

# Pràctica 6: Servei web

Aplicacions i serveis d'internet — Enginyeria de Sistemes TIC

Francisco del Àguila López      Aleix Llusà Serra

24 d'abril de 2023

## Índex

<b>1</b>	<b>Organització</b>	<b>2</b>
1.1	Objectius . . . . .	2
1.2	Condicions . . . . .	2
1.3	Lliurables . . . . .	2
1.4	Material necessari . . . . .	2
<b>2</b>	<b>Introducció al servei web</b>	<b>2</b>
2.1	Escenari . . . . .	3
2.2	Eines web . . . . .	3
2.3	La xarxa local virtual i l'accés des d'un navegador web gràfic . . . . .	3
2.3.1	Forwarding de les X . . . . .	4
2.3.2	Socks . . . . .	4
<b>3</b>	<b>Servidor web simple amb Python</b>	<b>5</b>
<b>4</b>	<b>Servidor web Apache: servei del sistema</b>	<b>6</b>
4.1	Fitxers de configuració i registre . . . . .	6
4.2	Configuració per defecte: Servidor de fitxers . . . . .	7
4.3	Definició del <i>site</i> de la pràctica . . . . .	7
4.4	Registre . . . . .	8
4.5	Fitxers, URL i permisos . . . . .	8
4.5.1	Àlies . . . . .	9
4.5.2	Redirecció . . . . .	9
4.5.3	Reverse Proxy . . . . .	10
4.6	Servidor per noms virtuals . . . . .	11
<b>5</b>	<b>Extensions</b>	<b>12</b>
5.1	Seguretat: autenticació, autorització i xifratge . . . . .	12
5.2	Proxy i Caching . . . . .	12
5.3	WebDav . . . . .	13

## Resum

Protocols d'internet: HTTP, DNS

# 1 Organització

## 1.1 Objectius

El objectius d'aquesta pràctica són:

1. Entendre el servei web.
2. Usar Apache com a servidor web.
3. Usar el mòdul `http.server` de la llibreria de Python3 com a servidor web.
4. Servir fitxers de forma genèrica via web.
5. Servir un portal web dissenyat amb HTML.
6. Configurar serveis web amb noms virtuals.

## 1.2 Condicions

- La pràctica està calibrada per a ésser treballada en equips de dues persones.
- La durada de la pràctica és d'1 setmana.

## 1.3 Lliurables

Heu d'entregar els fitxers de configuració que tingueu a `/etc/apache2/sites-enabled/`.

## 1.4 Material necessari

Per tal de poder treballar còmodament amb els serveis d'internet i no barrejar-los amb els de la vostra màquina, utilitzarem les eines de màquines i xarxes virtuals descrites a la pràctica de *Virtualització*. Per dur a terme la pràctica cal crear una màquina virtual i instal·lar-hi un servidor web. En el cas de GNU/Debian i equiparables cal que instal·leu el paquet `apache2`.

**TASCA PRÈVIA 1** Creeu una màquina virtual nova segons les condicions descrites a la pràctica de virtualització i instal·leu-hi els paquets anteriors.

Per dur a terme la pràctica cal tenir instal·lades eines de navegació web. En el cas de GNU/Debian i equiparables cal que instal·leu els paquets `wget`, `curl`, `lynx` i `firefox` o `firefox-esr`.

**TASCA PRÈVIA 2** Instal·leu els paquets anterior en el vostre computador o en una màquina virtual de proves; cal que la màquina on ho instal·leu tingui accés a la xarxa virtual on hi ha el servidor web.

# 2 Introducció al servei web

El servei web s'encarrega d'atendre les peticions HTTP dels clients. De manera genèrica, un client sol·licita fer una acció a un recurs, el qual s'identifica amb una adreça URL, i el servidor li respon amb un missatge.

En el cas més simple, aquests missatges són fitxers. Aleshores el servei web simplement estableix un mapa d'adreces URL cap a un sistema de fitxers. Aquest mapa pot incloure casos més elaborats com per exemple àlies, redireccions, noms virtuals, execucions d'scripts, etc.

Els fitxers poden ser de qualsevol tipus. Habitualment tenen format HTML per tal que els navegadors els sàpiguen presentar o altres formats estàndards per intercanviar informació entre màquines com per exemple XML.

El servei web també pot fer funcions complementàries com per exemple proxy, seguretat, gestió de cookies, etc.

## 2.1 Escenari

En el vostre domini d'autoritat `gXX.asi.itic.cat` heu d'establir un servei web de fitxers i portals web. Conseqüentment, heu d'aconseguir els requisits següents:

1. En primer lloc, heu d'establir un servidor web que llisti un seguit de fitxers i l'usuari els pugui descarregar.
2. En segon lloc, heu de crear un portal web amb els documents HTML de la pràctica anterior.
3. En tercer lloc, heu de configurar la resolució de noms i les redireccions web necessàries per a servir el portal web amb un nom virtual.

## 2.2 Eines web

Us proposem tres eines web de naturalesa diferent:

- `wget` o `curl`: un intercanviador de dades web de terminal.
- `lynx`: un navegador textual
- `firefox` o `firefox-esr`: un navegador gràfic [Moz23].

TASCA 3 Consulteu el manual de cada ordre per saber com funciona i familiaritzeu-vos-hi consultant les adreces URL que vagin sortint a la pràctica.

## 2.3 La xarxa local virtual i l'accés des d'un navegador web gràfic

Amb el muntatge que es té per fer les pràctiques s'ha de tenir en compte que:

- Existeix una xarxa local virtual on estan connectades totes les màquines virtuals de tots els participants de l'assignatura.
- La màquina host on estan aquestes màquines virtuals també té una interfície connectada a aquesta xarxa virtual, però per poder accedir a Internet també té interfície connectada a alguna xarxa que permeti accés a Internet. En general, aquesta connexió està gestionada de manera automàtica per un gestor de connexions. Això implica que, per exemple, la configuració del DNS local (resolver), com a client, ve donada en general pel servidor DHCP associat a la interfície que dona accés a Internet. En general, aquest gestor és el `NetworkManager`.

- Si la configuració del DNS local apunta a un servidor d'Internet, no es podran resoldre les consultes del vostre domini d'autoritat (ja que està en una xarxa privada). Per això no funcionaria correctament el DNS des del host.
- Des de qualsevol màquina virtual sí que funciona correctament el servidor de DNS local ja que apunta a un servidor de dins de la xarxa virtual i per tant obrir un navegador web en una màquina virtual per veure qualsevol servidor de dins de la xarxa virtual no dóna cap problema.

De totes maneres les màquines virtuals, a priori, no disposen d'entorn gràfic i per tant no es pot executar un navegador web gràfic.

Les possibles solucions a aquest problema poden ser:

1. Configurar el servidor de DNS local de la màquina host per a que apunti al vostre servidor de DNS de dins de la xarxa local virtual. Per tant, s'hauria d'afegir un nou perfil de NetworkManager a la màquina host que utilitzés com a servidor de DNS local el que heu creat en pràctiques anteriors i mantingui la resta de paràmetres de configuració de xarxa com estaven.
2. Fer que la sortida gràfica d'una aplicació en una màquina sense entorn gràfic es redireccioni cap a una altra màquina amb entorn gràfic. Això es pot fer amb la funcionalitat de *Forwarding* de les X de `ssh`.
3. Establir una connexió de tipus proxy genèrica amb una màquina virtual i fer que una aplicació (en aquest cas el navegador web) que s'executa en una altra màquina es comporti com si s'executés en la màquina virtual. Això es pot fer igualment amb la funcionalitat de *proxy SOCKS* que també contempla el `ssh`. Aquesta opció, que també serveix per altres aplicacions de xarxa (no només per al navegador web) es deixa opcional a interès dels estudiants.

### 2.3.1 Forwarding de les X

Les X són el sistema gràfic d'una màquina. Aquest sistema gràfic en les màquines UNIX actua de manera que hi ha un client i un servidor. Aquest client i servidor no té perquè estar en la mateixa màquina. Per aquest motiu, es pot enviar l'ordre d'execució d'una aplicació en una màquina des de una altra màquina remota i fer que el resultat gràfic es mostri a la màquina remota. Això és fer el *Forwarding* de les X.

Un exemple seria:

```
ssh -X usuari@maq_remota comanda_remota
```

Per a més detall consulteu el manual de `ssh`.

### 2.3.2 Socks

La shell remota i segura `ssh` permet establir un proxy genèric en una màquina remota. Si aquest proxy es fa servir per la navegació web, això permet que a efectes pràctics el punt de sortida de les peticions web sigui aquest proxy i per tant aquesta màquina remota. Per tant, en el cas que ens interessa per aquesta pràctica, permet utilitzar el navegador web de la màquina host i fer una

navegació com si l'origen de les connexions web sigui la màquina remota (en aquest cas virtual) això farà que el DNS es resolgui des de la màquina virtual i per tant funcionarà bé pels dominis d'autoritat propis.

Per fer servir el proxy socks:

Establiu la connexió SOCKS:

```
ssh -D 8080 usuari@maq_remota
```

Connecteu-vos al vostre Proxy Socks.

- Amb firefox: Configureu la xarxa amb el vostre servidor intermediari SOCKS a les preferències del navegador. A més a més, per a la resolució de DNS activeu `network.proxy.socks_remote_dns`, consulteu <http://sockslist.net/articles/firefox-socks-dns-resolve>.

### 3 Servidor web simple amb Python

Recupereu els documents en format HTML de la pràctica anterior. Recordeu que els consultàveu localment des d'un navegador. Ara servirem aquests documents HTML via web.

En el directori on tingueu els fitxers HTML, executeu un servidor web amb el mòdul `http.server` de la llibreria de Python:

```
python3 -m http.server
```

Això llença el servei web com una simple aplicació d'usuari. Per tant tindrà tots els privilegis i permisos associats a l'usuari que ha executat aquesta ordre.

Observareu que per defecte serveix en qualsevol adreça IP (0.0.0.0) i en un port privat (8000):

```
Serving HTTP on 0.0.0.0 port 8000 ...
```

En un navegador, visiteu doncs l'adreça URL `http://localhost:8000/` a on trobareu la pàgina web principal. A la sortida del servidor Python veureu el registre d'accés als recursos web

```
127.0.0.1 - - [23/Jan/2013 11:10:41] "GET / HTTP/1.1" 200 -
```

en aquest cas la màquina d'origen (127.0.0.1), la data, el recurs que s'ha demanat (GET /) i el codi de retorn (200, amb èxit).

Com podeu deduir, hi ha un pacte que serveix el fitxer `index.html` quan es demana per un directori. Si en el directori no hi ha el fitxer `index.html` es llista el directori, comproveu-ho per exemple a `http://localhost:8000/directori/`.

Ara proveu de visitar el portal web amb altres adreces URL i observeu què apareix al registre:

- `http://127.0.0.1:8000/` el mateix que localhost però amb adreça IP
- `http://<iphost>:8000/` consulteu l'adreça IP de la vostra màquina amb `/sbin/ip` a.
- `http://<ipcompany>:8000/` demaneu l'adreça IP d'un company vostre.
- `http://<nommaquina>.gX.asi.itic.cat:8000/` si esteu executant el servidor web en la màquina virtual, proveu de visitar pel nom que heu donat a la màquina en el DNS.

## 4 Servidor web Apache: servei del sistema

Com en altres serveis, el servei web també es pot desplegar encaixant en el sistema operatiu. Aleshores la seva configuració és responsabilitat i autoritat de l'administració de sistemes.

En aquesta pràctica veureu la configuració en el sistema del servidor web Apache [Apa12], actualment en la versió 2.

### 4.1 Fitxers de configuració i registre

Els fitxers i directoris de configuració implicats en el servidor web són els següents (a Debian i similars):

**/etc/apache2/** Directori de configuració d'Apache

**/etc/apache2/apache2.conf** Configuració principal del servidor, inclou els altres fitxers i directoris de configuració.

**/etc/apache2/ports.conf** Configuració general de a quines connexions d'entrada escolta el dimoni.

**/etc/apache2/{sites-available;/sites-enabled/}** Directoris de les configuracions d'accés disponibles i habilitades, respectivament. Cada configuració es defineix en un fitxer \*.conf a **available** i s'activa o desactiva amb les ordres **a2ensite** i **a2dissite**. És a dir, les que apareixen a **enabled** són les que actualment estan en funcionament.

**/etc/apache2/sites-available/default.conf** Fitxer inicial d'exemple que configura un servidor de fitxers de **/var/www/html** entre d'altres.

**/etc/apache2/{mods-available;/mods-enabled/}** Directoris semblants als de **sites** però per als mòduls; és a dir les extensions que amplien el funcionament d'Apache. En aquest cas l'activació es controla amb **a2enmod** i **a2dismod**.

**/etc/apache2/{conf-available;/conf-enabled/}**

**/etc/apache2/conf.d/** Directori de les configuracions d'accés per a altres paquets.

**/var/log/apache2/** Directori del registre d'Apache

**/var/log/apache2/access.log** Registre d'accés als recursos

**/var/log/apache2/error.log** Registre d'errors en el servei

Quan s'engega el servidor web, ha d'escoltar a una adreça i un port de la màquina. Aquesta adreça i port es configuren a **/etc/apache2/ports.conf**:

```
Listen 80
```

per defecte, s'escolta el port 80 de totes les adreces. Comproveu amb **ss -tul** que sigui així; a **/etc/services** podeu trobar una llista de ports públics i alguns usos habituals en els privats com és el cas dels ports registrats.

## 4.2 Configuració per defecte: Servidor de fitxers

Per defecte un cop instal·lat l'Apache us trobareu amb un servidor genèric de fitxers. Això de per sí és una eina de molt gran utilitat! Se serveixen els fitxers a partir de la configuració de `/etc/apache2/sites-available/000-default.conf`. Fixeu-vos que ens indica que se serviran tots els fitxers que hi hagi a `/var/www/html`:

```
DocumentRoot /var/www/html
```

Observeu que els fitxers que es troben originalment en aquest path poden pertànyer a l'usuari *root* o bé a l'usuari especial *www-data*. Aquest és l'usuari de sistema reservat al servei web Apache. Té just els permisos necessaris per funcionar correctament. D'aquesta manera, no cal que sigui *root* el responsable d'executar el servei. Per tant, tot el contingut col·locat aquí hauria de ser accessible (o pertànyer) a aquest usuari.

TASCA 4 Poseu fitxers amb diferents formats (application/PDF, text/plain, text/x-rst, text/x-python, audio/, image/, etc.) a `/var/www/html`. Creeu un directori amb el contingut HTML de l'anterior pràctica (format text/html i d'altres). Descarregueu i visiteu els fitxers amb el nom de la màquina com a adreça base:

- `http://<nommaquina>.gX.asi.itic.cat/fitxer.pdf`
- `http://<nommaquina>.gX.asi.itic.cat/directori/`
- `http://<nommaquina>.gX.asi.itic.cat/directori/doc1.html`
- etc.

Cal fer notar que l'accés als objectes referenciats per les URLs anteriors també són accessibles quan canviem `http://<nommaquina>.gX.asi.itic.cat` per `http://<iphost>`. De la mateixa manera, si el client web (navegador) es troba a la mateixa màquina on està el servidor web, es pot accedir al contingut web amb `http://localhost` o bé `http://127.0.0.1`.

Màgicament (`/etc/apache2/magic`) cada document se serveix amb el tipus MIME correcte. Per aquesta raó, el navegador sap interpretar correctament com ha de gestionar cada fitxer.

A la documentació d'Apache de `http://httpd.apache.org/docs/current/urlmapping.html` trobareu tota la informació de com es determina la localització del fitxer a dins del sistema de fitxers del sistema operatiu on està instal·lat Apache a partir de la URL sol·licitada. Cal remarcar que el punt de partida dels fitxers que s'estan servint ve determinat per la directiva d'Apache *DocumentRoot*.

## 4.3 Definició del *site* de la pràctica

Apache està preparat per servir diferents *llocs web* a la mateixa màquina. Això evita que calgui un Apache per a cada *lloc web*. Cada *site* o *lloc web* queda definit a `/etc/apache2/sites-available/` dins un fitxer de configuració amb extensió `.conf`. Cada *site* es pot habilitar o deshabilitar sense que afecti a la resta de *sites* amb les comandes `a2ensite` o `a2dissite`.

TASCA 5 Per no barrejar la definició del *site* de la pràctica amb el *site* que hi ha per defecte `000-default.conf` creeu-vos un nou fitxer (p.ex. `/etc/apache2/sites-available/practweb.conf`) que sigui còpia de `000-default.conf`. D'aquesta manera tindrem la mateixa configuració de partida.

Modifiqueu les directives que ja apareixen amb la configuració adequada. A part de *DocumentRoot*, definiu l'adreça de contacte del gestor del servei (apareix en pàgines d'error):

```
<VirtualHost *:80>
    DocumentRoot <path_que_vulgueu>
    ServerAdmin <usuari_webmaster>@gX.asi.itic.cat
</VirtualHost>
```

Per a que la vostra configuració sigui la que s'aplica l'heu d'activar amb `a2ensite practweb` i desactiveu la de per defecte amb `a2dissite 000-default`. Cada cop que feu canvis, actualitzeu la configuració del servidor amb `systemctl reload apache2`.

Per a entendre bé la sintaxi de configuració d'Apache visiteu-ne la documentació [Apa12], en particular els resums de <http://httpd.apache.org/docs/current/sections.html> i <http://httpd.apache.org/docs/current/configuring.html>.

#### 4.4 Registre

El registre dels servei web informa de l'activitat que duu a terme el servidor i dels error que hi hagin. Teniu documentació sobre el registre d'Apache a <http://httpd.apache.org/docs/current/logs.html>. El format i localització de cara registre és configurable; normalment es concentren al directori `/var/log/apache2/` i s'usa format *common* o *combined*, els quals són semblants al que ja s'ha vist pel servidor web amb Python.

Per exemple:

```
ErrorLog ${APACHE_LOG_DIR}/error.log

LogLevel warn

CustomLog ${APACHE_LOG_DIR}/access.log combined
```

TASCA 6 Consulteu els fitxers de registre i relacioneu-los amb les peticions que heu fet anteriorment. Recordeu que els registres són de molta ajuda quan no funciona alguna configuració en el servidor web.

#### 4.5 Fitxers, URL i permisos

El servidor web estableix un mapa entre les URL i el sistema de fitxers a on hi ha el servidor. Hi ha directives que treballen sobre les URL, p.ex. la directiva *Location* (<http://httpd.apache.org/docs/current/mod/core.html#location>), i hi ha directives que treballen sobre el sistema de fitxers, p.ex. *Files* i *Directory* (<http://httpd.apache.org/docs/current/mod/core.html#directory>).

L'objectiu del servidor web és configurar el mapa entre les URL i el sistema de fitxers i establir-hi mètodes i permisos d'accés, els quals poden ser diferents per a les URL o per al sistema de fitxers. Seguiu el resum de <http://httpd.apache.org/docs/current/sections.html> per a més detall.

Per exemple, el més senzill és establir un directori base del sistema de fitxers:

```
DocumentRoot /var/www/html
```



Però podeu establir permisos i mètodes d'accés en aquest directori. Per exemple, en el cas del directori `/var/www` el trobareu ja configurat globalment al fitxer `/etc/apache2/apache2.conf` amb permís d'accés per a tothom i amb l'opció *Indexes*, que mostra el contingut del directori quan no hi ha declarat un fitxer `index.html`:

```
<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
```

Apache s'executa en el sistema com a usuari i grup *www-data*. Per tant, si configureu accessos a altres directoris del sistema heu d'assegurar-vos que l'usuari o el grup *www-data* hi té permisos de lectura.

#### 4.5.1 Àlies

Els àlies d'Apache serveixen per mapar un path de la URL sobre un path del sistema de fitxers del servidor. Els àlies s'han de configurar en els fitxers de configuració dels *sites*, en el vostre cas a `practweb.conf`. Per poder fer servir la directiva **alias** d'Apache cal que l'extensió `mod_alias` d'Apache estigui habilitada. Consulteu-ne la documentació a [http://httpd.apache.org/docs/current/mod/mod\\_alias.html](http://httpd.apache.org/docs/current/mod/mod_alias.html).

**TASCA PRÈVIA 7** Comproveu a `mods-enabled/` que l'Apache tingui habilitada l'extensió en qüestió. Si no la trobeu, utilitzeu la comanda `a2enmod` adequadament.

La sintaxis és la següent:

```
Alias /path-URL/ /path-sistema-de-fitxers/
```

Vigileu com ha de ser la sintaxis respecte als `'/'`.

**TASCA 8** Creeu un àlies que amb la URL `http://<nommaquina>.g<X>.asi.itic.cat/docserver/` es serveixi el contingut de `/usr/share/doc/` de la màquina que fa de servidor web. Observeu que a `/usr/share/doc/` es troba la documentació de la distribució de Linux de qual-sevol màquina.

#### 4.5.2 Redirecció

Les redireccions també necessiten tenir habilitada l'extensió `mod_alias` d'Apache. En el cas de les redireccions el que passa és que el servidor informa al client que el contingut que sol·licita es troba en una altra URL. Quan passa això el codi de status del missatge de resposta és del tipus `3XX` i el client automàticament estableix una nova petició HTTP cap a la nova URL on hi ha el contingut.

La sintaxis:

```
Redirect [status] /path-URL NOVA-URL
```

Vigileu com ha de ser la sintaxis respecte als `'/'`. El paràmetre `[status]` pot ser de diferent tipus (permanent, temp, ...). Consulteu la documentació de referència.

TASCA 9 Definiu una redirecció al vostre fitxer de configuració de *site* tal que amb la URL `http://<nommaquina>.g<X>.asi.itic.cat/epsem/` mostri el contingut de la web principal de l'Escola.

TASCA 10 Tal com s'ha explicat a la secció 3 serviu en el servidor Python de la màquina host un contingut HTML arbitrari i redirigiu la connexió web del vostre *site* d'Apache de manera que s'hi pugui accedir amb la URL base `http://<nommaquina>.g<X>.asi.itic.cat/docspy/`. Reflexioneu i justifiqueu quin serà el nom de màquina de la NOVA-URL de la redirecció. Comproveu que es fa la redirecció per a qualsevol fitxer amb aquesta URL base.

### 4.5.3 Reverse Proxy

En el cas dels *proxy invers* cal tenir habilitada l'extensió `mod_proxy` d'Apache. Consulteu-ne la documentació a `http://httpd.apache.org/docs/current/mod/mod_proxy.html`. Comproveu que l'Apache tingui habilitada l'extensió en qüestió. Noteu que també us farà falta algun submòdul com per exemple `mod_proxy_http`.

*TASCA PRÈVIA 11 Habilitau els mòduls que calguin per implementar *proxy invers* al vostre Apache. Recordeu les comendes `a2enmod` i `a2dismod`.*

En el cas del *proxy invers*, a diferència de la redirecció, és el mateix servidor Apache el que s'encarrega d'anar a buscar el contingut a la NOVA-URL (fent de client amb una nova petició HTTP) per lliurar-li el contingut al client com a resposta a la única petició HTTP del client.

La sintaxis:

```
ProxyPass /path-URL NOVA-URL
ProxyPassReverse /path-URL NOVA-URL
```

o bé:

```
<Location /path-URL>
  ProxyPass NOVA-URL
  ProxyPassReverse NOVA-URL
</Location>
```

El fet de fer servir més d'una directiva `ProxyPass` i `ProxyPassReverse` és perquè cadascuna realitza tasques diferents que es complementen. En particular `ProxyPassReverse` ajusta les URL que conté la capçalera HTTP de resposta. Si a més s'han d'ajustar les URL del contingut HTML de resposta s'ha d'utilitzar l'extensió `mod_proxy_html`.

TASCA 12 Definiu un proxy invers al vostre fitxer de configuració de *site* tal que amb la URL `http://<nommaquina>.g<X>.asi.itic.cat/ocwitic/` mostri el contingut de `http://ocwitic.epsem.upc.edu/assignatures/asi/`.

TASCA 13 Serviu, ara, un contingut web arbitrari en el servidor Python de la vostra màquina host. Configureu en el vostre *site* un proxy invers de manera que accedint a `http://<nommaquina>.gX.asi.itic.cat/docproxy/` quedi servit per Apache el contingut del servidor Python. De nou reflexioneu sobre quin ha de ser el nom de màquina de la NOVA-URL del proxy.

TASCA 14 Avalueu les diferències entre una redirecció i un proxy invers observant el comportament amb les eines de *Desenvolupador Web* de Firefox.

## 4.6 Servidor per noms virtuals

Un mateix servidor web pot servir peticions per a diferents noms virtuals (*Name-based Virtual Host*); és a dir per a noms que no siguin el FQDN de la màquina. Cal configurar en el DNS aquests noms virtuals per tal que apuntin al servidor web i aquest extreu de la capçalera HTTP el nom virtual original pel qual el client pregunta.

Teniu un agrupament de documentació sobre Virtual Hosts d'Apache a <http://httpd.apache.org/docs/current/vhosts/>.

Per a definir els noms virtuals, cal configurar el DNS per tal que el nom del domini resolgui a l'adreça IP del servidor web, ja sigui amb registres *A* o *CNAME* del DNS. Per defecte, Apache respon amb el primer bloc `<VirtualHost>` que troba a la configuració. Així doncs, un cop configurat el DNS, el servei web ja respon per a diferents noms virtuals.

Per exemple, si únicament teniu un fitxer de configuració:

```
<VirtualHost *:80>
    DocumentRoot /var/www/html
</VirtualHost>
```

aleshores aquest respon a totes les peticions sigui quin sigui el domini original sol·licitat.

Ara bé, el servei web també permet atendre de manera diferent segons el nom virtual que se li preguntí, el qual s'ha emmagatzemat en les capçaleres HTTP. Per a aquest ús dels noms virtuals, en el servidor web cal configurar la directiva *ServerName* dels blocs `<VirtualHost>` amb el nom de domini.

Per exemple:

```
<VirtualHost *:80>
    ServerName example.com
    DocumentRoot /var/www/html
</VirtualHost>
<VirtualHost *:80>
    ServerName www.example.com
    DocumentRoot /var/www/doc
</VirtualHost>
```

Recordeu que per a noms virtuals que no apareguin a les directives, Apache respon amb el primer bloc `<VirtualHost>` que troba a la configuració.

La directiva *ServerName* només permet declarar un nom de domini per cada bloc `<VirtualHost>`. La directiva *ServerAlias* permet ampliar els noms virtuals que s'atenen en un bloc `<VirtualHost>`.

Per exemple:

```
<VirtualHost *:80>
    ServerName web.example.com
```

```

    DocumentRoot /var/www/html
</VirtualHost>
<VirtualHost *:80>
    ServerName www.example.com
    ServerAlias example.com doc.example.com
    DocumentRoot /var/www/doc
</VirtualHost>

```

El sentit de les directives *ServerName* i *ServerAlias* depèn de la directiva *UseCanonicalName*, la qual per defecte està *Off*:

- Si *UseCanonicalName* està a *Off* aleshores *ServerName* i *ServerAlias* configuren els noms virtuals al mateix nivell.
- Si *UseCanonicalName* està a *On* aleshores *ServerName* declara el nom canònic i *ServerAlias* declara els noms com a àlies. En algunes sol·licituds dels àlies, en particular quan el servidor ha de construir la base de la URL, es respon amb una redirecció cap al nom canònic. Consulteu <http://httpd.apache.org/docs/current/mod/core.html#usecanonicalname> per a més detall.

TASCA 15 Configureu el servei web i el DNS per tal que s'atenguin les URL següents del vostre domini:

- <http://g<X>.asi.itic.cat> Amb la vostra pàgina personal
- <http://www.g<X>.asi.itic.cat> Amb la vostra pàgina personal
- <http://web2.g<X>.asi.itic.cat> Amb un altre lloc web on tingueu altres documents.
- <http://ocw.g<X>.asi.itic.cat> Amb el contingut de <http://ocwitic.epsem.upc.edu/>

## 5 Extensions

El servei web té moltes més funcionalitats que les explorades en aquesta pràctica. Si voleu ampliar els vostres coneixements, la documentació d'**Apache** [Apa12] és una bona referència. A continuació us proposem algunes extensions de la pràctica.

### 5.1 Seguretat: autenticació, autorització i xifratge

L'autenticació és el procés que identifica els usuaris. L'autorització és el procés que controla el què poden fer els usuaris. El xifratge és el procés que transmet informació de forma segura.

En el servei web pot ser necessari establir mètodes de seguretat i controlar l'accés dels usuaris als recursos. En el cas d'**Apache** teniu una documentació de partida a <http://httpd.apache.org/docs/current/howto/auth.html>. Per a aquesta extensió és necessari entendre els temes de Seguretat a Internet.

### 5.2 Proxy i Caching

Un servidor Proxy consisteix a servir contingut d'un altre servidor. El Caching consisteix a emmagatzemar les respostes que s'han servit per tal que si es tornen a demanar se serveixin

d'una còpia local en comptes de consultar a l'origen. Les dues tècniques se solen fer servir alhora per construir 'Caching Proxy Servers'.

Per a establir Proxy i Caching amb Apache consulteu les extensions `mod_proxy` i `mod_cache`. També podeu provar d'establir-ho amb productes específics com per exemple el paquet `Squid` de la distribució.

### 5.3 WebDav

<http://en.wikipedia.org/wiki/WebDAV>

### Referències

- [Apa12] Apache Software Foundation. *The Apache HTTP Server Project*. 2012. URL: <http://httpd.apache.org/> (consultat 23 de gen. de 2013).
- [Moz23] Mozilla Foundation. *Firefox Project*. 2023. URL: <https://support.mozilla.org/es/kb/elegir-un-canal-de-actualizacion-de-firefox> (consultat 18 d'abr. de 2023).