شبکه‌های کامپیوتری ۲

درس دوم:

لایه کاربرد(Application Layer)

رئوس مطالب:

✅ **اصول برنامه‌های کاربردی شبکه**

- ❑ **وب : پروتکل HTTP**
- ❑ **انتقال فایل: FTP**
- ❑ **پست الکترونیکی:SMTP**
- ❑ **سرویس دایرکتوری اینترنت: DNS**
- ❑ **برنامه‌های کاربردی P2P**
- ❑ **برنامه نویسی سوکت با UDPوTCP**

لایه کاربرد(Application Layer)

- طبق مدل TCP/IPبالاترین لایه، لایه کاربرد است.
- برنامه های کاربردی دو دسته‌اند:برنامه‌های کاربردی عمومی-برنامه‌های کاربردی اختصاصی
  -ایمیل، وب، اشتراک‌گذاری فایل، اشتراک‌گذاری ویدیو،ارتباط در بستروب و شبکه‌های اجتماعی

- برای ارایه هر سرویسی به کاربران،تبادل داده توسط لایه کاربرد با یک سیستم انتهایی(میزبان) انجام می‌شود.

- برای ارایه سرویس لایه کاربرد تبادل داده بین دو سیستم انتهایی(میزبان) را انجام می‌دهد.

ایجاد برنامه‌های کاربردی شبکه

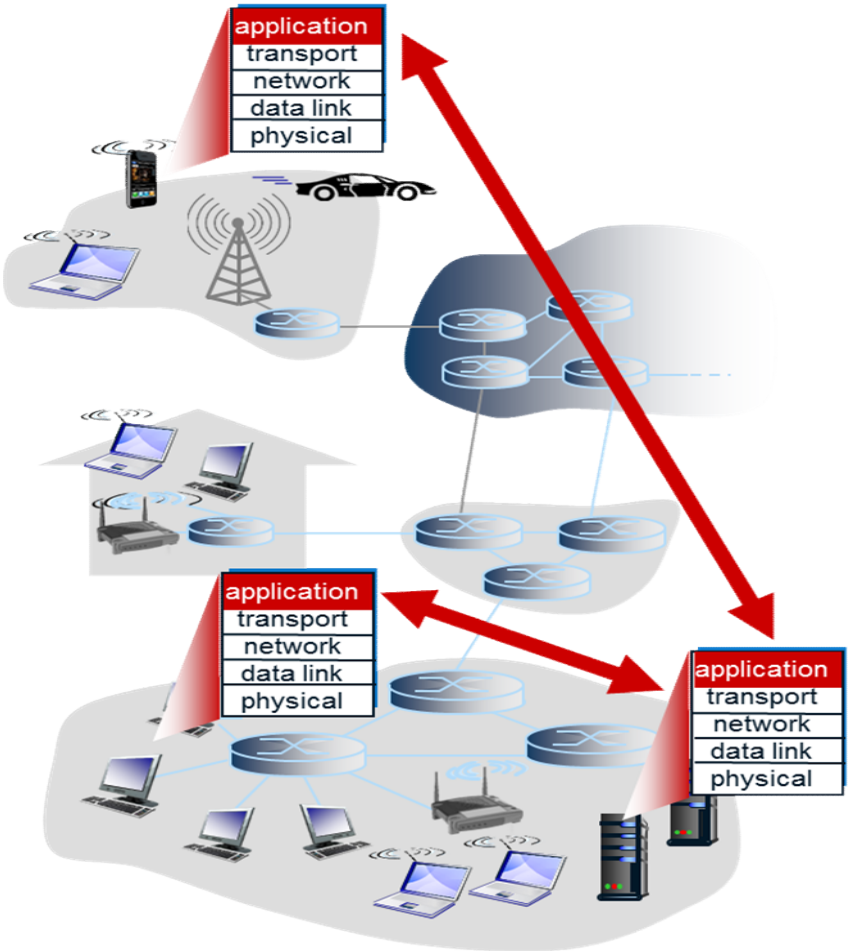
برنامه‌ای باید بنویسیم که:

- روی سیستم‌های انتهایی مختلف اجرا شود.
- باید در بستر شبکه با برنامه‌های دیگر ارتباط برقرار کند.
- بعنوان مثل سرویس دهنده وب با مرورگرها (مشتریان وب) ارتباط برقرار می‌کند

*نیازی به نوشتن برنامه برای سوئیچ‌های لایه پیوند و یا مسیریاب‌ها نیست.

- دستگاه‌های هسته شبکه برنامه‌های کاربردی

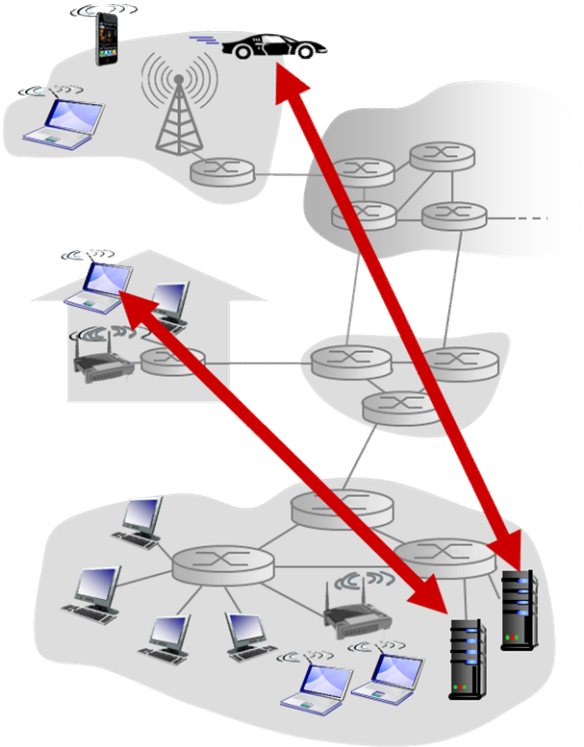
  را اجرا نمی‌کنند.
- مسیریاب‌ها و سوئیچ‌های لایه پیوند کار خورد را در لایه‌های پایینتر مثل شبکه و لایه پیوند انجام می‌دهند.


application
transport
network
data link
physical
application
transport
network
data link
physical
application
transport
network
data link
physical

معماری برنامه‌های کاربردی شبکه

- معماری مشتری-خدمتگزار
- معماری همتا به همتا(P2P)

معماری مشتری-سرویس‌دهنده



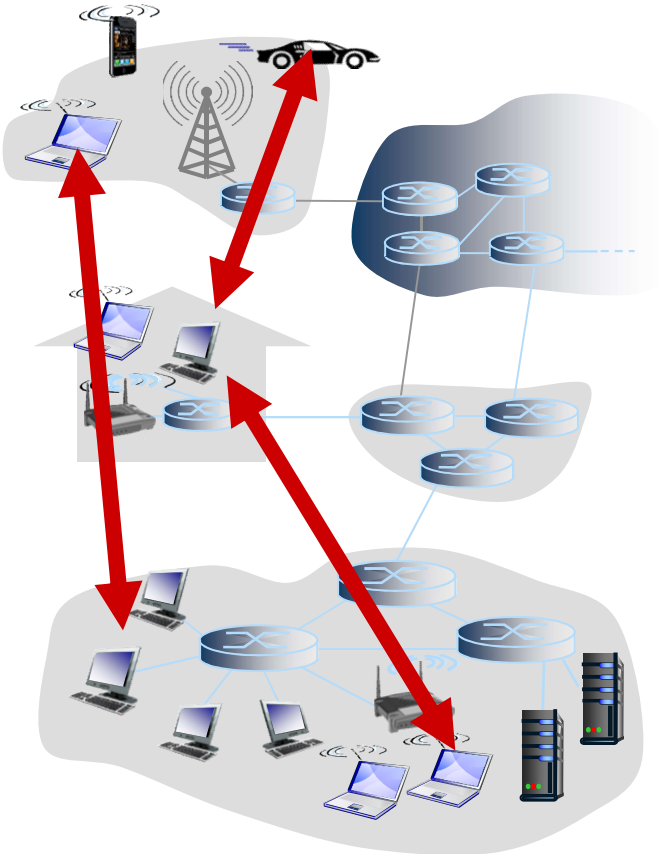
سرویس‌دهنده:

- یک میزبان همیشه فعال(روشن) است.
- همواره دارای دارای یک آدرس ثابت و شناخته شده بنام IP است.
- بعنوان یک مرکز داده بزرگ عمل می‌کند.

مشتری:

- با سرویس دهنده برای درخواست اطلاعات ارتباط برقرار می‌کند
- همواره فعال و متصل نیست.
- امکان دارد که دارای ادرس ثابتی نباشد.

معماری نظیربه‌نظیر



- هردو موجودیت،در ارتباط همتراز هستند.
- هیچگونه سرویس‌دهنده مرکزی و مشخصی وجود ندارد.
- همتایی که درخواست سرویس از یک همتادیگر می‌کند خودش بعدا به عنوان سرویس دهنده عمل خواهد کرد.

-خودمقیاس‌پذیری: هرهمتایی که به سیستم اضافه می‌شود در ادامه می‌تواند ظرفیت کارایی سیستم را بالا می‌برد.

- همتاها می‌توانند متناوبا متصل شده، و آدرس‌های مختلفی داشته باشند.

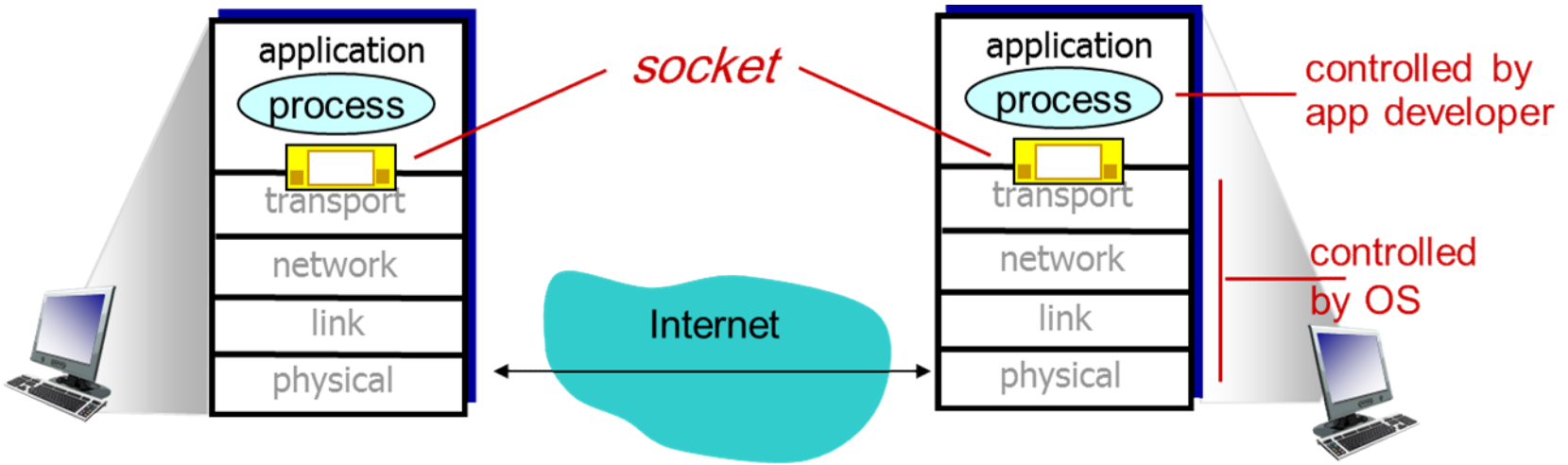
ارتباط بین فرایندها

فرایند: برنامه‌های اجرا شده بر روی میزبان‌ها

- فرایندهای موجود بر روی سیستم‌های انتهایی متفاوت از طریق پیام (message) باهم ارتباط برقرار می‌کنند.
- فرایند مشتری: فرایندی که در یک نشست ارتباطی بین یک زوج فرایند، ابتدا ارتباط را آغاز می‌کند.
- فرایند سرویس‌دهنده: فرایندی که منتظر ایجاد ارتباط با خود می‌باشد، فرایند سرویس‌دهنده است.

واسط بین فرایند و شبکه‌های کامپیوتری

- فرایندها از طریق سوکت پیام‌ها را ارسال و یا دریافت می‌کنند.و به نوعی سوکت واسط بین فرایندها و شبکه زیرین است.
- فرایند مانند خانه و سوکت مانند درب آن است.
  - فرایند پیام خود را به طرف دیگر درب می‌فرستد و انرا به زیرساخت ارتباطی می‌سپارد.
  - پیام انتقال داده شده از طریق زیرساخت ارتباطی به سوکت(درب) فرایند گیرنده در میزبان مقصد می‌رسد.


application
process
transport
network
link
physical
socket
Internet
application
process
transport
network
link
physical
controlled by
app developer
controlled
by OS

آدرس‌دهی فرایندها

- برای فرستادن پیام از یک فرایند به فرایند دیگر به دو چیز نیاز داریم:

  ۱-آدرس میزبان فرایند مقصد ۲-شناسه فرایند دریافت کننده

- بدلیل اینکه ممکن است چندین برنامه‌کاربردی بر روی یک میزبان اجرا شود، لذا وجود شناسه (شماره پورت) ضروری است.
- برنامه‌های معروف دارای شماره پورت مشخص هستند:

  -سرویس‌دهنده وب دارای شماره پورت ۸۰ است

  -سرویس دهنده ایمیل دارای شماره پورت ۲۵ است.

انتخاب سرویس‌های انتقال برای برنامه کاربردی

انتقال داده مطمئن

- بعضی از برنامه‌ها مانند وب و انتقال فایل نیاز دارند که داده‌ها ۱۰۰٪ و تضمینی به مقصد برسند.
- بعضی از برنامه‌ها قابلیت تحمل تلفات را دارند، و از بین رفتن بخش جزیی از داده‌ها تاثیری در عملکرد کلی ندارد.

انتخاب سرویس‌های انتقال برای برنامه کاربردی..

گذردهی (Throughput)

- حجم داده‌ای که منتقل می‌شود، مهم خواهد بود؟
- چه میزان گذردهی و پهنای باند نیاز داریم؟
- برنامه‌های حساس به پهنای باند به حدی از گذردهی نیاز دارند.
- بعضی از برنامه‌ها می‌توانند خود را با حداقل گذردهی موجود تطبیق دهند، به این برنامه‌ها کشسان می‌گویند.

انتخاب سرویس‌های انتقال برای برنامه کاربردی..

زمانبندی(Timing)

- آیا بسته‌ها سر زمان مقرر تحویل داده می‌شوند؟
- این سرویس در برنامه‌های کاربردی زمان واقعی حیاتی است.
- در برنامه‌های کاربردی تلفن اینترنتی،در بازی‌های چندنفره که حساسیت به تاخیر زمانی وجود دارد، لازم می‌باشد.
- در برنامه‌ها با حساسیت زمانی کمتر نیز حدی از تاخیر قابل قبول خواهد بود.

انتخاب سرویس‌های انتقال برای برنامه کاربردی..

امنیت

- سرویس بتواند محرمانگی در داده‌های ارسالی را رعایت کند.
- محرمانگی با رمزگذاری داده‌ها در مبدا توسط پروتکل لایه انتقال ایجاد می‌شود، و در مقصد دوباره داده‌ها رمزگشایی و بعد تحویل داده می‌شوند.

حساسیت زمانی (offline) حساسیت به زمان نیستند.

* سرویس‌های نامحکم

برنامه کاربردی	تلفات داده	گذردهی	حساسیت زمانی
file transfer	no loss	elastic	no
e-mail	no loss	elastic	no
Web documents	no loss	elastic	no
real-time audio/video	loss-tolerant	audio: 5kbps-1Mbps video:10kbps-5Mbps	yes, 100's msec
stored audio/video	loss-tolerant	same as above	
interactive games	loss-tolerant	few kbps up	yes, few secs
text messaging	no loss	elastic	yes, 100's msec yes and no

کیفیت

سرویس‌های انتقال اینترنت

سرویس‌های TCP

- سرویس اتصال‌گرا است.و انتقال داده مطمئن دارد.
- TCPدارای سازوکار کنترل ازدحام و کنترل جریان است.
- TCP تضمین گذردهی و زمانبندی ندارد.

سرویس‌های انتقال اینترنت...

سرویس‌های UDP

- پروتکل UDP غیراتصال‌گرا است.و انتقال داده نیز مطمئن نیست.
- در انتقال داده با UDP، امکان دارد که ترتیب تحویل بسته‌ها نیز حفظ نشود.
- UDPسازوکار کنترل ازدحام ندارد.

UDP : برای ارسال بسته‌های کوچک مناسب است.

TCP

اتصال گرا

قابل اطمینان

کنترل ازدحام

کنترل جریان

رهایابندی ندارد

امنیت ندارد

صحت تحویل داده‌ها را تضمین می‌کند

UDP

غیر اتصال گرا

سریع‌تر

کنترل جریان و ازدحام ندارد

رهایابندی ندارد

امنیت ندارد

قابلیت اطمینان ندارد

تحویل داده‌ها را تضمین نمی‌کند

برنامه کاربردی	پروتکل لایه کاربرد	پروتکل لایه انتقال
e-mail	SMTP [RFC 2821]	TCP
remote terminal access	Telnet [RFC 854]	TCP
Web	HTTP [RFC 2616]	TCP
file transfer	FTP [RFC 959]	TCP
streaming multimedia	HTTP (e.g., YouTube),	TCP or UDP
Internet telephony	SIP, RTP, proprietary (e.g., Skype)	TCP or UDP
DNS		UDP

رئوس مطالب:

- ❑ اصول برنامه‌های کاربردی شبکه
- ✅ وب : پروتکل HTTP
- ❑ انتقال فایل: FTP
- ❑ پست الکترونیکی:SMTP
- ❑ سرویس دایرکتوری اینترنت: DNS
- ❑ برنامه‌های کاربردی P2P
- ❑ برنامه نویسی سوکت با UDPوTCP

وب:پروتکل HTTP

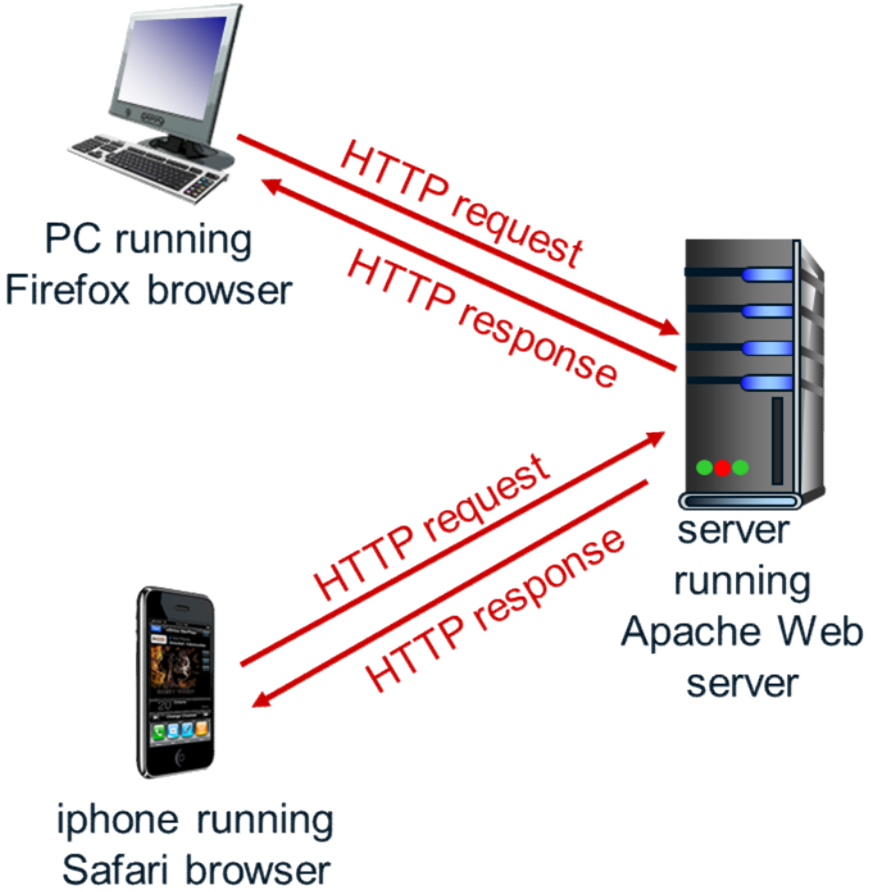
- هرصفحه وب شامل تعدادی شیء(Object) است.
- اشیا شامل فایل‌های HTML، تصاویر با فرمت‌های مختلف، فیلم‌ها،اپلت‌ها و انیمیشن‌ها است.
- در صفحه وب، یک صفحه اصلی(مبنا)وجود دارد که از طریق آن میتوان به اشیا و صفحات دیگر دسترسی پیدا کرد.
- هر شیء در صفحه وب دارای یک آدرس است:

www.someschool.edu/someDept/pic.gif

host name | path name

وب:پروتکلHTTP...

- ❖ HTTP دارای مدل مشتری-سرویس‌دهنده است.
  - مشتری همان مرورگرهای وب مانند فایرفاکس و اینترنت اکسپلورر هستند.
  - سرویس دهنده نیز میزبان اشیای وب است.IISوApache


HTTP request
HTTP response
PC running
Firefox browser
HTTP request
HTTP response
server
running
Apache Web
server
iphone running
Safari browser

وب: پروتکل HTTP...

- HTTP از TCP برای ارتباط استفاده می‌کند.
- ابتدا یک اتصال TCP توسط مشتری با سرویس دهنده روی پورت ۸۰ ایجاد می‌شود.
- بعد از ایجاد اتصال TCP فرایندهای مشتری و سرویس‌دهنده می‌توانند باهم ارتباط درخواست پاسخ داشته باشند.
- درخواست‌ها در قالب آدرس اشیا به سرویس‌دهنده داده می‌شوند.

اتصالات ماندگار و غیرماندگار

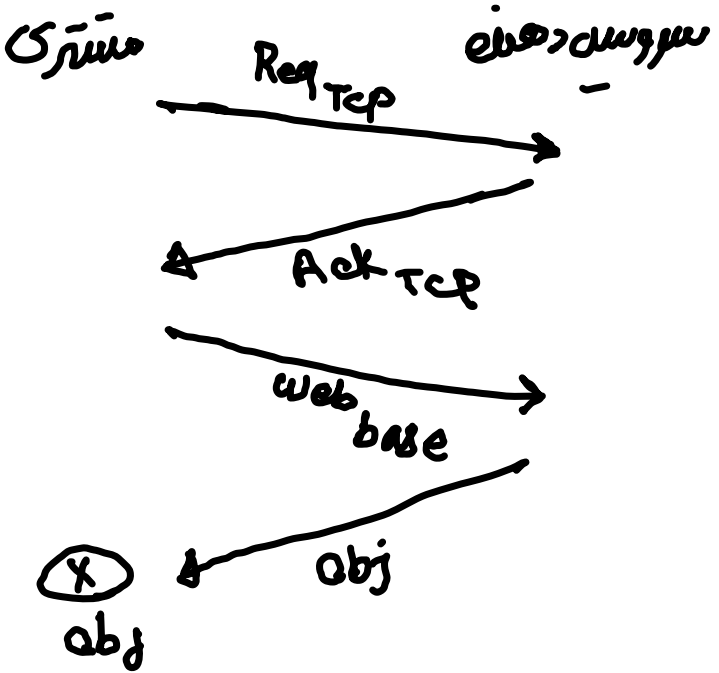
Persistent – Non Persistent

اتصال غیرماندگار

I. فرایند مشتری روی پورت ۸۰ یک اتصال TCP با سرویس‌دهنده برقرار می‌کند.

II. مشتری HTTP از طریق سوکت خود یک پیام درخواست HTTP به سرویس‌دهنده می‌دهد.

III. سرویس‌دهنده HTTP بعد از دریافت پیام درخواست، از طریق سوکت خود شیء درخواستی را به مشتری برمی‌گرداند.


Req TCP
Ack TCP
web base
obj
X
obj

IV. فرایند سرویس‌دهنده بعد از اطمینان از دریافت پیام توسط مشتری اتصال را می‌بندد.

V. فرایند مشتری هنگامیکه پیام پاسخ را دریافت می‌کند،اتصال TCP را می‌بندد.و شی داخل پیام را که همان فایل HTML پایه است را استخراج و می‌خواند.

IV. مشتری HTTP به ازای هر یک اشیا موجود در فایل HTML پایه، مراحل ۱ تا ۴ را دوباره انجام می‌دهد.

`www.someSchool.edu/someDepartment/home.index` → 1o obj

time

1a. HTTP client initiates TCP connection to HTTP server (process) at www.someSchool.edu on port 80

Req_tcp

1b. HTTP server at host www.someSchool.edu waiting for TCP connection at port 80. "accepts" connection, notifying client

Ack

2. HTTP client sends HTTP request message (containing URL) into TCP connection socket. Message indicates that client wants object someDepartment/home.index

Req_web

3. HTTP server receives request message, forms response message containing requested object, and sends message into its socket

Res

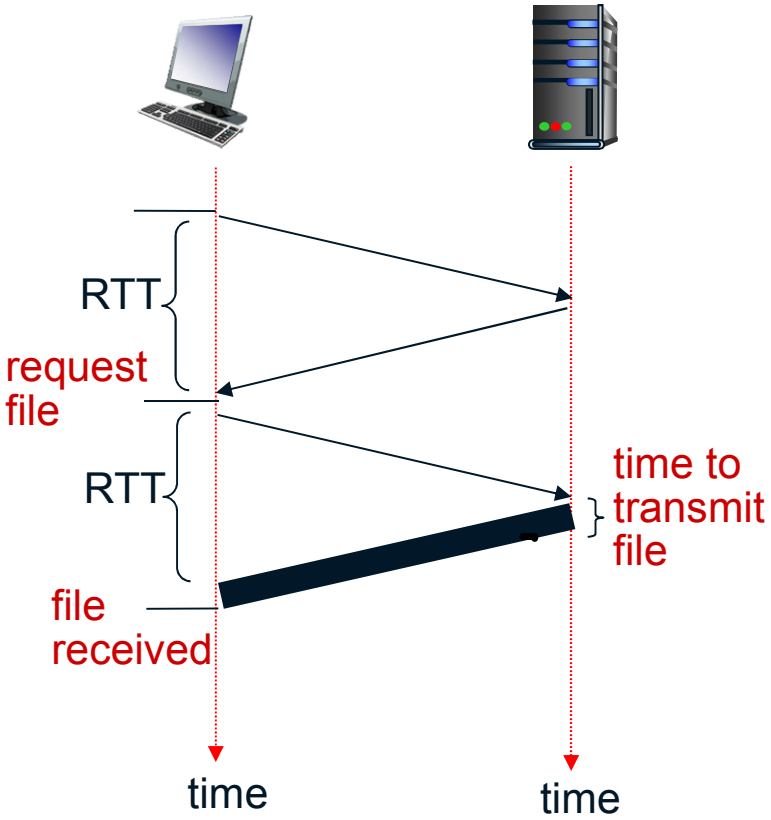
time

4. HTTP server closes TCP connection.

5. HTTP client receives response message containing html file, displays html. Parsing html file, finds 10 referenced jpeg objects

6. Steps 1-5 repeated for each of 10 jpeg objects

زمان رفت و برگشت(RTT)

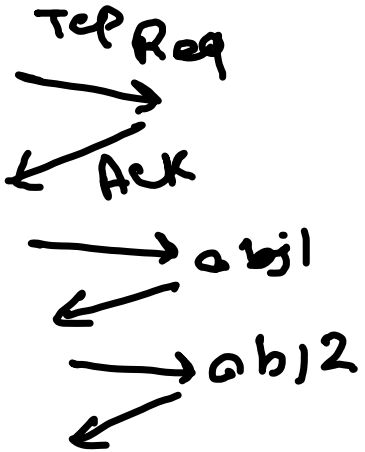


در اتصال غیرمانندگار:

زمان ارسال فایل + 2RTT

اتصال ماندگار

در این روش برخلاف اتصال غیرماندگار سرویس دهنده بعد از برگرداندن پاسخ همچنان اتصال TCP را برای درخواست/پاسخ‌های بعدی باز نگه می‌دارد.



مثال (۱):برای دانلود یک فایل از اینترنت که شامل ۱۲ شی می‌باشد،چقدر طول می‌کشدتا صفحه وب اصلی ظاهر شود؟فرض کنید سرعت انتقال لینک 100kbps و سرعت انتشار 10^6m/s و فاصله بین سرویس دهنده و مرورگر وب 100kmاست.طول بسته‌های کنترلی 200 بیت و اندازه هر شی 100kbاست.

$RTT_{TCP} = 2\left(\frac{200}{10^5}\right) + 2(0.1) = 4\times10^{-3} + 2\times10^{-1}$

$= 204\,ms$

$RTT_{H} = \frac{200}{10^5} + 1 + 2(0.1) = 2\times10^{-3} + 1000\times10^{-3} + 2\times10^{-1}$

$= 1202\,ms$

کل زمان $= 1202 + 204 = 1406\,ms \cong 1.5s$

- **مثال (۲):** یک لینک کوتاه ۱۰ متری درنظربگیرید که فرستنده می‌تواند با آهنگ ۱۵۰bit/sec در هردوجهت ارسال می‌کند.اگر بسته‌های داده 10^5بیت و بسته‌های کنترلی ۲۰۰بیت باشند.حال پروتکل HTTP با اشیایی به بزرگی ۱۰۰kb درنظربگیرید، که درشی HTMLپایه به ۱۰ شی دیگر ارجاع داده شده است.درهرکدام از حالات زیر زمان دریافت را محاسبه نمایید.

  تأخیر انتشار : T_{Pd}

  الف)اتصالات HTTP غیرماندگار و موازی
  ب)اتصالات HTTP ماندگار

الف

- اتصال Tcp

مشتری — Req → سرویس دهنده

Ack ← سرویس دهنده

$$2\left(\frac{200}{150} + T_{Pd}\right)$$

- درخواست یک HTML

Req (Html) →

$$\frac{200}{150} + T_{Pd}$$

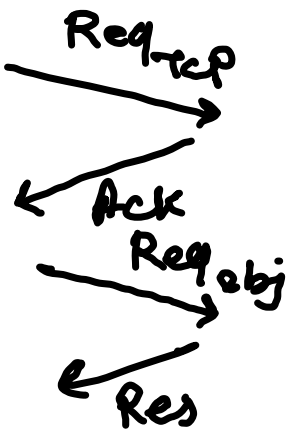
- پاسخ سرویس دهنده برای HTML

Res ←

$$\frac{100000}{150} + T_{Pd}$$

زمان کل دریافت یک HTML: $3\left(\frac{200}{150} + T_{Pd}\right) + \frac{10^5}{150} + T_{Pd}$

- درخواست موازی برای ۱۰ شیء :



$$2\left(\frac{200}{15} + T_{Pd}\right)$$

$$\left(\frac{200}{15} + T_{Pd}\right) + \left(\frac{10^5}{15} + T_{Pd}\right)$$

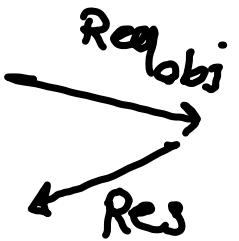
$$T_{کل} = 7377 + 8T_{Pd}$$

(ب)

- انتقال Tcp و دریافت HTML پایه

$\rightarrow 3\left(\frac{200}{150} + T_{Pd}\right) + \left(\frac{10^5}{150} + T_{Pd}\right)$

- دریافت 10 شیء (obj)



$10\left(\frac{200}{150} + T_{Pd} + \frac{10^5}{150} + T_{Pd}\right)$

$T_{کل} = 7351 + 24 T_{Pd}$

سرعت انتقال: فیبر نوری $3 \times 10^8 \frac{m}{s}$

$$T_{Pd} = \frac{10}{3 \times 10^8} = 3.3 \times 10^{-8}$$

پیام درخواست HTTP

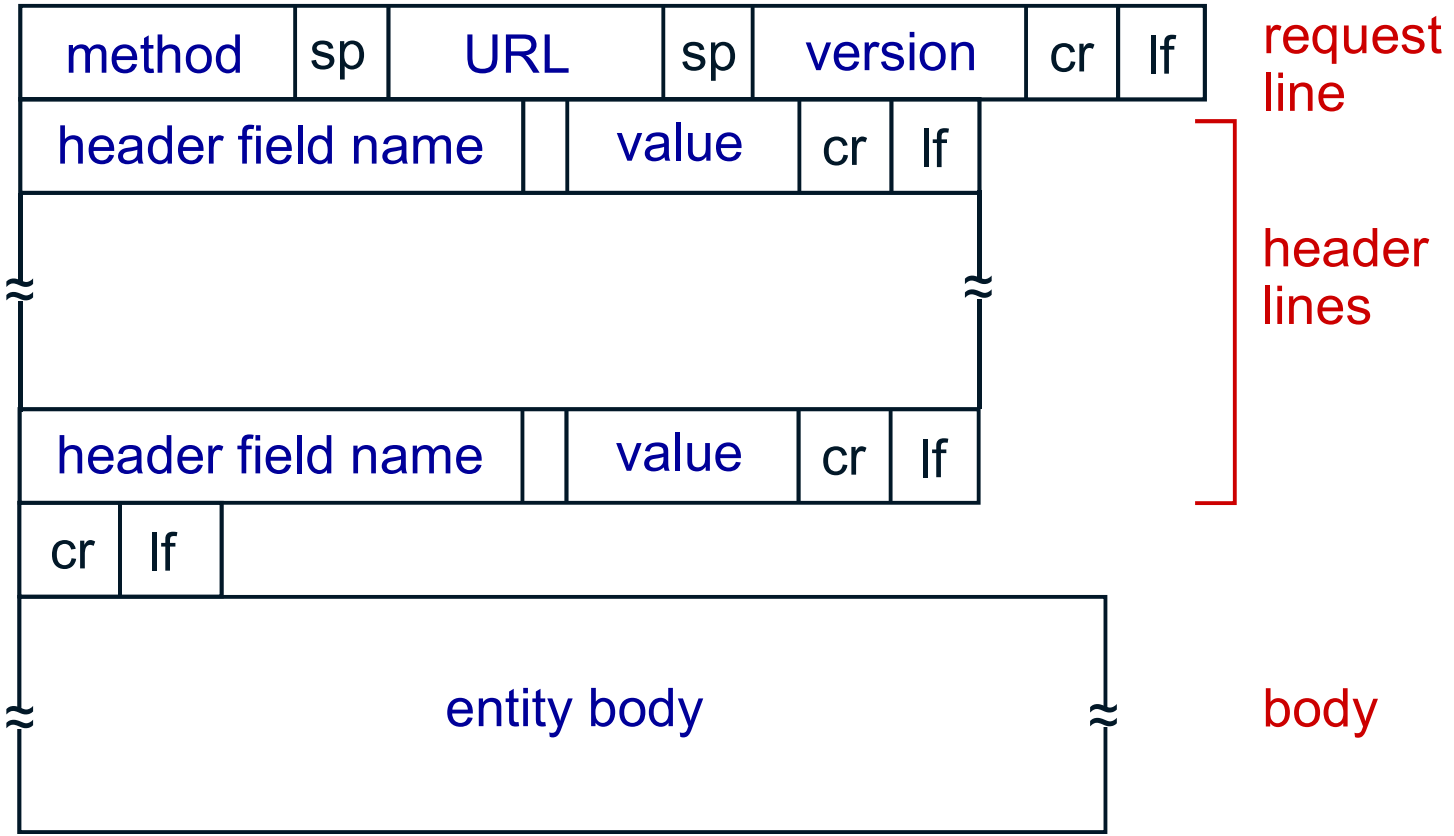
ارسال پارامترهای فرم

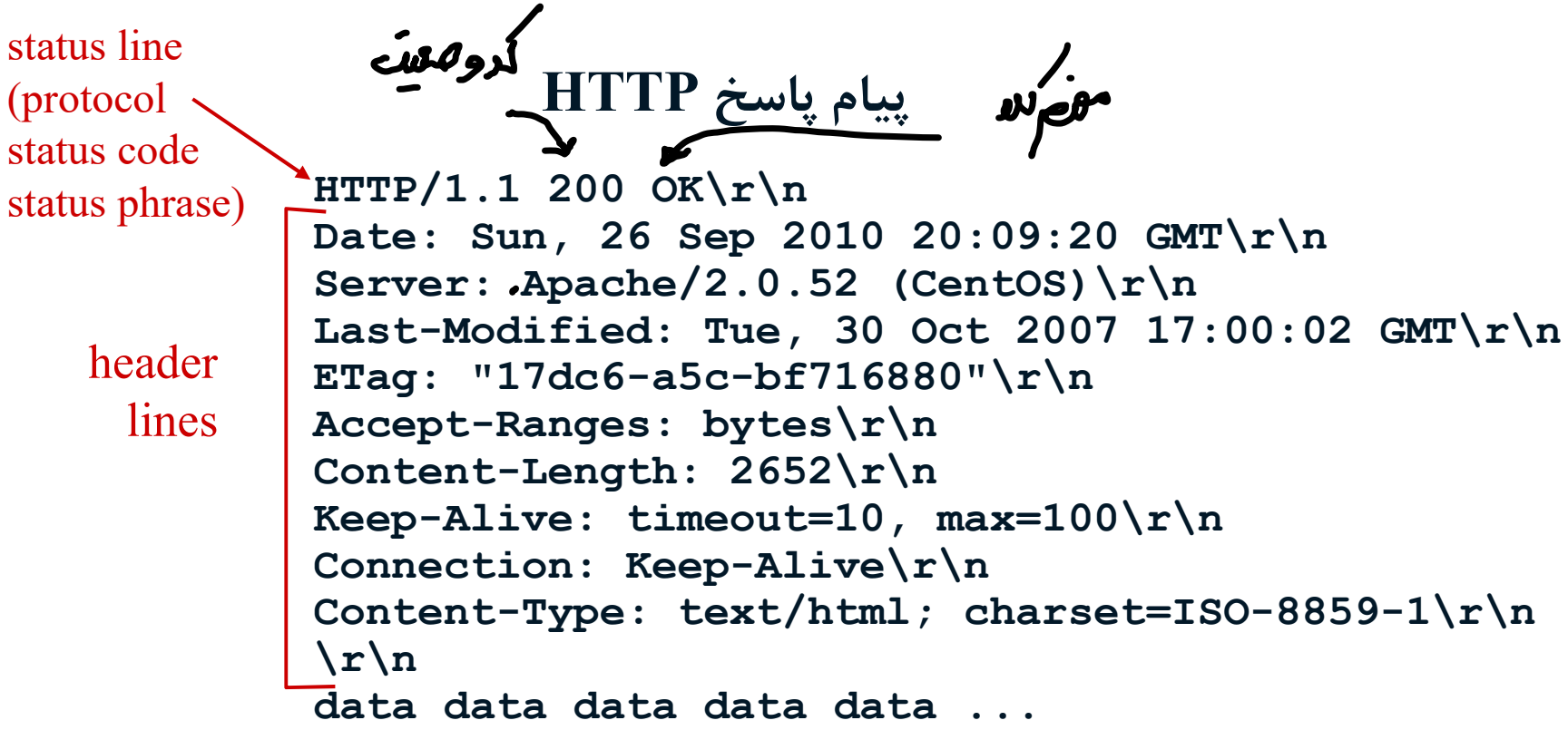
request line
(GET, POST,
HEAD commands)

header
lines

```
GET /index.html HTTP/1.1\r\n
Host: www-net.cs.umass.edu\r\n
User-Agent: Firefox/3.6.10\r\n
Accept: text/html,application/xhtml+xml\r\n
Accept-Language: en-us,en;q=0.5\r\n
Accept-Encoding: gzip,deflate\r\n
Accept-Charset: ISO-8859-1,utf-8;q=0.7\r\n
Keep-Alive: 115\r\n
Connection: keep-alive\r\n
\r\n
```

فرمت کلی پیام درخواست




status line
(protocol
status code
status phrase)
header
lines
HTTP/1.1 200 OK\r\n
Date: Sun, 26 Sep 2010 20:09:20 GMT\r\n
Server: Apache/2.0.52 (CentOS)\r\n
Last-Modified: Tue, 30 Oct 2007 17:00:02 GMT\r\n
ETag: "17dc6-a5c-bf716880"\r\n
Accept-Ranges: bytes\r\n
Content-Length: 2652\r\n
Keep-Alive: timeout=10, max=100\r\n
Connection: Keep-Alive\r\n
Content-Type: text/html; charset=ISO-8859-1\r\n
\r\n
data data data data data ...

کدهای وضعیت در پیام پاسخ HTTP

200 OK

- اجرای درخواست موفقیت آمیز بوده است،و اطلاعات در هرحال ارسال است.

301 Moved Permanently

Location: مسیر جدید

- شیء درخواست شده از این سرویس‌دهنده منتقل شده است.

400 Bad Request

- درخواست کاربر برای سرویس‌دهنده مفهوم نیست.

404 Not Found

- سند خواسته شده در این سرویس‌دهنده موجود نیست.

505 HTTP Version Not Supported

- نسخه پروتکل HTTP درخواست شده توسط سرویس دهنده پشتیبانی نمی‌شود.

تقابل کاربر-سرویس‌دهنده:کوکی

- اغلب سایت‌های تجاری از کوکی استفاده می‌کنند.

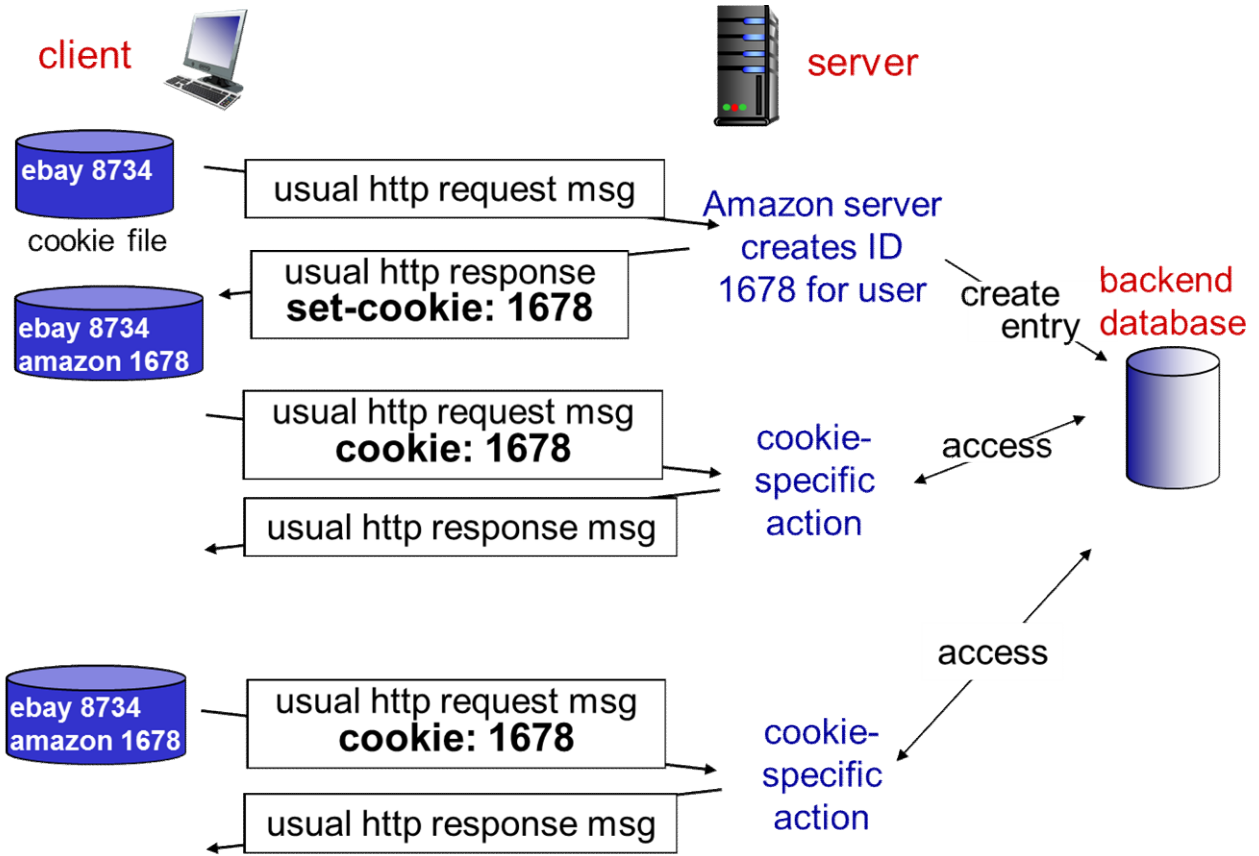
کوکی دارای چهار مولفه است:

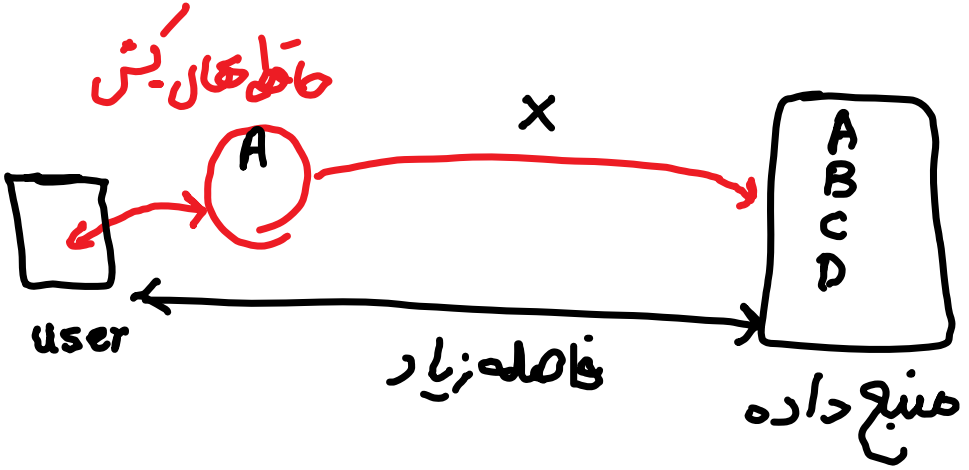
۱)یک خط سرآیند کوکی در پیام پاسخHTTP

۲)یک خط سرایند کوکی در پیام درخواست HTTP

۳)یک فایل کوکی در سیستم میزبان مشتری که توسط مرورگر وب مدیریت می‌شود.

۴)یک پایگاه داده پشتیبان در سایت وب.

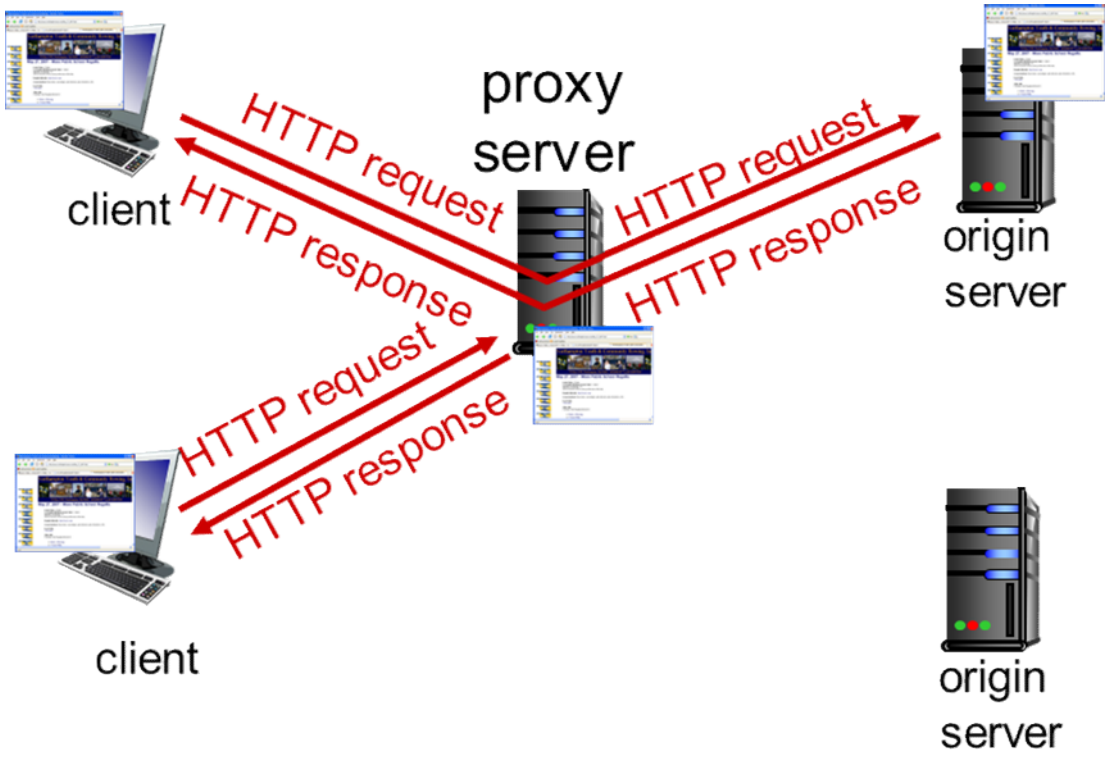

client
server
ebay 8734
cookie file
usual http request msg
Amazon server
creates ID
1678 for user
usual http response
set-cookie: 1678
create
entry
backend
database
ebay 8734
amazon 1678
usual http request msg
cookie: 1678
cookie-
specific
action
access
usual http response msg
access
ebay 8734
amazon 1678
usual http request msg
cookie: 1678
cookie-
specific
action
usual http response msg


حافظه‌های پیش
A
X
A
B
C
D
user
فاصله زیاد
منبع داده

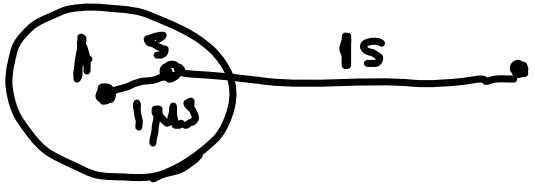
Proxy Server Web Cache
حافظه نهان وب(سرویس‌دهنده پروکسی)

- با استفاده از آن درخواست‌های HTTP بدون مراجعه به سرویس‌دهنده اصلی پاسخ داده می‌شوند.
- حافظه نهان وب با تجهیزات ذخیره‌سازی خود یک نسخه از هر شیء درخواستی از سرویس‌دهنده اصلی را برای مراجعات بعدی مشتری نزد خود نگه می‌دارد.
- مرورگرهای وب را می‌توان طوری پیکربندی کرد که ابتدا درخواست‌های HTTP را به حافظه نهان وب بدهند.
- درصورت ارسال درخواست‌ها به حافظه نهان وب:

-شیء درخواستی در حافظه نهان وب موجود است، و به مرورگر برگشت داده می‌شود
- حافظه نهان وب به سرویس دهنده اصلی درخواست داده و سپس به مرورگر برگشت می‌دهد.


proxy
server
client
HTTP request
HTTP response
HTTP request
HTTP response
origin
server
client
HTTP request
HTTP response
origin
server

حافظه نهان وب(سرویس‌دهنده پروکسی)...



چرا از حافظه نهان وب استفاده می‌شود:

- **زمان پاسخ برای درخواست مشتریان را کاهش می‌دهد.**
- **ترافیک لینک‌های دسترسی به اینترنت را در سازمان‌ها کاهش می‌دهد.**
- **ترافیک کل اینترنت را کاهش داده، و عملکرد برنامه‌های کاربردی را بهبود می‌بخشد.**

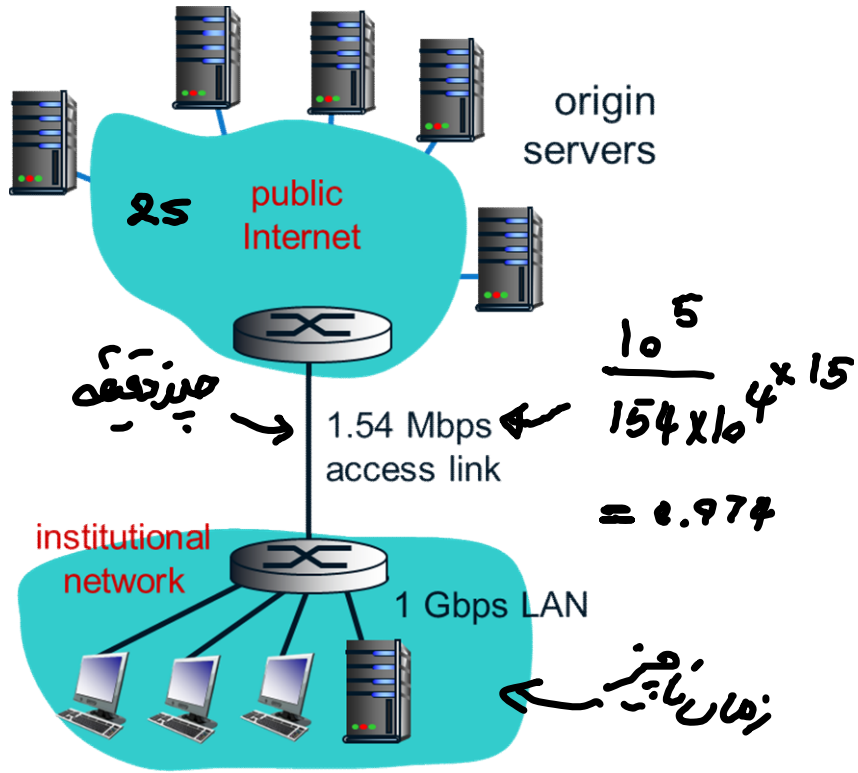
یک مثال برای نشان دادن کارایی کش

فرض کنید:

- میانگین اندازه بسته‌ها: ۱۰۰kb
- تعداد درخواست‌ها از مرورگرها به سرویس‌دهنده اصلی بطور میانگین: ۱۵ درخواست در ثانیه.
- میانگین نرخ داده به مرورگرها: ۱/۵Mbps
- لحظه ارسال درخواست از مسیریاب اینترنت تا دریافت پاسخ برابر: ۲ ثانیه

درنتیجه:

- شدت ترافیک در شبکه محلی: ٪ ۰/۱۵
- شدت ترافیک در لینک بین مسیریاب‌ها: ۰/۹۷۴
- زمان پاسخ کل برابرست با: ۲ثانیه + تاخیر در حد چندین دقیقه + تاخیر ناچیز شبکه محلی



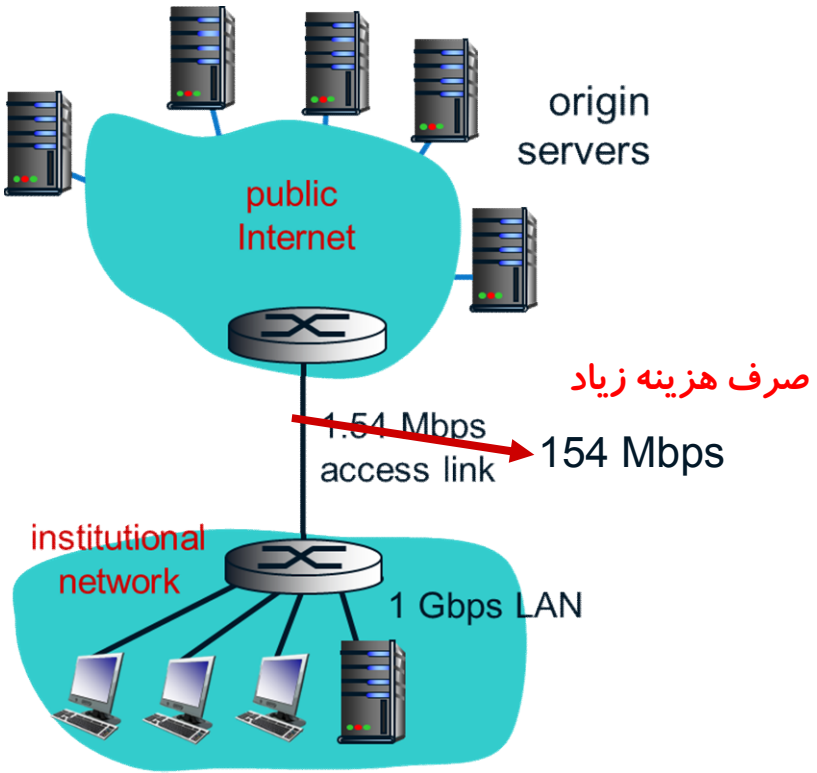
یک مثال برای نشان دادن کارایی کش...

فرض کنید:

- میانگین اندازه بسته‌ها: **۱۰۰kb**
- تعداد درخواست‌ها از مرورگرها به سرویس‌دهنده اصلی بطور میانگین: ۱۵ درخواست در ثانیه.
- میانگین نرخ داده به مرورگرها:**۱/۵Mbps**
- لحظه ارسال درخواست از مسیریاب اینترنت تا دریافت پاسخ برابر: ۲ ثانیه

درنتیجه:

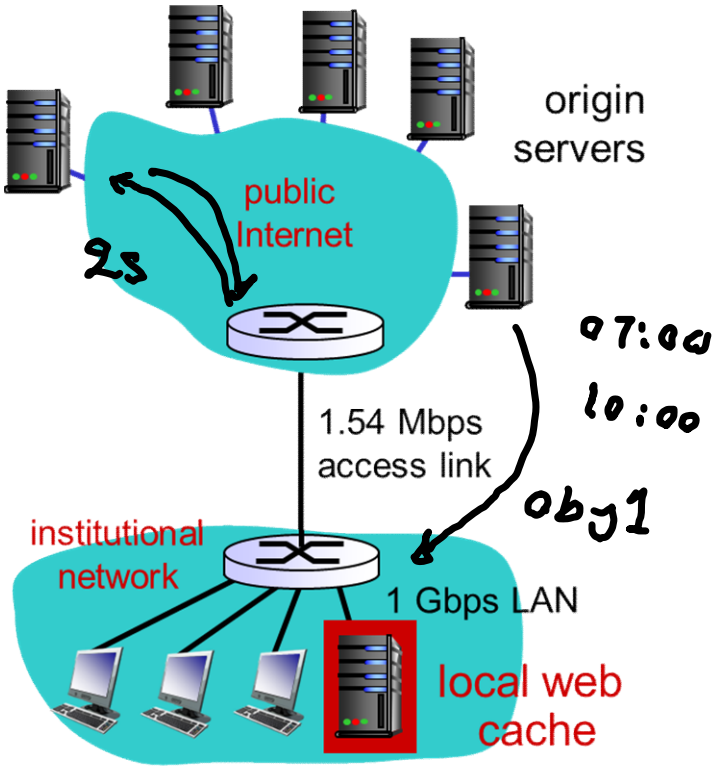
- شدت ترافیک در شبکه محلی: ٪ ۰/۱۵
- شدت ترافیک در لینک بین مسیریاب‌ها: ۰/۹۴ ٪۹
- زمان پاسخ کل برابرست با: تقریبا ۲ ثانیه



یک مثال برای نشان دادن کارایی کش...

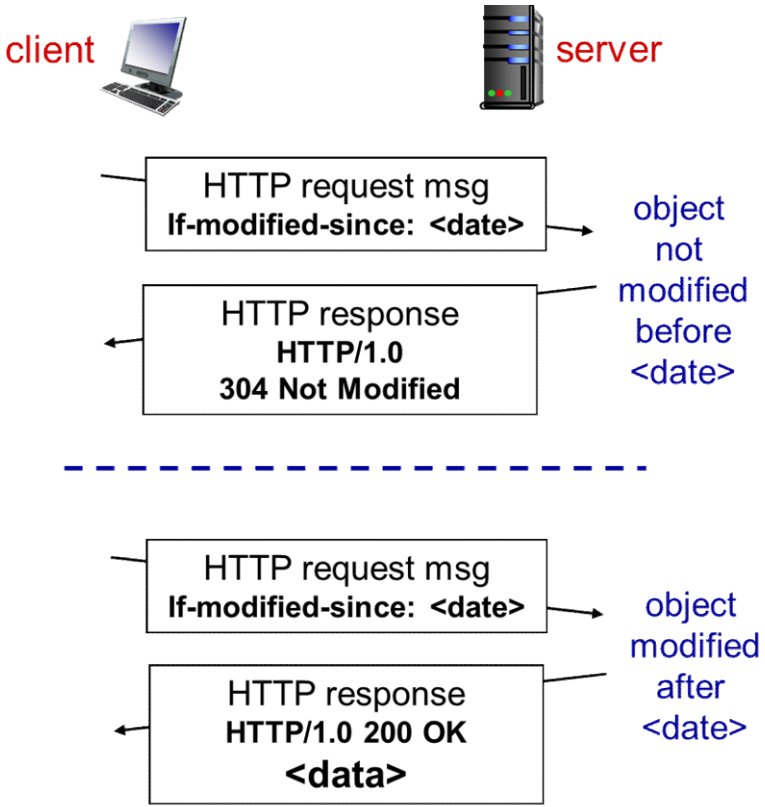
فرض کنید:
- میانگین اندازه بسته‌ها: ۱۰۰kb
- تعداد درخواست‌ها از مرورگرها به سرویس‌دهنده اصلی بطور میانگین: ۱۵ درخواست در ثانیه.
- میانگین نرخ داده به مرورگرها: ۱/۵Mbps
- لحظه ارسال درخواست از مسیریاب اینترنت تا دریافت پاسخ برابر: ۲ ثانیه
- فرض کنید ۴۰٪ مواقع در مراجعه به کش برخورد رخ دهد.

درنتیجه:
- تاخیرکل برابر است با:
- ۰/۶ تاخیرات در مراجعه به سرویس‌دهنده اصلی + ۰/۴ تاخیر در شبکه محلی



0.6 (200ms + 2S) + 0.4x0 = 1.32S

فرمان GET شرطی



- هدف از فرمان GETشرطی چک نمودن شیء موجود در حافظه نهان است. درصورتیکه شیء در مدت زمان قرار گرفتن در حافظه نهان، در سرویس دهنده اصلی تغییر نکرده باشد، آنگاه سرویس دهنده فقط یک پیام مبنی بر تغییرنکردن شیء به حافظه نهان ارسال می‌کند.
- حافظه نهان در پیام GETشرطی به سرویس دهنده در آن از خط سرایند زیر همراه با تاریخ آخرین ویرایش استفاده می‌کند.

```
If-modified-since: <date>
```

- درصورتیکه کپی موجود در حافظه نهان به‌روز باشد؛ انگاه سرویس‌دهنده از خط وضعیت زیر استفاده می‌کند.

```
HTTP/1.0 304 Not Modified
```

مثال (۳): در شکل اسلاید بعد، اگر اندازه متوسط اشیاء درخواستی ۸۵۰۰۰۰ بیت، و مرورگرها در هر ثانیه ۱۳/۵ درخواست بطور میانگین به سرویس‌دهنده اصلی می‌فرستند.اگر ارسال هر درخواست HTTP و برگشت ان از طریق مسیریاب اینترنت ۳ثانیه طول بکشد.

الف) زمان پاسخ میانگین کل را بیابید.

ب)اگر در شبکه LAN این سازمان یک حافظه نهان با نرخ برخورد ۰/۴ وجود داشته باشد.زمان پاسخ میانگین کل چقدر است؟

میانگین تاخیر دسترسی در لینک دسترسی برابر است با:

$$\frac{\text{میانگین زمان ارسال یک شی}}{\text{۱-شدت ترافیک}}$$

الف) مدت زمان ارسال = تأخیر انتقال + تأخیر انتشار

$$\text{تأخیر انتقال} = \frac{85 \times 10^4}{15 \times 10^6} = 567 \times 10^{-4}\,s$$

$$\text{تأخیر انتشار} = \frac{10^3}{10^5} = 10^{-2}\,S = 100 \times 10^{-4}\,s$$

تأخیر کل در لینک لایه متاطی $= 667 \times 10^{-4}\,S$

نشدت ترافیک $= 13.5 \times 667 \times 10^{-4} = 0.9$

میانگین تأخیر دسترسی $= \frac{667 \times 10^{-4}}{1 - 0.9} = 0.667\ S$

زمان پاسخ میانگین کل $= 3 + 0.667 = 3.667S$

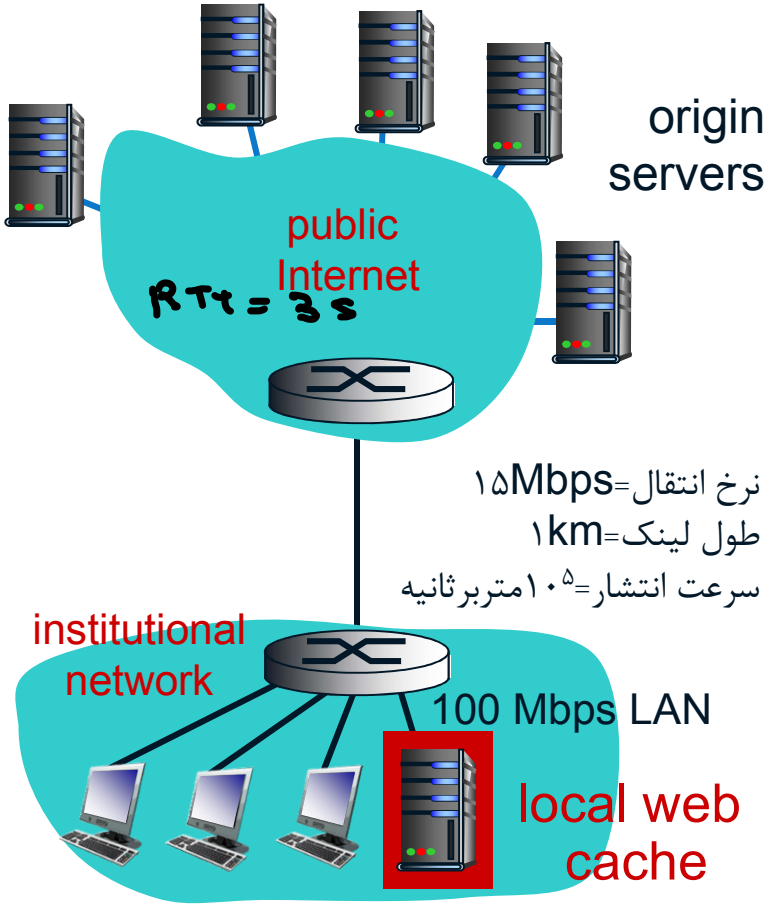
ب) چون نرخ برخورد برابر 0.4 است لذا در 0.6 مواقع به سرویس دهنده اصلی ارجاع می‌شود.

$$\text{شدت ترافیک} = 0.6 \times 0.9 = 0.54$$

$$\text{تأخیر دسترسی} = \frac{667 \times 10^{-4}}{1 - 0.54} = 0.145$$

$$\text{زمان دسترسی به سرویس دهنده اصلی} = 3 + 0.145 = 3.145$$

$$\text{زمان کل} = 0.4 \times 0 + 0.6 \times 3.145 = 1.8875$$

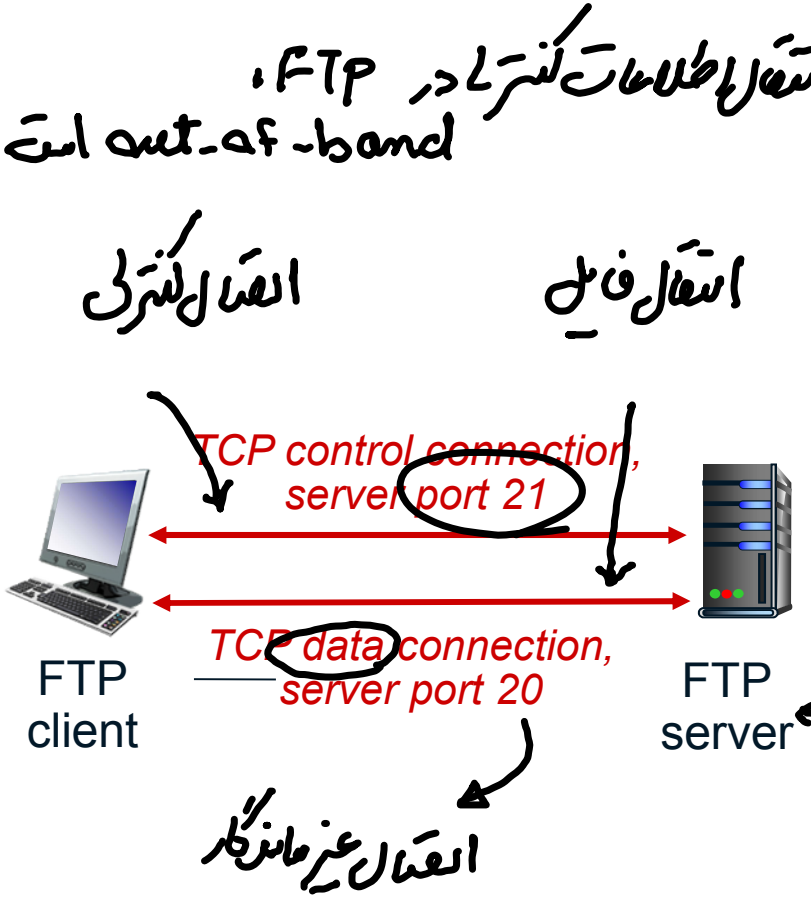

origin
servers
public
Internet
RTt = 3s
نرخ انتقال=۱۵Mbps
طول لینک=۱km
سرعت انتشار=۱۰^۵متربرثانیه
institutional
network
100 Mbps LAN
local web
cache

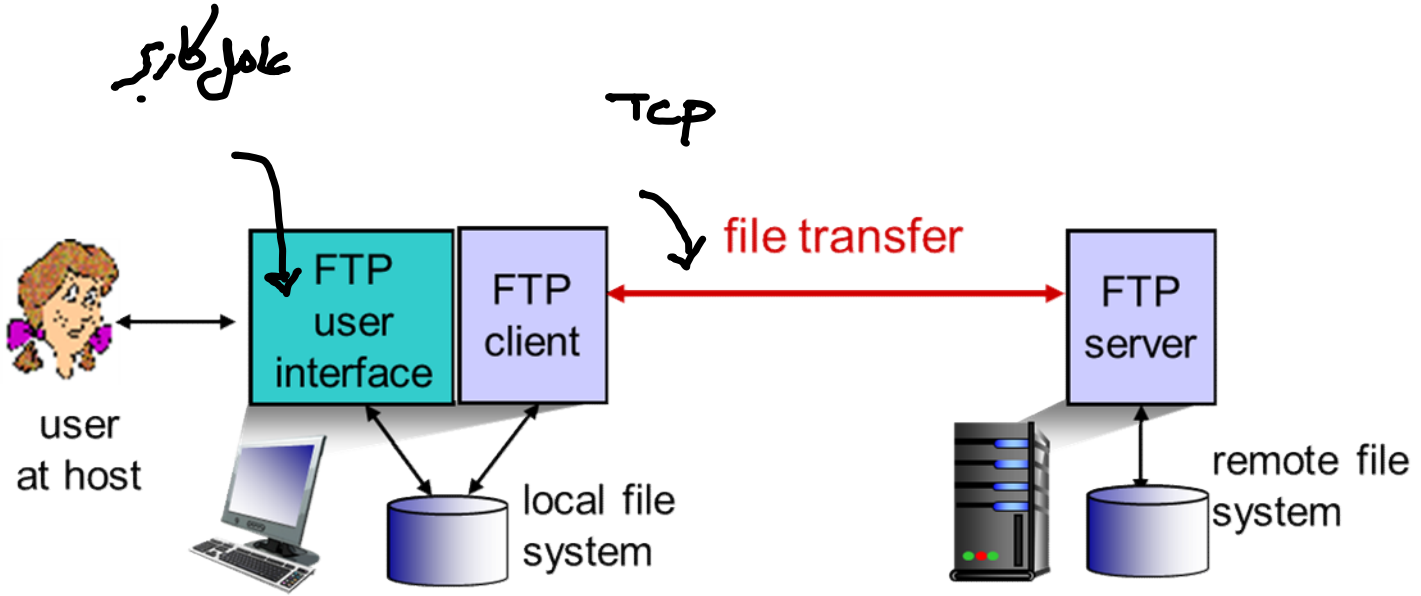
رئوس مطالب:

- ❑ اصول برنامه‌های کاربردی شبکه
- ❑ وب : پروتکل HTTP
- ✅ **انتقال فایل: FTP**
- ❑ **پست الکترونیکی:SMTP**
- ❑ **سرویس دایرکتوری اینترنت: DNS**
- ❑ **برنامه‌های کاربردی P2P**
- ❑ **برنامه نویسی سوکت با UDPوTCP**

FTP:پروتکل انتقال فایل

- انتقال فایل به/از میزبان راه دور
- مدل مشتری-سرویس‌دهنده
  - مشتری: سمتی که انتقال را شروع می‌کند.
  - سرویس دهنده: میزبان راه دور
- همانند پروتکل HTTP یک پروتکل انتقال فایل است.
- برای ارتباط از TCP بر روی پورت ۲۱ استفاده می‌کند.
- برخلاف HTTP از دو اتصال موازی استفاده می‌کند.




عامل کاربر
TCP
FTP
user
interface
FTP
client
file transfer
FTP
server
user
at host
local file
system
remote file
system

فرمان‌ها و پاسخ‌های FTP

برخی از فرامین

- همانند HTTP فرامین آن قابل فهم هستند.

- **USER *username***
- **PASS *password***
- **LIST**: درخواست فهرست تمامی فایل‌های دایرکتوری جاری
- **RETR filename**: درخواست ارسال فایل مشخص شده از دایرکتوری جاری(ایجاد یک اتصال غیرماندگار از سرویس‌دهنده به مشتری)
- **STOR filename** : ارسال فایل مشخص شده در دایرکتوری جاری به سرویس‌دهنده تا در دایرکتوری جاری راه دور ذخیره شود.(ایجاد یک اتصال غیرماندگار از مشتری به سرویس‌دهنده)

فرمان‌ها و پاسخ‌های FTP...

RFC 959

- همانند HTTP پیام‌های پاسخ یک عدد ۳ رقمی با یک توضیح مختصر هستند.
- `331 Username OK, password required`
- `125 data connection already open; transfer starting`
- `425 Can't open data connection`
- `452 Error writing file`

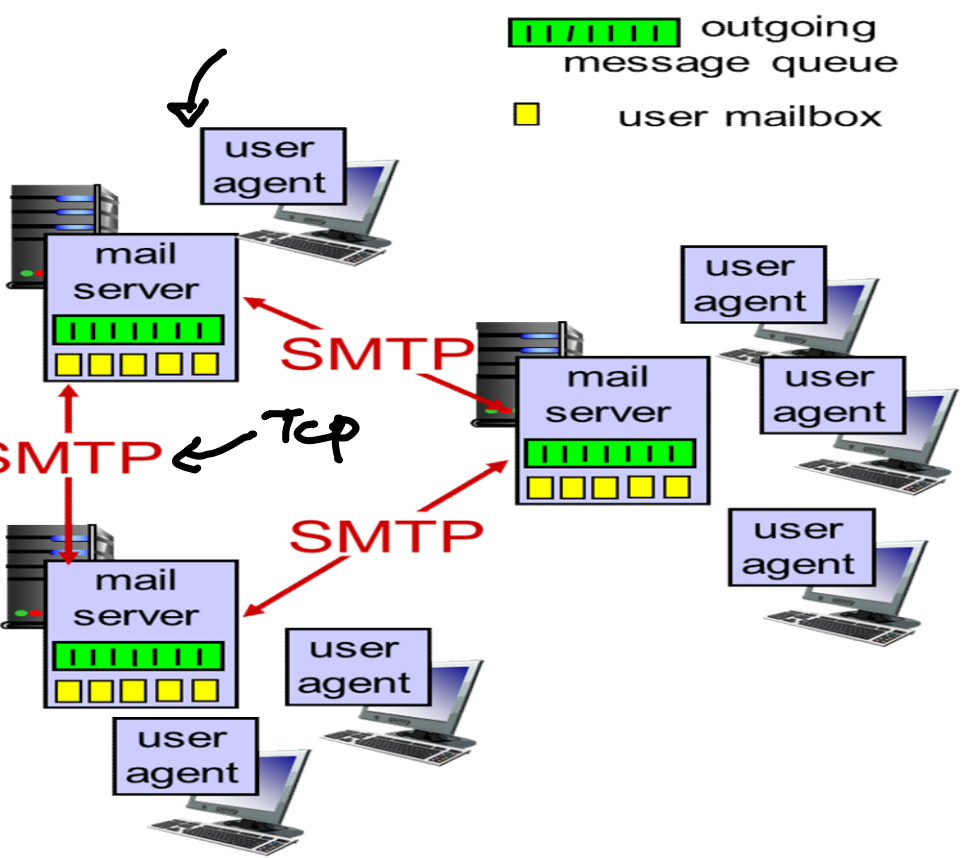
رئوس مطالب:

- ❑ اصول برنامه‌های کاربردی شبکه
- ❑ وب : پروتکل HTTP
- ❑ انتقال فایل: FTP
- ✅ پست الکترونیکی:SMTP
- ❑ سرویس دایرکتوری اینترنت: DNS
- ❑ برنامه‌های کاربردی P2P
- ❑ برنامه نویسی سوکت با UDPوTCP

پست الکترونیک در اینترنت

۳ مولفه:
۱- عامل کاربر UA
۲- سرویس دهنده ایمیل
۳- پروتکل انتقال نامه

UA
outlook
تحت وب

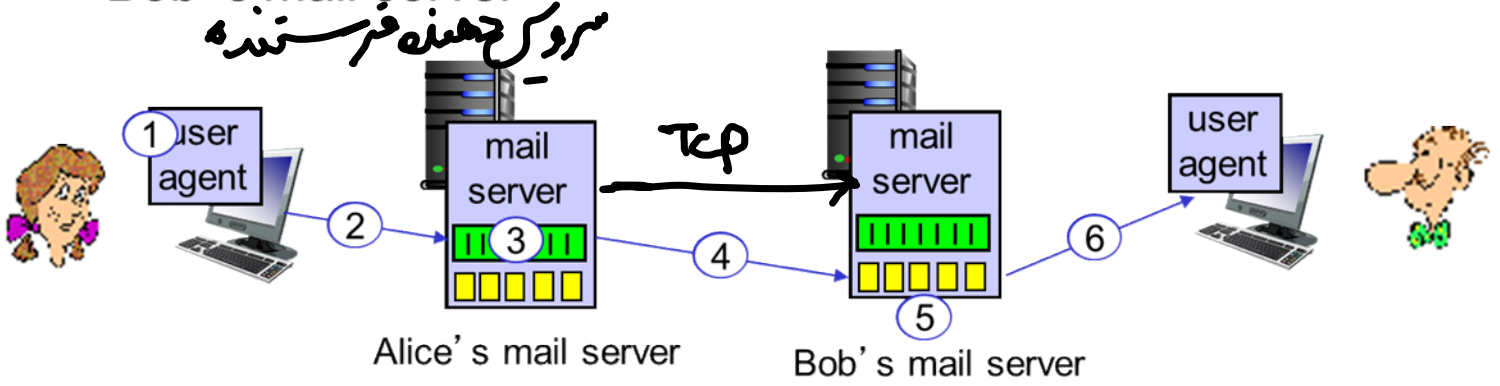


پروتکل انتقال نامه:SMTP

- از TCP برای انتقال مطمئن روی پورت ۲۵ استفاده می‌کند.
- نامه را از سمت سرویس‌دهنده فرستنده به سرویس‌دهنده گیرنده انتقال می‌دهد.
- انتقال فایل شامل سه مرحله است:
  - دست تکانی(Handshaking) –معرفی اولیه مشتری و سرویس‌دهنده.
  - انتقال پیام
  - بستن ارتباط
- همانند HTTP و FTP یک پروتکل فرمان/پاسخ است.
- پیام‌ها باید اسکی ۷-بیتی باشند.

1) Alice uses UA to compose message "to" `bob@someschool.edu`
2) Alice's UA sends message to her mail server; message placed in message queue
3) client side of SMTP opens TCP connection with Bob's mail server

4) SMTP client sends Alice's message over the TCP connection
5) Bob's mail server places the message in Bob's mailbox
6) Bob invokes his user agent to read message



S:
C. } handshaking

C: MAIL FROM: آدرس ایمیل فرستنده

S: —

C: RCPT TO. آدرس ایمیل گیرنده

S: —

C. داده‌ها

⋮

C: .

S: Ack

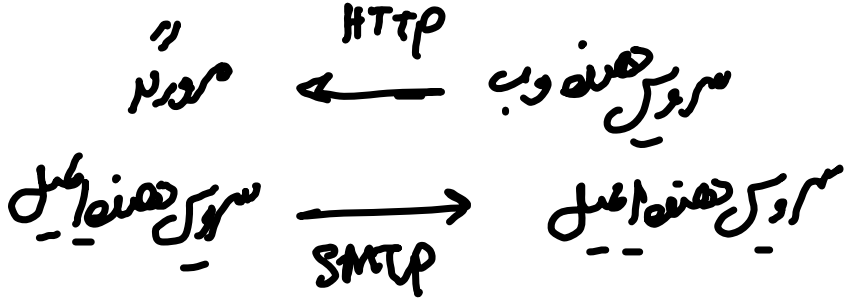
C. بسته شدن ارتباط

S: بسته شدن ارتباط

مقایسه SMTP با HTTP:

وجه تشابه:

1- هردو پروتکل انتقال فایل



2- هردو از پروتکل TCP استفاده می‌کنند.

انتقال فایل: SMTP: مانوگار
HTTP: مانوگار یا غیرمانوگار

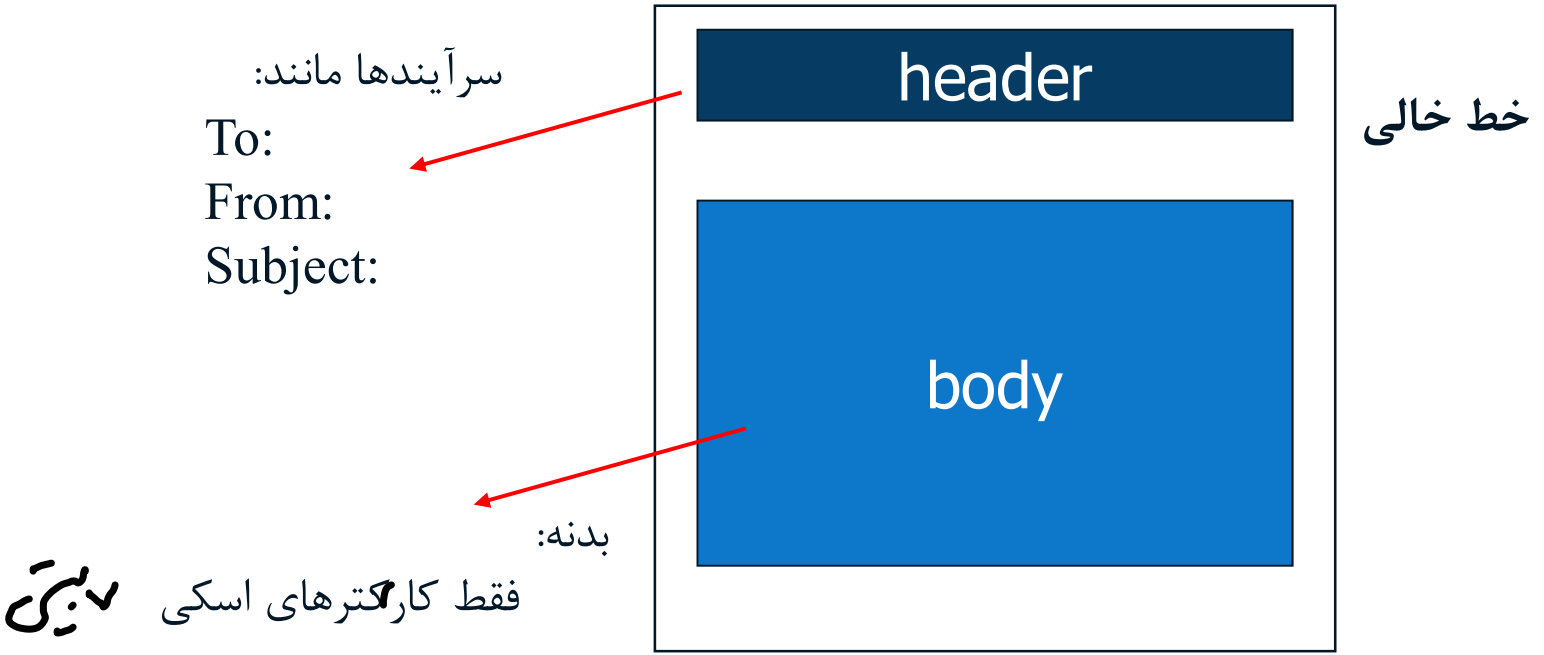
وجه تمایز

۱- پروتکل HTTP، گرفتنی است (pull). پروتکل SMTP فرستنی است (push).

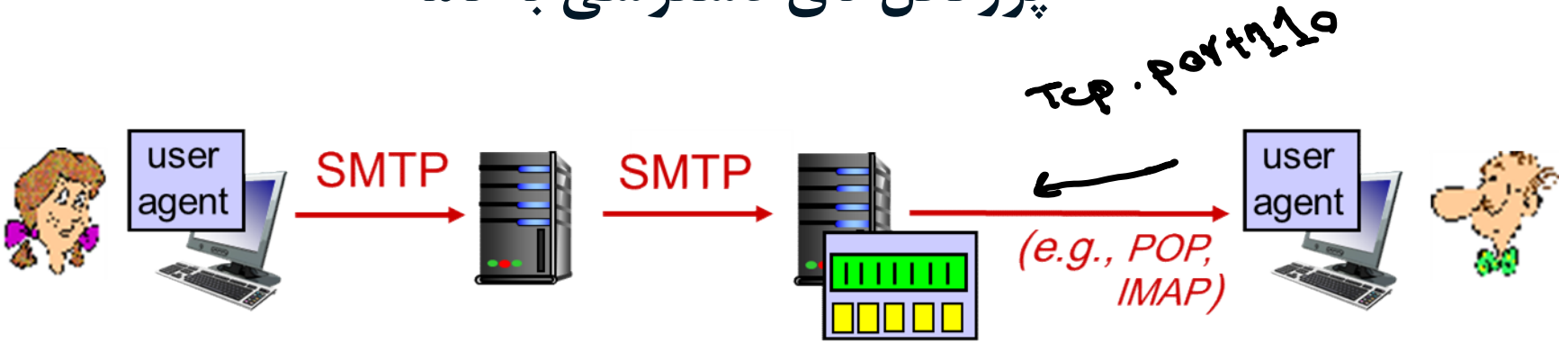
2- در SMTP تمام پیام‌ها (سرآیندها، بدنه پیام و ریپوست‌ها) باید به اسکی ۷ بیتی تبدیل شوند در حالیکه در HTTP این محدودیت وجود ندارد.

3- در SMTP تمام اجزای یک ایمیل چندقالب یک پیام واحد ارسال می‌شوند. اما HTTP هر شیء را با یک پیام جداگانه ارسال می‌کند.

فرمت پیام‌های نامه



پروتکل‌های دسترسی به نامه



- SMTP پروتکل فرستادنی است، و برای گرفتن نامه‌ها از سرویس‌دهنده کاربرد ندارد.
- پروتکل‌های دسترسی به نامه‌ها در سرویس‌دهنده گیرنده:
  - POP3: یک پروتکل دسترسی به نامه بسیار ساده است.
  - IMAP: یک پروتکل دسترسی به نامه با قابلیت‌هایی مثل دسته‌بندی پیام‌ها روی سرویس‌دهنده است.
  - HTTP: دسترسی از طریق HTTP و عامل کاربر همان مرورگر می‌باشد.

UA ⟶ Server

Tcp . port 110

3 حالت:

1) مجوز دسترسی : User pass

+OK
-ERR

2) تراکنش = ماربابی ایمیل ها : حذف پیام ها

3) تأیید حذف پیام ها - همگام سازی

پروتکل POP3

```
S: +OK POP3 server ready
C: user bob
S: +OK
C: pass hungry
S: +OK user successfully logged on
```

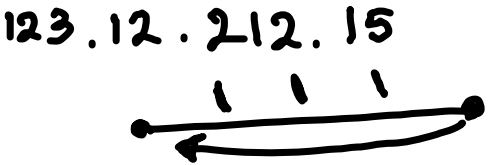
```
C: list
S: 1 498
S: 2 912
S: .
C: retr 1
S: <message 1 contents>
S: .
C: dele 1
C: retr 2
S: <message 1 contents>
S: .
C: dele 2
C: quit
S: +OK POP3 server signing off
```

c. telnet mailServer 110

رئوس مطالب:

- ❑ اصول برنامه‌های کاربردی شبکه
- ❑ وب : پروتکل HTTP
- ❑ انتقال فایل: FTP
- ❑ پست الکترونیکی:SMTP
- ✅ **سرویس دایرکتوری اینترنت: DNS**
- ❑ **برنامه‌های کاربردی P2P**
- ❑ **برنامه نویسی سوکت با TCPوUDP**

DNS: domain name System



- چگونه نام یک میزبان به آدرسی موسوم به IP تبدیل خواهد شد؟

- سیستم نام دامنه (DNS) شامل دو بخش است:

  ۱- یک پایگاه داده توزیع شده در یک ساختار سلسله مراتبی از سرویس‌دهنده‌های DNS

  ۲-یک پروتکل لایه کاربرد که امکان جستجو در پایگاه داده توزیع شده را فراهم می‌کند.

- پروتکل‌های لایه کاربرد مانند HTTP،SMTP و FTP از برای تبدیل نام میزبان به آدرسIP از DNS استفاده می‌کنند.

مثالی از ترجمه آدرس توسط DNS در HTTP

۱-ماشین کاربر برنامه کاربردی سمت مشتری HTTP، DNS را اجرا می‌کند.
۲-مرورگر وب نام سرویس‌دهنده میزبان وب را از URL استخراج نموده و انرا به برنامه سمت مشتری DNS می‌دهد.
۳-مشتری DNS با نام میزبان موجود یک پرس‌وجو به سرویس‌دهنده DNS ارسال می‌کند.

۴-مشتری DNS پاسخ را از سرویس‌دهنده DNS دریافت می‌کند.این پاسخ شامل آدرس IP مربوط به میزبان HTTP است.
۵-مرورگر با دریافت آدرس IP از DNS، یک اتصال TCP روی پورت ۸۰ با فرایند سرویس‌دهنده HTTP ایجاد می‌کند.

سرویس‌های DNS

۱-تبدیل نام میزبان به آدرس IP

۲-بدست آوردن نام مستعار میزبان

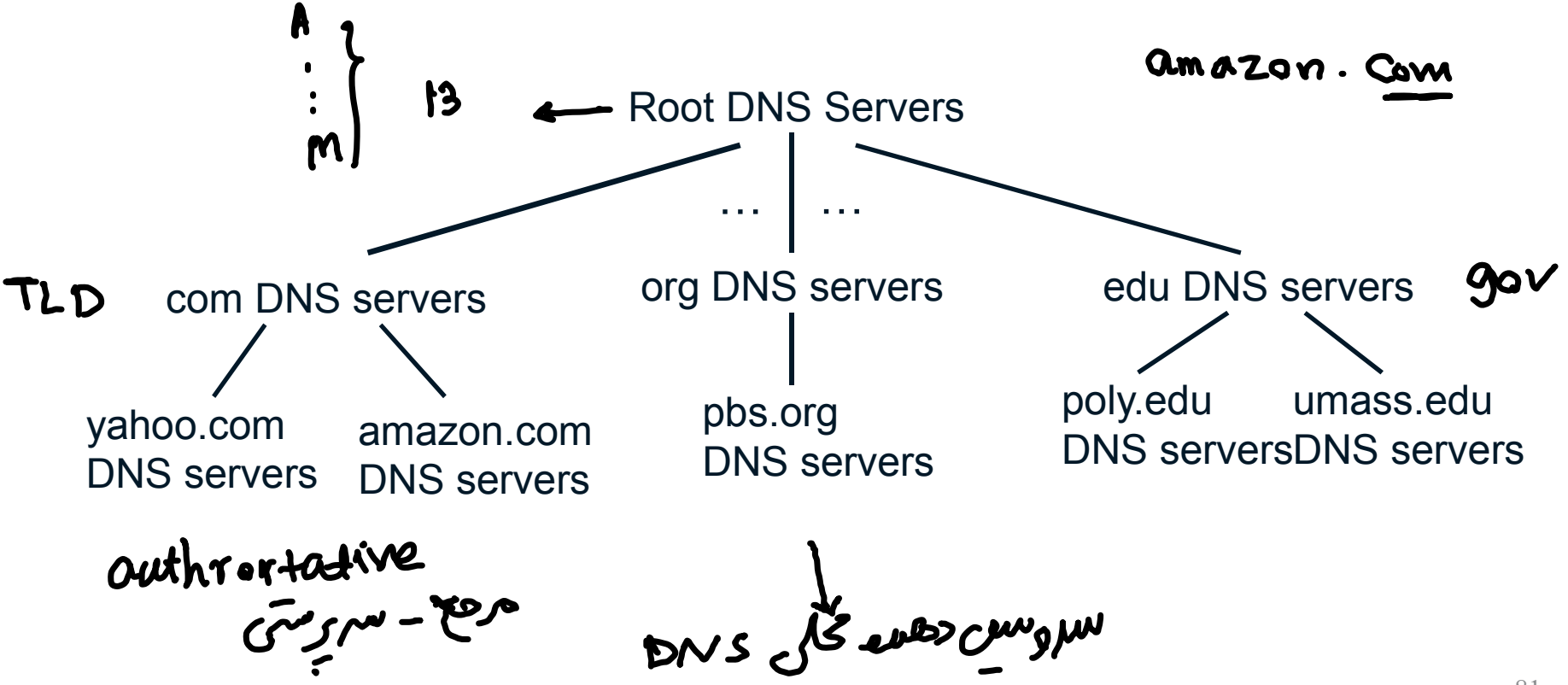
۳-بدست آوردن نام مستعار سرویس‌دهنده پست الکترونیکی

۴-توزیع بار

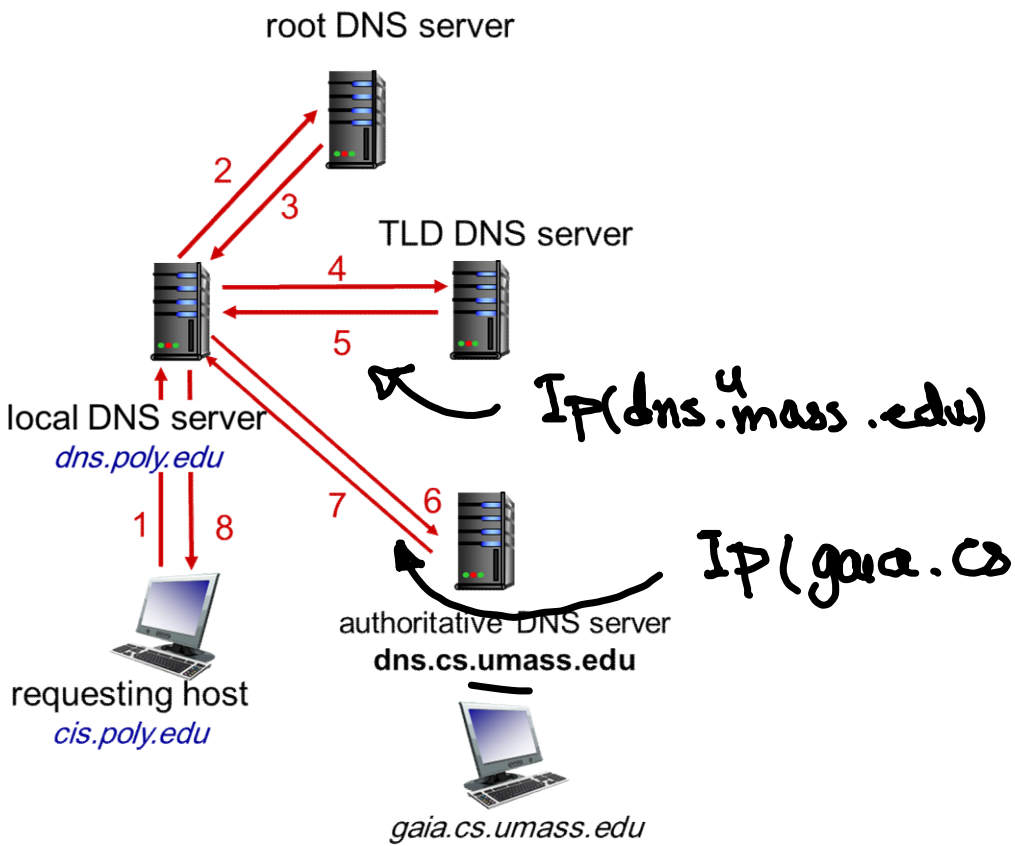
مروری بر طرز کارDNS

- ساده ترین مدل برای DNS بدین صورت است، که تمام کاربران به یک سرویس‌دهنده متمرکز ارسال کنند اما این طرح چندین مشکل اساسی دارد:
  - نقطه شکست واحد: درصورت ازکار افتادن سرویس‌دهنده کل اینترنت از کار خواهد افتاد.
  - حجم ترافیک: سرویس‌دهنده واحد باید تمامی درخواست‌های تبدیل آدرس را پاسخ دهد.
  - فاصله زیاد بعضی مشتریان از سرویس‌دهنده
  - بهنگام سازی و نگهداری با هزینه سنگین

پایگاه داده سلسله مراتبی توزیع شده



یک مثال از ترجمه آدرس

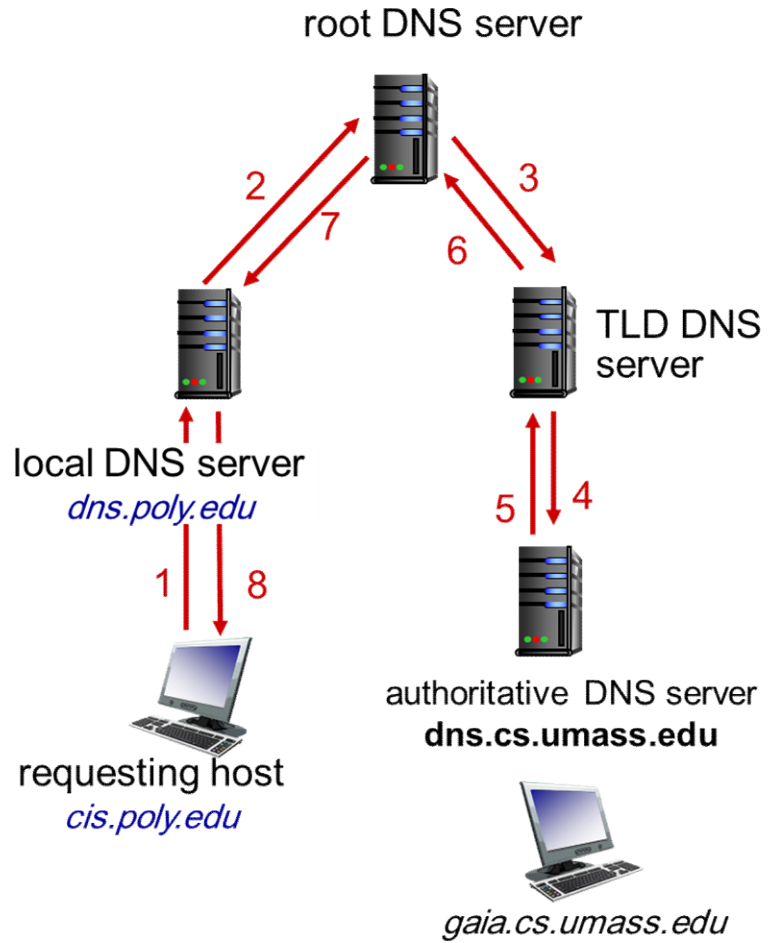


- فرض کنید که میزبان cis.poly.edu آدرس IP میزبان gaia.cs.umass.edu را درخواست کند:
- سرویس‌دهنده محلی دانشگاه پلی‌تکنیک dns.poly.edu:
- سرویس‌دهنده اصلی دانشگاه ماساچوست: dns.umass.edu

دو نوع پرس وجو وجود دارد :

1- بازگشتی : در صورتیکه یک سرور دهنده به نیابت از پرس جو کننده دنبال آدرس IP مانند بازگشتی است .

2 - تکرار شونده : در صورتیکه پاسخ پرس وجو به سوال کننده برگردد تکرار شونده است


root DNS server
2
7
3
6
TLD DNS server
local DNS server
dns.poly.edu
5
4
1
8
authoritative DNS server
dns.cs.umass.edu
requesting host
cis.poly.edu
gaia.cs.umass.edu

رکوردها و پیام‌های DNS

DNS: یک پایگاه داده توزیع شده شامل رکوردهای منبع(RR)

RR format: **(name, value, type, ttl)**

type=A

- **Name** نام میزبان
- **Value** **IP** میزبان

(faradars.org,130.185.74.123,A, 10794s)

type=NS

- **Name** نام دامنه
- **Value** نام میزبان سرویس دهنده اصلی

(faradars.org, noah.ns.cloudflare.com,NS)

alt3.aspmx.l.google.com

type=CNAME

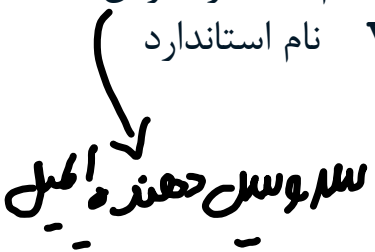
- **Name** نام مستعار میزبان
- **Value** نام استاندارد

(YAHOO.COM, atsv2-fp.wg1.b.yahoo.com ,CNAME)

Type=MX

- **Name** نام مستعار میزبان
- **Value** نام استاندارد

(GOOGLE.COM, alt3.aspmx.l.google.com,MX)

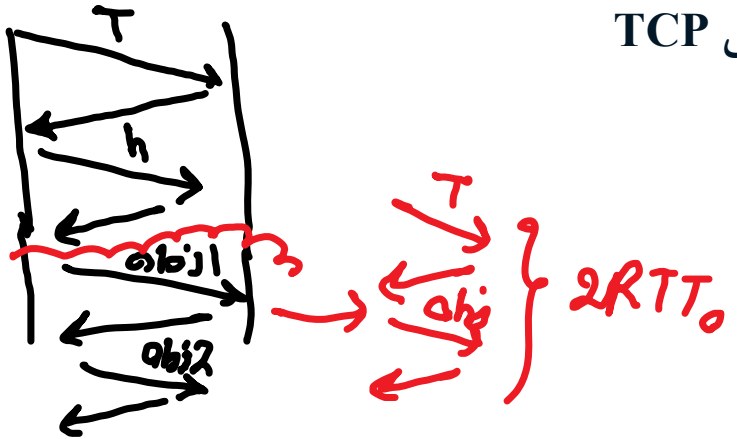


مثال (۴): یک کاربر در مرورگر خود یک صفحه وب را باز می‌کند.در این صفحه یک لینک را کلیک می‌کند،که آدرس IPمتناظربا URL آن در حافظه نهان DNS موجود نیست.اگر ماشین کاربر برای بدست آوردن آدرس IP از nسرویس‌دهنده DNS استفاده کند،و زمان رفت و برگشت درخواست‌ها RTT_1 تا RTT_n است. اگر در فایل HTML این صفحه به ۸ شی دیگر روی همان سرویس‌دهنده ارجاع شود.زمان دریافت این صفحه وب در هر حالت را محاسبه کنید.

الف) Non-Persistent HTTP

ب) Non-Persistent HTTP با ۳اتصال موازی TCP

ج) Persistent HTTP



الف)
$$\sum_{i=1}^{n} RTT_i + 2RTT_0 + 8(2RTT_0)$$

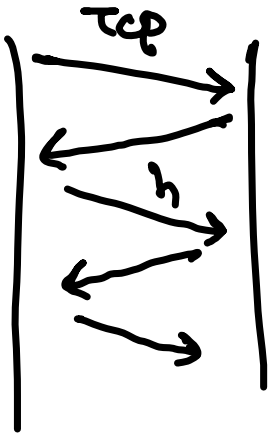
ایجاد ارتباط TCP

درخواست و دریافت html با سه

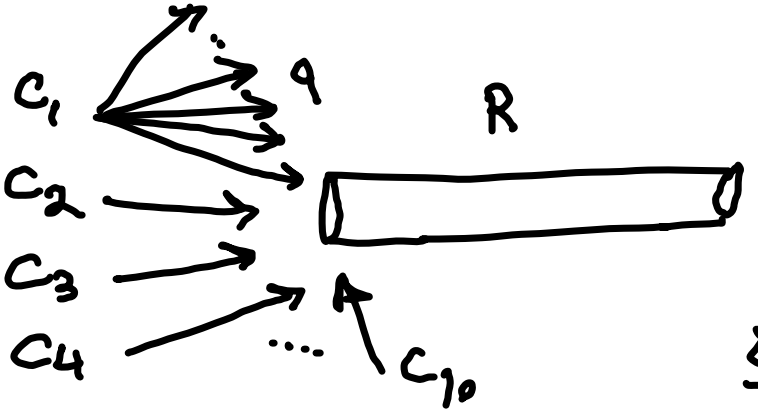
ب)
$$\sum_{i=1}^{n} RTT_i + 2RTT_0 + \underbrace{2RTT_0}_{\text{obj مورد 3}} + 2RTT_0 + 2RTT_0$$

$$= \sum_{i=1}^{n} RTT_i + 8RTT_0$$

c) $\sum_{i=1}^{n} RTT_i + 2RTT_0 + 8RTT_0 = 10RTT_0 + \sum_{i=1}^{n} RTT_i$



مثال (۴): فرض کنید ۱۰ کلاینت با استفاده از پروتکل **FTP** بطور همزمان در حال دریافت فایل‌هایی با حجم زیاد از یک سرویس‌دهنده فایل هستند، و لینک متصل به سرور گلوگاه می‌باشد.اگریکی از کلاینتها از برنامه **Download Manager** استفاده کند، که بطور همزمان ۹ اتصال موازی باز می‌کند،سرعت دانلود این کلاینت چند برابر می‌شود؟

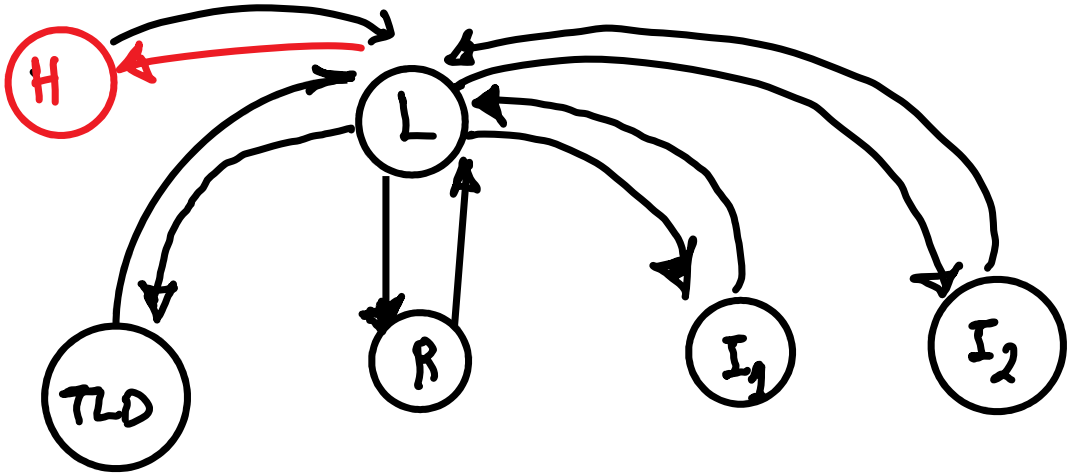


سریع = $\frac{R/2}{R/10} = \frac{10}{2} = 5$

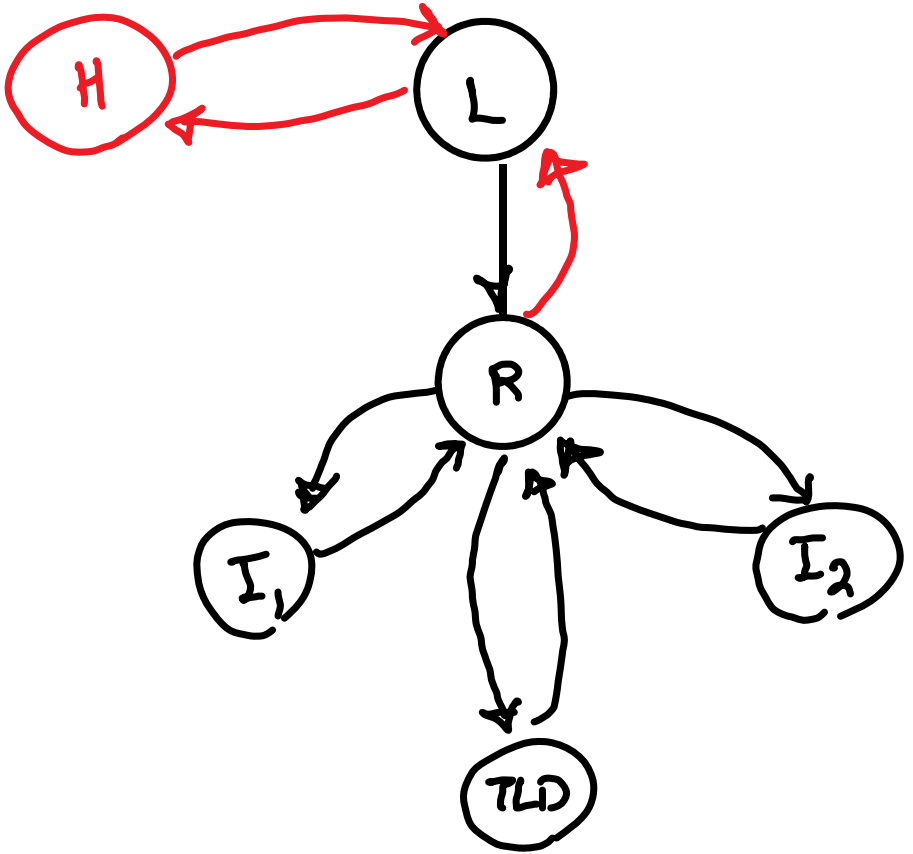
سرعت دانلود قبل از اتصالات موازی = $\frac{R}{10}$

سرعت دانلود بعد از اتصالات موازی = $\frac{9R}{18} = \frac{R}{2}$

- **مثال (۴): برای مشخص شدن آدرس IP متناظر با یک URL به سرویس‌دهنده محلی DNS یک پرس‌وجو ارسال می‌شود.و این سرویس‌دهنده پرس‌وجو را برای یک سرویس‌دهنده ریشه ارسال می‌کند.و سرویس‌دهنده ریشه باید درخواست را از سرویس‌دهنده مرجع بپرسد،اما بین سرویس‌دهنده مرجع و ریشه دو سرویس‌دهنده DNS میانی و یک سرویس‌دهنده TLD وجود دارد.درهرکدام ازحالات زیر سرویس‌دهنده محلی چند پرس‌وجو در مجموع ارسال می‌کند؟**
  الف) پرس جوها همگی تکراری(Iterative) هستند.
  ب)پروس جو مربوط به سرویس‌دهنده محلی بازگشتی (Recursive) باشد.



الف)

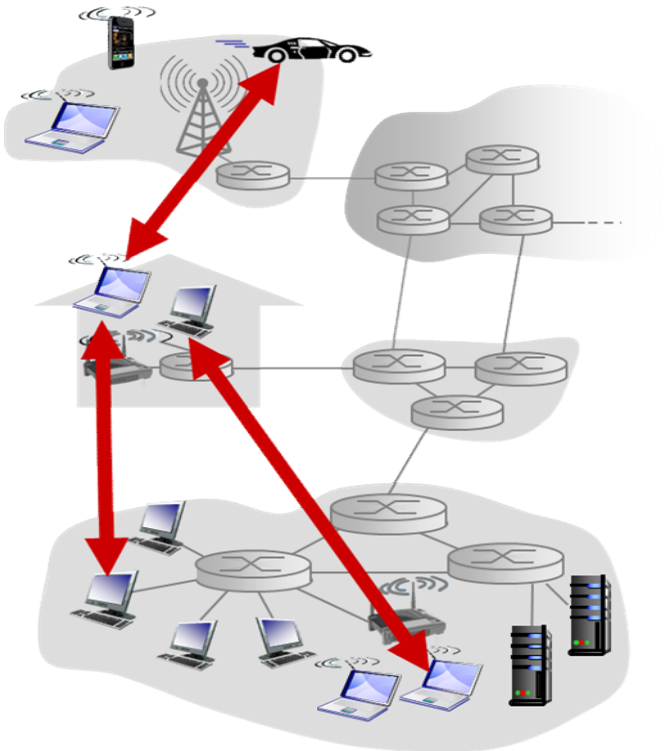


(ب

رئوس مطالب:

- ❑ اصول برنامه‌های کاربردی شبکه
- ❑ وب : پروتکل HTTP
- ❑ انتقال فایل: FTP
- ❑ پست الکترونیکی:SMTP
- ❑ سرویس دایرکتوری اینترنت: DNS
- ✅ **برنامه‌های کاربردی P2P**
- ❑ **برنامه نویسی سوکت با UDPوTCP**

برنامه‌های کاربردی P2P



- وابستگی به زیرساخت سرویس‌دهنده‌های همیشه روشن ندارد.
- میزبان‌ها بصورت مستقیم باهم ارتباط برقرار می‌کنند.

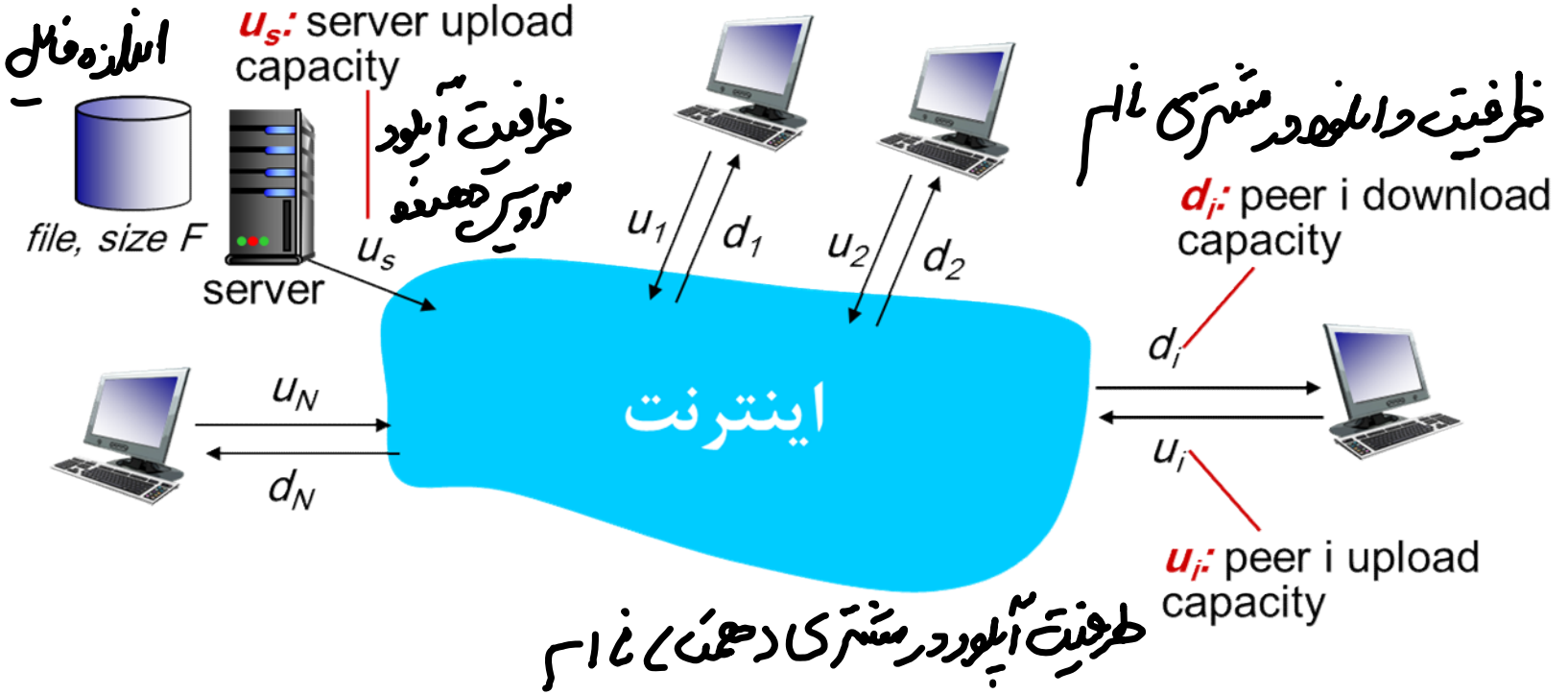
- دو کاربرد مهم از برنامه‌های P2P :

-توزیع فایل مانند بیت تورنت

- توزیع پایگاه‌داده روی چندین همتا

توزیع فایل: P2P یا مشتری-سرویس‌دهنده

- زمان توزیع یک فایلFبیتی در بین N مشتری توسط سرویس‌دهنده چقدر است؟



زمان توزیع در معماری مشتری-سرویس‌دهنده

- سرویس‌دهنده بایدNFبیت را ارسال کند.در نتیجه زمان توزیع برابر NF/u_s است.
- حال اگر $d_{min}=\min\{d_1,d_2,\ldots,d_N\}$ باشد. آنگاه مدت زمان دریافت ضعیفترین همتا برابر F/d_{min} است.
- لذا مدت زمان توزیع در مدل مشتری-سرویس‌دهنده برابر است با حداکثر دو مقدار بالا:

$$D_{cs} \geq \max\left\{\frac{NF}{u_s}, \frac{F}{d_{\min}}\right\}$$

کران پایین

$$D_{cs} = \max\left\{\frac{NF}{u_s}, \frac{F}{d_{min}}\right\}$$

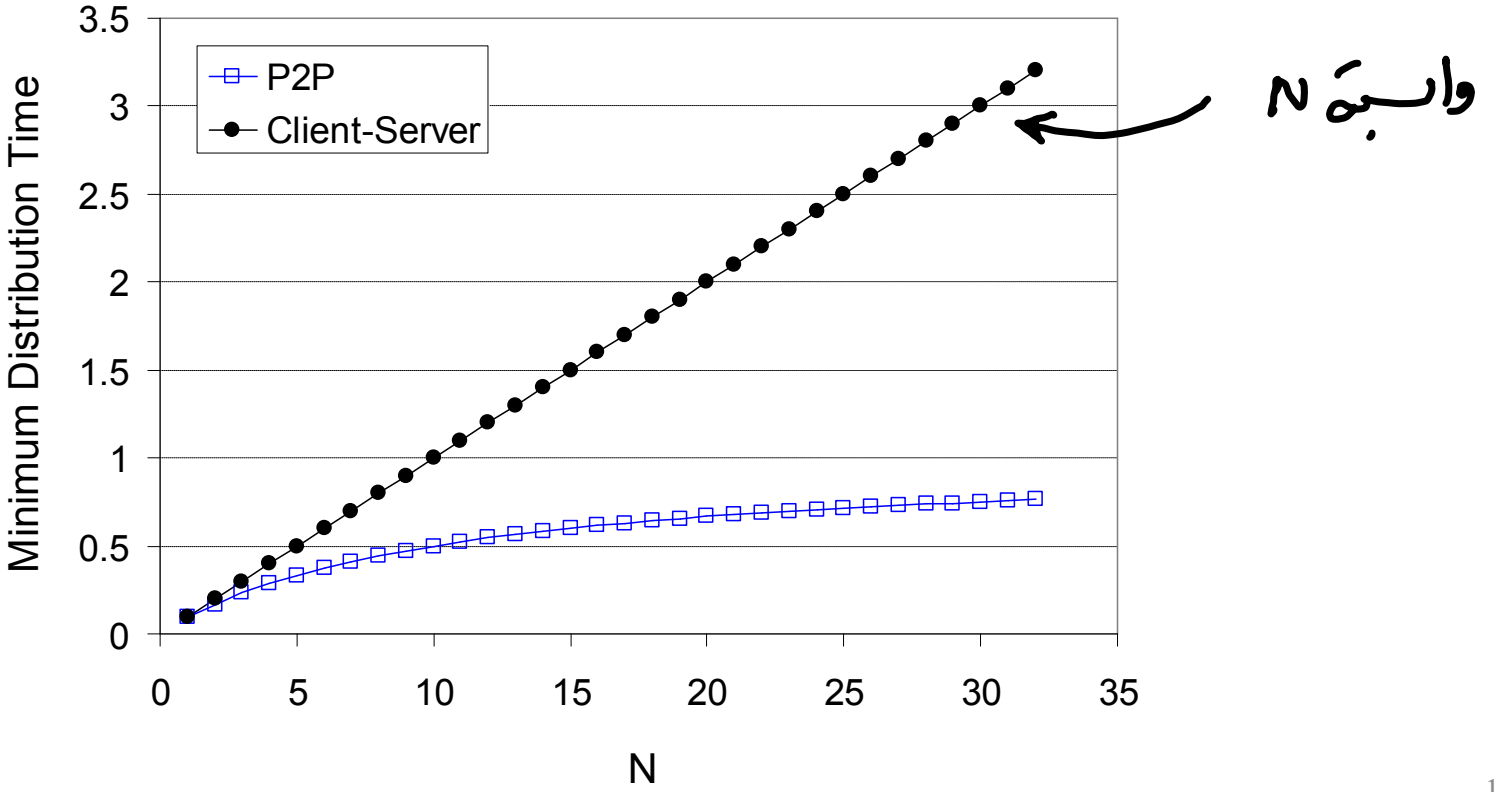
زمان توزیع در معماری P2P

- سرویس‌دهنده باید حداقل یکبار فایل را اپلود کند.درنتیجه حداقل زمان توزیع برابر F/u_s است.
- حال اگر $d_{min}=\min\{d_1,d_2,\ldots,d_N\}$ باشد. آنگاه مدت زمان دریافت ضعیفترین همتا برابر F/d_{min} است.
- چون تمام همتاها بعد دریافت بیت‌ها می‌توانند انها را بین خودشان به اشتراک بگذارند لذا ظرفیت آپلود کل سیستم برابر است با:$u_{total}=u_s+u_1+u_2+\ldots+u_N$.
- در سیستم NFبیت جابجا خواهد شد.و این کار با حداکثر نرخ u_{total} انجام می‌شود.لذا کمترین زمان توزیع برابر است با: NF/u_{total} .

$$D_{P2P} \geq \max\left\{\frac{F}{u_s},\frac{F}{d_{\min}},\frac{NF}{u_{total}}\right\}$$

$$u_{total} = u_s + \sum_{i=1}^{N} u_i$$

client upload rate = u, F/u = 1 hour, $u_s = 10u$, $d_{min} \geq u_s$



واسبة N