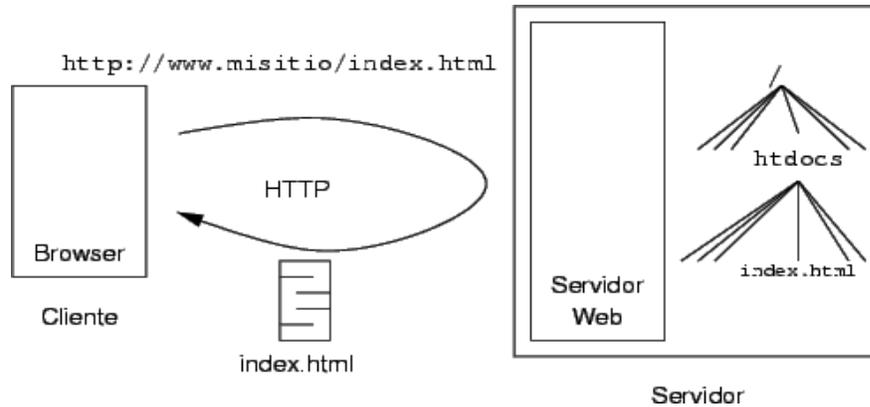


## HTTP (HyperText Transfer Protocol)



- ✓ Protocolo de transferencia de Hipertextos (Páginas Web, o paginas HTML)
- ✓ HTTP define la sintaxis y la semántica que utilizan los elementos software de la arquitectura web (clientes, servidores, proxies) para comunicarse.
- ✓ Es un esquema orientado a transacciones (petición/respuesta, cliente/servidor)
- ✓ La información transmitida se conoce como recurso y se identifica como URL (Uniform Resource Locator) Localizador de recurso uniforme.
- ✓ Los recursos pueden ser archivos, resultados de consultas de bases de datos, imágenes, etc.
- ✓ Utiliza el puerto de conexión 80
- ✓ HTTP funciona íntimamente ligado a HTML (HyperText Markup Language)

### Servidor WEB

- ✓ Un servidor WEB es un programa que implementa el protocolo HTTP
- ✓ Se mantiene a la espera de peticiones por parte de los clientes (navegador web) y responde a las peticiones adecuadamente mediante una página Web que se exhibira en el navegador
- ✓ Ej: Al ingresar [www.hotmail.com](http://www.hotmail.com) en nuestro navegador, el servidor WEB ante la petición del cliente, responde enviando el código HTML de una página en especifico solicitada. El cliente una vez recibido el código HTML lo interpreta y lo exhibe en pantalla.
- ✓ El cliente es el encargado de interpretar el código HTML

**Aplicaciones del lado del cliente:** Son ejecutadas por el cliente Web.

- JavaScript, Applets en Java
- Flash (lenguaje ActionScript)
- HTML (HyperText Markup Language)

**Aplicaciones del lado del servidor:** El servidor web ejecuta la aplicación; esta una vez ejecutada genera un código HTML, el cual es enviado al cliente.

- PHP
- ASP
- Perl
- JSP

### **Algunos servidores WEB**

- Apache
- IIS
- Tomcat
- Resin
- JBoss

## Servidor WEB APACHE

- Proviene de la palabra Patchy Server (Servidor Parchado), el cual fue conformado por diversos parches del servidor WEB NCSA servidor de ese entonces desarrollado por National Center Supercomputing
- El desarrollo de Apache se remonta al año de 1995, y en los años de 1998 incluyo soporte para plataformas Windows
  
- Actualmente se encuentra en su versión 2, que incluye entre otras características
  - Modo Híbrido
  - Nuevo sistema de configuración y compilación
  - Soporte IPv6
  - Mensajes de error en diferentes idiomas
  - Otras mejoras
  
- Permite crear sitios virtuales cuando se trata de dar hospedaje a varios sitios WEB dentro del mismo servidor

### Configuración del Servicio

Una vez instalado el servicio

***Apt-get install apache2***

Se inicia el servicio

***/etc/init.d/apache2 start/stop/restart/status***

***Service apache2 start/stop/restart/status***

El archivo principal de configuración es */etc/apache2/apache2.conf*, el cual se puede administrar mediante directivas de configuración, organizadas en 3 bloques:

- Global Environment: Aspectos globales de configuración
- Main Server Configuration: Aspectos del servidor principal
- Virtual Host

### Global Environment

Las directivas que se muestran en esta sección crean el entorno global para el servidor Apache. Este rige el funcionamiento general de Apache, como puede ser el número de peticiones concurrentes que se pueden manejar, la ubicación de los ficheros de configuración, etc.

<b>ServerType {standalone   inetd}</b>	Permite indicar el tipo de servidor a ejecutar.
<b>ServerRoot /etc/apache</b>	Donde se ubica toda la información de configuración y registro que necesita el servidor para su correcto funcionamiento
<b>Timeout 300</b>	Tiempo en segundos que el servidor esperará para recibir y enviar peticiones durante la comunicación, tras los cuales el servidor cierra la conexión
<b>KeepAlive On</b>	Indicar si se activarán las conexiones persistentes; es decir, el poder hacer más de una petición por conexión.
<b>MaxKeepAliveRequests 100</b>	Máximo número de peticiones que se pueden realizar en una conexión persistente (peticiones por conexión)
<b>Listen 80</b> <b>Listen 8080</b>	Especifica qué puertos se utilizará para atender las peticiones Para atender dos direcciones distintas por diferente puerto <b>Listen 192.168.255.5:80</b> <b>Listen 192.168.255.8:8080</b>
<b>LoadModule nombreModulo</b>	Permite cargar módulos que incluyen distintas funcionalidades

<b>ubicacionFichero</b>	
<b>MaxClients 200</b>	Permite especificar la cantidad máxima de clientes conectados simultáneamente al servidor
<b>MaxRequestsPerChild 80</b>	Cantidad de pedidos que puede atender un proceso servidor por hijo antes de que muera

### Main Server Configuration

Esta es la sección principal de configuración del servidor. En ella podemos encontrar las siguientes directivas:

<b>ServerAdmin</b> <b>admin@sitioweb.com</b>	Dirección de correo electrónico del administrador, Esta dirección aparece en los mensajes de error
<b>ServerName www.seguros.com</b>	Especifica cuál es el Nombre de Dominio Totalmente calificado, que responderá el sitio Web
<b>DocumentRoot /var/www/html</b>	Carpeta donde se colocan las páginas que se desea publicar.
<b>DirectoryIndex index.html index.php</b>	Especifica el fichero por defecto que buscará en cada directorio al subir un sitio, por defecto es index.html. Se puede especificar la prioridad si se tiene más de uno.
<b>ErrorLog</b>	Especifica la ubicación del fichero que contiene el registro de errores. Por defecto en la carpeta logs.
<b>CustomLog</b>	Especifica el fichero donde se anotan las peticiones realizada al servidor
<b>ServerTokens Pord</b>	Delimita qué información muestra el servidor sobre el servicio y componentes que está ejecutando: Pord, Min, Os y Full.
<b>ErrorDocument</b>	Establece la configuración del servidor para cuando se produzca un error, Sacar un texto de error, Redirigir a un fichero en el mismo directorio, etc.
<b>CacheRoot</b>	Establece el directorio donde se encontrarán los ficheros de la caché de Apache.
<b>CacheSize</b>	Tamaño de la caché en Kilobytes.

### Virtual Hosts (Hosts Virtuales)

Esta opción de Apache es muy útil en el caso de que contemos con más de un dominio en nuestro host; es decir, hosts virtuales. Hay que tener en cuenta que muchas de las directivas que hemos visto en la sección de Directivas básicas de configuración también se pueden aplicar a hosts virtuales.

#### **NameVirtualHost**

Cuando es necesario definir una dirección IP y/o puerto para el host virtual. **NameVirtualHost addr[:port]**

```
NameVirtualHost *:80
NameVirtualHost 172.20.30.40
NameVirtualHost 172.20.30.40:8080
```

Para un host virtual llamado host.virtual.com que utilice la dirección 172.20.30.40 y escuche por el puerto 8080

```
NameVirtualHost 172.20.30.40:8080
<VirtualHost 172.20.30.40:8080>
    ServerName www.host.virtual.com
    #Aquí ponemos otras directivas correspondientes a la directiva VirtualHost
</VirtualHost>
```

#### **ServerAlias**

Establece nombres alternativos para un host

```
<VirtualHost *:80>
    ServerName aula.server.com
    ServerAlias aula aula2.server.com aula2
    #Otras directivas
</VirtualHost>
```

### <VirtualHost>

La directiva del contenedor <VirtualHost> especifica una configuración para el host virtual. Todas las directivas encontradas entre <VirtualHost> y </VirtualHost> se aplican únicamente a dicho host virtual

```
<VirtualHost *:80>
<VirtualHost 172.20.30.50>
<VirtualHost 192.168.1.1 172.20.30.40>
<VirtualHost 172.20.30.40:8080>
<VirtualHost *:*>
```

Para dos sitios virtuales.

```
NameVirtualHost *:80

<VirtualHost *:80>
    ServerName publico.com
    DocumentRoot /var/www/publico
    .... Otras directivas
</VirtualHost>

<VirtualHost *:80>
    ServerName seguridad.com
    DocumentRoot /var/www/seguridad
    .... Otras directivas
</VirtualHost>
```

Para dos sitios con diferente Ip y Puerto

```
Listen 80
Listen 8080

NameVirtualHost 172.20.30.50:80
NameVirtualHost 172.20.30.40:8080

<VirtualHost 172.20.30.50:80>
    ServerName example1.com
    DocumentRoot /www/domain80
</VirtualHost>

<VirtualHost 172.20.30.40:8080>
    ServerName example2.com
    DocumentRoot /www/domain8080
</VirtualHost>
```

### **Activa y desactivar un Sitio**

**a2ensite, a2dissite:** Activa y desactiva un sitio web en Apache

a2ensite ejemplo.com.conf

**a2enmod, a2dismod:** Activa y desactiva un modulo para Apache

a2enmod ssl

**a2enconf, a2disconf:** Activa o desactiva un archivo de configuración en Apache

### Configurar un bloque <VirtualHost> con SSL

- Escucha las peticiones por el puerto 443
- Se deben Activar el modulo:

*a2enmod ssl*

Implemetar un VirtualHost

```
<VirtualHost 192.168.0.1:443>
    DocumentRoot /var/www/html2
    ServerName su.dominio.com
    SSLEngine on
    SSLCertificateFile /ruta/a/su_dominio.crt
    SSLCertificateKeyFile /ruta/a/su_dominio.key
</VirtualHost>
```