

Networking 101

Utah Open Source Conference
October 8, 2009

Corey Edwards
Email: tensai@zmonkey.org
Twitter: [@hey tensai](https://twitter.com/hey tensai)
IRC: tensai



OSI Network Model

- ⇒ Created by ISO in 1977
- ⇒ Abstract model for how to write network software
- ⇒ Competed with IETF model, aka we-don't-need-no-steenkin'-models model
- ⇒ Is an ideal model and therefore has failures
- ⇒ Nobody actually implements it but it's useful for reference
- ⇒ ...and for test questions :)



OSI Model Layers

- ➔ Layer 1 – Physical
- ➔ Layer 2 – Data link
- ➔ Layer 3 – Network
- ➔ Layer 4 – Transport
- ➔ Layer 5 – Session
- ➔ Layer 6 – Presentation
- ➔ Layer 7 – Application



Layer 1

- ⇒ Physical
 - Cabling
 - Connectors
 - Voltages
 - Timing



Layer 2

- ⇒ Data link
 - Ethernet (802.3)
 - Token ring
 - 802.11a/b/g/n



Ethernet

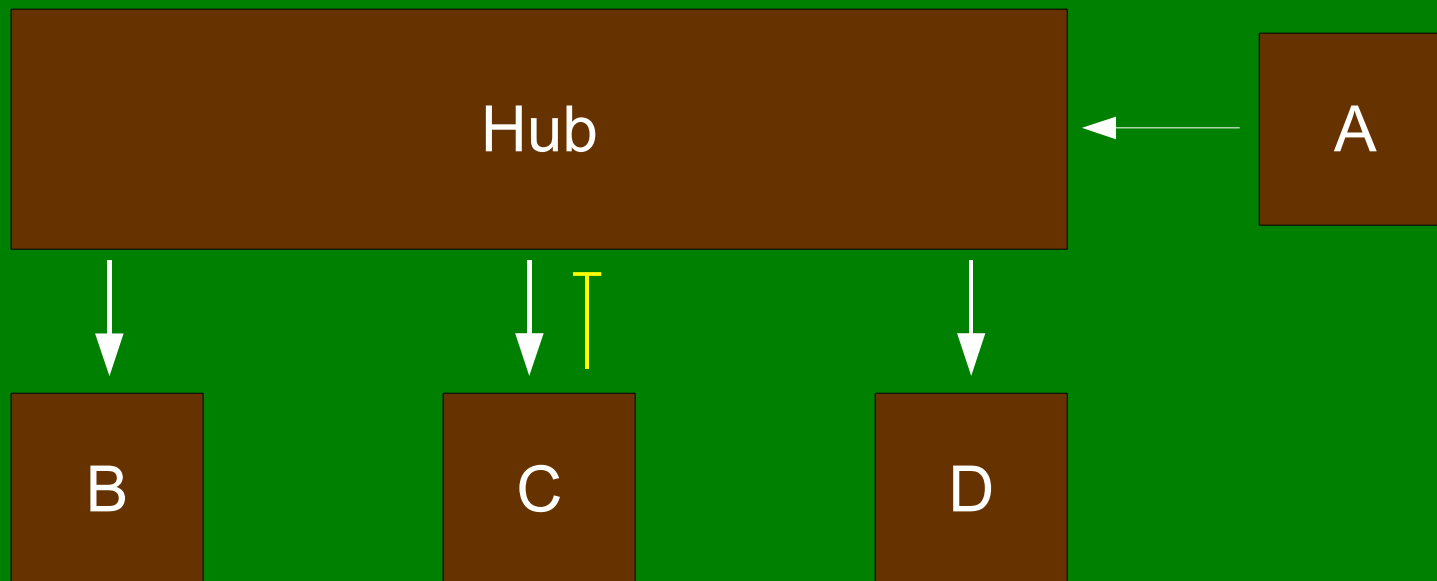
- ⇒ 48 bit address – media access control (MAC)
- ⇒ 24 bit vendor + 24 bit sequence
- ⇒ Example: 00:14:22:d9:4b:26
- ⇒ CSMA/CD
 - Carrier Sense, Multiple Access with Collision Detection
- ⇒ Broadcast: ff:ff:ff:ff:ff:ff



Ethernet Switching

⇒ Hubs

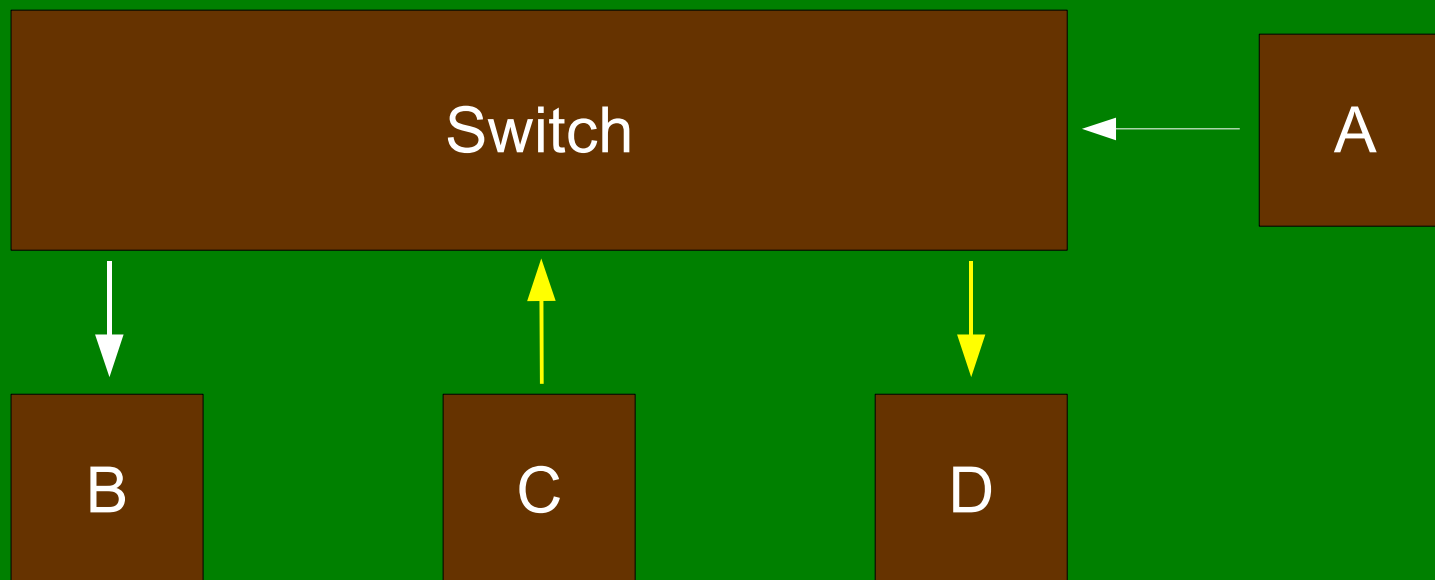
- Like a crowded room



Ethernet Switching

⇒ Switch

- “Cone of silence”



Layer 3

- ⇒ Network
 - IP
 - IPX



IPv4

⇒ 32 bit address

- Format: 172.16.0.1
- Old school: classful addressing
 - Class A: 10.0.0.0 – 10.255.255.255 (4 million)
 - Class B: 172.16.0.0 – 172.16.255.255 (65 thousand)
 - Class C: 192.168.0.0 – 192.168.0.255 (256)
 - Dead, dead, dead
- New school: Classless Interdomain Routing (CIDR)
 - (network bits) : (host bits)
 - 10.144.87.0/24
 - 24 network bits
 - $32 - 24 = 8$ host bits
 - $2^8 = 256$ hosts



IPv4

⇒ Subnetting

- 192.168.44.0/24
 - Network address: 192.168.45.0
 - Broadcast address: 192.168.45.255
 - Usable: 192.168.45.1 – 192.168.45.254
- 192.168.44.0/22 – 1024 addresses (1022 usable)
- 192.168.44.0/30 – 4 addresses (2 usable)



IPv6

⇒ Changes from IPv4

- 128 bit addresses
 - 3×10^{38} total addresses
 - 5×10^{28} per person
- Subnetting improved
 - No more unusable addresses
- Autoconfiguration
 - 64 bits provided by router
 - 64 bits generated, generally from MAC
- IPsec mandatory
- Multicast
- No fragmentation

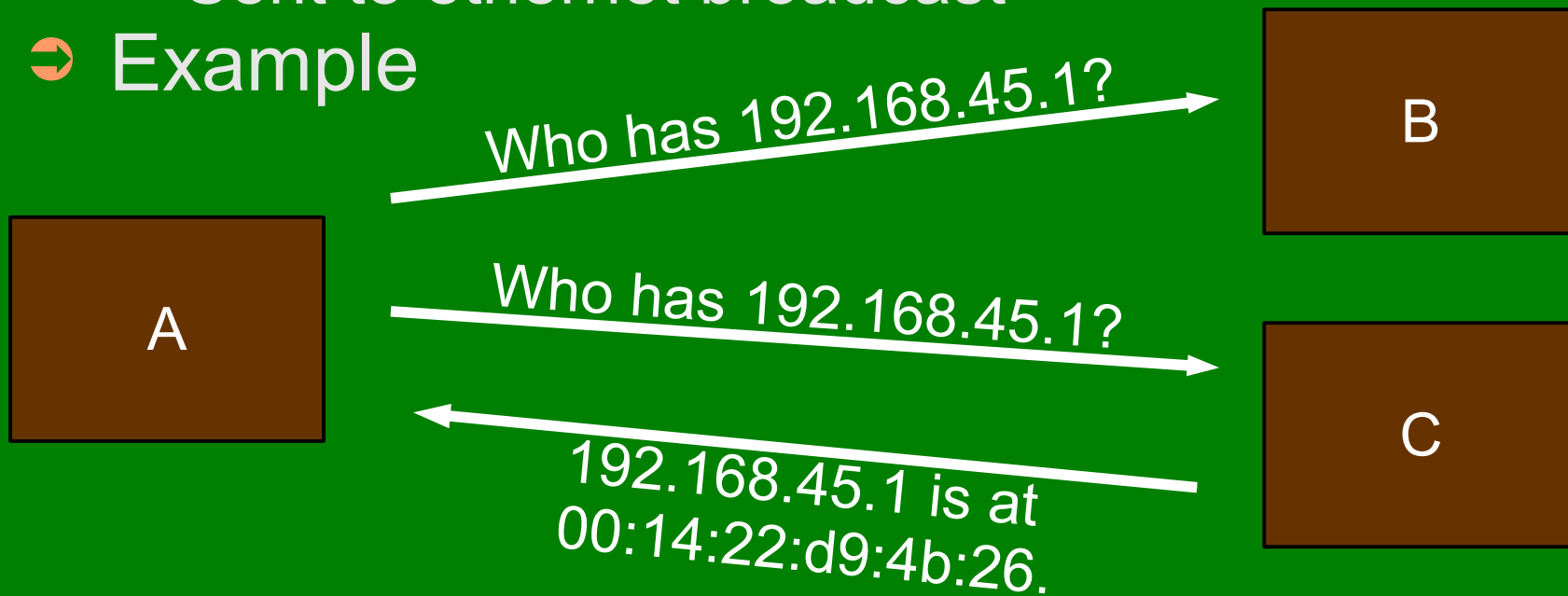


Bridging The Gap

⇒ ARP

- Address Resolution Protocol
- Layer 3 → Layer 2
- Sent to ethernet broadcast

⇒ Example



Layer 4

- ⇒ Transport
 - TCP
 - UDP
 - ICMP



ICMP

⇒ Error reporting

- Type, code
 - Code is optional
- Examples
 - 0 – echo response (aka pong)
 - 3 – destination unreachable
 - 5 – redirect
 - 8 – echo request (aka ping)
 - 11 – time to live (TTL) exceeded



UDP

⇒ Features

- Datagram oriented
- Source port
- Destination port
- Checksum

⇒ Disadvantages

- No flow control, no guarantees, no reliability

⇒ Advantages

- Lightweight
- Good for real time data (voice, video)
- DHCP, TFTP



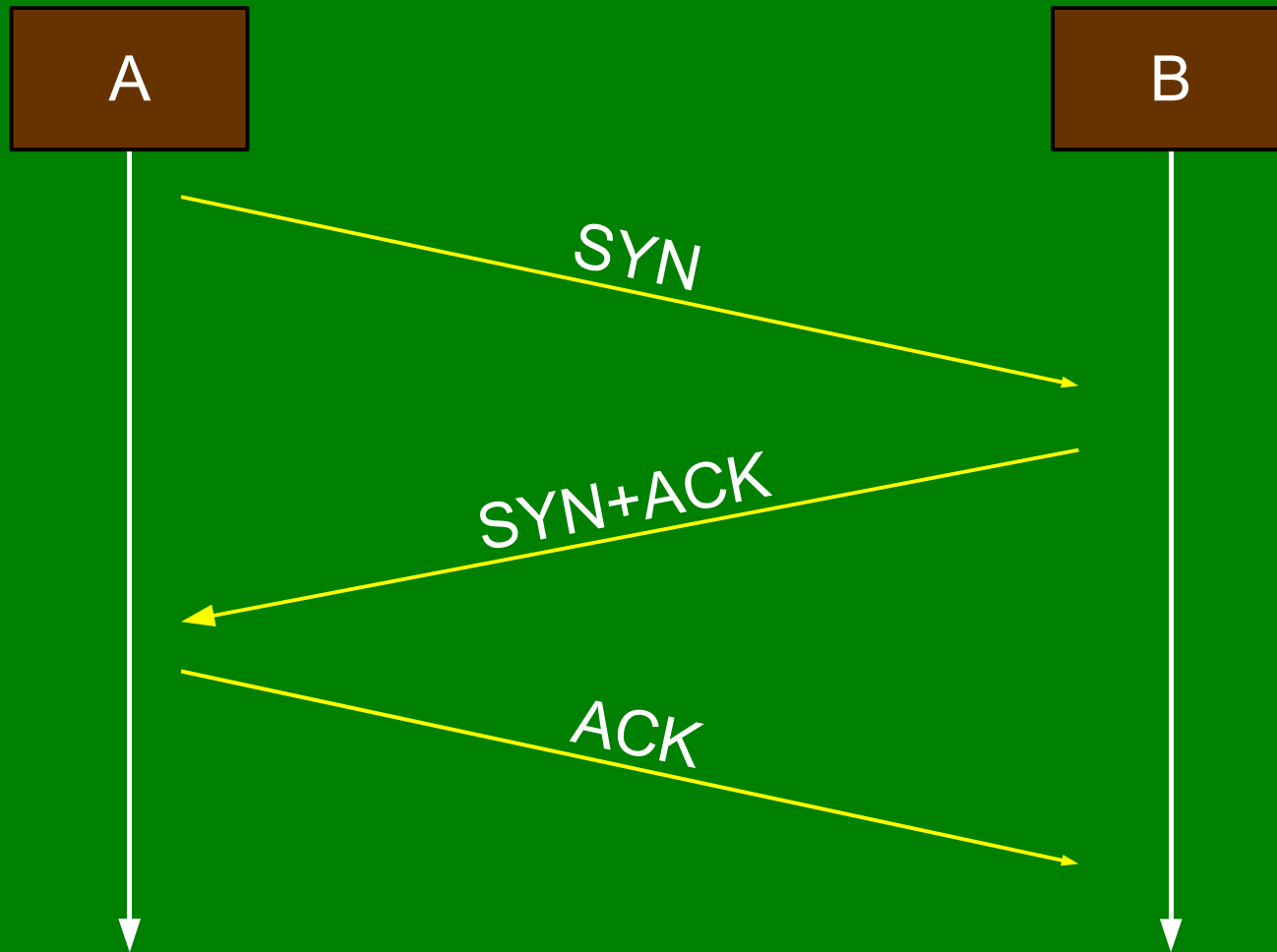
TCP

➔ Features

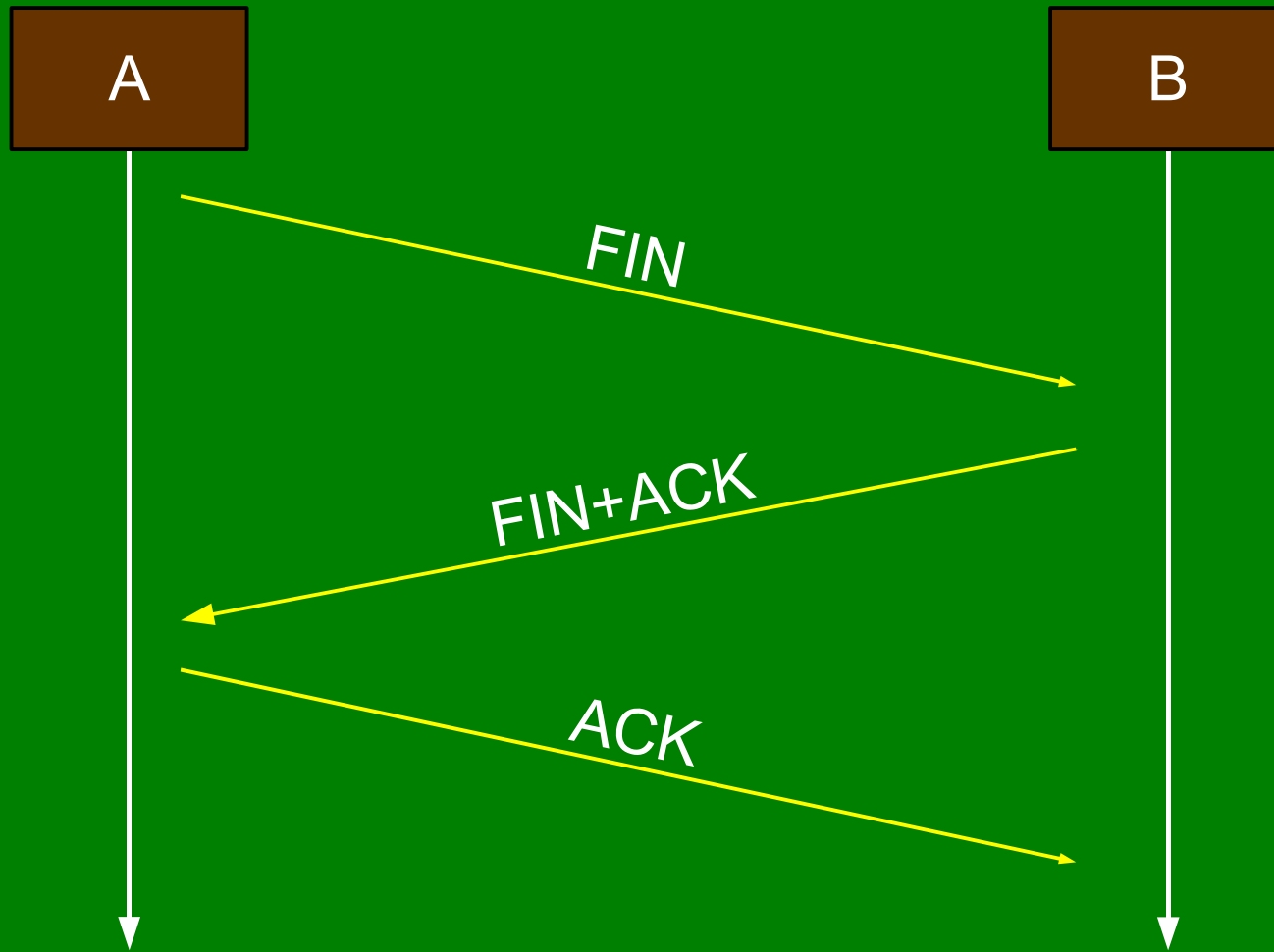
- Stream (byte) oriented
- Source port
- Destination port
- Sequence number
- Acknowledgement number
- Checksum
- 6 bit flags
 - SYN, ACK, FIN, RESET, PUSH, URG



TCP Setup



TCP Teardown



TCP

⇒ Advantages

- Reliable delivery
- Ordered delivery
- Flow control

⇒ Disadvantages

- Slow start
- Congestion avoidance
- Overhead
- Reordering/retransmission delay



Layer 5, 6, 7

- ⇒ Session, Presentation, Application
 - “meh”



HTTP

⇒ Features

- Text based
- Stateless
- Similar to POP, IMAP, SMTP, FTP, etc.



HTTP

➔ Example Request:

GET /index.html HTTP/1.1

Host: example.com

Connection: keep-alive

User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.9.0.13)
Gecko/2009080315 Ubuntu/9.04 (jaunty) Firefox/3.0.13

Referer: http://www.example.com/

Cookie: cookie1=foo; cookie2=bar



HTTP

➔ Example Response:

HTTP/1.1 200 OK

Date: Thu, Jan 1 2009 01:00:00 GMT

Connection: close

Server: Apache/2.2.3 (Debian) OpenSSL/0.9.8c

Content-Type: text/plain

Content-Length: 14

Hello World!



DNS

- ⇒ IP addresses are great and all, but who can remember them all?
- ⇒ DNS translates names into numbers, and numbers into names
- ⇒ Distributed system, no single point of failure
- ⇒ Names are resolved hierarchically
- ⇒ Each record has a timeout
- ⇒ Servers cache records as they pass through
- ⇒ Authoritative vs recursive

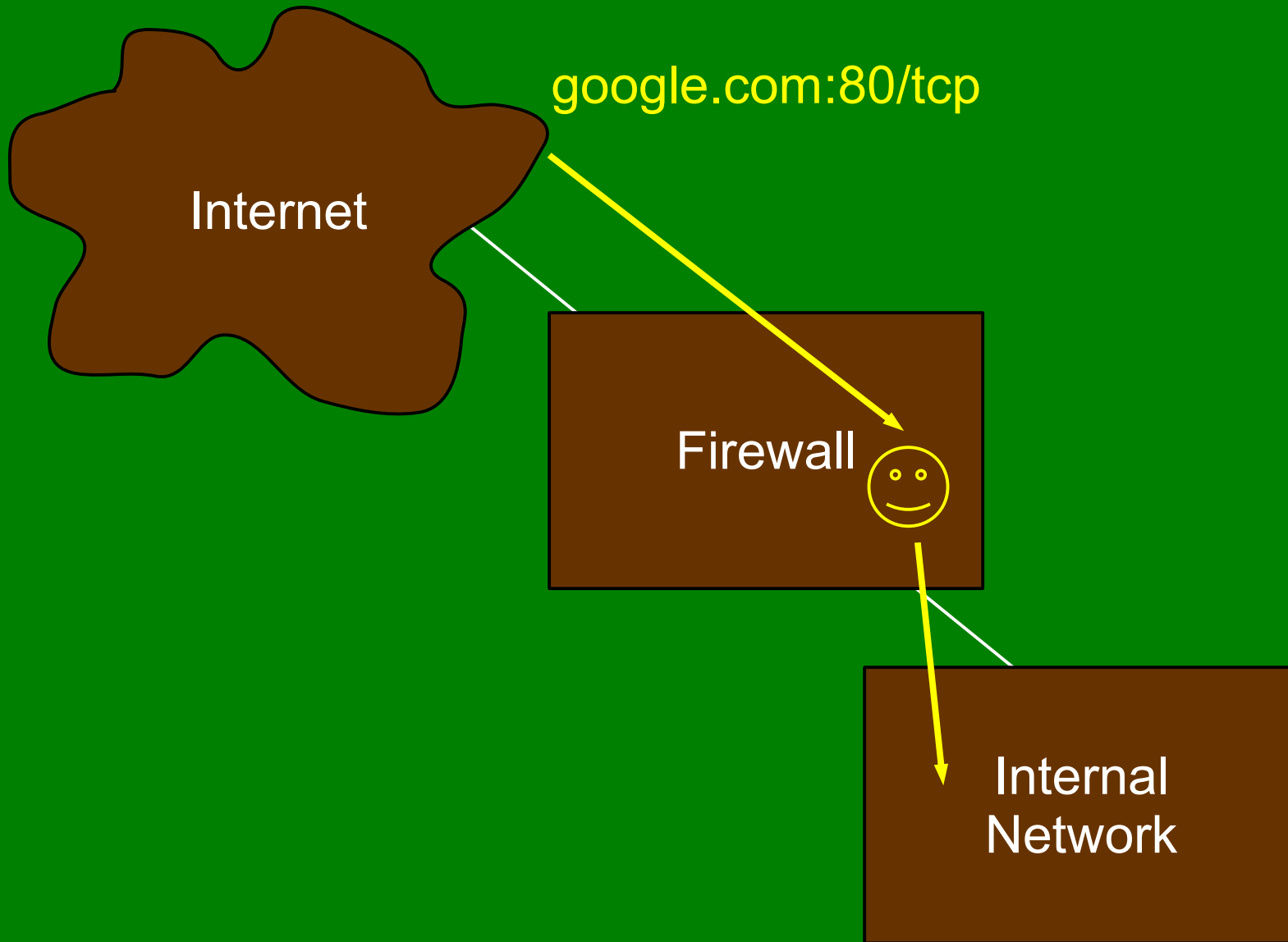


DNS Records

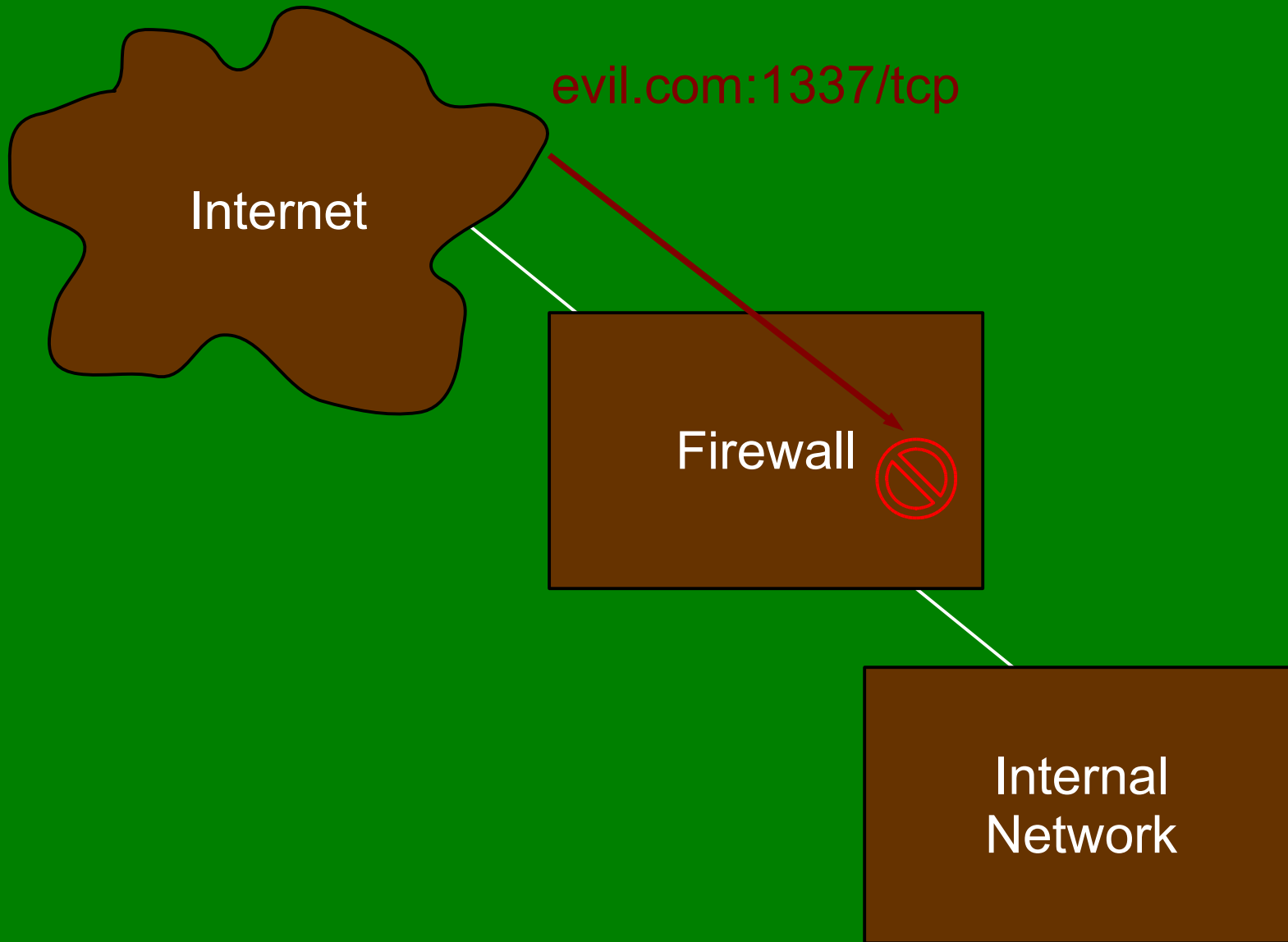
- ➔ Different types of DNS records
 - A - IP address
 - CNAME - Name alias
 - SOA - Start of authority. Zone details, e.g. time-outs, serial number.
 - NS - Name server
 - MX - Mail server
 - SRV - Service. General purpose MX record.
 - PTR - Reverse DNS
 - TXT - General purpose text entry



Firewalls



Firewalls



Linux Firewalls

⇒ iptables

- 3 tables: filter, nat, mangle
- filter table
 - INPUT
 - Packets destined for the host
 - OUTPUT
 - Packets generated by the host
 - FORWARD
 - Packets merely passing through
- Targets
 - ACCEPT, DROP, REJECT, RETURN, DNAT, SNAT, MASQUERADE



Linux Firewalls

⇒ Example:

- iptables -A INPUT -p tcp --dport 22 -s 10.0.0.0/24 -j ACCEPT
- iptables -A INPUT -p tcp --dport 80 -s 0/0 -j ACCEPT
- iptables -A INPUT -s 212.0.0.0/8 -j DROP
- iptables -P INPUT -j REJECT



The End

Questions?

Corey Edwards

Email: tensai@zmonkey.org

Twitter: [@hey tensai](https://twitter.com/hey tensai)

IRC: tensai

