

CS 4410  
Operating Systems

Networking:  
Application Layer

Summer 2016  
Cornell University

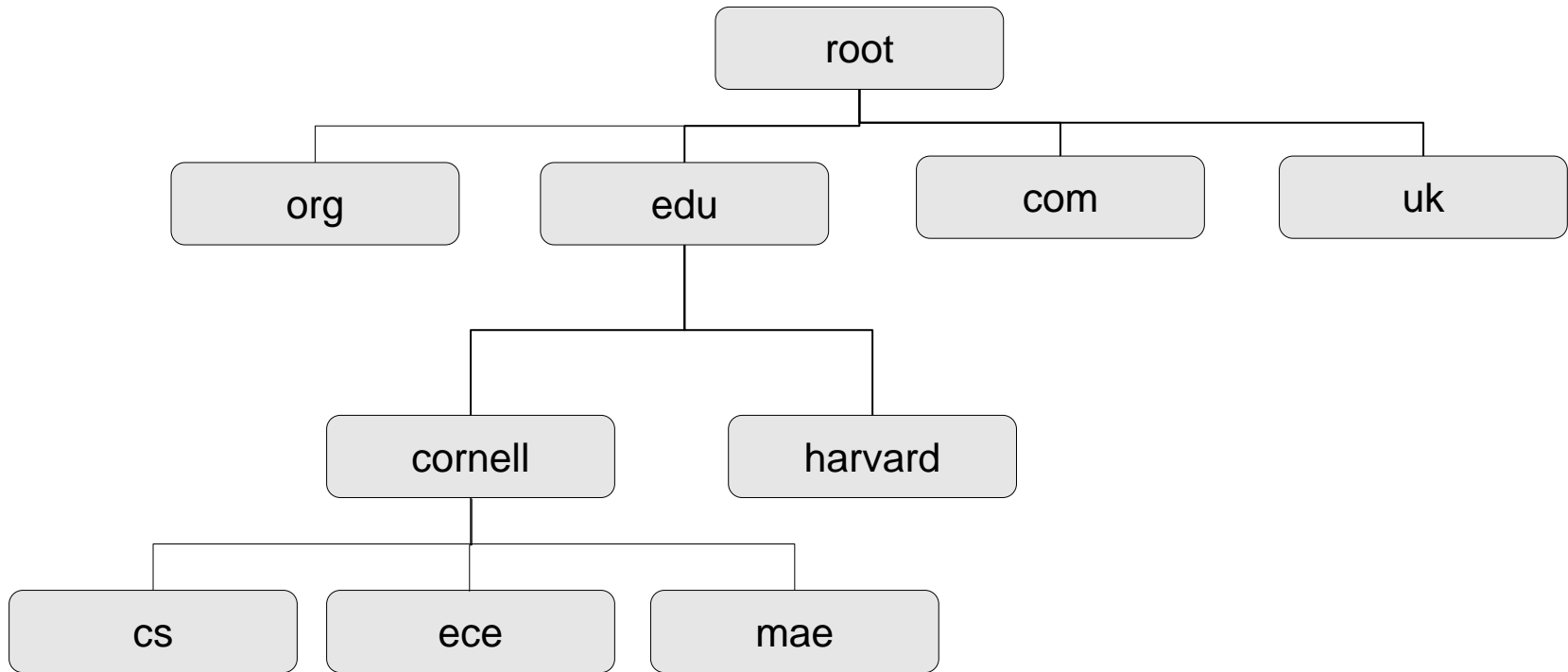
# Today

- Two application-layer protocols:
  - DNS
  - HTTP

# Domain Name Service

- When a user wants to communicate with a remote node, is it easier to remember 69.63.176.13 or [www.facebook.com](http://www.facebook.com)?
- Human-friendly names are given to nodes.
- Simultaneously, a translation mechanism should exist between names and IP addresses.
  - Packets need IP addresses to be transmitted.
- Will we use arbitrary or well-structured names? Why?
  - It affects the efficiency of the translation mechanism.
- Nowadays, we use Domain Names.
  - Well-structured strings.
  - Multiple labels separated by dots.
  - They create a hierarchical Domain Name Space.
- Use:
  - Emails, web sites

# DNS



- Hierarchical Domain Space
  - Every domain name should belong to this tree.
- When a process wants to send data to [www.facebook.com](http://www.facebook.com), somebody needs to provide the IP translation. Who?
- Domain Name Service (DNS)

# DNS

- Can we have only one server in the network which would provide the translations of all domain names? Why?
- DNS uses multiple DNS resolvers (servers) in the network and the translations are distributed among them. How?
- Each DNS resolver is responsible for a subset of the Domain Name space.

# DNS

- Example:
  - Suppose we want to access [www.cs.cornell.edu](http://www.cs.cornell.edu).
  - First, the DNS resolver in the OS contacts root DNS server and asks if it knows the translation of [www.cs.cornell.edu](http://www.cs.cornell.edu).
  - The root does not know the translation, but it knows the DNS server that is responsible for .edu addresses. Thus, it forwards the request there.
  - The DNS server for .edu does not know the translation, but it knows the DNS server responsible for the cornell.edu addresses. Thus, it forwards the request there.
  - The DNS server for cornell.edu does not know the translation, but it knows the DNS server responsible for the cs.cornell.edu addresses. Thus, it forwards the request there.
  - Finally, the DNS server for cs.cornell.edu addresses sends the IP address of the web site [www.cs.cornell.edu](http://www.cs.cornell.edu).
- Improvement:
  - One or more of these DNS servers may have cached the translation from previous requests, accelerating the DNS query.

# HTTP

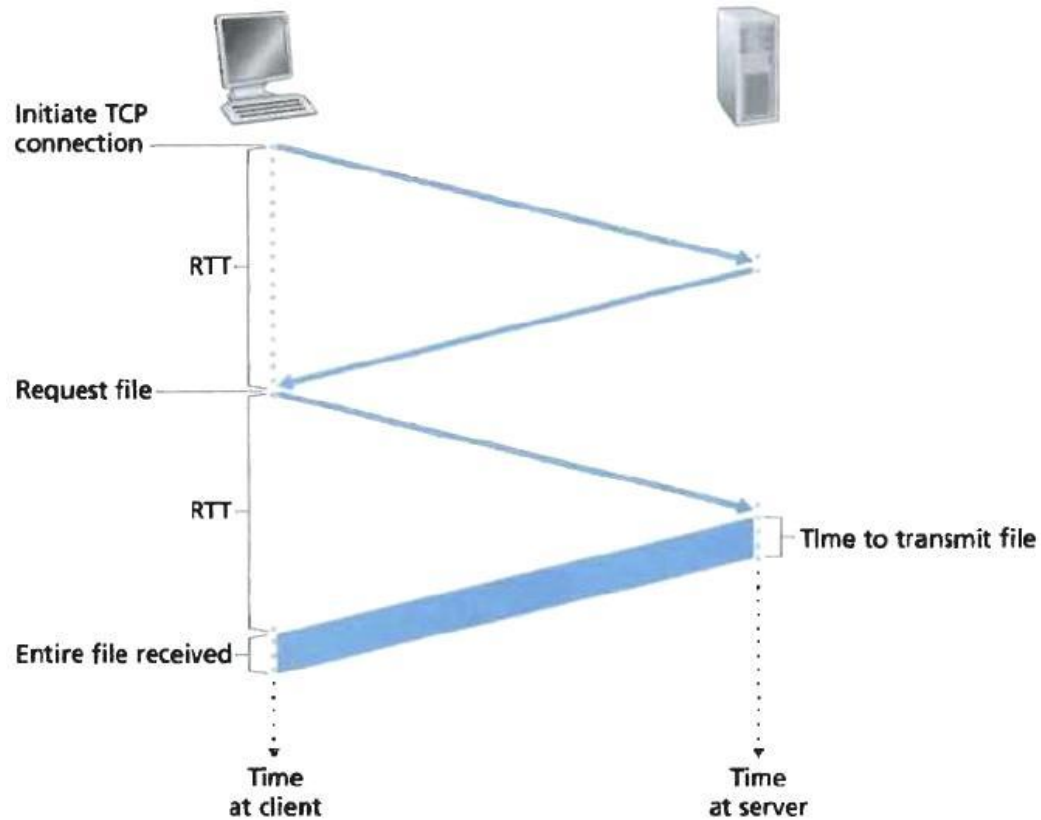
- HyperText Transfer Protocol
  - The Web's applications-layer protocol.
- <https://studentessentials.cornell.edu/>
- HTTP defines how Web clients request Web pages from Web servers and how servers transfer Web pages to clients.
- HTTP uses TCP as its underlying transport protocol.

# HTTP

- A user requests a Web page.
- The browser sends HTTP requests messages for the objects in the page to the server.
- The server receives the requests and responds with HTTP response messages that contain the objects.



# HTTP



# HTTP

- Request Message:

```
GET /somedir/page.html HTTP/1.1
Host: www.someschool.edu
Connection: close
User-agent: Mozilla/4.0
Accept-language: en
```

- Response Message:

```
HTTP/1.1 200 OK
Connection: close
Date: Thu, 07 Jul 2007
    12:00:15 GMT
Server: Apache/1.3.0
Last-Modified: Thu, 07
    Jul 2007 12:00:15 GMT
Content-Length: 6821
Content-Type: text/html

(data data ... data)
```

# End-to-End Argument

- Should the network guarantee packet delivery?
  - Think about a file transfer program.
  - Read file from disk, send it, the receiver reads packets and writes them to the disk.
- If the network guaranteed packet delivery, one might think that the applications would be simpler
  - No need to worry about retransmits.
  - But still need to check that the file was written to the remote disk intact.
- A check is necessary if nodes can fail
  - Consequently, applications need to be written to perform their own retransmits.
  - No need to burden the internals of the network with properties that can, and must, be implemented at the periphery.

# End-to-End Argument

- Application-specific properties are best provided by the applications, not the network.
  - Guaranteed, or ordered, packet delivery, duplicate suppression, security, etc.
- The Internet performs the simplest packet routing and delivery service it can.
  - Packets are sent on a best-effort basis.
  - Higher-level applications do the rest.

# Today

- Two application-layer protocols:
  - DNS
  - HTTP

# Coming up...

- Next lecture: Security