RESUMEN TEMA 2. Configuración y administración de servidores Web.

Albert Einstein: "Se debe hacer todo tan sencillo como sea posible, pero no más sencillo."

## 1.- Funcionamiento de un servidor Web

## 1.1.- Servicio de ficheros estáticos

Debemos diferenciar cuando accedemos a una página web entre código ejecutable en el lado del servidor y en el lado del cliente.

Si al acceder a una página web no es necesaria la intervención de código en el lado del servidor, entonces entenderemos que la página es estática, si por el contrario la intervención en el lado servidor y/o en el lado del cliente entenderemos que la página es dinámica.

Ofrecer páginas estáticas es simple, puesto que solamente se necesita que el servidor web disponga de soporte html/xhtml/css o solo html/xhtml.

La configuración y administración del servidor es simple: soporte mínimo base de instalación del servidor Apache.

El rendimiento del servidor, es bajo, como no se ejecuta código en el lado servidor, el coste de CPU y memoria es bajo y se traduce en mayor rapidez.

Para ofrecer páginas estáticas mediante el servidor Apache simplemente copias la página en la ruta correspondiente donde quieres.

Estructura de ficheros de Apache:



En la instalación de Apache se crea una página web en **/var/www/html/index.html** referenciada a través del archivo **/etc/apache2/sites-avaliable/000-default.conf** (contiene la configuración por defecto.

Ejercicio:

Copiar el archivo .zip al servidor ubuntuserver en el directorio raíz del usuario Ubuntu.

Scp ruta.zip ubuntu@dirip-server:/home/usuario

El fichero nuevo, descomprimimos y copiamos el contenido del directorio web-estatica al directorio **/var/www/html.**

Comprobar el despliegue desde el navegador.

## 1.2.- Contenido dinámico

Una página dinámica, necesita más recursos del servidor web que una página estática, ya que consume más tiempo de CPU y más memoria.

La configuración y administración del servidor web será más compleja, cuantos más módulos tengamos que soportar, más tendremos que configurar y actualizar.

Repercusión en la seguridad del servidor web, cuántos más módulos más posibilidades de problemas de seguridad. Algunos módulos son: *mod\_actions, mod\_cgi\_ mod\_cgid, mod\_ext\_filter, mod\_include, mod\_ldap, mod\_perl, mod\_php, mod\_python.*

***ACTIVAR O DESACTIVAR MODULOS****:* a2enmod , a2ensite, a2dismod, a2dissite

Debemos controlar que es lo que se ejecuta en el servidor.

Ejercicio:

Copiar y descomprimir el fichero .zip en el directorio raíz.

Copiar el contenido de la web dinámica al directorio /var/www/html.

Comprobamos el funcionamiento accediendo desde un navegador a la URL.

*Composer: es una herramienta de administración de dependencias para PHP que facilita la gestión de bibliotecas y paquetes en tus proyectos utilizado por los desarrolladores PHP.*

* *Instalación de dependencias: permite definir las bibliotecas y paquetes que tu proyecto necesita y los instala automáticamente a partir de registros de paquetes en línea.*
* *Autocarga de clases: genera un archivo de autocarga que simplifica la inclusión de clases en tu código eliminando la necesidad de incluir manualmente archivos de clase.*
* *Resolución de conflictos: resuelve automáticamente conflictos entre diferentes versiones de dependencias para garantizar la compatibilidad de las bibliotecas en tu proyecto.*
* *Actualización de dependencias: puedes actualizar las dependencias de tu proyecto fácilmente con Composer, lo que te permite mantener tu código actualizado con las últimas versiones de las bibliotecas.*
* *Creación de paquetes personalizados: te permite crear y distribuir tus propias bibliotecas y paquetes PHP de manera sencilla.*
* [*https://packagist.org/*](https://packagist.org/)

|  |
| --- |
| * sudo apt update * sudo apt install php-cli unzip * cd ~ * curl -sS [https://getcomposer.org/installer -o /tmp/composer-setup.php](https://getcomposer.org/installer%20-o%20/tmp/composer-setup.php) * HASH=`curl -sS [https://composer.github.io/installer.sig`](https://composer.github.io/installer.sig%60) * echo $HASH * php -r "if (hash\_file('SHA384', '/tmp/composer-setup.php') === '$HASH') { echo 'Installer verified'; } else { echo 'Installer corrupt'; unlink('composer-setup.php'); } echo PHP\_EOL;" * sudo php /tmp/composer-setup.php --install-dir=/usr/local/bin --filename=composer * composer |