

## 2º ASIR. IAW

### Tema 2: Instalación de Servidores

#### **Práctica 2.1** **Instalación y Configuración de Apache**



## Introducción

El punto de partida será una máquina virtual con Ubuntu server 14.04. En el aula utilizaremos KVM como plataforma de virtualización, aunque la práctica podría realizarse perfectamente también con VirtualBox.

El objetivo principal de esta práctica es familiarizarse con la configuración de apache, pero además, conseguiremos que nuestra máquina virtual quede configurada para usarla más adelante como entorno de pruebas para la programación en PHP.

## Configuración de la red.

Utilizaremos dos tarjetas de red.

La primera la configuraremos en modo “Red enrutada” sobre la red de trabajo del aula.

La segunda tarjeta estará conectada a la DMZ (red host only)

El método de trabajo será el que emplearíamos contra un servidor remoto: nos conectaremos mediante ssh desde nuestro puesto de trabajo (la máquina real) contra la IP del servidor y trabajaremos “dentro de un screen”.

El servidor debe estar correctamente actualizado, y, para el trabajo en el aula, es muy recomendable configurar el sources.list para usar el mirror del instituto como se describió en la practica '0'

## 1 Instalación de Apache

Instalaremos apache desde el repositorio oficial con

```
apt-get install apache2
```

Podemos comprobar que el servicio se ha instalado correctamente abriendo un navegador contra la IP del server.

Ejercicio 1. Una vez instalado apache2, contesta a las siguientes preguntas.

Ejercicio 1.1 ¿Cómo qué usuario está ejecutándose el servidor apache? ¿Por qué crees que no se ejecuta como root? ¿Que implicaciones puede tener sobre los contenidos y ficheros que pueden servirse mediante apache?

## 2 DocumentRoot y DirectoryIndex

Ejercicio 2.1 Indica cuál es el “DocumentRoot” por defecto de apache. Crea un subdirectorio llamado 'apuntes' dentro del mismo y copia algunos ficheros de prueba en él, por ejemplo “fichero1.txt” y “fichero2.pdf”.

Indica la URL que debes usar para visualizar su contenido y navegar por los archivos. Comprueba que puedes hacerlo y acompaña una captura.

Ejercicio 2.2 Crea un fichero index.html dentro de apuntes. Puedes ser algo más creativo o simplemente usar este sencillísimo ejemplo:

```
<html>
  <head>
    <title>Directorio de Apuntes</title>
  </head>
  <body>
    Este directorio contiene apuntes
  </body>
</html>
```

¿Qué ocurre ahora al acceder con el navegador al directorio? ¿Por qué?  
¿Podemos seguir accediendo a los ficheros? ¿Con que URL accederíamos a por ejemplo “fichero1.txt”?

### 3 Activar/Desactivar Módulos. Ejemplo 'userdir'

Cómo primer ejemplo de módulo sencillo usaremos el módulo 'userdir'. Este módulo permite que cualquier usuario del sistema cree su propio espacio web en un subdirectorio de su “home”. Cada uno de estos espacios estarán accesibles como subdirectorios de la forma “~username”

La activación desde línea de comandos se hace mediante la instrucción:

```
sudo a2enmod userdir
```

Ejercicio 3. Activa el módulo userdir, entra al directorio de configuración de apache y contesta a las siguientes preguntas:

Ejercicio 3.1 ¿Cuáles son los ficheros o directorios de la configuración de apache que debemos comprobar para verificar que el módulo userdir está activado? Explica como lo has comprobado e incluye alguna prueba.

¿Cuál es el fichero de configuración del módulo userdir? Indica el nombre y adjunta el contenido del mismo. ¿Cuál es el nombre por defecto del directorio que cada usuario debe crear en su 'home' para colocar los ficheros html y que sean servidos por 'userdir'?

Ejercicio 3.2 Crea el directorio requerido en el home de nuestro usuario en el servidor (ausias). Crea un fichero index.html en dicho directorio (puedes simplemente usar algo parecido al del ejercicio anterior cambiando el texto para que indique que es el directorio personal del usuario ausias)

Accede desde el navegador a [http://IP\\_SERVIDOR/~ausias](http://IP_SERVIDOR/~ausias) y comprueba que funciona. Adjunta una captura.

Modifica ligeramente el html. Para que se muestren los cambios ¿es necesario reiniciar apache o basta con refrescar la página en el navegador? ¿Por qué?

### 4 Alias, Directorios y Redirecciones

Ejercicio 4.1 Comprueba si disponemos del módulo 'alias' y si está activo. Justifica la respuesta indicando los ficheros o directorios en los que has buscado. Activa el módulo si fuese necesario.

Crea un fichero de nombre 'alias-ausias' en el subdirectorio de configuración conf-available con el siguiente contenido:

```
Alias /personal /home/ausias/public_html/
```

Activa la configuración con:

```
sudo a2enconf alias-ausias
```

Fuerza a que apache vuelva a leer su configuración y prueba a acceder a

```
http://IP_SERVIDOR/personal
```

Indica el resultado obtenido

Crea el directorio `/var/www/html/practica/web` y **mueve** el directorio “apuntes” del ejercicio 2 para que quede en `/var/www/html/practica/web/apuntes`.

Ejercicio 4.2 Abre el navegador y comprueba que ahora aparece un error al acceder a [http://IP\\_SERVIDOR/apuntes](http://IP_SERVIDOR/apuntes). Crea un fichero de configuración en `conf-available` llamado 'apuntes' y actívalo. En este fichero debes utilizar la directiva 'alias' para conseguir que vuelva a mostrarse el contenido del directorio `apuntes` con [http://IP\\_SERVIDOR/apuntes](http://IP_SERVIDOR/apuntes)

¿Podrías conseguirse que este mismo directorio está accesible con más de una URL, por ejemplo [http://IP\\_SERVIDOR/apuntes](http://IP_SERVIDOR/apuntes) y [http://IP\\_SERVIDOR/clase](http://IP_SERVIDOR/clase). Adjunta las directivas utilizadas finalmente

Ejercicio 4.3 Investiga si hay otros alias ya predefinidos en alguno de los ficheros de configuración de `apache2`. Prueba las urls e indica los alias que has encontrado.

Ejercicio 4.4 Utiliza la directiva 'Redirect' para que la URL “[http://IP\\_SERVIDOR/buscar](http://IP_SERVIDOR/buscar)” sea redirigida automáticamente a <http://www.google.es>. Comprueba que funciona la redirección con el navegador. Muevete a un directorio temporal y utiliza `wget` con la opción `-S` para acceder a “[http://IP\\_SERVIDOR/buscar](http://IP_SERVIDOR/buscar)” y visualizar las cabeceras:

```
wget -S http://IP_SERVIDOR/buscar
```

Adjunta la salida del comando `wget`.

*OJO: Es MUY IMPORTANTE ejecutar este comando desde un directorio temporal, ya que wget descarga la página y crea ficheros en el directorio de trabajo actual.*

## 5 Restricciones de acceso

Añade otro fichero de configuración en `conf-available` llamado 'logfiles'. Utiliza de nuevo la directiva `alias` para poder acceder al directorio de logs del sistema (`/var/log`) mediante la siguiente url: [http://IP\\_SERVIDOR/log](http://IP_SERVIDOR/log)

Ejercicio 5.1 Intenta acceder a `/log` con el navegador. ¿Por qué no funciona? Fijate bien en el mensaje que se muestra y razona lo que está ocurriendo.

Ejercicio 5.2 Incluye una directiva “Directory” en el fichero “logfiles” para el directorio “log”. ¿Qué directiva “Options ...” tendrás que incluir también para poder acceder a los distintos ficheros y subdirectorios que existen en `/var/log`? Incluye el contenido completo del fichero 'logfiles' que has utilizado.

Ejercicio 5.3 Accede mediante la consola al directorio /var/log y compara con el resultado de acceder mediante apache. ¿Por qué no aparecen todos los ficheros y directorios?

Ejercicio 5.4 ¿Qué directivas puedes añadir para restringir el acceso de manera que el directorio “/log” pueda accederse con el navegador únicamente desde la máquina real y no desde otros equipos? Incluye el contenido del fichero de configuración 'logfiles' tal y como queda finalmente.

## 6 Directiva Location

Comprueba si disponemos del módulo 'info' y si está activo. Activa el módulo si fuese necesario.

Podemos usar este módulo utilizando la directiva 'Location'. Esta directiva permite restringir el alcance de las directivas que incluyamos a una URL concreta, de manera similar a “Directory” o “Files” pero se utiliza para contenidos completamente ajenos al sistema de archivos que estén gestionados por un módulo u otros programas.

Crea un fichero de nombre 'server-info' en el subdirectorio de configuración conf-available con el siguiente contenido y actívalo:

```
<Location /server-info>
  SetHandler server-info
</Location>
```

Ejercicio 6.1 Reinicia el servidor apache y prueba a acceder a [http://IP\\_SERVIDOR/server-info](http://IP_SERVIDOR/server-info). ¿Qué es lo que ocurre? Fíjate bien en el mensaje que obtienes.

El módulo info entraña ciertos riesgos de seguridad, por lo que su configuración por defecto es muy restrictiva. Comprueba el fichero de configuración del módulo y deduce que directiva debemos añadir a nuestro fichero server-info para conseguir que funcione. Fuerza a que apache vuelva a leer su configuración y prueba de nuevo a acceder a [http://IP\\_SERVIDOR/server-info](http://IP_SERVIDOR/server-info). Indica el resultado obtenido.

## 7 Hosts virtuales

### 7.1 El problema de la resolución de nombres

Los hosts virtuales permiten que un único servidor apache pueda dar servicio a varios sitios web diferentes. Cada uno de estos hosts virtuales estará asociado a un nombre diferente.

El funcionamiento de los hosts virtuales requiere la configuración tanto de Apache como de la resolución de nombres a fin de permitir diferentes nombres de host e incluso de dominios para un único servidor.

Se desprende que la creación y configuración de hosts virtuales, requiere la configuración tanto de apache como del servicio DNS, sin el cual no es posible disponer de hosts virtuales accesibles públicamente en Internet.

No obstante, si nos limitamos a realizar pruebas con dominios locales en nuestro propio PC o máquinas virtuales, podemos empezar a trabajar con hosts virtuales en nuestro equipo sin necesidad de estudiar previamente el servicio DNS.

Aunque más adelante en el curso, instalaremos un servidor DNS y utilizaremos una herramienta de administración para configurarlo, de momento manipularemos la resolución de nombres en nuestro ordenador, evitando la complejidad del DNS. Estos servidores virtuales no estarán accesibles en Internet ni desde otros equipos, ya que los nombres de los dominios virtuales sólo

podrán resolverse desde los ordenadores “manipulados” pero así podemos empezar ya a realizar todos tipo de pruebas con nuestro ordenador y nuestras MV.

## 7.2 Fichero 'hosts'

La configuración habitual tanto en sistemas linux como Windows para el sistema de resolución de nombres, consiste en examinar primero un fichero del sistema (/etc/hosts en linux o c:\windows\system32\drivers\etc\hosts en Windows). Este fichero mantiene una sencillísima tabla de correspondencias entre nombres y IPs. Si se encuentra en el fichero el nombre correspondiente, se obtiene la IP desde el fichero y no se realiza ninguna consulta a los servidores DNS. Este mecanismo permite acelerar el acceso a hosts y dominios habituales, y también permite sustituir al servicio DNS para entornos de prueba o pequeñas redes.

Cada línea consta de la dirección IP y separado por uno o mas espacios o tabuladores, el nombre o nombres que deben asociarse, por ejemplo:

```
127.0.0.1      localhost
127.0.1.1      aurex-laptop
192.168.100.10 pruebas      servidor.local
...
```

La manipulación del fichero 'hosts' requiere privilegios de administrador y no podemos hacerlo en nuestro ordenador de clase, por lo que **para probar los siguientes ejercicios, usaremos un navegador desde una máquina virtual con Lubuntu Desktop.**

Editaremos el fichero /etc/hosts del Lubuntu Desktop y asignaremos el nombre 'prueba' a la IP de nuestra MV Ubuntu server, digamos:

```
IP_SERVIDOR      prueba
```

El efecto debería ser inmediato, por lo que si abrimos la url <http://prueba> desde el navegador accederemos a nuestro servidor apache.

De momento no instalaremos el servidor DNS en la MV y seguiremos estos mismos pasos para definir los servidores virtuales que vamos a ir creando.

## 7.3 Creación hosts virtuales

Ejercicio 7.1 Crea y activa los hosts virtuales 'prueba' y 'servidor'. Crea ficheros de configuración separados para cada uno. Utiliza como DocumentRoot '/srv/web/prueba' y '/srv/web/servidor' respectivamente. Crea una página sencilla 'index.html' diferente en cada caso.

Añade las líneas correspondientes en el fichero /etc/hosts y comprueba que puedes acceder a ambos servidores con el navegador desde el Ubuntu desktop.

Adjunta como resultado del ejercicio el contenido de los respectivos ficheros de configuración y alguna captura que demuestre que puedes acceder.

Ejercicio 7.2 Crea un nuevo host virtual 'ausias' y personaliza la configuración de manera que al acceder se produzca directamente la redirección hacia la página web del instituto: <http://www.ausiasmarch.net>  
¿Qué sentencia tendrás que escribir exactamente? Como resultado del ejercicio, debes indicar que sentencias de configuración has usado. Prueba con el navegador la redirección. Usa wget -S desde un terminal para que se muestren las cabeceras que provocan la redirección. Adjunta también la salida del comando como comprobación de la práctica.