

# Configuración do servidor web

## 1.1 Introducción

Na actividade que nos ocupa aprenderanse os seguintes conceptos e manexo de destrezas sobre un servidor web:

- Levar a cabo a configuración avanzada.
- Instalar novos módulos.
- Crear sitios virtuais.
- Realizar probas de funcionamento.
- Documentar a configuración do servidor.

## 1.2 Actividade

### 1.2.1 Ficheiros de configuración

A configuración de Apache ten lugar a través de arquivos textuais mediante diversas directivas que permiten escoller as distintas opcións dispoñibles. Se ben as directivas son as mesmas independentemente do sistema operativo, a distribución e localización dos arquivos, son diferentes.

Á hora de escribir os arquivos de configuración, debemos ter en conta que:

- Os ficheiros de configuración conteñen unha directiva por liña. Pode empregarse unha barra invertida \ como último carácter dunha liña para indicar que a directiva continúa na liña seguinte. Non pode haber outros caracteres ou espazos en branco entre a barra invertida e a fin de liña.
- Os argumentos das directivas sepáranse destas por medio de espazos en branco. Se un argumento contén espazos, debe ser escrito entre comiñas.
- As directivas non son sensibles a maiúsculas (case-insensitive), pero moitos dos argumentos si o son.
- As liñas que comezan polo carácter # considéranse comentarios. polo que son ignorados. Os comentarios non poden ser incluídos na mesma liña que unha directiva de configuración.
- Tanto os espazos en branco a principio de liña como as liñas en branco se ignoran, polo que poden empregarse para facilitar a lectura dos ficheiros de configuración.

### 1.2.1.1 Linux (Debian)

O ficheiro principal de configuración de Apache es `/etc/apache2/apache2.conf`, nel están incluídos os demais arquivos de configuración:

- `/etc/apache2/ports.conf` Configuración dos portos de escoita.
- `/etc/apache2/sites-available/000-default.conf` Configuración do servidor virtual por defecto.

### 1.2.1.2 Windows

A configuración do servidor atópase no ficheiro `C:\Apache24\conf\httpd.conf`.

## 1.2.2 Lanzar e parar Apache

Cada vez que se modifican os ficheiros de configuración, para que os cambios teñan lugar, é preciso que se reinicie o servidor.

### 1.2.2.1 Linux

Para parar o servizo, emprégase o comando:

```
sudo service apache2 stop
```


Para lanzar o servizo, emprégase o comando:

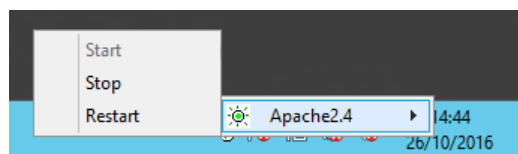
```
sudo service apache2 start
```

Se queremos parar e relanzar o servizo:

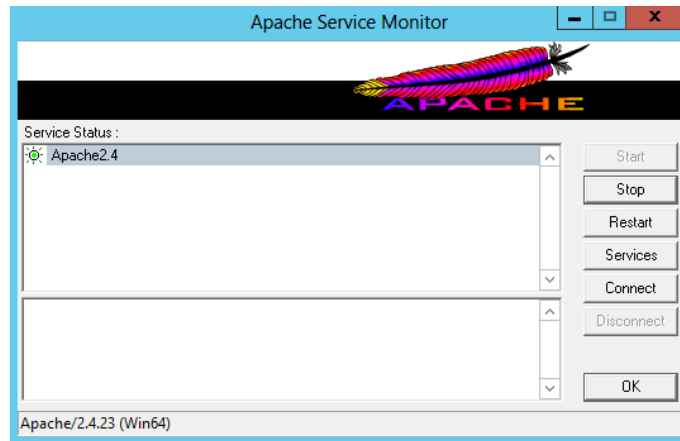
```
sudo service apache2 restart
```

### 1.2.2.2 Windows

Podemos parar e relanzar o servizo desde a interface gráfica, ben na icona que aparece na área de notificacións ,



ou no Monitor de Apache.



## 1.2.3 Directivas de configuración básicas

### 1.2.3.1 Listen

Esta directiva indica a través de que portos e interfaces IP aceptará peticións. Por defecto, responde peticións en todas as as interfaces, no porto que se indique. Na versión 2.4 de Apache é unha directiva obrigatoria.

#### Exemplos

- Para facer que o servidor acepte conexións nos portos 80 e 8080:  

```
Listen 80  
Listen 8080
```
- Para atender dous enderezos IP distintos, con distintos portos, empregárase:  

```
Listen 192.168.255.5:80  
Listen 192.168.255.8:8080
```

### 1.2.3.2 ServerRoot

Especifica a localización do directorio raíz onde se atopa instalado Apache. Esta directiva non debería cambiar a non ser que se mova o cartafol de instalación de Apache a outra localización.

#### Exemplo

```
ServerRoot /home/httpd
```

### 1.2.3.3 Include

Permite que se inclúan outros arquivos de configuración en tempo de execución. A ruta a estes arquivos de configuración poden ser absolutas ou relativas con respecto ao directorio indicado en `ServerRoot`.

#### Exemplo

```
Include ports.conf
```

#### 1.2.3.4 DocumentRoot

Indica o directorio desde o que Apache vai servir os arquivos. O servidor engade a ruta indicada na URL a este directorio.

Todos os directorios que vai servir Apache deben ter permisos de lectura e execución para todos os usuarios e todos os arquivos que serve deben ter permiso de lectura. Recordemos que os permisos de arquivos e directorios se cambian en Linux co comando `chmod`.

##### Exemplo

Para un valor

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

Se a URL solicitada é `http://www.meuservidor.com/proba/index.html` Apache servirá o ficheiro `index.html` que se atopa en `/var/www/html/proba`

#### 1.2.3.5 ErrorLog

Especifica a localización do ficheiro que contén o rexistro de erros. Se a ruta que se indica non é absoluta, considerarase relativa ao `ServerRoot`. Por defecto atópanse no cartafol `logs` dentro de `ServerRoot`.

##### Exemplo

```
ErrorLog /var/log/httpd/error_log
```

#### 1.2.3.6 DirectoryIndex

Especifica o ficheiro por defecto que se servirá para cada directorio, no caso de que non se especifique ningún na URL. Por defecto é `index.html`.

Poden indicarse varios ficheiros. A orde co que se especifica o nome de ficheiro determinará a prioridade á hora de decidir que ficheiro é o que se amosa.

##### Exemplo

```
DirectoryIndex index.html indice.html index.php
```

#### 1.2.3.7 ErrorDocument

Explica que debe facerse no caso de que se produza un erro. Se non se indica nada, amosarase a mensaxe por defecto. Se se indica, poderá amosarse un texto personalizado ou unha páxina web personalizada dentro ou fóra do noso servidor.

##### Exemplos

```
ErrorDocument 404 "Este ficheiro non está"  
ErrorDocument 404 erros/error404.html  
ErrorDocument 404  
http://www.outroservidor.com/erro404externo.html
```

#### 1.2.3.8 Alias

Permite a definición de directorios virtuais. Un directorio virtual é un directorio que se atopa nunha localización distinta da que se indica na URL. O directorio virtual non ten por que atoparse dentro da árbore de directorios que se crea a partir de `DocumentRoot`, senón que se pode atopar en calquera outra localización. A súa sintaxe é:

```
Alias nombreFicticio ubicaciónReal
```

##### Exemplo

```
Alias /ciclos/daw /home/usuario/daw
```

#### 1.2.3.9 Redirect

A directiva `Redirect` permite indicar ao cliente que un documento foi modificado ou actualizado. Ten preferencia con respecto á directiva `Alias`, independentemente da orde no que se sitúen no ficheiro de configuración.

##### Exemplo

```
Redirect /service http://outrodirectorio.exemplo.com/service
```

#### 1.2.3.10 Options

Controla que características do servidor están dispoñibles para un determinado directorio. As características fanse dispoñibles se se listan e non dispoñibles se se preceden por un signo `-`. Algunhas destas opcións son as seguintes:

- `Indexes`: Se se solicita unha URL que apunta a un directorio e no devandito directorio non se atopa ningún dos ficheiros indicados por `DirectoryIndex`, amosarase unha listaxe dos arquivos contidos nese directorio.
- `FollowSymLinks`: O servidor seguirá ligazóns simbólicas neste directorio.

##### Exemplos

```
Options Indexes -FollowSymLinks
```

### 1.2.4 Etiquetas de configuración por seccións

As directivas que poden poñerse dentro de etiquetas de bloque e así afectar unicamente a ese bloque. A continuación expoñemos algúns dos distintos tipos de bloques.

#### Directory

As directivas que se atopen dentro dunha etiqueta `<Directory>` só se aplican a ese directorio, os seu subdirectorios e os contidos.

O directorio debe indicarse coa súa ruta absoluta.

### Exemplo

```
<Directory /var/www/html/ciclos>
#Aquí as directivas
</Directory>
```

### Files

As directivas que se atopan dentro desta etiqueta unicamente se aplican aos ficheiros indicados.

### Exemplo

```
<Files "privado.html">
#Aquí as directivas
</Files>
```

### Location

As directivas que se atopan dentro desta etiqueta unicamente se aplican no ámbito da URL indicada na etiqueta.

### Exemplo

```
<Location /privado1>
#Aquí as directivas
</Location /privado1>
```



Realiza as tarefas Tarefa 1.2. “Configuración básica de Apache en Linux” e a Tarefa 1.3. “Configuración básica de Apache en Windows” nas que se empregarán as directivas expostas.

## 1.2.5 Módulos

O deseño de Apache é modular. O núcleo de Apache (core) prové a funcionalidade básica dun servidor web, pero existen moitos módulos adicionais que permiten engadir funcións extra.

Cada módulo ten un conxunto de directivas específicas que permiten a súa xestión.

A adición desas funcións pode realizarse de dúas formas:

- Estaticamente: no momento de compilar o executable do servidor, indícanse que funcións adicionais se desexan incorporar ao núcleo.
- Dinamicamente: no momento de lanzar o servidor, se cargan xunto a el aqueles módulos que implementen a configuración desexada. Os módulos cargados se xestionan na configuración do servidor. Estes módulos que se poden cargar dinamicamente son denominados obxectos dinámicos compartidos ou módulos DSO (Dynamic Shared Object).

### 1.2.5.1 Consulta de módulos estáticos

Ubuntu	Windows
<code>sudo apache2ctl -l</code>	<code>C:\Apache24\bin\httpd.exe -l</code>

### 1.2.5.2 Consulta de módulos dinámicos

Ubuntu	Windows
Lista os contidos do cartafol <code>/etc/apache2/mods-enabled</code>	Consultando as directivas <code>LoadModule</code> do ficheiro de configuración de Apache ( <code>C:\Apache24\conf\httpd.conf</code> )

### 1.2.5.3 Consulta de módulos dinámicos dispoñibles

Ubuntu	Windows
<ul style="list-style-type: none"><li>▪ Xa instalados: Listar o directorio <code>/usr/lib/apache2/modules</code></li><li>▪ Non instalados: <code>sudo apt-cache search libapache2-mod</code> (instálanse coma calquera outro paquete en Ubuntu).</li></ul>	Consultar o cartafol <code>C:\Apache24\modules</code>

### 1.2.5.4 Habilitar módulo dinamicamente

Ubuntu	Windows
<code>sudo a2enmod nome_modulo</code>	No ficheiro <code>httpd.conf</code> , eliminar o comentario das directivas <code>LoadModule</code> e <code>Include</code> correspondentes a ese módulo

### 1.2.5.5 Deshabilitar módulo dinamicamente

Ubuntu	Windows
<code>sudo a2dismod nome_modulo</code>	No ficheiro <code>httpd.conf</code> , comentar as directivas <code>LoadModule</code> e <code>Include</code> correspondentes a ese módulo.

### 1.2.5.6 Inserción de directivas correspondentes ao módulo

Ubuntu	Windows
No ficheiro de extensión <code>.conf</code> correspondente ao módulo, que se atopa no directorio <code>/etc/apache2/mods-available</code> Dentro das etiquetas <code>&lt;IfModule&gt;&lt;/IfModule&gt;</code>	No ficheiro de extensión <code>.conf</code> correspondente ao módulo que se atopa no directorio <code>C:\Apache24\conf\extra</code>

### 1.2.5.7 O módulo `userdir`

Userdir é un módulo de Apache que fai posible que todos os usuarios con acceso a un servidor teñan un cartafol propio no cal poidan aloxar as súas páxinas e arquivos.

Para acceder a estas páxinas desde o cliente, debe engadirse á IP ou URL o usuario. Por exemplo:

`http://192.168.0.1/~xurxo`

## Ficheiro de configuración

Ubuntu	Windows
/etc/apache2/mods-available/userdir.conf	C:\Apache24\conf\extra\httpd-userdir.conf

## Cartafol por defecto

Ubuntu	Windows
/home/usuario/public_html	C:\Users\usuario\My Documents\My Website (ollo!, na máquina virtual de Windows Server 2012, non se chama My Documents, senón Documents polo que hai que cambialo no ficheiro de configuración)



Realiza a Tarefa 1.4 “Instalación de Userdir en Linux” e a Tarefa 1.5 “Instalación de Userdir en Windows” nas que se habilitará e configurará un novo módulo en ambos sistemas operativos.

### 1.2.6 Aloxamento compartido (virtual hosting)

O termo aloxamento compartido fai referencia á práctica de aloxar máis dun sitio web nunha mesma máquina (por exemplo, un.dominioogus.es e dous.dominioextraño.com).

Pode estar baseado en dirección IP, de modo que cada sitio teña a súa propia dirección IP; ou baseado en nome de forma que se teñan varias direccións simbólicas para unha mesma dirección IP.

#### 1.2.6.1 Aloxamento compartido baseado en nome

O servidor confía no cliente para que lle indique o nome do host como parte das cabeceiras HTTP. Normalmente, este é máis sinxelo que o baseado en IP, xa que unicamente son necesarios dous pasos:

- Configurar o servidor DNS para que relacione o enderezo simbólico coa IP adecuada.
- Configurar o servidor Apache para que recoñeza os diferentes enderezos simbólicos.

Reduce o problema da escaseza de direccións IP. Por iso, debe utilizarse o aloxamento compartido baseado en nomes salvo que o equipamento a empregar requira o contrario.

## Pasos

- Designar en que IP e portos se aceptarán peticións coa directiva `NameVirtualHost`.

```
NameVirtualHost *:80
```

Indica que se aceptarán peticións en todas as IP no porto 80.



- Crear un bloque `<VirtualHost>` para cada un dos diferentes sitios que se van servir. O argumento deste bloque debe coincidir co dalgũa directiva `NameVirtualHost`.
- Dentro de cada un destes bloques se precisan, polo menos, dúas directivas:
  - `ServerName` para designar que sitio se serve
  - `DocumentRoot` para indicar onde se atopan os ficheiros dese sitio na árbore de directorios.
- Poden incluírse máis directivas que indiquen como se serve o devandito sitio: `DirectoryIndex`, directivas de control de acceso... Para saber se unha directiva pode ou non situarse nun bloque `VirtualHost` debe consultarse o apartado `Context` da documentación oficial de dita directiva.

Moitos sitios web queren ser accesibles a través de varios nomes, para iso se emprega a directiva `ServerAlias`.

### Exemplo

```
NameVirtualHost *:80

<VirtualHost *:80>
    ServerName uno.dominioogus.es
    ServerAlias uno.dominioogus.com
    DocumentRoot /var/www/uno
    DirectoryIndex indice.html
</VirtualHost>

<VirtualHost *:80>
    ServerName dos.dominioextraño.com
    DocumentRoot /var/www/dos
</VirtualHost>
```

### Onde situar as directivas

#### Ubuntu

A directiva `NameVirtualHost` debe situarse no ficheiro `/etc/apache2/ports.conf`

Os bloques `<VirtualHost>` deben situarse cada un nun arquivo separado dentro do directorio `/etc/apache/sites-available` e o seu nome deber ser significativo. Recoméndase utilizar o indicado na directiva `ServerName`. En Apache 2.4 é necesario que estes ficheiros teñan a extensión `.conf`

- Listado de sitios virtuais habilitados
 

```
ls /etc/apache2/sites-enabled
```
- Activación dun sitio virtual
 

```
sudo a2ensite nombreServidor
```

- Desactivación dun sitio virtual

```
sudo a2dissite nombreServidor
```

## Windows

Todas elas están no ficheiro `C:\Apache24\conf\extras\httpd-vhost.conf`. Para poder utilizar aloxamento compartido, debemos incluír o devandito ficheiro en `httpd.conf`, para iso, descomentamos a liña:

```
# Virtual hosts  
Include conf/extra/httpd-vhosts.conf
```



Realiza a Tarefa 1.6 “Sitios virtuais en Linux” e a Tarefa 1.7 “Sitios virtuais en Windows” nas que se crearán e configurarán os sitios virtuais en ambos sistemas operativos.