CSE461 Section

HTTP Tips and Tricks

First: Networking Jokes and Trivia

- UDP
- HTTP
- HTTP 418
  - “The HTCPCP server is a teapot; the resulting entity body may be short and stout.”
  - “There is a strong, dark, rich requirement for a protocol designed espressoly for the brewing of coffee.”

HTTP Request Types

- GET
  - Page, please
- HEAD
  - Metadata, please
- POST
  - Here's some data—update what you have
- PUT
  - Here's some data—create a new resource
- CONNECT
  - Make a tunnel for me through your proxy!

CONNECT and Tunneling

- CONNECT requests say, “give me a tunnel!”
- Used for communication via proxies
- It’s a virtual connection—it means “forward all of the packets that I send to the destination, and vice-versa
- You don’t *have* to use CONNECT with proxies…
- But you do when HTTPS/SSL is used… why?

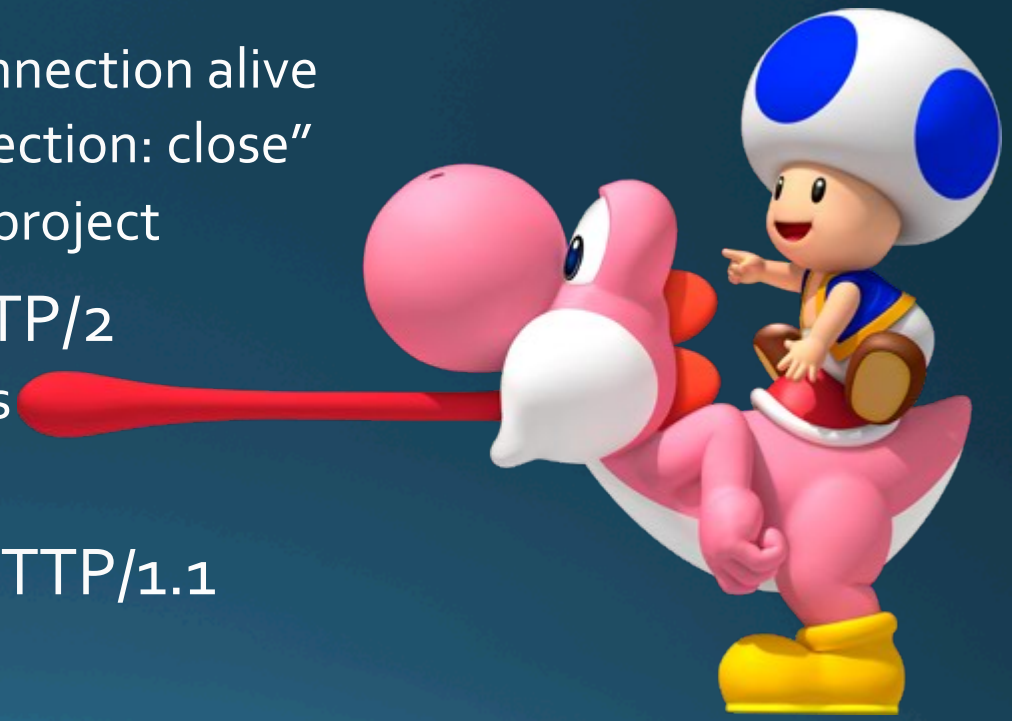
HTTP Headers: Host

- Specifies the hostname (and, optionally, the port) to which to send a request

- Example:
  GET /Passport.aspx?popup=1 HTTP/1.1
  Host: www.bing.com
  User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:34.0) Gecko/20100101 Firefox/34.0
  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
  Accept-Language: en-US,en;q=0.5

HTTP Headers: Connection

- Standard header for keep-alives
  - "Connection: keep-alive" is sent to keep a connection alive
  - If you don't want to keep it alive, send "Connection: close"
  - We'll rewrite packets to do this for the proxy project
- Also used to switch from HTTP /1.1 to HTTP/2
  - Client sends "Connection: upgrade" to do this
- Example:
  GET /ajax/libs/jquery/1.7.1/jquery.min.js HTTP/1.1
  Host: ajax.googleapis.com
  Connection: keep-alive
  Accept: */*



HTTP Headers: Proxy-Connection

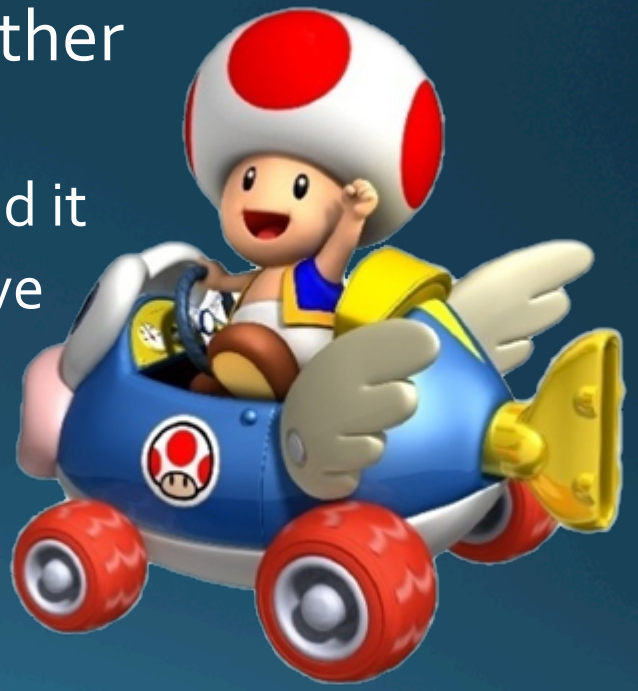
- Same functionality as a “Connection” field in the header
- Actually came from a mistake made by Netscape developers; it wasn’t in the standard but they used it anyway
- Now seen occasionally, but does the same thing as “Connection”

Buffering Requests

- When writing a proxy, you may want to buffer requests
- This is a good idea to keep your network traffic and thread consumption reasonable
- (Proxies that don't buffer often crash)

TCP Streams

- UDP sends a string of messages; no concept of "connection"
- TCP is like a direct pipe from one node to another
  - No concept of message boundaries
  - Easy to associate data with the data that preceded it
  - Gives you guarantees that data you send will arrive



Binary Streams vs. Text Streams

- Important when writing code to think about if you're reading text or arbitrary binary data
- If you use APIs for reading text buffers to read your binary headers, you're going to have a bad time
  - Binary bytes can be misinterpreted as text-related commands by text buffer readers and not properly received
  - Text buffer readers may try to read in UTF-8, which formats byte strings very differently and will cause issues
- In Java, this means use ByteStream instead of Inputstream
- With Python, if you use socket.recv() you shouldn't have problems

Gateways

- General term for network nodes that bridge a boundary between networks that use different protocols
- Often package internet packets to travel and get routed through networks where they wouldn't normally work
- In Tor, your Tor nodes will act as gateways

Real-Life Demo