

## **Pràctica 1: Breu estudi del 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 compona 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>

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

1. Posa en marxa **Wireshark** i 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) La capçalera IP, l'adreça IP del teu ordinador i la del servidor a la capçalera IP del missatge enviat pel programa telnet.
  - 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) Que indica la capçalera *Server* de la resposta donada pel servidor? Explica el resultat que has obtingut.
  - e) Que indica la capçalera *Content-Type* de la resposta donada pel servidor? Explica el resultat que has obtingut.
  - f) 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. Posa en marxa **Wireshark** i amb l'eina **telnet** connectat al port **80/tcp** del servidor **xtec.gencat.cat** i realitza la següent petició:

```
HEAD /ca/curriculum/professionals/ HTTP/1.1
Host: xtec.gencat.cat
User-Agent: telnet
Connection: close
```

Amb wireshark comprova:

- a) Quina reposta dóna el servidor. Indica el significat del codi de la reposta i del camp *Location*.
- b) Indica quina és la nova adreça del servidor.
- c) Indica el programa servidor de pàgines web utilitzat per [xtec.gencat.cat](http://xtec.gencat.cat).
- d) Quina és la diferència entre haver utilitzat **HEAD** i no **GET**?
- e) A quina hora s'ha generat el missatge HTTP en el servidor?

4. 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

**NOTA:** Abans de començar aquesta part:

- Accedeix a Preferències → Privadesa i seguretat
- Neteja Galetes i contingut de memòria cau
- Neteja historial i configura Firefox per no recordar mai l'historial
- A la barra de localització (a on escrius les adreces URL) escriu about:config per accedir a la configuració.
- Busca l'entrada de configuració browser.urlbar.autoFill. Fes que el seu valor sigui False.
- Reinicia Firefox

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

a) Des de "about:config", selecciona el paràmetre de configuració intl.accept\_languages i indica que s'utilitzaran com idiomes de treball les següent llengües en aquest ordre: Català (ca), Spanish(es), English (en). Esborra l'historial complet del teu navegador. Tanca i torna obrir el teu navegador. Connecta't a <http://www.collados.org>. Comprova ara la capçalera de petició realitzada pel navegador. Indica el valor del camp **Accept-Language** enviat pel navegador dins de la capçalera.

b) Quin és l'efecte del camp Accept-Language

c) Quin és el propòsit del valor q?

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

a) Connectat a [http://www.collados.org/daw2/m08/uf1/php\\_html\\_post\\_get/form.html](http://www.collados.org/daw2/m08/uf1/php_html_post_get/form.html). 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. Demostrea-ho.
- Comprova que les dades s'envien dins del cos del missatge. Demostrea-ho.
- Que les dades no són visibles a la barra d'adreces del navegador.
- Demostrea 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. Demosta-ho.
  - Comprova que les dades s'envien dins de la capçalera del missatge. Indica dins de quin camp es troben aquestes dades. Demosta-ho.
  - Que les dades són visibles a la barra d'adreces del navegador.
  - Demosta que el resultat es pot afegir a les adreces d'interès

### 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 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 noves peticions i respostes.
- 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 **22/10/24** penjaré les respostes a totes les preguntes del treball.
- 2- El dia **24/10/24** es dura a terme un examen validador de la pràctica
- 3- L'examen tindrà una part pràctica i una part teòrica.
- 4- Per la part pràctica caldrà utilitzar els programes telnet i wireshark. La part pràctica tindrà un valor del 40% de la nota.
- 5- La part teòrica tindrà un valor del 60% de la nota de la pràctica i es dividirà en 3 parts:
- a) 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 60% de la nota.
  - b) 5 preguntes curtes (màxim 1 línia - Màxim 4 o 5 paraules)
  - c) 5 preguntes llargues (màxim 2 o 3 línies)