



Redes de Computadoras

Profesores: César Luis Zaccagnini, Sergio Daniel Loyola y

Leonardo Jose Balbiani

Comisión: 1038-1-G14

Estudiante: Flores Rea Frank Matias

Trabajo Práctico N°2:

Problema 2-

El recurso / :

GET /HTTP/1.1

Encabezado del recurso /tac :

HEAD/tac HTTP/ 1.1

El recurso /logo.jpg si no fue modificado desde una determinada fecha :

GET /logo.jpg HTTP/1.1

Host: www.dc.uba.ar

If-Modified-Since: Tue, 29 Aug 2024 12:00:00 GMT

Problema 3- La diferencias entre los dos comandos que si bien se conectan al mismo equipo, que se conectan por telnet por el puerto 80, en el primer comando pide la informacion del www.inta.gob.ar y el segundo comando pide la informacion de www.mercosurt.gob.ar

Problema 4:

4.1- Solo se necesitaria 1 solo servidor web

4.2- Para el servidor una sola dirección IP alcanza para que los cliente pueda conectarse

4.3-Para que todos los clientes puedan conectarse solo se necesitaria 1 solo puerto con modularización

4.4- Para 150.000.000 de clientes necesitaria las mismas configuraciones porque en el protocolo HTTP sería lo mismo tener unos o miles de clientes, sin tener en cuenta las condiciones del hardware que se tiene para el servidor.

Problema 5: Salida:

HTTP/1.1 200 OK ➡ me indica la versión de protocolo que utiliza(HTTP/1.1), el código de estado(200) y el significado del código(OK), en este caso se realizo la conexion con éxito

Accept-Ranges: bytes ➡ me indica que tipo de solicitudes acepta

Date: Thu, 24 Mar 2011 23:15:43 GMT ➡ es la fecha en la que comienza la transferencia de datos

Content-Length: 286 ➡ Me indica la longitud en bytes de la página

Content-Type: text/html ➡ Me indica que el tipo del cuerpo es texto/html

Last-Modified: Wed, 06 Oct 2010 12:46:39 GMT ➡ Es la fecha de la última modificación de la página

ETag: "749338855465cb1:0" ➡ Es una etiqueta que sirve como un nombre para el contenido de la página

Server: Microsoft-IIS/7.0 ➡ Me indica que el servidor utilizado es Microsoft-IIS/7.0

X-Powered-By: ASP.NET ➡ me indica que el servidor está potenciado o utiliza la tecnología ASP.NET



Redes de Computadoras

Profesores: César Luis Zaccagnini, Sergio Daniel Loyola y

Leonardo Jose Balbiani

Comisión: 1038-1-G14

Estudiante: Flores Rea Frank Matias

Problema 6.

Si se utiliza HTTP/1.0 se necesita 1 petición y por cada petición por cada objeto que se pide, en este caso son 7 peticiones y 7 conexiones porque cada vez que pide un elemento cierra la conexión y si se utiliza HTTP/1.1 serian 3 conexiones y 7 peticiones porque una vez que se conecta con el servidor mantiene la conexión y trae todos los elementos del servidor uno tras otro (pipelining) seria una conexión por casas servidor que se tiene que conectar y 1 petición por cada elemento que esté en el servidor.

Problema 7. Cuando se visite el sitio index.html y que a su vez tiene que traer dos archivos.jpg de 2 servidores distintos, va a necesitar 3 conexiones de nivel de transporte(TCP) usando HTTP/1.1, porque para cada servidor establece una conexión nueva hasta que obtiene todos los archivos se pide del servidor y cierra la conexión.

Problema 8: Las conexiones a nivel de transporte(TCP) que usará el navegador para transferir la totalidad de los recursos son 3 conexiones TCP porque como son tres recursos(index.html; logo_transp y logo_mail) los que me tiene que traer, abre una conexión nueva para cada recurso sin importar si están en el mismo servidor como el index.html y el logo que están en el servidor unq.edu.ar.

Problema 9:

- Teniendo en cuenta los siguientes recursos que tiene que transferir:
index.html; style.css; searchline.png, home.png, search_icon.gif
- **A:** En HTTP/1.0 se necesita mínimo 2 RTT para abrir y cerrar la conexión TCP y la transferencia de recursos, tomando en cuenta esto y que en total son 5 los recursos que tiene que transferir, el tiempo sería de 10 RTTs usando HTTP/1.0
- **B:** Utilizando HTTP/1.1 como no necesita abrir y cerrar la conexión cada vez que hace un pedido, va a tener 1 RTT menos por pedido que en HTTP/1.0, el tiempo de transferencia sería de 5 RTTs

Problema 10.

a. El recurso index.html del servidor web conteniendo a su vez los recursos 1.jpg, 2.jpg, 3.jpg y 4.jpg hospedados en el Servidor Web, y los recursos 1.gif, 2.gif y 3.gif hospedados en el Servidor de Ads.

- **a:** Como la caché está vacía, va a tener que realizar el recorrido completo hasta el servidor, en este caso el tiempo que va a tardar en traer todos los objetos va a ser de **5*310**: se va a tardar en traer el index.html y las imágenes 1,2,3,4.jpg del servidor Web
3*410: se va a tardar en traer los recursos 1,2,3.gif del servidor ADS
 $(5*310) + (3*410) = 2780$ ms es el tiempo total en realizar el primer pedido



Redes de Computadoras

Profesores: César Luis Zaccagnini, Sergio Daniel Loyola y

Leonardo Jose Balbiani

Comisión: 1038-1-G14

Estudiante: Flores Rea Frank Matias

b. El recurso comprar.php del servidor web conteniendo a su vez los recursos 3.jpg, 4.jpg y 5.jpg hospedados en el Servidor Web, y los recursos 2.gif y 3.gif hospedados en el Servidor de Ads.

- En este pedido como en la cache del proxy ya están cacheadas los recursos 3 y 4.jpg y del servidor ADS ya están cacheadas los recursos 2 y 3.gif, estos recursos solo los va a buscar al proxy y no al servidor. El tiempo total sería el siguiente:

(4*10): Es el tiempo que se tarda en traer los recursos 3 y 4.jpg, 2 y 3.gif del proxy

(2*310): Es el tiempo que tarda en pedir el recurso comprar.php y 5.jpg

(4*10)+(2*310): **660 ms** es el tiempo total de transferencia total de recursos

c. El recurso gracias-por-comprar.html del Servidor Web conteniendo a su vez los recursos 1.jpg, 4.jpg y 6.jpg hospedados en el Servidor Web, y los recursos 1.gif y 3.gif hospedados en el Servidor de Ads.

- En este pedido solo va a buscar el recurso gracias-por-comprar.html y el 6.jpg y los demás recursos ya están cacheadas en el proxy por ende los va a buscar al proxy y no al servidor. Estos son los tiempos de transferencia de archivos:

(2*310): son los tiempos que tarda en pedir los recursos gracia-por-comprar.html y 6.jpg

(4*10 ms): es el tiempo que tarda en traer los recursos que están cacheados en el proxy

(2*310 ms)+(4*10 ms) = **660 ms** es el tiempo total