Networking with TCP and HTTP

OR

How To Build The Internet

LIGHTHOUSE LABS

AGENDA

Layers of the Internet

TCP/IP & Below

Simple Chat Server

HTTP Fundamentals

LIGHTHOUSE LABS

Layers of the Internet

LIGHTHOUSE LABS

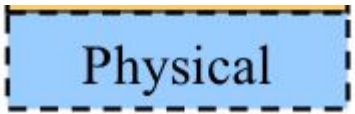
The Internet

- A the global system of interconnected computer networks

The Internet

- A the global system of interconnected computer networks

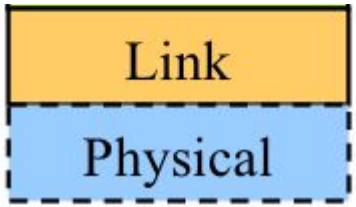
1. A Physical Network
    a. Copper Cables
    b. Fiber Optic Cables
    c. Satelites



The Internet

- A the global system of interconnected computer networks

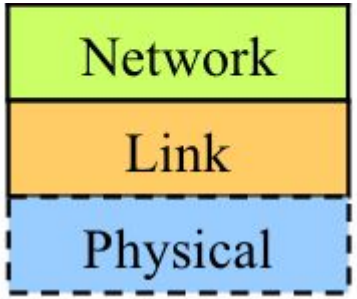
1. Physical Infrastructure
    a. Copper Cables
    b. Fiber Optic Cables
    c. Satelites
2. Protocols to link to the Physical Infrastructure
    a. Ethernet
    b. WiFi



The Internet

- A the global system of interconnected computer networks

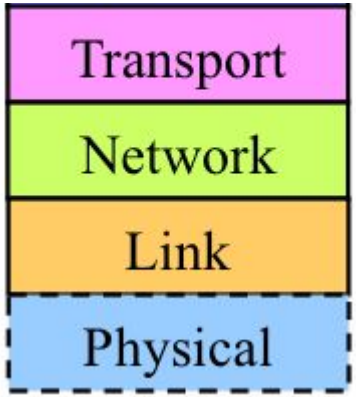
1. Physical Infrastructure
    a. Copper Cables
    b. Fiber Optic Cables
    c. Satelites
2. Protocols to link to the Physical Infrastructure
    a. Ethernet
    b. WiFi
3. The Software Network
    a. IPv4
    b. IPv6



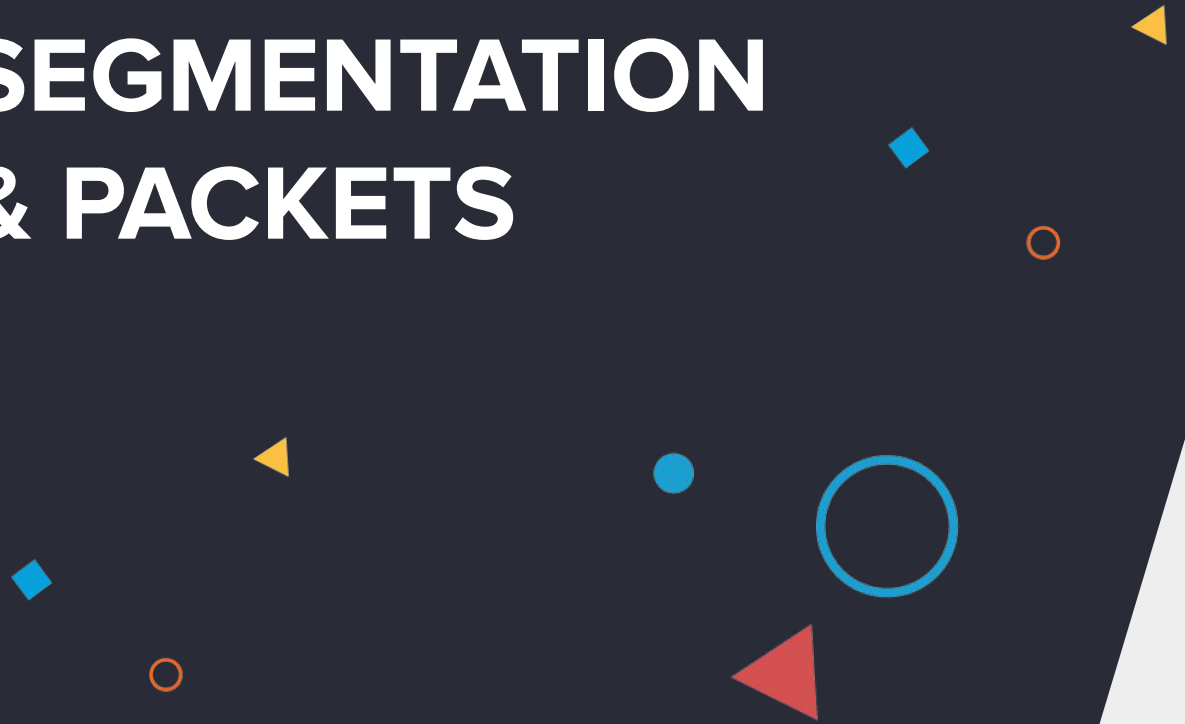
The Internet

- A the global system of interconnected computer networks

1. Physical Infrastructure
    a. Copper Cables
    b. Fiber Optic Cables
    c. Satelites
2. Protocols to link to the Physical Infrastructure
    a. Ethernet
    b. WiFi
3. Addresses and Routing Across the Network [Software]
    a. IPv4
    b. IPv6
4. Transport Protocols [Software]
    a. TCP
    b. UDP



SEGMENTATION & PACKETS



LIGHTHOUSE LABS

TCP/IP Segmentation

It’s much simpler to send fixed sized messages:

- Easy to confirm that the whole message has been received
- Each message takes less memory
- Less time tied up in sending/receiving

Segmentation

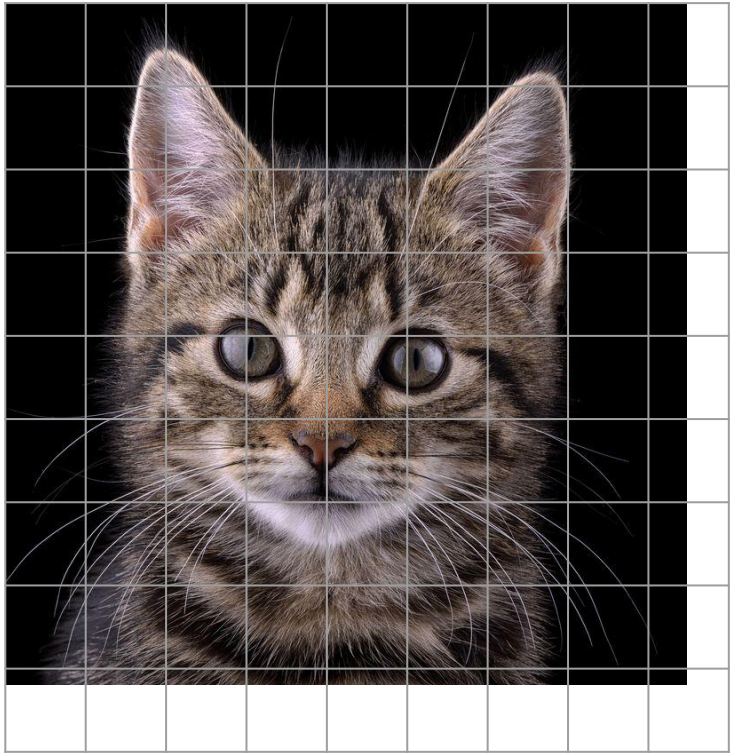
Segmentation

Segmentation

Segmentation

Segmentation

Segmentation



Packets - like an envelope

- A packet is made up of the segment of data and a variety of headers
  - IP information - who sent it, and where is it destined?
  - Sequence number - what segment of data is this
  - Acknowledgement number - to ensure reception
  - Checksum data - to ensure data integrity

TCP/IP

LIGHTHOUSE LABS

TCP

- TCP allows us to send a message made of **many** packets with guarantees.

- Your message will arrive **quickly and reliably**
  - Unreliable network retransmission - Dupack/Timeout
  - Error Detection - Hash functions
  - Packet Ordering - Sequence Numbers
  - Flow Control - Network Speed Mismatches
  - Congestion Control - Various Mechanism

Client / Server Communication

LIGHTHOUSE LABS

Demo

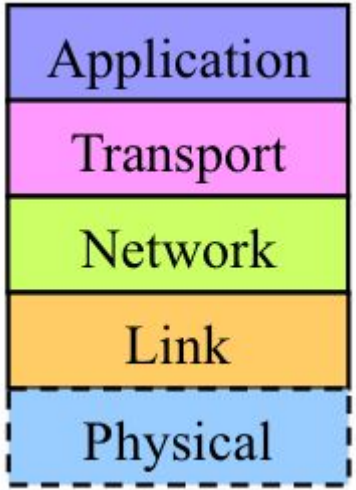
HTTP

LIGHTHOUSE LABS

The Internet

- A the global system of interconnected computer networks

1. Physical Infrastructure
   - Copper Cables, Fiber Optic Cables, Satelites
2. Protocols to link to the Physical Infrastructure
   - Ethernet, WiFi
3. Addresses and Routing Across the Network [Software]
   - IPv4, IPv6
4. Transport Protocols [Software]
   - TCP, UDP
5. Application Layer [Software]
   a. HTTP
   b. FTP
   c. telnet
   d. NFS
   e. SMTP
   f. SSH



What is HTTP

- HTTP is a request-response communication protocol
- It utilizes TCP for client-server communication
- HTTP is stateless

Your browser is the client:

- Request a website, image, js, css
- Upload a file
- Submit a form

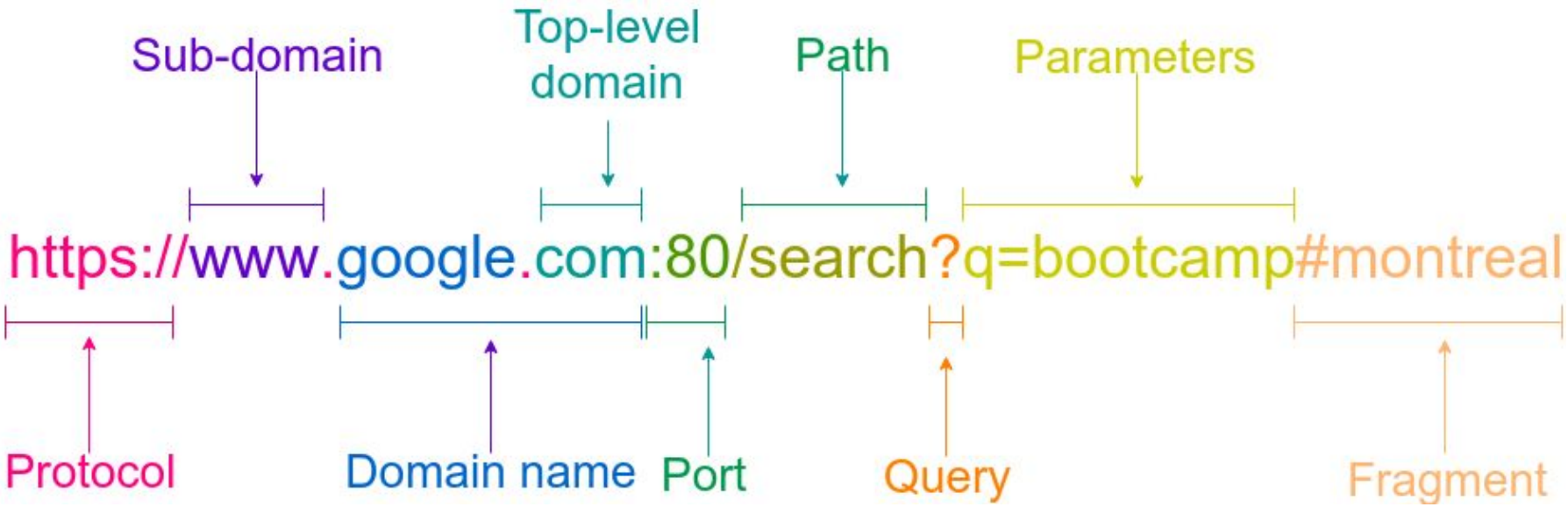
The website’s backend is the server:

- Respond with website, image, js, css
- Receive a form submission or file upload

Anatomy of HTTP request

- A URL
- A METHOD
- HEADERS [optional]
- BODY [optional]

URL


Sub-domain
Top-level domain
Path
Parameters
https://www.google.com:80/search?q=bootcamp#montreal
Protocol
Domain name
Port
Query
Fragment

METHOD

- GET - A request to **read** resource
- POST - A request to **write** resource
- PUT - A request to **update** resource
- PATCH - A request to **partially update** resource
- DELETE - A request to **delete** resource
- OPTION - A request to **read** information about a path

HEADERS

HTTP headers allow us to specify additional information in a request.

- Authorization - used to specify credentials
- Cookie - used to maintain session between client and server
- Caching - used to improve performance
- Accept - used to specify acceptable response types
- CORS - used to specify security preferences

BODY

HTTP allows you to specify a body along with your request.

- Usually used with POST/PUT/PATCH requests to **write** a resource
- This payload could contain
  - Form data - automatically generated from a form submission or through JS/AJAX
  - structured data like JSON through JS/AJAX
  - Multipart form data - for uploading larger images through your browser

Anatomy of HTTP response

- A STATUS
- HEADERS [optional]
- BODY [optional]

HTTP STATUS CODE

- 200 range: success
  - 200 OK - generic success
  - 201 CREATED - your resource was created
  - 202 ACCEPTED - request received, but not yet completed

HTTP STATUS CODE

- 200 range: success
  - 200 OK - generic success
  - 201 CREATED - your resource was created
  - 202 ACCEPTED - request received, but not yet completed
- 300 range: redirect
  - 301 MOVED PERMANENTLY - your browser will automatically redirect
  - 302 FOUND - temporary redirection, your browser will automatically redirect

HTTP STATUS CODE

- 200 range: success
  - 200 OK - generic success
  - 201 CREATED - your resource was created
  - 202 ACCEPTED - request received, but not yet completed
- 300 range: redirect
  - 301 MOVED PERMANENTLY - your browser will automatically redirect
  - 302 FOUND - temporary redirection, your browser will automatically redirect
- 400 range: error
  - 400 BAD REQUEST - something is wrong with your request
  - 401 UNAUTHORIZED - server couldn’t validate your identity
  - 404 NOT FOUND - server doesn’t recognize path

HTTP STATUS CODE

- 200 range: success
  - 200 OK - generic success
  - 201 CREATED - your resource was created
  - 202 ACCEPTED - request received, but not yet completed
- 300 range: redirect
  - 301 MOVED PERMANENTLY - your browser will automatically redirect
  - 302 FOUND - temporary redirection, your browser will automatically redirect
- 400 range: error
  - 400 BAD REQUEST - something is wrong with your request
  - 401 UNAUTHORIZED - server couldn’t validate your identity
  - 404 NOT FOUND - server doesn’t recognize path
- 500 range: server error
  - 500 INTERNAL SERVER ERROR - something is wrong with the server

BODY

The body of an HTTP response is the data you requested

- HTML/CSS/JS
- JSON - structured data
- Image

Your browser will receive this data, and do it’s best to display it to the user

Questions?

LIGHTHOUSE LABS