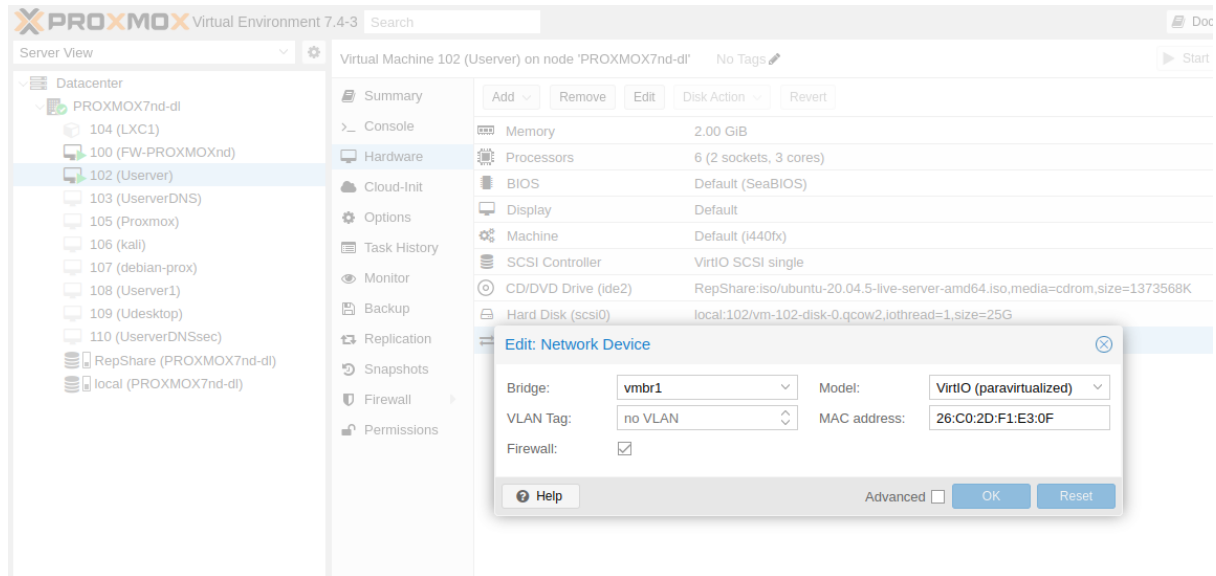


Práctica Apache: Multisite

Trabajo realizado por: [David López](#)



Configuración de entorno



Para empezar, yo he hecho la práctica en proxmox, así que lo primero que hago es configurar la interfaz de red de mi Servidor Apache, la cual yo asigno en la “vmbr1”, es decir, estará en la DMZ.

```
david@userver:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 26:c0:2d:f1:e3:0f brd ff:ff:ff:ff:ff:ff
    inet 172.16.1.12/24 brd 172.16.1.255 scope global ens18
        valid_lft forever preferred_lft forever
    inet6 fe80::24c0:2dff:fef1:e30f/64 scope link
        valid_lft forever preferred_lft forever
david@userver:~$ ping -c 1 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=114 time=7.11 ms

--- 8.8.8.8 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 7.110/7.110/7.110/0.000 ms
```

En esta captura demuestro que estoy en la red “172.16.1.0”. Tengo una ip estática, que yo elijo que sea la “172.16.1.12/24” y además, muestro que tengo salida a internet.

```
david@userver:~$ hostnamectl set-hostname apacheserver
==== AUTHENTICATING FOR org.freedesktop.hostname1.set-static-hostname ====
Authentication is required to set the statically configured local host name, as well as the pretty host name.
Authenticating as: david
Password:
==== AUTHENTICATION COMPLETE ====
david@userver:~$ sudo reboot
```

A continuación, “seteo” el “hostname” de la máquina a “apacheserver”, con lo que realizo un reboot para aplicar los cambios.

Instalación de Apache2 Server

```
david@apacheserver:~$ sudo apt update && sudo apt upgrade
Obj:1 http://es.archive.ubuntu.com/ubuntu focal InRelease
Obj:2 http://es.archive.ubuntu.com/ubuntu focal-updates InRelease
Obj:3 http://es.archive.ubuntu.com/ubuntu focal-backports InRelease
Obj:4 http://es.archive.ubuntu.com/ubuntu focal-security InRelease
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Todos los paquetes están actualizados.
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Calculando la actualización... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  linux-headers-5.4.0-163 linux-headers-5.4.0-163-generic linux-image-5.4.0-163-generic linux-modules-5.4.0-163-generic linux-modules-extra-5.4.0-163-generic
Utilice «sudo apt autoremove» para eliminarlos.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
```

Para comenzar con la instalación de apache2 hay que, primero, actualizar nuestros repositorios y aplicar esos cambios con el comando “sudo apt update && sudo apt upgrade”.

```
david@apacheserver:~$ sudo apt install apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
apache2 ya está en su versión más reciente (2.4.41-4ubuntu3.14).
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  linux-headers-5.4.0-163 linux-headers-5.4.0-163-generic linux-image-5.4.0-163-generic linux-modules-5.4.0-163-generic linux-modules-extra-5.4.0-163-generic
Utilice «sudo apt autoremove» para eliminarlos.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
```

Y ahora sí que podremos instalar apache con el comando “sudo apt install apache2”. Como yo ya lo tengo instalado me reporta que no hay nuevos paquetes a instalar.

```
david@apacheserver:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2023-11-16 18:10:52 CET; 49s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 30647 (apache2)
    Tasks: 55 (limit: 2255)
   Memory: 6.5M
   CGroup: /system.slice/apache2.service
           └─30647 /usr/sbin/apache2 -k start
             └─30679 /usr/sbin/apache2 -k start
               └─30680 /usr/sbin/apache2 -k start

nov 16 18:10:52 apacheserver systemd[1]: Starting The Apache HTTP Server...
nov 16 18:10:52 apacheserver apachectl[30633]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.16.1.12. Set
nov 16 18:10:52 apacheserver systemd[1]: Started The Apache HTTP Server.
```

Y por último, en este apartado he añadido la salida del “status” de apache2 para que se pueda comprobar que realmente está corriendo.

Configuración de Apache2 Server

```
david@apacheserver:~$ sudo a2enmod -m ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Module socache_shmcb already enabled
Module ssl already enabled
```

Como se van a utilizar certificados, hay que habilitar el módulo de seguridad con “sudo a2enmod -m ssl”.

```
david@apacheserver:~$ cat /etc/apache2/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.conf

#Listen 80

<IfModule ssl_module>
    Listen 443
</IfModule>

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

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Ahora, hay que comentar la línea donde pone “Listen 80” porque no es el puerto que vamos a usar.

```
david@apacheserver:~$ sudo a2dissite 000-default.conf
Site 000-default disabled.
To activate the new configuration, you need to run:
  systemctl reload apache2
david@apacheserver:~$ systemctl reload apache2
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to reload 'apache2.service'.
Authenticating as: david
Password:
==== AUTHENTICATION COMPLETE ====
david@apacheserver:~$ sudo a2dissite 000-default.conf
Site 000-default already disabled
```

Lo siguiente es deshabilitar el sitio por defecto que viene en apache2 habilitado, para habilitar el nuestro.

```
david@apacheserver:~$ cat /etc/hosts
127.0.0.1 localhost
127.0.1.1 userver
172.16.1.12 david.com www.david.com publica.com intranet.empresa.com
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Antes de empezar a configurar los sitios, hay que añadir una línea nueva en nuestro “/etc/hosts”, en la que incluiremos nuestra IP más las direcciones a las que resolverá la IP.

```
david@apacheserver:~$ sudo mkdir -p /var/www/publica/david.com
david@apacheserver:~$ touch /var/www/publica/david.com/index.html
touch: cannot touch '/var/www/publica/david.com/index.html': Permission denied
david@apacheserver:~$ sudo touch /var/www/publica/david.com/index.html
```

En esta captura creo el directorio donde se alojará mi pagina publica, y creo un fichero que será tomado por defecto por Apache para mostrar la página principal “index.html”.

```
david@apacheserver:/etc/apache2/sites-available$ sudo a2ensite david.com.conf
Enabling site david.com.
To activate the new configuration, you need to run:
  systemctl reload apache2
david@apacheserver:/etc/apache2/sites-available$ sudo systemctl reload apache2
david@apacheserver:/etc/apache2/sites-available$ sudo a2ensite david.com.conf
Site david.com already enabled
```

Y ahora hay que habilitar el sitio web creado con el comando “**sudo cp default-ssl.conf david.com.conf**” (No lo muestro en la captura) que está en la ruta “/etc/apache2/sites-available/” con “sudo a2ensite david.com.conf”. Después, para aplicar los cambios “sudo systemctl reload apache2”. Me doy cuenta que vuelvo a poner el comando anterior pero este no es necesario.

```
david@apacheserver:/etc/apache2/sites-available$ cat /etc/apache2/sites-available/david.com.conf | grep -v "#"
<IfModule mod_ssl.c>
<VirtualHost *:443>
    ServerAdmin webmaster@david.com

    DocumentRoot /var/www/publica/david.com
    ServerName www.david.com
    ServerAlias empresa.com

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    SSLEngine on

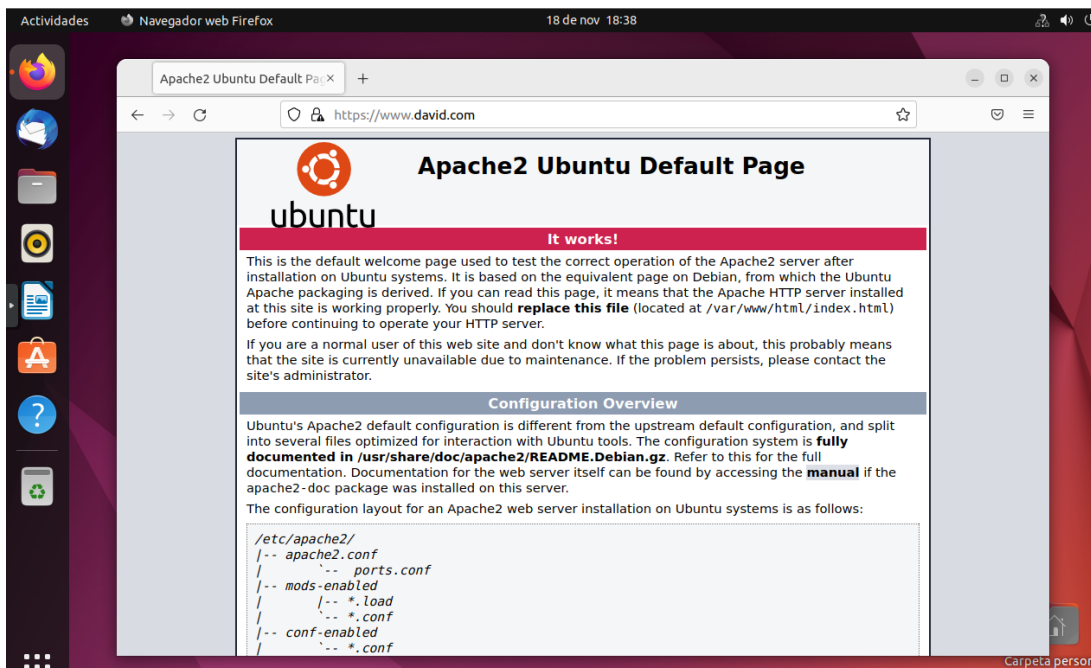
    SSLCertificateFile      /etc/ssl/certs/ssl-cert-snakeoil.pem
    SSLCertificateKeyFile   /etc/ssl/private/ssl-cert-snakeoil.key

    <FilesMatch "\.(cgi|shtml|phtml|php)$">
        SSLOptions +StdEnvVars
    </FilesMatch>
    <Directory /usr/lib/cgi-bin>
        SSLOptions +StdEnvVars
    </Directory>
    <Directory /var/www/publica/david.com>
    </Directory>

</VirtualHost>
</IfModule>
```

Aquí muestro el fichero de configuración de mi sitio público, el cual está en el servicio del puerto 443 por cualquier IP, especifico el correo, el directorio de la página ("/var/www/publica/david.com"), el nombre del servidor (www.david.com) y un alias(empresa.com), después vienen los registros, la activación SSL, los certificados y clave, y las opciones y permisos de directorio (/var/www/publica/david.com).

Verificación rápida de funcionamiento Apache desde un cliente ubuntu



Simplemente quería probar que estaba funcionando correctamente y dando servicio, pero la prueba definitiva de página pública y privada está al final del documento.

Configuración de sitio privado

```
david@apacheserver:~$ sudo mkdir -p /var/www/privada/intranet.com
david@apacheserver:~$ sudo touch /var/www/privada/intranet.com/index.html
```

Ahora creo la ruta donde almacenaré mi página privada y su archivo por defecto al acceder a ella (index.html).

```
david@apacheserver:~$ cd /etc/apache2/sites-available/
david@apacheserver:/etc/apache2/sites-available$ sudo cp default-ssl.conf intranet.david.com.conf
```

Me vuelvo a la ruta “/etc/apache2/sites-available/” y creo el fichero de configuración con el comando “sudo cp default-ssl.conf intranet.david.com.conf”, lo mismo que realicé en la pública pero con el nombre del sitio de la privada.

```
david@apacheserver:/etc/apache2/sites-available$ cat /etc/apache2/sites-available/intranet.david.com.conf | grep -v "#"
<IfModule mod_ssl.c>
  <VirtualHost *:443>
    ServerAdmin webmaster@david.com

    DocumentRoot /var/www/privada/intranet.com
    ServerName intranet.david.com
    ServerAlias intranet.com intranet

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    SSLEngine on

    SSLCertificateFile      /etc/ssl/certs/ssl-cert-snakeoil.pem
    SSLCertificateKeyFile   /etc/ssl/private/ssl-cert-snakeoil.key

    <FilesMatch "\.(cgi|shtml|phtml|php)$">
      SSLOptions +StdEnvVars
    </FilesMatch>
    <Directory /usr/lib/cgi-bin>
      SSLOptions +StdEnvVars
    </Directory>
    <Directory /var/www/privada/intranet.com>
      Options Indexes
      AuthType Basic
      AuthName "ACCESO RESTRINGIDO A LA INTRANET"
      AuthBasicProvider file
      AuthUserFile "/opt/apache2/intranet.com/passwords"
      Require valid-user
    </Directory>

  </VirtualHost>
</IfModule>
```

Aquí muestro la configuración ya terminada del sitio privado, el cual está en el servicio del puerto 443 por cualquier IP, especifico el correo, el directorio de la página (“/var/www/privada/intranet.com”), el nombre del servidor (intranet.david.com) y un alias(intranet.com o intranet), después vienen los registros, la activación SSL, los certificados y clave, y las opciones y permisos de directorio (/var/www/privada/intranet.com), además como es un sitio privado, hay que añadir una contraseña a cierto usuario para acceder, lo cual se almacenará en “/opt/apache2/intranet.com/passwords”. Más adelante configuro las contraseñas.

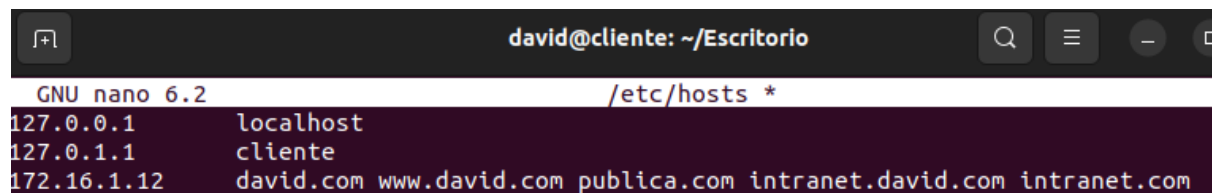
```
david@apacheserver:/etc/apache2/sites-available$ sudo a2ensite intranet.david.com.conf
Site intranet.david.com already enabled
david@apacheserver:/etc/apache2/sites-available$ sudo systemctl restart apache2
```

Ahora que la configuración ya está realizada, habilito el sitio y restarteo el servicio de apache2.

```
david@apacheserver:/etc/apache2/sites-available$ sudo htpasswd -c /opt/apache2/intranet.com/passwords david
New password:
Re-type new password:
Adding password for user david
```

Por último, creo el directorio “/opt/apache2/intranet.com/passwords” con mkdir y una vez creado, con el comando “sudo htpasswd -c /opt/apache2/intranet.com/passwords david”, me deja crear el usuario david para acceder al sitio y lo siguiente que genera es una contraseña para el usuario. Con esto concluiría la configuración del servidor.

Configuración del cliente



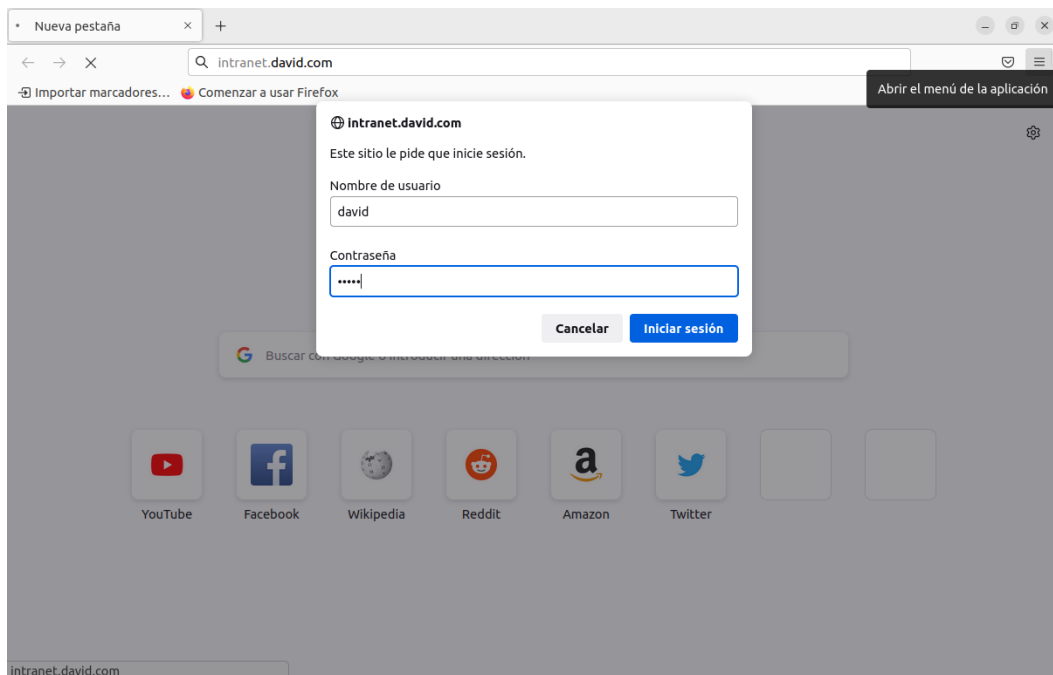
The screenshot shows a terminal window titled "david@cliente: ~/Escritorio". The terminal is running GNU nano 6.2 editing the file /etc/hosts. The content of the file is as follows:

IP Address	Hostnames
127.0.0.1	localhost
127.0.1.1	cliente
172.16.1.12	david.com www.david.com publica.com intranet.david.com intranet.com

Para que apunte a los sitios web que antes he creado tengo que ir al “/etc/hosts” del cliente y añadirle una nueva línea definiendo la IP del servidor y los sitios web a los que apunta.

Verificación del funcionamiento de los sitios desde el cliente

Privada:

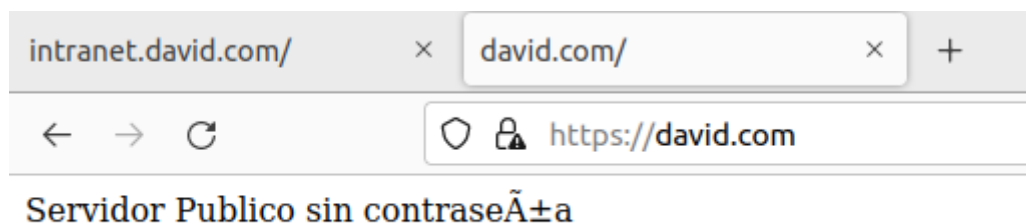


Se puede ver que al escribir el nombre de mi intranet en la url me muestra un “pop-up” de login, en el que pongo mi usuario y contraseña antes generados



Y entro a la intranet con éxito!! =)

Pública



Ahora lo mismo pero en la url introduzco david.com y... Funciona, estoy en la pública!! =)