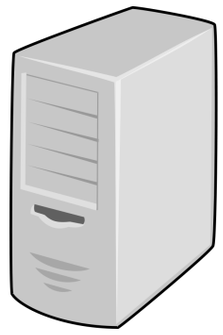
I am going to remember this in case anybody else asks!



Root name server



.com name server



google.com name server

DNS attack: poisoning

- You can point users to wherever you want!

- Google in Malaysia was redirected to a Madleets website

http://techcrunch.com/2013/10/10/google-malaysia-site-hacked-credit-claimed-by-team-madleets/

- Some countries (e.g., China, Turkey,…) use DNS poisoning for censorship

How to make it work

1. Query ID (QID) attack

   $ dig web.mit.edu mx

- Every DNS query has a **QID**
- DNS in connectionless (UDP)
- If Darth responds to query with the right QID, then it wins, over the real nameserver's response!
- **FIX**: randomize QID (16 bits)
- But Darth can send 1000s of fake responses. If client is forced to ask the same q 1000s of times (e.g., embed fake images) Darth wins!

```
; <<>> DiG 9.10.6 <<>> web.mit.edu mx
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 54503
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 1, ADDITIONAL: 1

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

;; ANSWER SECTION:
web.mit.edu.      831   IN CNAME www.mit.edu.edgekey.net.
www.mit.edu.edgekey.net. 5 IN CNAME e9566.dscb.akamaiedge.net.

;; AUTHORITY SECTION:
dscb.akamaiedge.net. 1000  IN SOA   n0dscb.akamaiedge.net. hostmaster.akamai.com.
1575726098 1000 1000 1000 1800

;; Query time: 2 msec
;; SERVER: 137.73.254.10#53(137.73.254.10)
;; WHEN: Sat Dec 07 13:41:38 GMT 2019
;; MSG SIZE  rcvd: 177
```

How to make it work

2.RRSet attack

- DNS response contains different **R**esource **R**ecord **S**ets, or RRSets. In particular, an “additional” section, where name server can give additional info that may be “useful” for future lookups.
  - e.g., in an iterative query, the .com nameserver says you can ask ns.example.com for the IP address of example.com. To help next request, an additional record might give IP for ns.example.com.
- Darth abuses this feature and adds an additional record that says the IP address of victim.com is 6.6.6.6
- **FIX**: Bailiwick checking (when asking for www.google.com, do not allow www.victim.com to be accepted as an additional record)

How to make it work

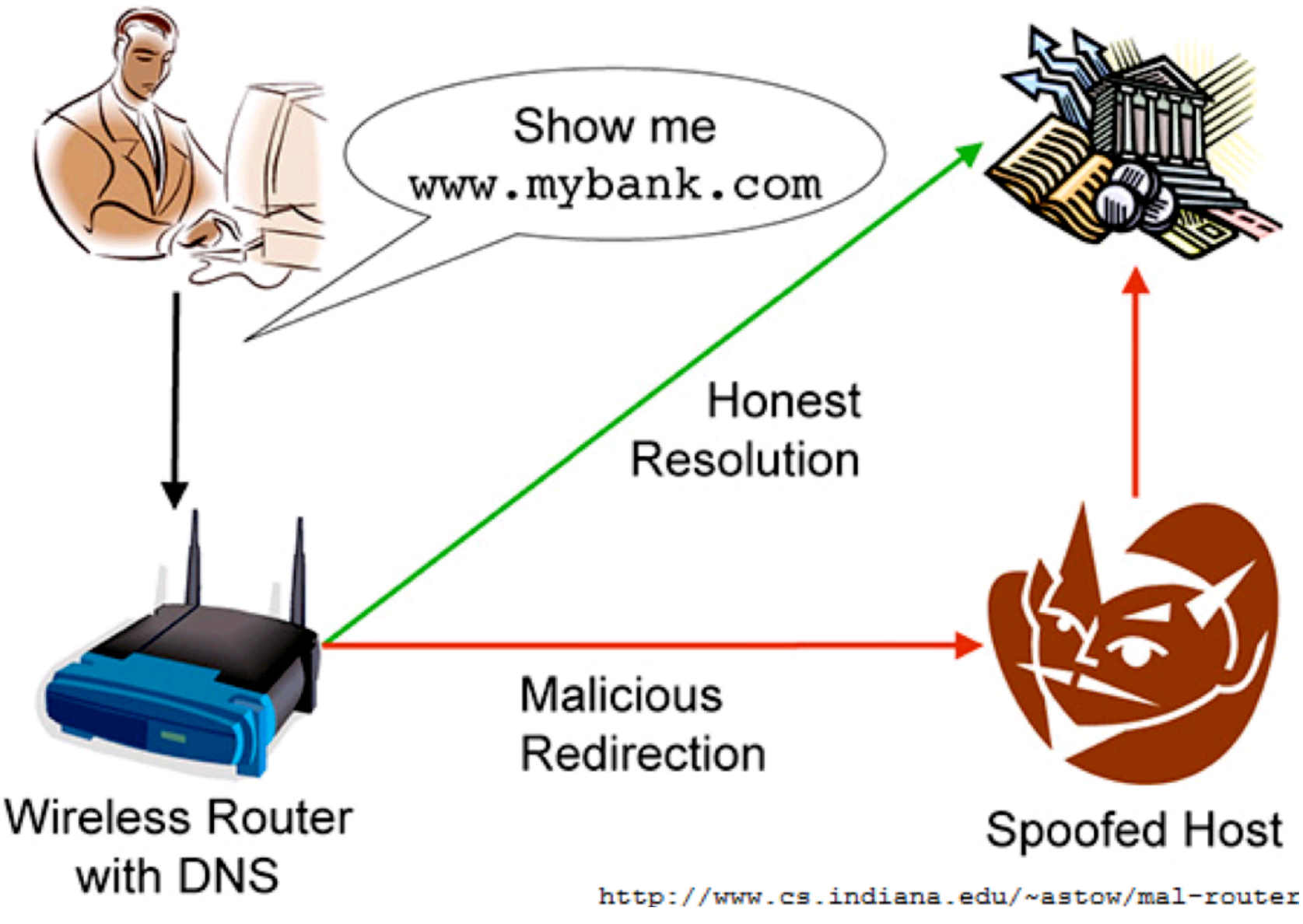
3.Dan Kaminsky's attack (2008)

- Bailiwick checking prevent Darth from creating 1000s of fake responses, or duping an innocent Alice from making 1000s of queries. But, for efficiency - when the resolver asks for www.google.com, additional records such as mail.google.com can still be set.
- Kaminsky's idea was to force resolver to query non-existent domains: aaaa.google.com, aaab.google.com etc... (e.g., thru embedded images)
- At the same time, Darth can send answers to each of these queries; with additional record for google.com in each case. In many cases, google.com's authoritative servers will beat Darth. But Darth only need to succeed once!

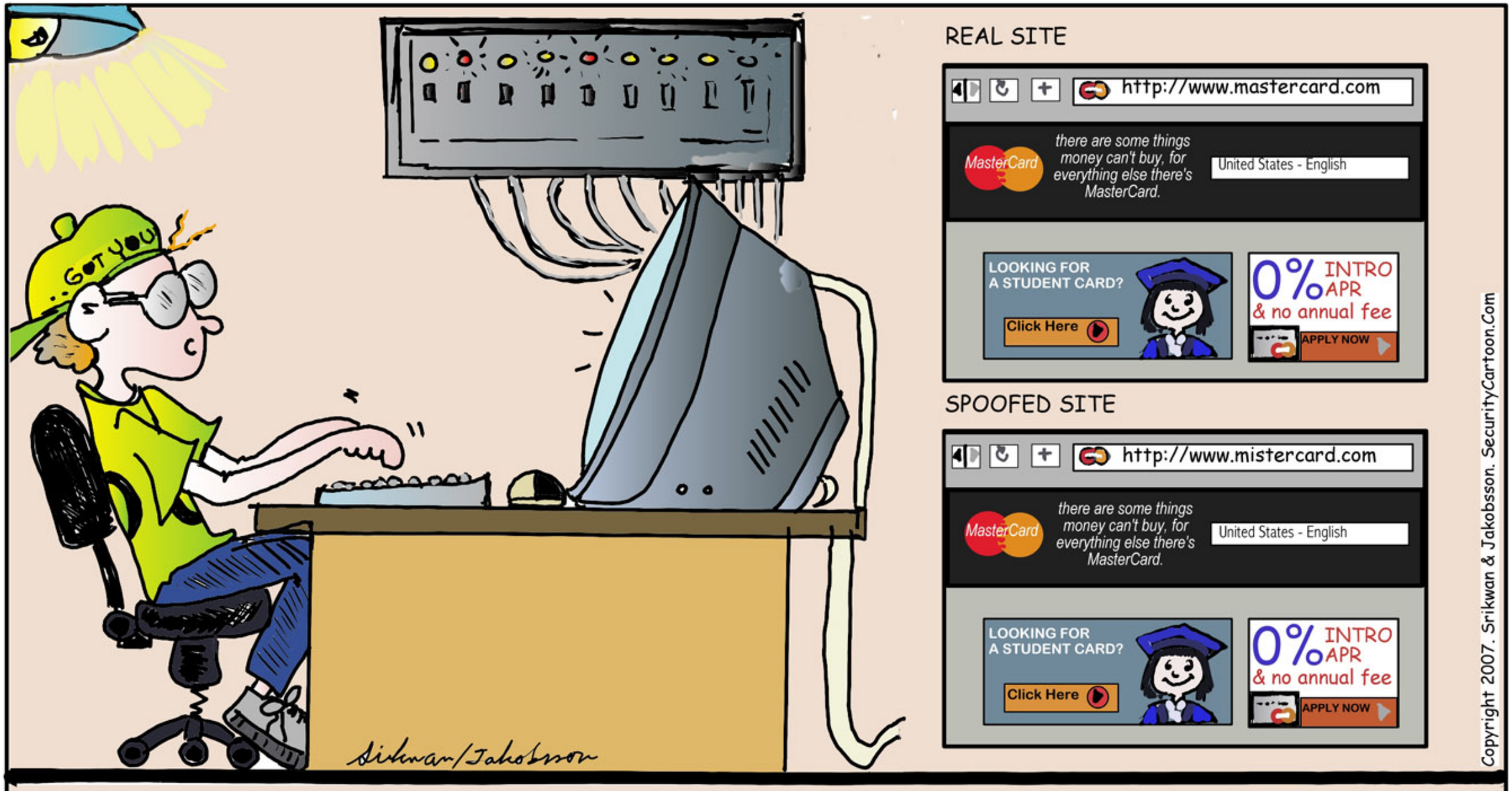
Details here: http://unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html

DNS attack: poisoning

- DNS poisoning typically is combined with website spoofing, to yield “results"



Phishing could be easier than poisoning



Bigger picture and countermeasures

- How to protect oneself on the internet?
- Where should identity be?
- Which IDs should be trusted?
- Some answers: IPSec, TLS,...

RECAP: Spoofing vs hijacking vs poisoning

- **Spoofing**: imitating someone else (humorous!)
- **Hijacking**: taking over what belongs to someone else, especially at “run time”
- **Poisoning**: contaminating some source of information (that is usually trusted to be good)

RECAP: Spoofing vs hijacking vs poisoning

- There are very subtle semantic differences, and also a matter of perspective. For example:

  1. Spoofing -> hijacking someone's identity?

  2. BGP Route hijacking aka Routing Table Poisoning?

  3. DNS Poisoning -> spoofing a domain owner?

- Name does not matter; simply an easy "handle" to think about and remember a kind of attack...

LAB time (EPISODE 3)

- DDoS attacks
  - `command: hping3`
  - `options: –flood`
- IP Spoofing
- DoS + IP Spoofing