ECCS-3631
Networks and Data Communications

Module 6-2
Cache, FTP, DHCP, DNS

Dr. Ajmal Khan

User-Server State: Cookies

many Web sites use cookies

four components:

    1) cookie header line of HTTP *response* message

    2) cookie header line in next HTTP *request* message

    3) cookie file kept on user's host, managed by user's browser

    4) back-end database at Web site

example:

- visits specific e-commerce site for first time

- when initial HTTP requests arrives at site, site creates:
  - unique ID
  - entry in backend database for ID

Cookies: Keeping "state" (cont.)

client

server

ebay 8734

cookie file
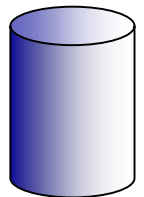
usual http request msg

Amazon server creates ID 1678 for user

create entry

backend database

usual http response
set-cookie: 1678

ebay 8734
amazon 1678

usual http request msg
cookie: 1678

cookie-specific action

access

usual http response msg

one week later:

access

ebay 8734
amazon 1678

usual http request msg
cookie: 1678

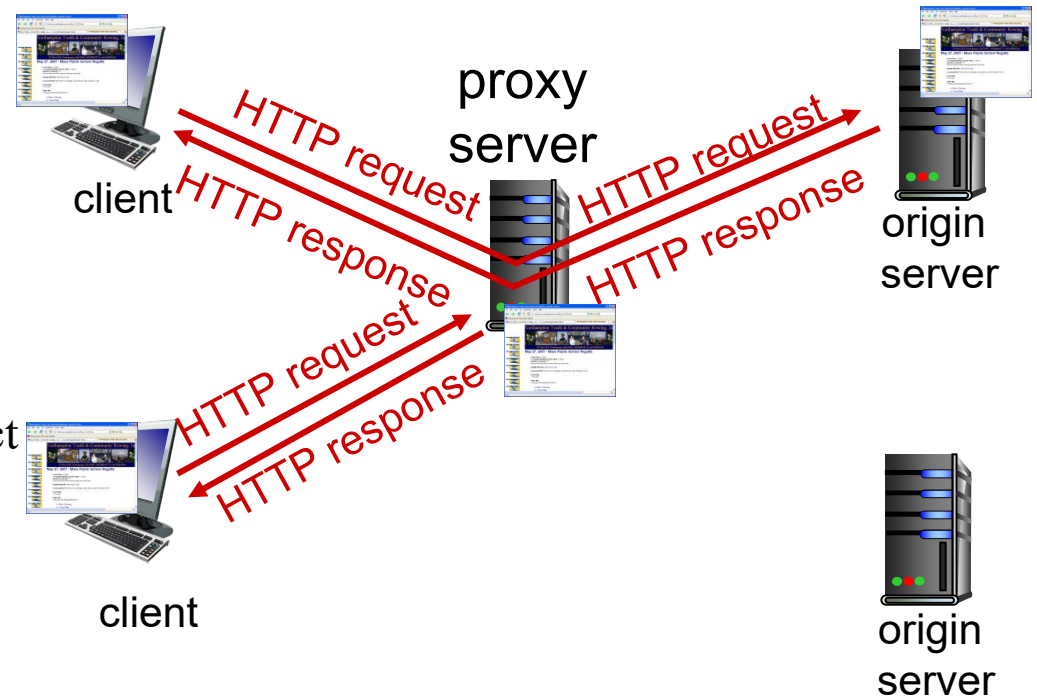cookie-specific action

usual http response msg

Web Caches (proxy server)

goal: satisfy client request without involving origin server

user sets browser: Web accesses via cache

browser sends all HTTP requests to cache

- object in cache: cache returns object
- else cache requests object from origin server, then returns object to client

More about Web Caching

cache acts as both client and server

- server for original requesting client
- client to origin server

typically cache is installed by ISP (university, company, residential ISP)

why Web caching?

➢ reduce response time for client request

➢ reduce traffic on an institution's access link

➢ Internet dense with caches: enables "poor" content providers to effectively deliver content

Conditional GET

Goal: don't send object if cache has up-to-date cached version

◦ no object transmission delay

◦ lower link utilization

cache: specify date of cached copy in HTTP request

If-modified-since: <date>

server: response contains no object if cached copy is up-to-date:

HTTP/1.0 304 Not Modified

client | server

HTTP request msg
If-modified-since: <date>

→ object not modified before <date>
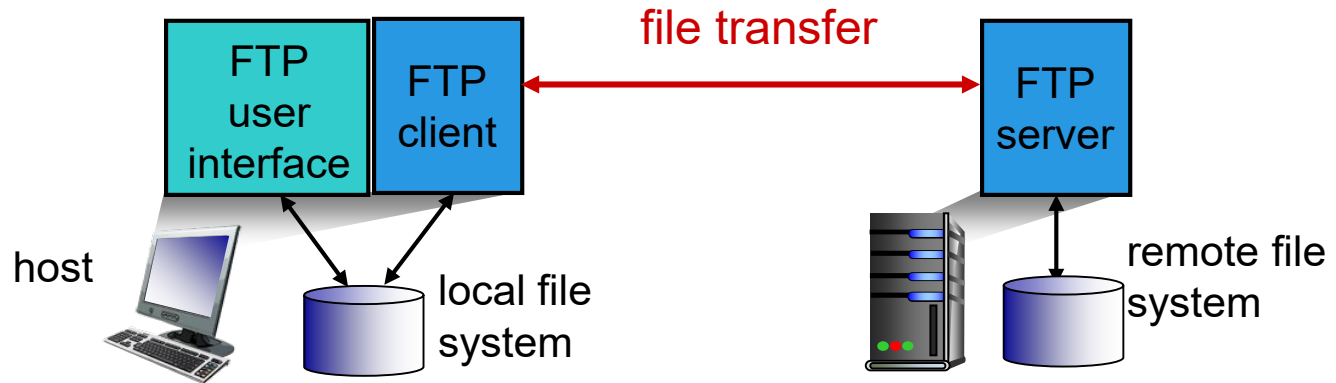
HTTP response
**HTTP/1.0
304 Not Modified**

HTTP request msg
**If-modified-since: <date>**

→ object modified after <date>

HTTP response
**HTTP/1.0 200 OK
<data>**

FTP: File Transfer Protocol

- ❖ transfer file to/from remote host
- ❖ client/server model
  - ▪ *client:* side that initiates transfer (either to/from remote)
  - ▪ *server:* remote host
- ❖ ftp server: port 21, 20
- ❖ FTP uses two parallel TCP connections: ***control connection*** and ***data connection***.
- ❖ The control connection is used for sending control information, such as username and password, and commands to put and get files.
- ❖ The data connection is used to send a file.
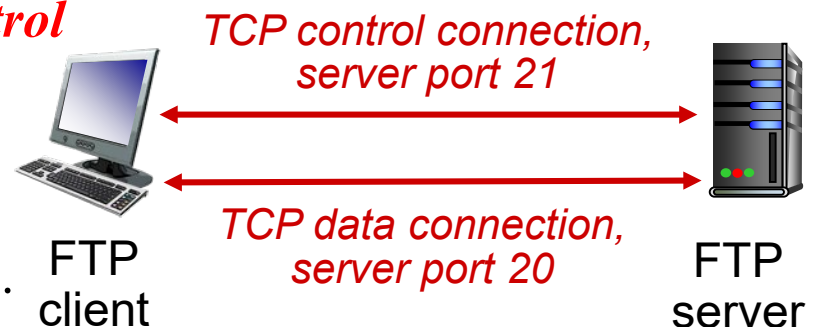
DHCP Protocol

➢ Note that Dynamic Host Configuration Protocol (DHCP) is not a Routing Protocol.

➢ DHCP is a client-server protocol.

➢ A network administrator configures DHCP so that a host receives IP address when it connects to the network.

➢ In addition, DHCP also allows a host to learn additional information, such as its subnet mask, default gateway, and DNS server.

➢ DHCP is also used in residential Internet access network and in wireless LANs, where hosts join and leave the network frequently.

➢ Each time a host joins, the DHCP server allocates an arbitrary address from its current pool of available addresses. Each time a host leaves, its address is returned to the pool.

DNS: Domain Name System

people: many identifiers:
- SSN, name, passport #

Internet hosts, routers:
- IP address (32 bit) - used for addressing datagrams
- "name", e.g., www.yahoo.com - used by humans

Domain Name System:

application-layer protocol: hosts, name servers communicate to *resolve* names (address/name translation)

DNS Name Resolution Example

host wants IP address for onu.edu

iterated query:
- ❖ contacted server replies with name of server to contact
- ❖ "I don't know this name, but ask this server"

recursive query:
- ❖ puts burden of name resolution on contacted name server
- ❖ heavy load at upper levels of hierarchy?

root DNS server

2

3

TLD DNS server

4

5

local DNS server

1 8

7 6

requesting host

authoritative DNS server
ns3.onu.edu

onu.edu