

## **Pràctica 1: Protocol HTTP**

### **Introducció**

L'objectiu d'aquesta pràctica és l'estudi del funcionament del protocol petició-resposta HTTP. S'han de tenir clars els elements principals que intervenen en les comunicacions web i els protocols TCP/IP i HTTP)

### **A) Busca informació i respon les següents preguntes indicant les referències utilitzades:**

1. A quin port es reben normalment les peticions del protocol HTTP? A quina capa del model TCP/IP es troba el protocol HTTP? I els protocols TCP, UDP, i IP?
2. Respon les següents preguntes:
  - a) El protocol HTTP és un protocol client-servidor? Raona la resposta.
  - b) A quin camp de la capçalera d'un missatge HTTP pots trobar l'adreça IP del servidor al qual va dirigida una petició.
  - c) A quin camp de la capçalera d'un missatge HTTP pots trobar informació sobre el programa client que ha realitzat una petició.
  - d) Què és el Request Payload Body d'un missatge HTTP?
  - e) Què significa que HTTP és un protocol sense estat (stateless)?
3. Indica de quina manera el client faria una petició de la pàgina estàtica index3.html que és troba a la carpeta "exam" que penja de l'arrel de l'arbre de directoris del servidor web. S'ha d'indicar mètode, fitxer, protocol i versió del protocol.
4. Indica l'esquema URI per accedir a un fitxer que s'anomena ex1m08uf1.pdf que es troba a la carpeta /home/daw2/examen d'un servidor FTP al qual s'ha d'accedir amb el login daw i password m08uf1pr1. El nom del servidor és ftp.fjeclot.edu i escolta pel port 21.
5. Indica quines són les diferències existents entre utilitzar un mètode POST i un mètode PUT
6. Indica quines són les parts de les quals es compon una entitat i el propòsit de cadascuna d'elles.
7. Indica quins són els 5 grups de missatges que pot enviar un servidor al client i quins són els propòsits de cadascun d'ells.
8. Indica quins mètodes farien que el servidor respongués amb el missatge 201.
9. Respon amb verdader o fals les següents afirmacions:
  - a) El mètode POST és menys segur perquè envia dades dins de la URL. Raona la resposta.
  - b) El mètode GET reenvia la informació quan es recarrega una web i el navegador hauria d'avisar-nos d'aquest fet. Raona la resposta.
  - c) Les peticions de tipus POST no poden ser desades a les adreces d'interès (bookmark)
  - d) Una petició de tipus POST envia les dades dins del cos del missatge
  - e) Les peticions de tipus GET no admeten qualsevol tipus de dades (texte, imatge, video, etc..)
10. Quin és el significat de la següent resposta d'un servidor:  
HTTP/1.1 302 Found  
Location: http://www.example.com/test/index2.php
11. Indica quin és el propòsit del següent codi PHP

```
<?php
    header("Cache-Control: no-cache, must-revalidate"); //HTTP 1.1
    header("Pragma: no-cache"); //HTTP 1.0
    header("Expires: 0"); // Date in the past
?>
```

## B) Peticions i respostes HTTP treballant amb Telnet i Wireshark

1. Filtra el contingut de les captures de manera que només es mostrin els continguts dels missatges HTTP. Fes la següent petició HTTP utilitzant: **telnet www.google.cat 80**

```
GET / HTTP/1.1
Host: www.google.cat
User-Agent: telnet
```

Mostra el resultat obtingut amb Wireshark, i troba:

- a) Dins de la capçalera IP, l'adreça IP del teu ordinador i la del servidor.
  - b) El port (tipus i valor) utilitzat pel programa telnet i pel servidor web
  - c) Quin codi de resposta dóna la capçalera HTTP enviada pel servidor? Què significa?
  - d) En el cas d'obtenir una capçalera *Location* a la resposta del servidor, explica el seu significat.
  - e) Que indica la capçalera *Server* de la resposta donada pel servidor? Explica el resultat que has obtingut.
  - f) Que indica la capçalera *Content-Type* de la resposta donada pel servidor? Explica el resultat que has obtingut.
  - g) Que hi ha al cos del missatge de la resposta donada pel servidor?. El servidor t'ha tornat la pàgina web inicial? Què t'ha tornat?.
  - h) Comprova si la connexió es tanca immediatament després de rebre la resposta del servidor. Hi ha alguna manera de tancar la connexió?
2. Fes la següent petició HTTP utilitzant: **telnet www.google.com 80**

```
GET / HTTP/1.1
Host: www.google.es
User-Agent: telnet
Connection: close
```

Mostra el resultat obtingut amb Wireshark, i respon a les següents preguntes:

- a) Per què serveix la capçalera *Connection: close*? Què passa al executar la petició amb *Connection:close* a diferència de l'exercici anterior?
  - b) Fixa't que a la petició el *Host* ara és [www.google.es](http://www.google.es). Això ha fet que la resposta sigui diferent? Per què? Mostra la resposta i comenta el resultat obtingut i les diferències amb l'exercici anterior.
  - c) Què significat pel navegador que el *Expires* valgui -1 i que el camp *max-age* de *Cache-Control* sigui igual a 0?
3. Amb l'eina telnet connectat a la web **www.pangea.org**. Amb Wireshark comprova:
    - a) Quina resposta dóna el servidor. Indica el significat de la resposta.
    - b) Indica quina és la nova adreça del servidor i quin camp de la capçalera dóna la nova URL
    - c) Indica el programa servidor de pàgines web utilitzat per [www.pangea.org](http://www.pangea.org).
  4. Com ja sabeu els navegadors tenen memòria cau per agilitzar les peticions dels recursos web. Aquesta gestió de la memòria cau (*cache*) es fa gràcies a la informació de les capçaleres HTTP. Cada objecte de la memòria cau està identificat i també marcat amb la data d'obtenció. Connectat per *telnet* al servidor **www.mecd.gob.es** al port **80** i realitza la següent petició:

```
GET /portada-mecd/ HTTP/1.1
Host: www.mecd.gob.es
User-Agent: telnet
Connection: close
```

Hauries d'obtenir una resposta amb una capçalera HTTP semblant a aquesta:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-Magnolia-Registration: Registered
Pragma:
Cache-Control: max-age=600, public
Expires: Mon, 08 Oct 2018 11:09:02 GMT
Last-Modified: Mon, 08 Oct 2018 10:57:08 GMT
Content-Type:text/html;charset=UTF-8
Content-Length: 39637
Date: Mon, 08 Oct 2018 10:59:01 GMT
Connection: close
Set-Cookie: BIGipServerpool_web=1124081580.36895.0000; path=/
```

(Nota: Les dates i hores poden canviar en funció del moment de realització de la pràctica)

- Realitza un connexió i mostra la informació de la capçalera obtinguda. Quina informació dona la capçalera *Last-Modified* de la resposta? Veus alguna utilitat en saber aquesta informació?
- A partir de quina data la validesa del recurs expira? Què significa això?
- Aquest apartat s'ha de fer immediatament després de l'anterior petició per poder observar el resultat desitjat. Torna a fer la mateixa petició però utilitzant una capçalera addicional (*If-Modified-Since*), el valor ha de ser la data de la darrera modificació que es troba a la resposta anterior (**a partir dels valors de la teva execució i no utilitzant els valors de l'exemple**).

```
GET /portada-mecd/ HTTP/1.1
Host: www.mecd.gob.es
User-Agent: telnet
If-Modified-Since: Mon, 08 Oct 2018 10:57:08 GMT
Connection: close
```

Quin codi de resposta has obtingut? Explica el significat del codi i raona el motiu pel qual obtens aquest codi a partir d'explicar el funcionament de les capçaleres utilitzades.

- Torna a fer la mateixa petició però utilitzant el valor de la capçalera addicional *If-Modified-Since* igual a *Mon, 08 Oct 2018 8:57:08 GMT* (o sigui, dues hores abans). Comprova ara si el servidor torna la pàgina web sencera i també el codi de resposta obtingut.

5. Analitza la següent "conversa" HTTP entre un client i un servidor i descriu allò que està passant:

```
GET /private/index2.html HTTP/1.1
Host: localhost
```

```
HTTP/1.1 401 Authorization Required
Server: HTTPd/1.0
Date: Sat, 27 Nov 2006 10:18:15 GMT
WWW-Authenticate: Basic realm="Secure Area"
Content-Type: text/html
Content-Length: 311
```

```
GET /private/index2.html HTTP/1.1
Host: localhost
Authorization: Basic QWxhZGRpbjpvYVUHNlc2FtZQ==
```

```
HTTP/1.1 200 OK
Server: HTTPd/1.0
Date: Sat, 27 Nov 2006 10:19:07 GMT
Content-Type: text/html
Content-Length: 10476
```

## C) Peticions i respostes HTTP treballant amb Firefox i wireshark

### 1.- Treballant amb la capçalera *Accept-Language*

a) Treballa amb Firefox. Selecciona com idioma únic de treball l'Anglès, o sigui English[en]. Esborra l'historial complet del teu navegador. Tanca i torna obrir el teu navegador. Posa en marxa wireshark de manera que només capturi paquets HTTP. Connecta't a la web:

[http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=/com.ibm.itame.doc\\_6.1/am61\\_webseal\\_admin134.htm](http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=/com.ibm.itame.doc_6.1/am61_webseal_admin134.htm)

Comprova la capçalera de petició realitzada pel navegador. Indica el valor del camp Accept-Language enviat pel navegador dins de la capçalera.

b) Comprova l'idioma de la pàgina web mostrada.

c) Selecciona ara com idioma únic de treball l'Espanyol, o sigui Spanish[es]. Esborra l'historial complet del teu navegador. Tanca i torna obrir el teu navegador. Connecta't a la mateixa web de l'apartat a). Comprova ara la capçalera de petició realitzada pel navegador. Indica el valor del camp Accept-Language enviat pel navegador dins de la capçalera. Comprova també l'idioma de la pàgina web mostrada. Què ha passat?

### 2- Comprovant el funcionament de les peticions POST i GET

a) Treballa amb Firefox. Connectat a **daw2.co.nf**. Comprova el valor del camp User-Agent del primer missatge HTTP enviat des del client.

b) Accedeix a la web Mètode Post. Omple el formulari i tramet la consulta. Comprova:

- Amb wireshark que s'ha creat una petició de tipus POST. Demuestra-ho.
- Comprova que les dades s'envien dins del cos del missatge. Demuestra-ho.
- Que les dades no són visibles a la barra d'adreces del navegador.
- Demostra que el resultat no es pot afegir a les adreces d'interès
- Comprova des de wireshark que quan recarreguem la pàgina es torna a enviar la petició i que el navegador dona un missatge d'avís.

c) Accedeix a la web Mètode Get. Omple el formulari i tramet la consulta. Comprova:

- Amb wireshark que s'ha creat una petició de tipus GET. Demuestra-ho.
- Comprova que les dades s'envien dins de la capçalera del missatge. Indica dins de quin camp es troben aquestes dades. Demuestra-ho.
- Que les dades són visibles a la barra d'adreces del navegador.
- Demostra que el resultat es pot afegir a les adreces d'interès
- Comprova des de wireshark que quan recarreguem la pàgina no es tornen a enviar la petició i que el navegador no dona cap missatge d'avís.

d) Instal·la el complement "Firebug" dins del navegador Firefox. Activa i habilita la pestanya "Xarxa" i selecciona l'opció "All" de la pestanya. Torna a fer l'apartat **b)** i captura amb el navegador les capçaleres enviades i rebudes. Comprova que pots també accedir a les capçaleres i el cos dels missatges HTTP des d'aquest complement del navegador. Comprova que hi ha a a "Enviat" i "Paràmetres de la URL" (NOTA: potser no hi són tots dos). Ara torna a fer l'apartat **c)** i comprova la diferència amb les dades que hi ha "Enviat" i "Paràmetres de la URL" (NOTA: potser no hi són tots dos). Quins són els motius d'aquestes diferències?

### 3- Múltiples connexions

a) Treballa amb Firefox. Connectat a <http://www.binefa.net/electronica/tutorial04/>. Comprova quantes peticions GET s'han generat per part del client i i quantes respostes respostes ha enviat el servidor.

b) Indica el motiu pel qual, a part de la petició inicial del client i la resposta inicial del servidor, s'han generat 4 peticions més i quatre respostes més.

c) Comprova que cada petició ha implicat la utilització d'un nou port per part del client. Per què?

## D) Taula comparativa de mètodes

- a) Què significa que un mètode sigui segur?. Quins mètodes són segurs?
- b) Què significa que un mètode sigui "Cacheable"?. Quins mètodes són "Cacheables"?
- c) Fes una taula comparativa dels mètodes GET, PUT, POST, HEAD i DELETE indicant per cada mètode si té les següents propietats:
- La petició envia dades al cos del missatge?
  - La resposta a la petició té dades al cos del missatge?
  - És un mètode segur?
  - És un mètode idempotent?
  - És un mètode cacheable?

### **Examen validador de la pràctica**

1- El dia **29/10/18** es dura a terme un examen validador de la pràctica

2- L'examen tindrà una part pràctica i una part teòrica.

3- Per la part pràctica caldrà utilitzar els programes telnet i wireshark. La part pràctica tindrà un valor del 50% de la nota.

4- La part teòrica serà un test de 20 preguntes a fer en 15 minuts amb 4 opcions i només una resposta vàlida. La part teòrica tindrà un valor del 50% de la nota.