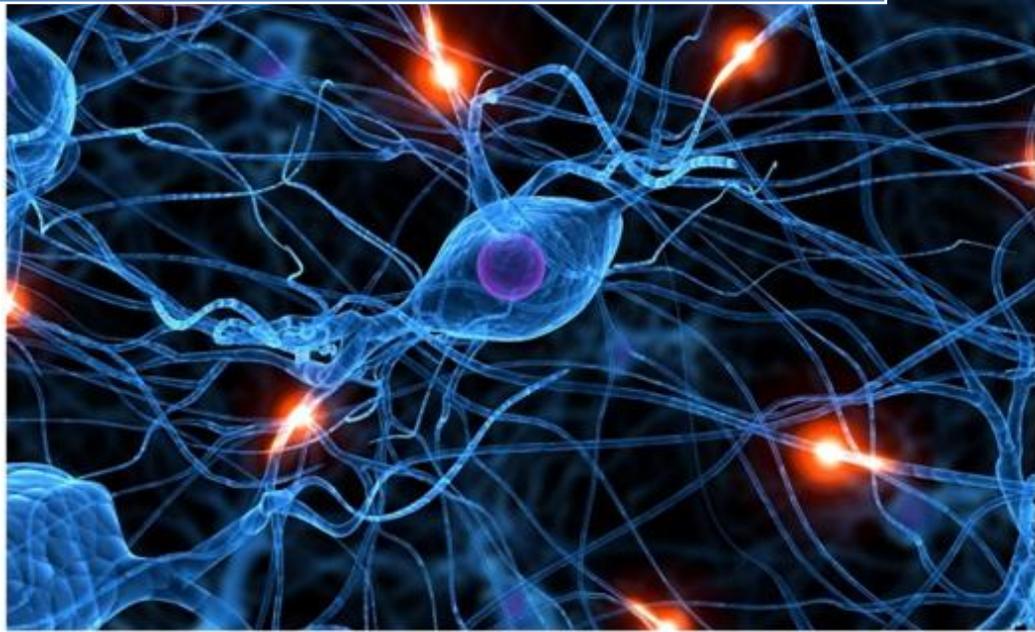


2011-
2012

Actividades 3 y 4 Instalación y configuración de un servidor HTTP (Apache2) en Ubuntu Server (configuraciones por defecto)



José Jiménez Arias
IS Gregorio Prieto
2011-2012

En primer lugar actualizamos los repositorios y a continuación instalamos la aplicación:

```
Legendo lista de paquetes... Hecho
root@ubuntusrv04:/home/josejimenez# apt-get install apache2
Legendo lista de paquetes... Hecho
Creando árbol de dependencias
Legendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
 apache2-mpm-worker apache2-utils apache2.2-bin apache2.2-common libapr1
 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap ssl-cert
Paquetes sugeridos:
 apache2-doc apache2-suexec apache2-suexec-custom
Se instalarán los siguientes paquetes NUEVOS:
 apache2 apache2-mpm-worker apache2-utils apache2.2-bin apache2.2-common
 libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap ssl-cert
```

Comprobamos que se inicia el servicio:

```
Configurando apache2-mpm-worker (2.2.14-5ubuntu8.7) ..
* Starting web server apache2
```

Observamos los directorios que crea el programa tras su instalación:

```
root@ubuntusrv04:/home/josejimenez# cd /var
root@ubuntusrv04:/var# ls
backups cache crash lib local lock log mail opt run spool tmp www
root@ubuntusrv04:/var# cd www
root@ubuntusrv04:/var/www# ls
index.html
```

Observamos los ficheros de gestión de usuarios y grupos y observamos que se crean el usuario y grupo www-data.

```
root@ubuntusrv04:/var/www# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
proxu:x:13:13:proxu:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
dip:x:30:
www-data:x:33:
backup:x:34:
operator:x:37:
list:x:38:
```

Observamos que el propietario es root:

```
root@ubuntusrv04:/var/www# ls -l /var
total 44
drwxr-xr-x  2 root root  4096 2011-07-15 15:48 backups
drwxr-xr-x  9 root root  4096 2012-01-09 09:54 cache
drwxrwxrwt  2 root root  4096 2010-04-13 22:52 crash
drwxr-xr-x 34 root root  4096 2011-11-10 12:25 lib
drwxrwsr-x  2 root staff 4096 2011-07-15 15:48 local
drwxrwxrwt  4 root root    80 2012-01-09 09:54 lock
drwxr-xr-x 10 root root  4096 2012-01-09 09:54 log
drwxrwsr-x  2 root mail  4096 2011-09-28 10:58 mail
drwxr-xr-x  2 root root  4096 2011-09-28 10:58 opt
drwxr-xr-x 10 root root   400 2012-01-09 09:55 run
drwxr-xr-x  4 root root  4096 2011-09-28 10:59 spool
drwxrwxrwt  2 root root  4096 2011-07-15 15:48 tmp
drwxr-xr-x  2 root root  4096 2012-01-09 09:54 www
```

Realizamos un ls en el directorio apache2 para observar su contenido:

```
root@ubuntusrv04:/etc/apache2# ls
apache2.conf  envvars      magic          mods-enabled  sites-available
conf.d        httpd.conf   mods-available ports.conf     sites-enabled
root@ubuntusrv04:/etc/apache2#
```

Comprobamos que el servicio apache está iniciado:

```
root@ubuntusrv04:/etc/apache2# ps -ef | grep apache2
root      1388      1  0 09:54 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1409    1388  0 09:55 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1410    1388  0 09:55 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1411    1388  0 09:55 ?        00:00:00 /usr/sbin/apache2 -k start
root      1500    939  0 10:07 tty1    00:00:00 grep --color=auto apache2
root@ubuntusrv04:/etc/apache2#
```

Comprobamos también que el puerto 80 está escuchando:

```
root@ubuntusrv04:/etc/apache2# netstat -ltn
Conexiones activas de Internet (solo servidores)
Proto Recib Enviad Dirección local      Dirección remota      Estado
tcp    0      0 0.0.0.0:80           0.0.0.0:*              ESCUCHAR
tcp    0      0 10.6.8.50:53         0.0.0.0:*              ESCUCHAR
tcp    0      0 127.0.0.1:53         0.0.0.0:*              ESCUCHAR
tcp    0      0 127.0.0.1:953       0.0.0.0:*              ESCUCHAR
tcp6   0      0 :::53                :::*                   ESCUCHAR
tcp6   0      0 :::1:953             :::*                   ESCUCHAR
root@ubuntusrv04:/etc/apache2#
```

Entramos en el fichero de configuración /etc/apache2/ports.conf y comprobamos que están habilitados los servidores virtuales por nombre [Name Virtual Host *:80] en todas las direcciones Ip (con *) y en el puerto 80. Esto anula el servidor principal en todas las direcciones Ip.

```
root@ubuntusrv04:/etc/apache2# cat ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default
# This is also true if you have upgraded from before 2.2.9-3 (i.e. from
# Debian etch). See /usr/share/doc/apache2.2-common/NEWS.Debian.gz and
# README.Debian.gz

NameVirtualHost *:80
Listen 80

<IfModule mod_ssl.c>
    # If you add NameVirtualHost *:443 here, you will also have to change
    # the VirtualHost statement in /etc/apache2/sites-available/default-ssl
    # to <VirtualHost *:443>
    # Server Name Indication for SSL named virtual hosts is currently not
    # supported by MSIE on Windows XP.
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>
```

A continuación entramos al directorio sites-available y comprobamos que existe default, que contiene la configuración del servidor virtual por defecto.

```
root@ubuntusrv04:/etc/apache2/sites-available# ls
default default-ssl
root@ubuntusrv04:/etc/apache2/sites-available#
```

Comprobamos que en sites-enabled existe 000-default.

```
root@ubuntusrv04:/etc/apache2# cd sites-enabled
root@ubuntusrv04:/etc/apache2/sites-enabled# ls
000-default
root@ubuntusrv04:/etc/apache2/sites-enabled#
```

Observamos el fichero default y comprobamos que el directorio raíz /var/www

```
GNU nano 2.2.2 Archivo: default
<VirtualHost *:80>
  ServerAdmin webmaster@localhost

  DocumentRoot /var/www
  <Directory />
```

Comprobamos a través de la terminal del cliente que contacta con el servidor:

```
root@alvaroniko:/home/joseyedu# ping 10.33.4.3
PING 10.33.4.3 (10.33.4.3) 56(84) bytes of data.
64 bytes from 10.33.4.3: icmp_req=1 ttl=64 time=3.89 ms
64 bytes from 10.33.4.3: icmp_req=2 ttl=64 time=0.492 ms
^Z
[3]+ Detenido ping 10.33.4.3
root@alvaroniko:/home/joseyedu#
```

Comprobamos a través del navegador del cliente que contacta con el servidor:



It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.

Entramos en /var/www (el directorio) y observamos el contenido de index.html para comprobar que sale lo mismo que en el navegador del cliente.

```
GNU nano 2.2.2 Archivo: index.html
<html><body><h1>It works!</h1>
<p>This is the default web page for this server.</p>
<p>The web server software is running but no content has been added, yet.</p>
</body></html>
```

A continuación creamos un nuevo archivo en el directorio /var/www de nombre red.html

```
GNU nano 2.2.2 Archivo: red.html
<html>
<body>
<h1> ESTE ES EL SERVIDOR WEB DE LA RED ASIRO4.</h1>
</body>
</html>
```

Nos situamos en el cliente y accedemos a la pagina creada anteriormente.



Creamos el directorio datos dentro de /var/www.

```
root@(none):/var/www# ls
index.html red.html
root@(none):/var/www# mkdir datos
root@(none):/var/www# ls
datos index.html red.html
root@(none):/var/www#
```

Dentro del directorio datos creamos en fichero datos1.html con el siguiente código:

```
GNU nano 2.2.2 Archivo: datos1.html
<html>
<body>
<b>MIS DATOS SON JOSE Y JUAN EDUARDO</b>
</body>
</html>
```

MONTAMOS EL SERVIDOR DNS EN EL MISMO EQUIPO SERVIDOR WEB.

Configuramos el cliente y le asociamos un servidor dns, que será el mismo que el servidor web.

```
GNU nano 2.2.4      Fichero: /etc/resolv.conf
domain asir04.
search asir04.
nameserver 10.33.4.3
```

Nos dirigimos al cliente Windows 7 y configuramos los parámetros de red.

Obtener una dirección IP automáticamente

Usar la siguiente dirección IP: _____

Dirección IP:

Máscara de subred:

Puerta de enlace predeterminada:

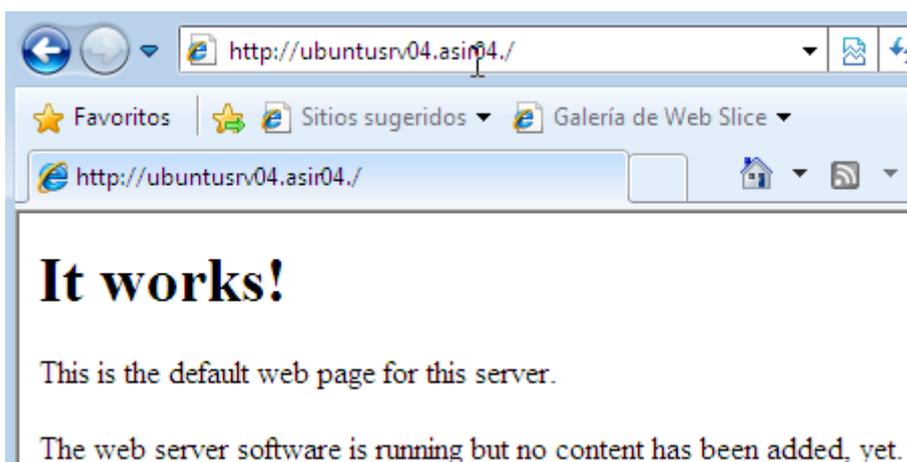
Obtener la dirección del servidor DNS automáticamente

Usar las siguientes direcciones de servidor DNS: _____

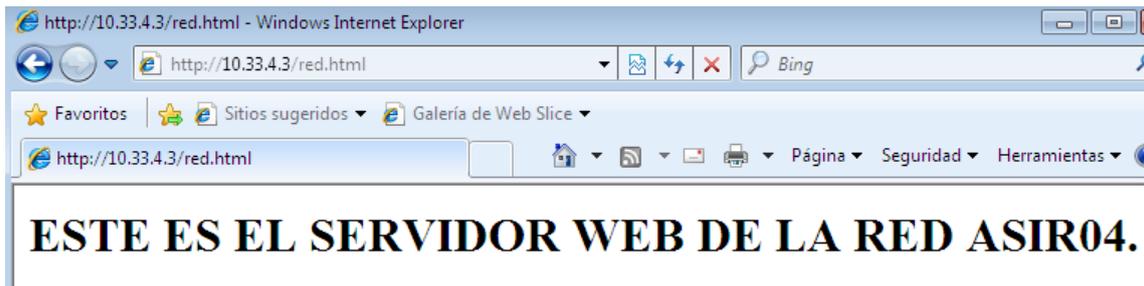
Servidor DNS preferido:

A continuación comprobamos mediante navegador que contacta con el servidor ubuntu04 y este facilita las páginas web.

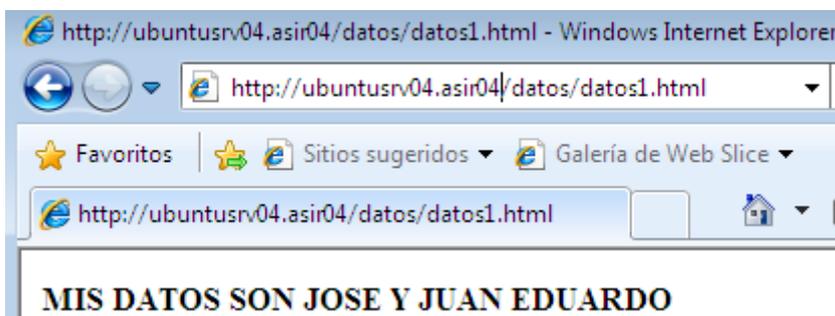
<http://ubuntu04.asir04.>



http://10.33.4.3/red.html



http://ubuntusrv04.asir04/datos/datos1.html



Continuamos realizando operaciones en el servidor:

DIRECTIVAS de apache2.conf

```
# Timeout: The number of seconds before receives and sends time out.  
#  
Timeout 300  
#  
#  
# KeepAlive: Whether or not to allow persistent connections (more than  
# one request per connection). Set to "Off" to deactivate.  
#  
KeepAlive On  
#  
# MaxKeepAliveRequests: The maximum number of requests to allow  
# during a persistent connection. Set to 0 to allow an unlimited amount.  
# We recommend you leave this number high, for maximum performance.  
#  
MaxKeepAliveRequests 100
```

El máximo de clientes es 150 por defecto.

```
<IfModule mpm_event_module>
  StartServers      2
  MaxClients        150
  MinSpareThreads   25
  MaxSpareThreads   75
  ThreadLimit       64
  ThreadsPerChild   25
  MaxRequestsPerChild 0
</IfModule>
```

Nos dirigimos a /etc/apache2/apache2.conf y buscamos la directiva Server Root

```
#
ServerRoot "/etc/apache2"
```

Buscamos dentro del documento los parámetros de las directivas User y Group.

```
# These need to be set in /etc/apache2/envvars
User ${APACHE_RUN_USER}
Group ${APACHE_RUN_GROUP}
```

Consultamos el fichero envvars que se encuentra en /etc/apache2

```
jose.jimenez@ubuntusrv04:/etc/apache2$ cat envvars
# envvars - default environment variables for apache2ctl

# Since there is no sane way to get the parsed apache2 config in scripts, some
# settings are defined via environment variables and then used in apache2ctl,
# /etc/init.d/apache2, /etc/logrotate.d/apache2, etc.
export APACHE_RUN_USER=www-data
export APACHE_RUN_GROUP=www-data
export APACHE_PID_FILE=/var/run/apache2.pid
```

El número máximo de peticiones simultáneas que permite el servidor por defecto es: 150

La directiva es MaxClients

La directiva contenedora de /var/www es la siguiente:

```
<Directory /var/www/>
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
</Directory>
```

Por último añadimos un CNAME o nombre canonico a nuestro servidor DNS para que podemos resolver las www.

```
asir04.      IN      SOA      asir04. root.asir04. (
                2          ; Serial
                604800     ; Refresh
                86400      ; Retry
                2419200    ; Expire
                604800 )   ; Negative Cache TTL
;
asir04.      IN      NS       ubuntu04.asir04.
ubuntu04.asir04. IN     A        10.33.4.3
windows7.asir04. IN     A        13.33.4.10
debian04.asir04. IN     A        10.33.4.30
opensuse04.asir04. IN   A        10.33.4.50
molinux04.asir04. IN   A        10.33.4.60
fedora04.asir04. IN    A        10.33.4.70
windowsxp04.asir04. IN  A        10.33.4.20
www.ubuntu04.asir04. IN  CNAME    ubuntu04.asir04.
```

Comprobamos que funciona correctamente de la siguiente forma:

