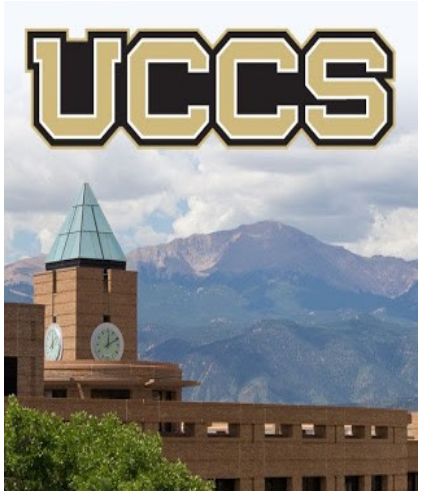
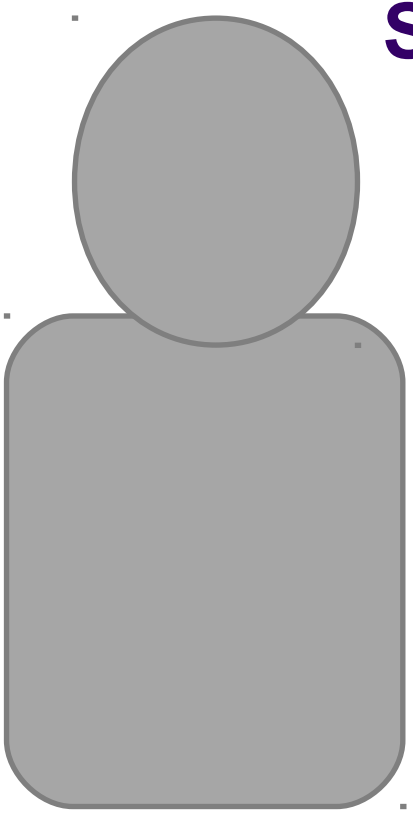


Unit 01.01.03
CS 5220:
COMPUTER COMMUNICATIONS

Examples of Protocols and Services

XIAOBO ZHOU, Ph.D.

Professor, Department of Computer Science



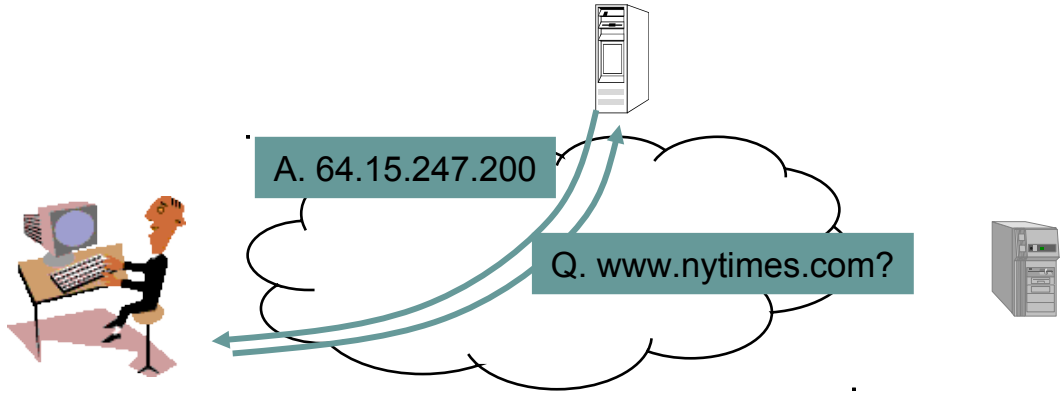
Services & Applications

- Service: information transfer capability
  - Internet transfer of individual block of information
  - Internet reliable transfer of a stream of bytes
  - Real-time transfer of a voice signal
- Applications build on communication services
  - E-mail & web build on reliable stream service
- New applications build on multiple networks
  - SMS builds on Internet reliable stream service and cellular telephone text messaging

Layers, Services & Protocols

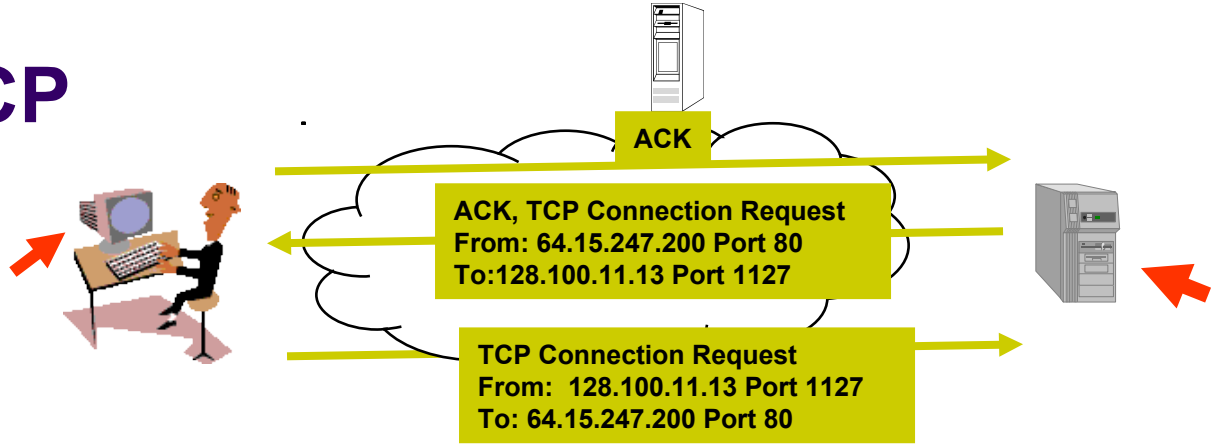
- The overall communications process between machines connected across one or more networks is very complex
- ***Layering*** partitions related communications functions into groups that are manageable
- Each layer provides a ***service*** to the layer above
- Each layer operates according to a ***protocol***

DNS



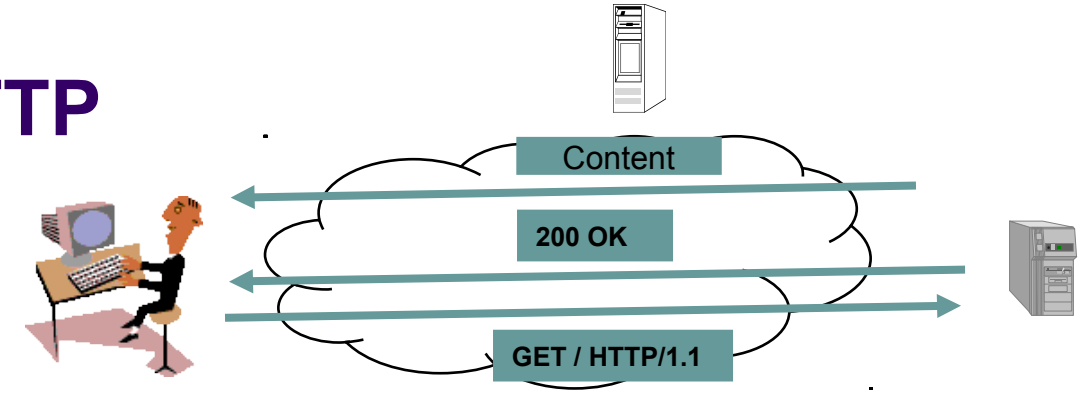
- User clicks on http://www.nytimes.com/
- URL contains Internet name of machine (www.nytimes.com), but not Internet address
- Internet needs Internet address to send information to a machine
- Browser software uses Domain Name System (DNS) protocol to send query for Internet address
- DNS system responds with Internet address

2. TCP

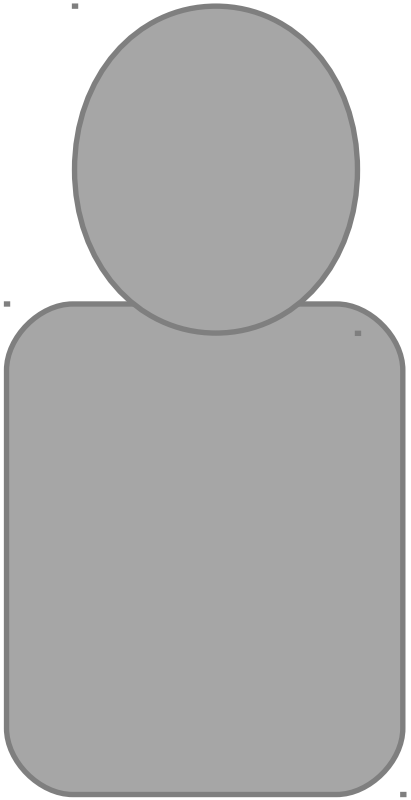


- Browser software uses HTTP to send request for document
- HTTP server waits for requests by listening to a well-known port number (80 for HTTP)
- HTTP client sends request messages through an “ephemeral port number,” e.g. 1127
- HTTP needs a Transmission Control Protocol (TCP) connection between the HTTP client and HTTP server to transfer messages reliably

3. HTTP



- HTTP client sends its request message: “GET comm.html …”
- HTTP server sends a status response: “200 OK”
- HTTP server sends requested file
- Browser displays document
- Clicking a link sets off a chain of events across the Internet!



Protocols

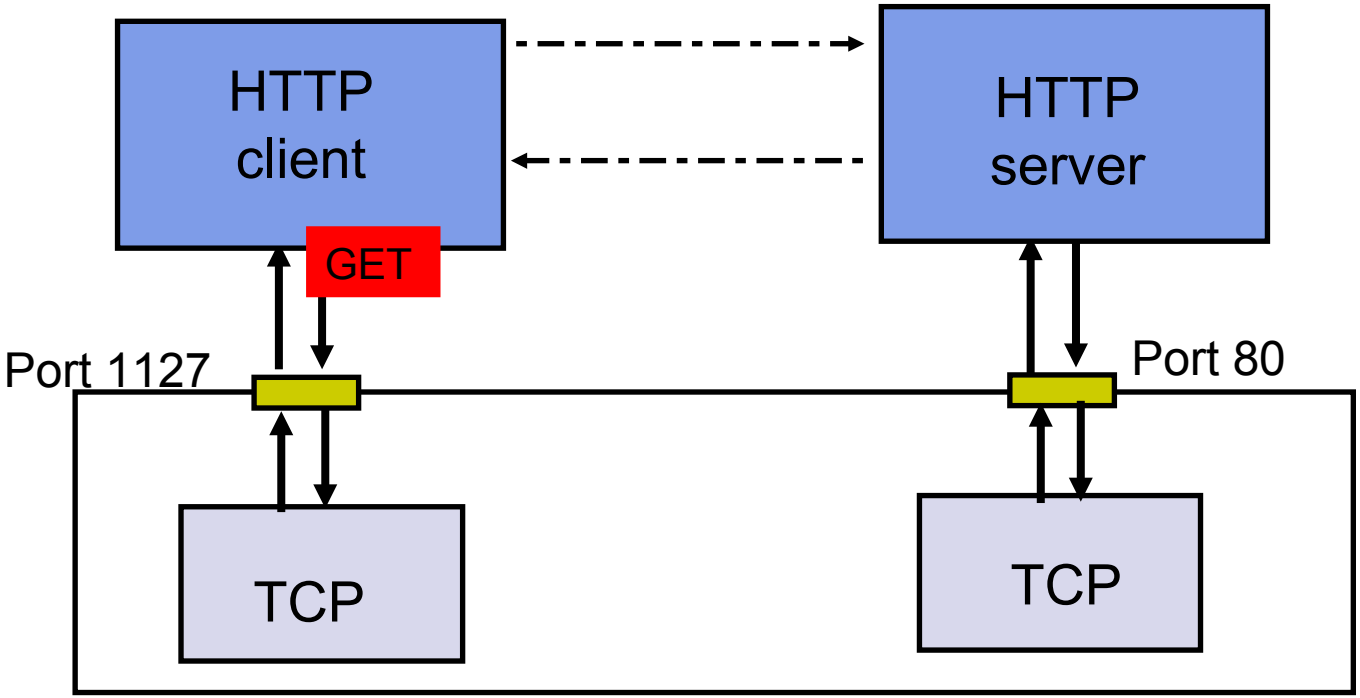
- A *protocol* is a set of precise & unambiguous rules that governs
  - how two or more communicating entities in a layer are to interact
  - *Messages* that can be sent and received
  - *Actions* that are to be taken when a certain event occurs

The purpose of a protocol is to provide a service to the layer above

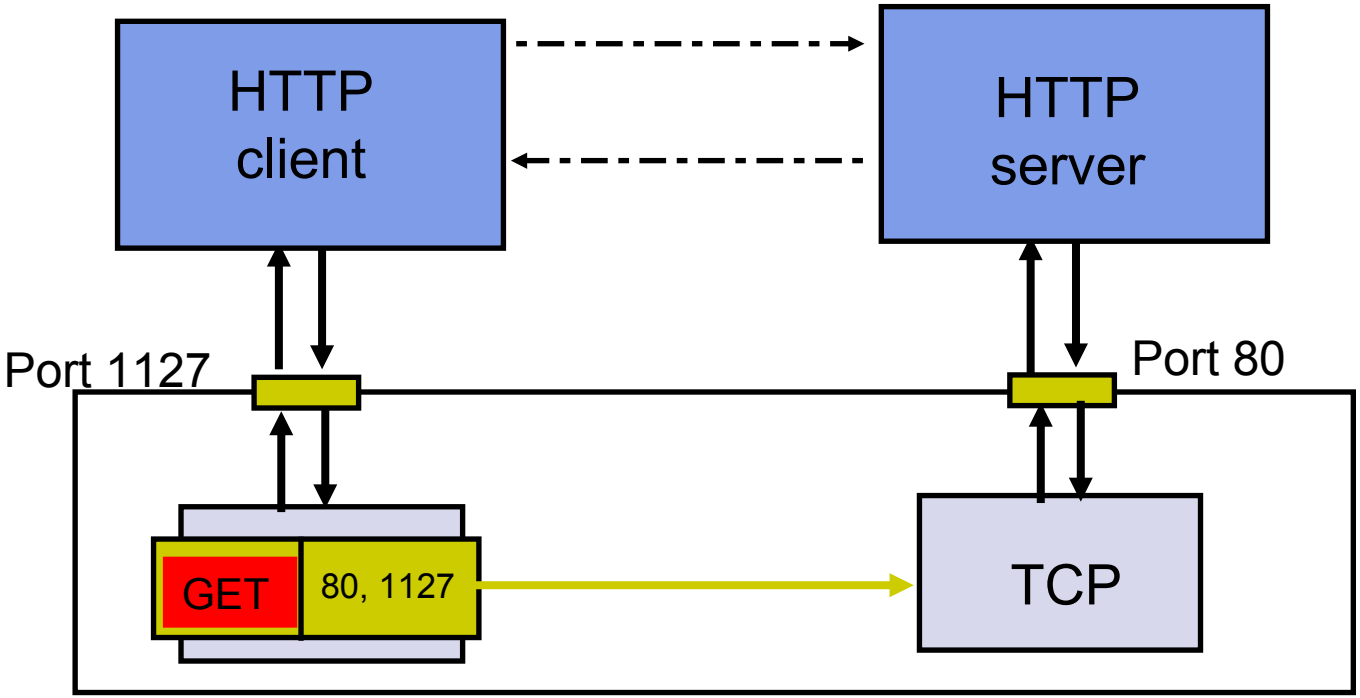
Example: HTTP

- HTTP is an application layer protocol
- Retrieves documents on behalf of a browser application program
- HTTP specifies fields in request messages and response messages
  - Request types; Response codes
  - Content type, options, cookies, …
- HTTP specifies actions to be taken upon receipt of certain messages

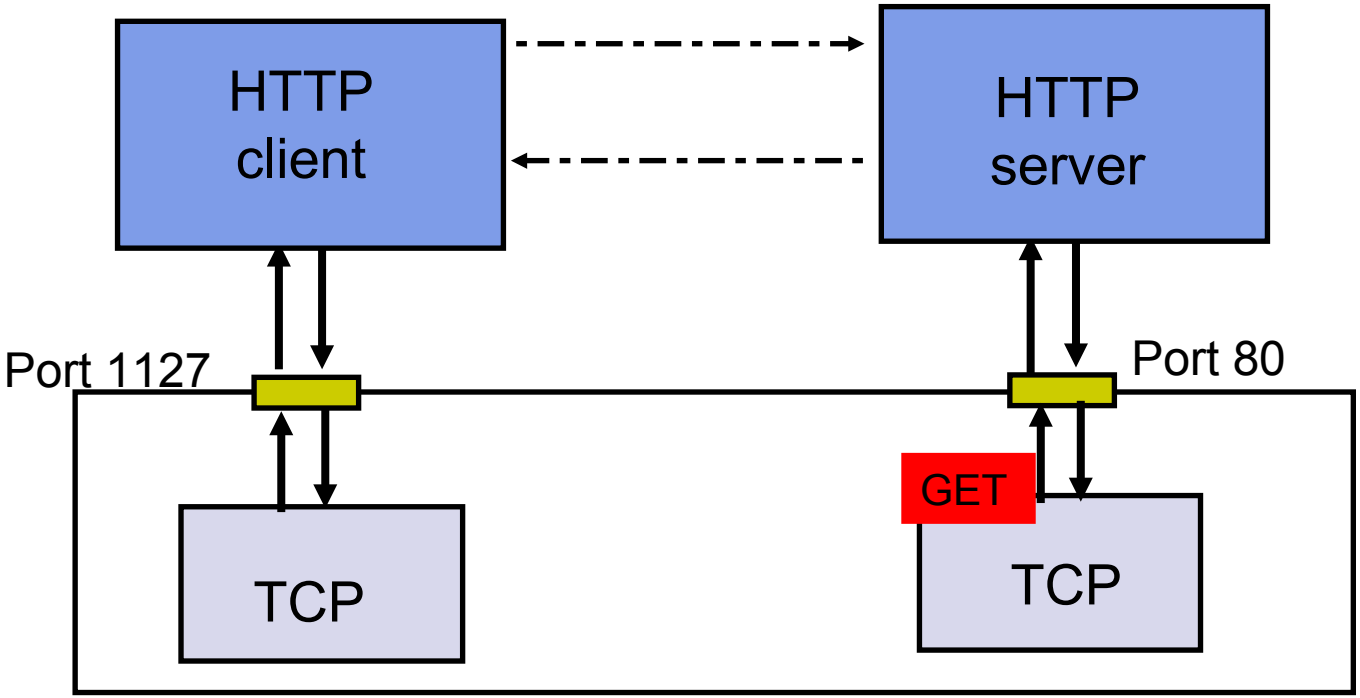
HTTP uses service of TCP



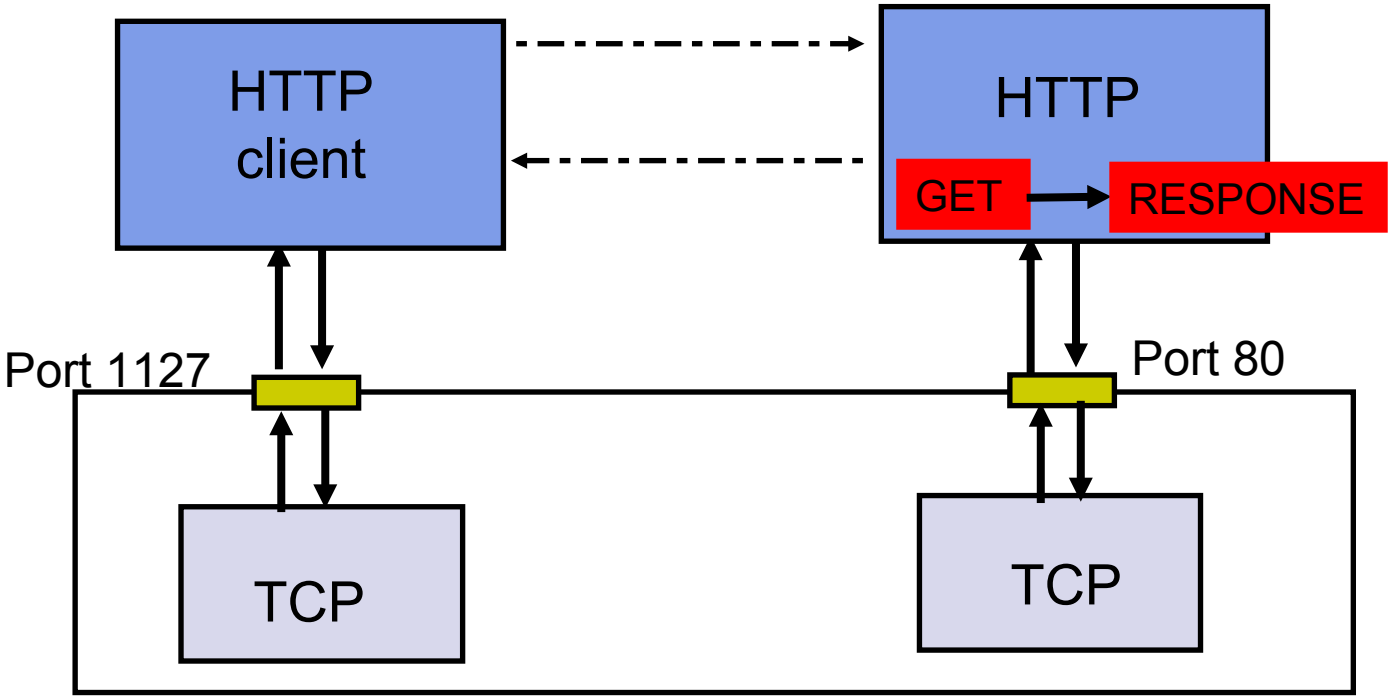
HTTP uses service of TCP - CONT



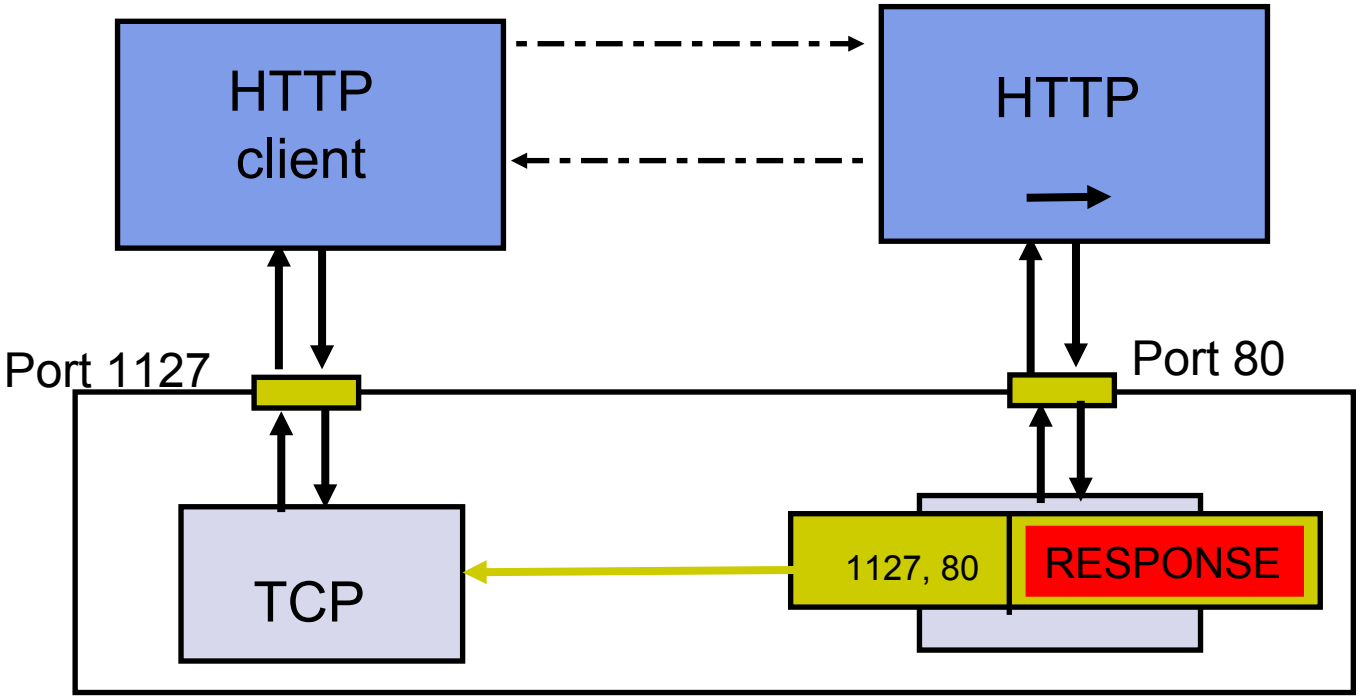
HTTP uses service of TCP- CONT



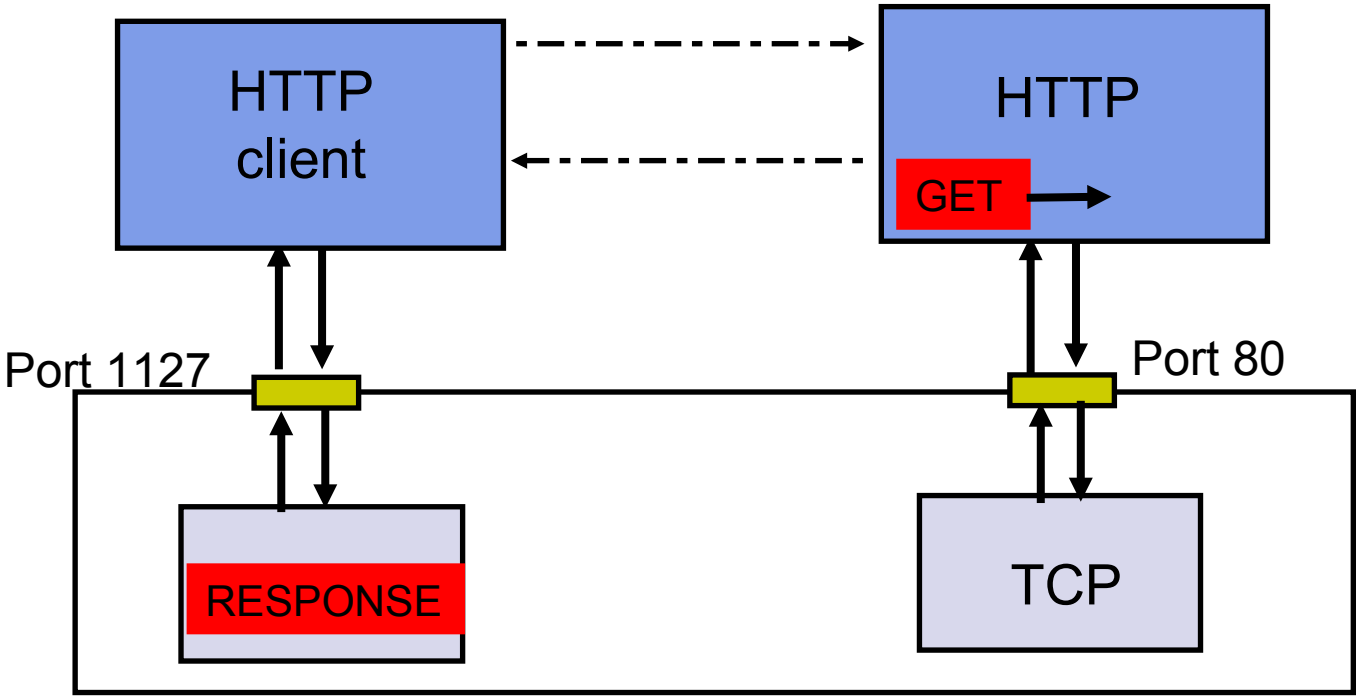
HTTP uses service of TCP- CONT

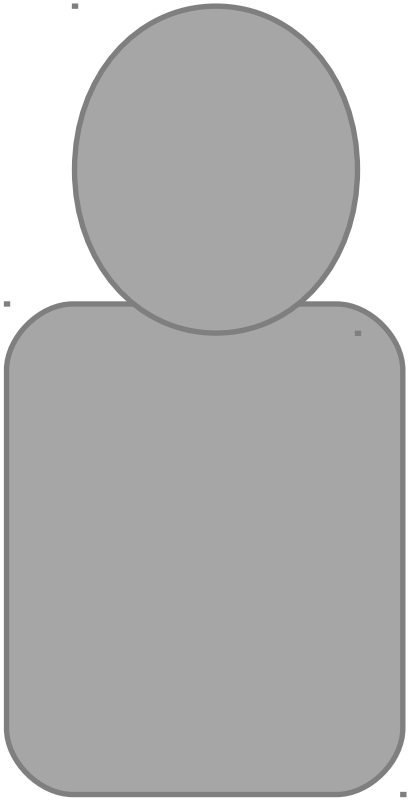


HTTP uses service of TCP- CONT



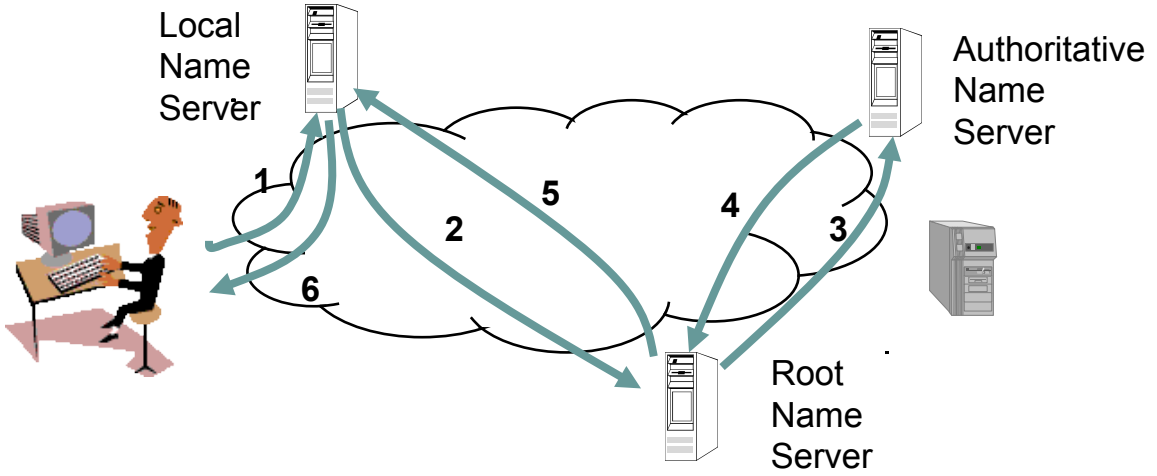
HTTP uses service of TCP- CONT





Example: DNS Protocol

- DNS protocol is an application layer protocol
- DNS is a distributed database that resides in multiple machines in the Internet
- DNS protocol allows queries of different types
- DNS usually involves short messages and so uses service provided by UDP
- Well-known port 53



- Local Name Server: resolve frequently-used names
  - E.g., University department, ISP
- Root Zone Name Servers
  - Resolves query or refers query to Authoritative Name Server
- Authoritative Name Server: last resort, 13 autoorities
  - Every machine must register its address with at least two servers

Summary

- Services: a protocol provides a communications service to the layer above
- DNS servers are one primary target of cyber attacks