

# Práctica 7

Hecho por Mateo Rico Iglesias - U0277172

## 1. Instalación

Cambio el hostname de la máquina linux con `hostnamectl` y con `uname -a` se puede comprobar que se ha cambiado correctamente.

```
[root@U0277172~]#hostnamectl set-hostname linux.as.local
[root@U0277172~]#uname -a
Linux linux.as.local 5.14.0-162.6.1.el9_1.x86_64 #1 SMP PREEMPT_DYNAMIC Tue Nov 15 07:49:10 EST 2022
x86_64 x86_64 x86_64 GNU/Linux
```

Podemos comprobar que desde la máquina WS2022 ya se puede resolver el ping a la dirección *linux.as.local*

```
C:\Users\Administrador>ping linux.as.local

Haciendo ping a linux.as.local [192.168.56.100] con 32 bytes de datos:
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 192.168.56.100:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
```

Al igual que en la anterior, en la máquina W10 también se puede

```
C:\Users\uoxxxxxx>ping linux.as.local

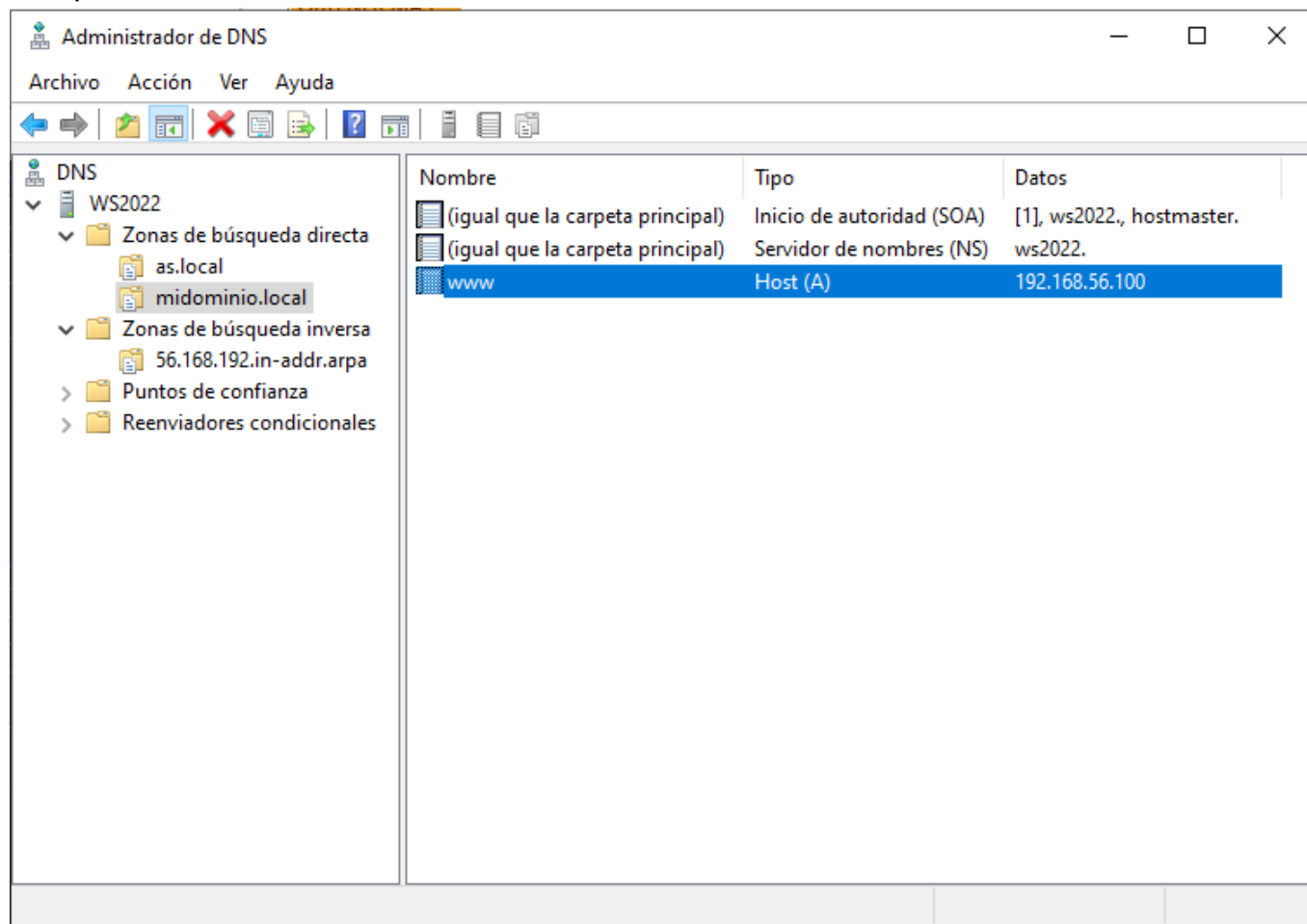
Haciendo ping a linux.as.local [192.168.56.100] con 32 bytes de datos:
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 192.168.56.100:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
```

Y por último si pruebo desde la máquina linux también obtengo respuesta del ping

```
[root@U0277172~]#ping linux.as.local
PING linux.as.local (192.168.56.100) 56(84) bytes of data.
64 bytes from w10.as.local (192.168.56.100): icmp_seq=1 ttl=64 time=0.028 ms
64 bytes from linux.as.local (192.168.56.100): icmp_seq=2 ttl=64 time=0.057 ms
^C
--- linux.as.local ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1006ms
rtt min/avg/max/mdev = 0.028/0.042/0.057/0.014 ms
```

Al igual que la práctica anterior creo otra zona de búsqueda directa en el dns de la máquina WS2022 con la terminación *midominio.local* y teniendo el host *www*. Esto me permitirá redirigir el tráfico a este nombre de dominio a la máquina linux.



Después de comprobarlo el servicio apache no está instalado por lo que lo instalo con `dnf install httpd`

```

Instalando      : apr-util-openssl-1.6.1-20.el9.x86_64      3/12
Instalando      : apr-util-1.6.1-20.el9.x86_64             4/12
Instalando      : httpd-tools-2.4.53-7.el9_1.1.x86_64      5/12
Instalando      : mailcap-2.1.49-5.el9.noarch               6/12
Ejecutando scriptlet: httpd-filesystem-2.4.53-7.el9_1.1.noarch 7/12
Instalando      : httpd-filesystem-2.4.53-7.el9_1.1.noarch 7/12
Instalando      : httpd-core-2.4.53-7.el9_1.1.x86_64       8/12
Instalando      : mod_http2-1.15.19-2.el9.x86_64           9/12
Instalando      : mod_lua-2.4.53-7.el9_1.1.x86_64          10/12
Instalando      : almalinux-logos-httpd-90.5.1-1.1.el9.noarch 11/12
Instalando      : httpd-2.4.53-7.el9_1.1.x86_64            12/12
Ejecutando scriptlet: httpd-2.4.53-7.el9_1.1.x86_64       12/12
[ 1295.697353] systemd-rc-local-generator[14891]: /etc/rc.d/rc.local is not marked executable, skipping.
Verificando     : almalinux-logos-httpd-90.5.1-1.1.el9.noarch 1/12
Verificando     : apr-1.7.0-11.el9.x86_64                 2/12
Verificando     : apr-util-1.6.1-20.el9.x86_64            3/12
Verificando     : apr-util-bdb-1.6.1-20.el9.x86_64        4/12
Verificando     : apr-util-openssl-1.6.1-20.el9.x86_64    5/12
Verificando     : httpd-2.4.53-7.el9_1.1.x86_64           6/12
Verificando     : httpd-core-2.4.53-7.el9_1.1.x86_64      7/12
Verificando     : httpd-filesystem-2.4.53-7.el9_1.1.noarch 8/12
Verificando     : httpd-tools-2.4.53-7.el9_1.1.x86_64     9/12
Verificando     : mod_http2-1.15.19-2.el9.x86_64          10/12
Verificando     : mod_lua-2.4.53-7.el9_1.1.x86_64         11/12
Verificando     : mailcap-2.1.49-5.el9.noarch              12/12

Instalado:
  almalinux-logos-httpd-90.5.1-1.1.el9.noarch      apr-1.7.0-11.el9.x86_64
  apr-util-1.6.1-20.el9.x86_64                   apr-util-bdb-1.6.1-20.el9.x86_64
  apr-util-openssl-1.6.1-20.el9.x86_64            httpd-2.4.53-7.el9_1.1.x86_64
  httpd-core-2.4.53-7.el9_1.1.x86_64              httpd-filesystem-2.4.53-7.el9_1.1.noarch
  httpd-tools-2.4.53-7.el9_1.1.x86_64             mailcap-2.1.49-5.el9.noarch
  mod_http2-1.15.19-2.el9.x86_64                  mod_lua-2.4.53-7.el9_1.1.x86_64

¡Listo!
[root@U0277172~]#

```

Activo el sistema con el systemctl para poder usarlo

```

[root@U0277172~]#systemctl start httpd
[root@U0277172~]#systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Wed 2023-03-29 12:45:51 CEST; 4s ago
     Docs: man:httpd.service(8)
  Main PID: 1366 (httpd)
    Status: "Started, listening on: port 80"
     Tasks: 213 (limit: 11064)
    Memory: 43.7M
       CPU: 148ms
    CGroup: /system.slice/httpd.service
            └─1366 /usr/sbin/httpd -DFOREGROUND
              └─1367 /usr/sbin/httpd -DFOREGROUND
                └─1368 /usr/sbin/httpd -DFOREGROUND
                  └─1369 /usr/sbin/httpd -DFOREGROUND
                    └─1370 /usr/sbin/httpd -DFOREGROUND

mar 29 12:45:50 linux.as.local systemd[1]: Starting The Apache HTTP Server...
mar 29 12:45:51 linux.as.local systemd[1]: Started The Apache HTTP Server.
mar 29 12:45:51 linux.as.local httpd[1366]: Server configured, listening on: port 80

```

Y actualizo las reglas del firewall para no tener problemas

```

[root@U0277172~]#firewall-cmd --zone=internal --permanent --add-service=http
success
[root@U0277172~]#firewall-cmd --reload
success

```

Tras hacer esto vuelvo a probar desde las tres máquinas que el ping funciona

correctamente a este nuevo nombre de dominio

```
[root@U0277172~]#ping www.midominio.local
PING www.midominio.local (192.168.56.100) 56(84) bytes of data.
64 bytes from w10.as.local (192.168.56.100): icmp_seq=1 ttl=64 time=0.038 ms
64 bytes from linux.as.local (192.168.56.100): icmp_seq=2 ttl=64 time=0.128 ms
64 bytes from www.midominio.local (192.168.56.100): icmp_seq=3 ttl=64 time=0.084 ms
64 bytes from w10.as.local (192.168.56.100): icmp_seq=4 ttl=64 time=0.142 ms
^C
--- www.midominio.local ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 0.038/0.098/0.142/0.040 ms
```

```
C:\Users\Administrador>ping www.midominio.local

Haciendo ping a www.midominio.local [192.168.56.100] con 32 bytes de datos:
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64

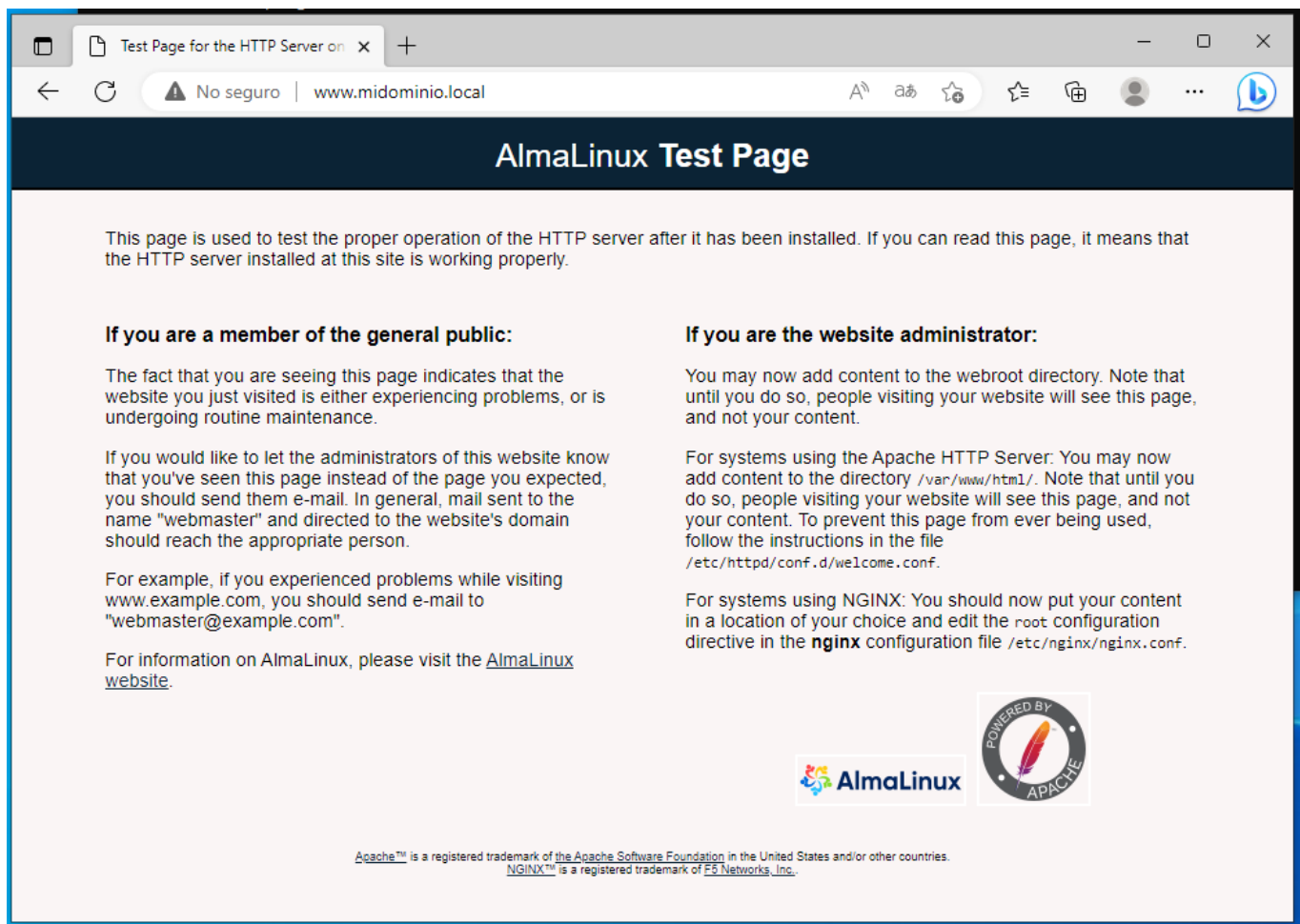
Estadísticas de ping para 192.168.56.100:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
```

```
C:\Users\uoxxxxxx>ping www.midominio.local

Haciendo ping a www.midominio.local [192.168.56.100] con 32 bytes de datos:
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.56.100: bytes=32 tiempo=1ms TTL=64

Estadísticas de ping para 192.168.56.100:
    Paquetes: enviados = 3, recibidos = 3, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 1ms, Media = 0ms
```

Y por último podemos ver que desde la máquina Windows ya se puede acceder a este dominio correctamente, mostrándose la página por defecto



Tras crear el archivo index.html con el comando `vi /var/www/html index.html` y editarlo con el código html que se puede encontrar en la práctica, si trato de volver a acceder al sitio se puede ver que ya se ha actualizado correctamente



## 2. Configuración de las páginas web de los usuarios

En este punto se nos permitirá crear páginas web independientes para cada usuario de la máquina linux.

Primero accedo al usuario `asuser` que ya tengo creado de la anterior práctica, para esto he tenido que usar el comando `usermod -s /bin/bash asuser` ya que en la anterior práctica debido a que el usuario solo era necesario para el servicio samba lo había creado sin bash con un `/sbin/nologin`. Tras hacer esto ya puedo acceder al usuario con normalidad o en mi caso usar el `su asuser` desde el usuario root para acceder más rápidamente.

Antes de acceder a este usuario voy a realizar una serie de cambios:

- Edito el fichero situado en `/etc/httpd/conf.d/userdir.conf` para comentar la línea que pone *UserDir disabled* y descomentar la línea *UserDir public\_html* lo que me permitirá crear diferentes html por usuario.

```
#
<IfModule mod_userdir.c>
#
# UserDir is disabled by default since it can confirm the presence
# of a username on the system (depending on home directory
# permissions).
#
# UserDir disabled

#
# To enable requests to ~/user/ to serve the user's public_html
# directory, remove the "UserDir disabled" line above, and uncomment
# the following line instead:
#
UserDir public_html
</IfModule>
```

- Aplico los permisos necesarios con `chmod 711 /home/asuser` al directorio, ejecuto `setsebool -P httpd_read_user_content on` y `setsebool -P httpd_enable_homedirs on` para permitir que Apache pueda leer contenidos localizados en los directorios de inicio de los usuarios locales y habilitar el uso del directorio `/public_html` en los usuarios, que como podemos ver en la anterior captura es el directorio de donde se buscará la información para la

página de cada usuario.

```
[root@U0277172 conf.d]#chmod 711 /home/asuser
[root@U0277172 conf.d]#setsebool -P httpd_read_user_content on
[ 531.231085] SELinux: Converting 337 SID table entries...
[ 531.234260] SELinux: policy capability network_peer_controls=1
[ 531.234422] SELinux: policy capability open_perms=1
[ 531.234568] SELinux: policy capability extended_socket_class=1
[ 531.234708] SELinux: policy capability always_check_network=0
[ 531.234842] SELinux: policy capability cgroup_seclabel=1
[ 531.234983] SELinux: policy capability mmp_nosuid_transition=1
[ 531.235122] SELinux: policy capability genfs_seclabel_symlinks=0
[root@U0277172 conf.d]#setsebool -P httpd_enable_homedirs on
[ 546.089971] SELinux: Converting 337 SID table entries...
[ 546.093845] SELinux: policy capability network_peer_controls=1
[ 546.094000] SELinux: policy capability open_perms=1
[ 546.094131] SELinux: policy capability extended_socket_class=1
[ 546.094285] SELinux: policy capability always_check_network=0
[ 546.094422] SELinux: policy capability cgroup_seclabel=1
[ 546.094549] SELinux: policy capability mmp_nosuid_transition=1
[ 546.094676] SELinux: policy capability genfs_seclabel_symlinks=0
```

- Ahora si ya entro como el usuario *asuser*, en este caso yo entraré con un `su asuser` desde el usuario root que estaba usando y creo en el directorio con `mkdir ./public_html` la carpeta necesaria para la página. Tras esto con el `chmod 755 -R public_html` daré los permisos necesarios en esa carpeta para continuar el proceso.

```
[asuser@U0277172 ~]#cd /home/asuser/
[asuser@U0277172 ~]#mkdir ./public_html
[asuser@U0277172 ~]#ls
public_html
```

```
[asuser@U0277172 ~]#chmod 755 -R public_html/
```

- Dentro de esta carpeta generaré el dichero `index.html` que se mostrará en la página del usuario *asuser* con `vi index.html`

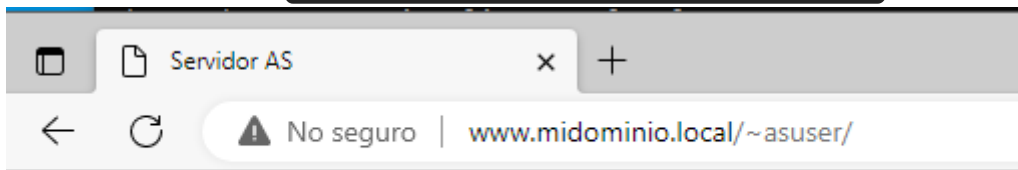
```
Archivo  Máquina  Ver  Entradas  Dispositivos  Ayuda
<?DOCTYPE html>
<head
    <meta charest="utf-8" />
    <title>Servidor AS</title>
</head>
<body>
    <h1>Administración de Sistemas</h1>
    <p>Usuario asuser</p>
</body>
</html>
```

```
[asuser@U0277172 public_html]#ls
index.html
```

- Por último reinicio el servicio con `systemctl restart httpd` y si pruebo desde la máquina W10 ya podré acceder a la página personal del usuario



con la dirección `http://www.midominio.local/~asuser`



## Administración de Sistemas

Usuario asuser

### 3. Configuración del servidor Apache

#### 3.a Ubicación

Primero creo la nueva ubicación para el servidor, en este caso */as/web*

```
[root@U0277172~]#mkdir /as
[root@U0277172~]#mkdir /as/web
```

Sustituyo la ruta por defecto en el documento *httpd.conf* a la ruta de la nueva carpeta

```
#
DocumentRoot "/as/web"

#
# Relax access to content within /var/www.
#
<Directory "/as/web">
    Options FollowSymLinks
    AllowOverride None
    # Allow open access:
    Require all granted
</Directory>
```

Y por último restauro el servicio httpd para aplicar los cambios

```
[root@U0277172~]#chcon -R -h -t httpd_sys_content_t /as/web/
```

#### 3.b ServerName

Modifico las directivas *ServerAdmin* y *ServerName* del archivo con la siguiente información como se me pide en la documentación. Primero cambio el *ServerAdmin* a mi UO con el dominio que utiliza mi servidor y después el *ServerName* a el dominio de mi servidor más el *puerto 9999* que he puesto nuevo



```
#  
# ServerAdmin: Your address, where problems with the server should be  
# e-mailed. This address appears on some server-generated pages, such  
# as error documents. e.g. admin@your-domain.com  
#  
ServerAdmin U0277172@www.midominio.local  
  
#  
# ServerName gives the name and port that the server uses to identify itself.  
# This can often be determined automatically, but we recommend you specify  
# it explicitly to prevent problems during startup.  
#  
# If your host doesn't have a registered DNS name, enter its IP address here.  
#  
#ServerName www.midominio.local:9999
```

Cambio el *Listen* para que escuche al nuevo puerto que quiero utilizar

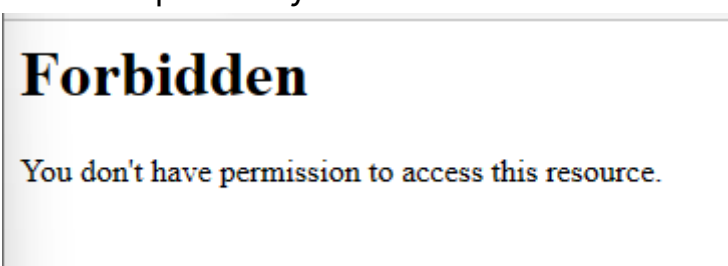
```
# page for more information  
#  
#Listen 12.34.56.78:80  
Listen 9999
```

Y podemos ver que tras reiniciar el servicio httpd y copiar mi *index.html* al nuevo directorio la página ya funciona con el puerto nuevo



### 3.c Repositorios

Renombro el archivo que moví a la nueva carpeta a índice.html lo que provoca que se vuelva a mostrar la página por defecto del servidor. Tras hacer esto comento por completo el archivo welcome.conf. Al ser este el archivo del que se saca la página por defecto entonces no se puede mostrar ninguna página tras hacer esto, por lo que obtengo un error de que no puedo acceder al recurso que estoy solicitando.



Después de probar esto, vuelvo al archivo de configuración de httpd y modifico el código donde había puesto el *as/web* y añado un **Indexes** en la primera línea. Tras recargar la página obtengo una página que me muestra todo lo contenido en el directorio base que tengo seleccionado, en este caso el *as/web* permitiéndome así seleccionar el html índices que tengo en la carpeta y visualizándolo.



Por último trato de acceder a una página que no existe, por ejemplo, trato de acceder al archivo *hola.html* el cual no existe. Si entro al archivo *access\_log* podemos ver que se ha guardado mi petición en la que trato de acceder a esta página.

```
192.168.56.110 - - [01/Apr/2023:14:19:41 +0200] "GET /hola.html HTTP/1.1" 404 196 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/111.0.0.0 Safari/537.36 Edg/111.0.1661.54"
```

## 4. Hosts Virtuales

Configuro un nuevo alias para *midominio.local* llamado *otraempresa*

 otraempresa	Alias (CNAME)	
 www	Host (A)	192.168.56.100

Para configurar mi servidor con un nuevo host virtual he hecho lo siguiente:

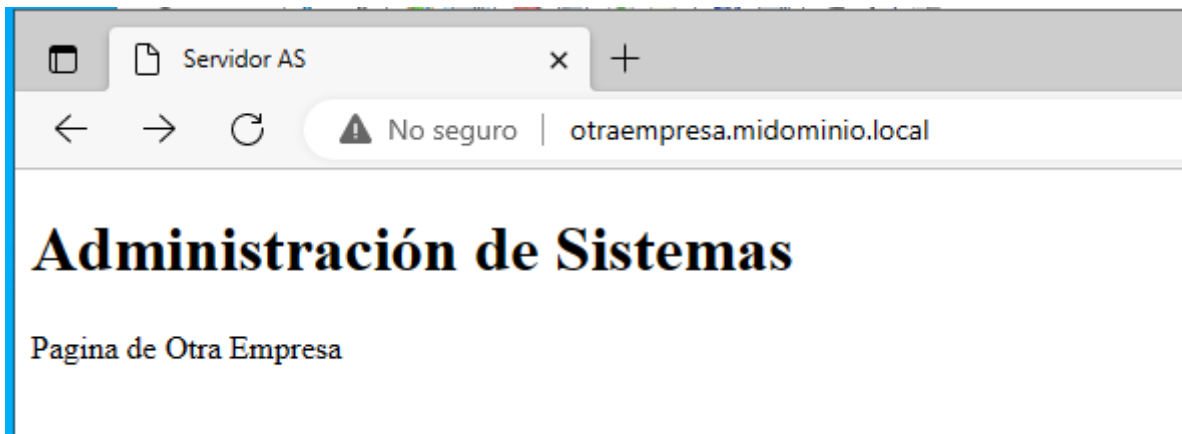
- Creo un nuevo directorio **as/web/** al que llamaré *otraempresa* y después muevo el anterior html a un directorio dentro de esta carpeta llamado *www* para tener todo más organizado.
- Creo dos directorios en **/etc/httpd/** uno llamado *sites-available* y otro llamado *sites-enabled* en los cuales haré la configuración para todos los hosts virtuales.

- Entro en el fichero de configuración de httpd y añado al final la siguiente línea `IncludeOptional sites-enabled/*.conf`
- Aquí en la carpeta sites-available crearé primero todos los ficheros de configuración y después los pasaré a sites-enabled para que empiecen a funcionar. La estructura en estos es una estructura simple

```
<VirtualHost *:80>
    ServerName otraempresa.midominio.local
    DocumentRoot /as/web/otraempresa
</VirtualHost>
```

```
<VirtualHost *:80>
    ServerName www.midominio.local
    DocumentRoot /as/web/www
</VirtualHost>
```

- Y después de esto con el comando `sudo ln -s /etc/httpd/sites-available/otraempresa.conf /etc/httpd/sites-enabled/otraempresa.conf` y `sudo ln -s /etc/httpd/sites-available/www.conf /etc/httpd/sites-enabled/www.conf` activo estos host virtuales
- Tras reiniciar la máquina para reiniciar el servicio y comprobar que está correctamente ejecutandose podemos ver que ya podemos entrar en el nuevo host virtual



## 5. Autenticación

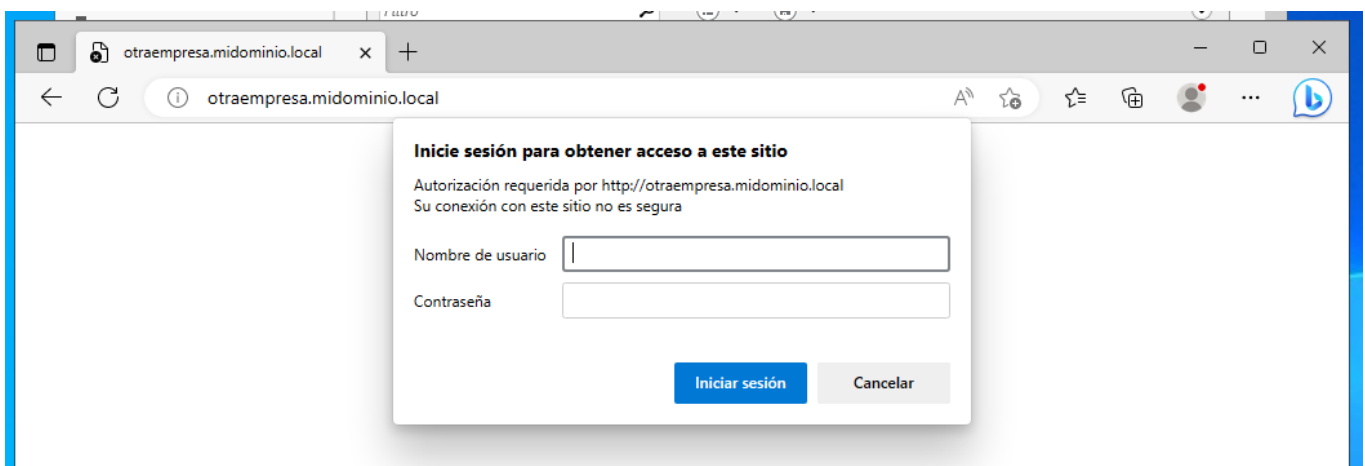
Primero añado la línea *AllowOverride AuthConfig* a la parte directory del virtual host dedicado a la dirección *otraempresa.midominio.local*

```
<VirtualHost *:80>
    ServerName otraempresa.midominio.local
    DocumentRoot /as/web/otraempresa
    <Directory "/as/web/otraempresa">
        AllowOverride AuthConfig
    </Directory>
</VirtualHost>
```

Después de esto creo el archivo *.htaccess* en la carpeta `as/web/otraempresa` donde se encuentra mi html para este host virtual y le añado lo siguiente

```
AuthType Basic
AuthName "Area Restringida"
AuthUserFile /etc/httpd/password.file
AuthGroupFile /dev/null Require valid-user
```

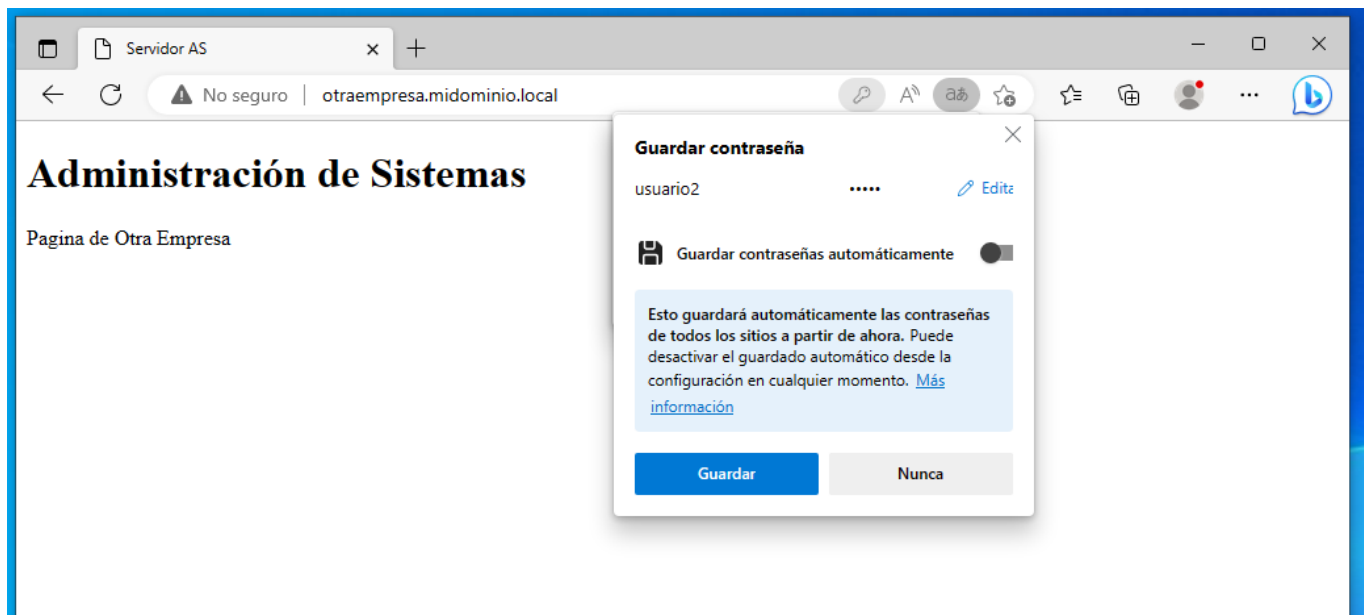
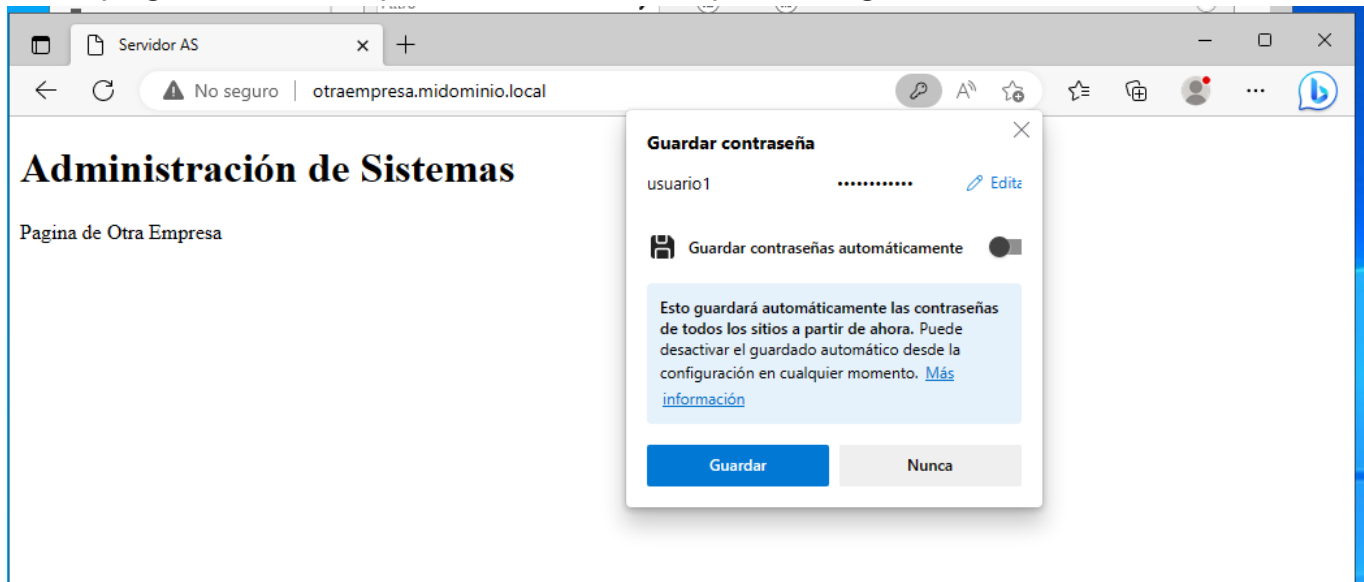
Tras esto la página ya me pide un usuario y una contraseña para poder visualizarla



Después de hacer esto tengo que añadir unos usuarios y contraseñas al archivo *password.file*, de esta manera si introduzco los datos de los nuevos usuarios podré ver la página. Esto lo haré con los comandos siguientes

```
htpasswd -c /etc/httpd/password.file usuario1  
htpasswd /etc/httpd/password.file usuario2
```

Tras hacer esto ya puedo iniciar sesión con ambos usuarios y ver el contenido de la página como se puede ver en las capturas siguientes



## 6. Servidor Proxy - squid

Primero de nada instalo el squid con el comando `dnf install squid`, con esto ya podré empezar este apartado.

Tras instalarlo inicio el servicio y lo habilito para que se inicie automáticamente con cada arranque de la máquina linux

```
[root@U0277172 ~]#systemctl start squid
```

```
[root@U0277172~]#systemctl enable squid
Created symlink /etc/systemd/system/multi-user.target.wants/squid.service → /usr/lib/systemd/system/squid.service.
[ 962.481436] systemd-rc-local-generator[4203]: /etc/rc.d/rc.local is not marked executable, skipping.
```

Añado el rango de red donde se puede encontrar el servidor al fichero de configuración de squid y descomento la línea del cache\_dir para que se me guarden todos los movimientos de la red que pasen a través del proxy

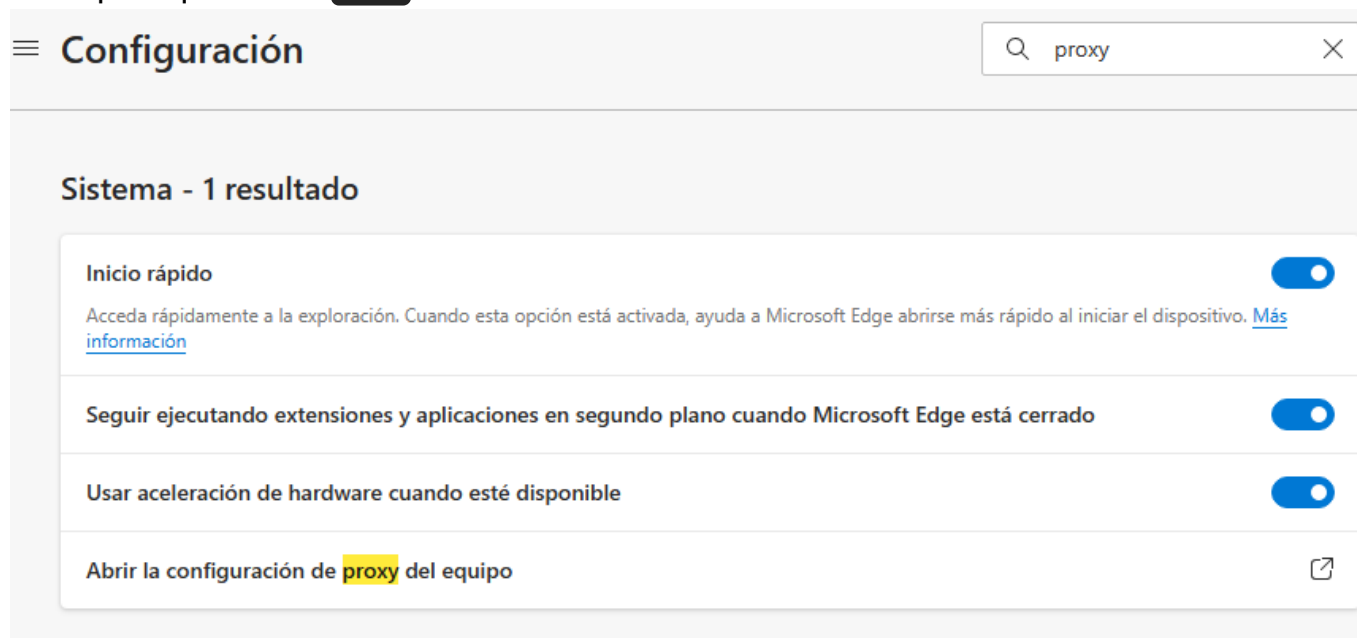
```
acl localnet src 10.0.0.0/24
acl localnet src 192.168.56.0/24
```

```
# Uncomment and adjust the following to add a disk cache directory.
cache_dir ufs /var/spool/squid 100 16 256
```

Creo una excepción en el firewall para que deje pasar todo lo relacionado con el puerto de squid.

```
[root@U0277172~]#firewall-cmd --zone=internal --add-service=squid
success
```

Después voy a la configuración del navegador, busco la opción de proxy y al clickar me manda a la configuración del proxy de Windows. En la pestaña que me sale activaré la opción de *Usar servidor proxy* y rellenaré los campos con el proxy que voy a usar, en este caso la dirección `www.midominio.local` y el puerto de squid que es el `3128`



Usar servidor proxy

☒ Activado

Dirección

http://www.midominio.lo

Puerto

3128

Tras hacer esto y probar a navegar en la máquina Windows con el comando tail podremos ir viendo que el proxy está funcionando y está pasando el tráfico por la máquina linux.

```
[root@00277172~]#tail -f /var/log/squid/access.log
1680445937.079      175 192.168.56.110 TCP_TUNNEL/200 214284 CONNECT prod-streaming-video-msn-com.akamaized.net:443 - HIER_DIRECT/96.16.84.23 -
1680445937.407      150 192.168.56.110 TCP_TUNNEL/200 39 CONNECT browser.events.data.msn.com:443 - HIER_DIRECT/13.89.179.9 -
1680445937.407      150 192.168.56.110 TCP_TUNNEL/200 39 CONNECT browser.events.data.msn.com:443 - HIER_DIRECT/13.89.179.9 -
1680445939.356      333 192.168.56.110 TCP_MISS/302 1700 GET http://www.google.es/ - HIER_DIRECT/142.250.200.131 text/html
1680445940.207      404 192.168.56.110 TCP_TUNNEL/200 4542 CONNECT www.gstatic.com:443 - HIER_DIRECT/142.250.184.163 -
1680445947.243     10211 192.168.56.110 TCP_TUNNEL/200 637 CONNECT r.bing.com:443 - HIER_DIRECT/23.200.66.148 -
1680445947.244     10213 192.168.56.110 TCP_TUNNEL/200 637 CONNECT r.bing.com:443 - HIER_DIRECT/23.200.66.148 -
1680445967.256     23348 192.168.56.110 TCP_TUNNEL/200 41826 CONNECT img-s-msn-com.akamaized.net:443 - HIER_DIRECT/96.16.84.28 -
1680445970.510         1 192.168.56.110 TCP_MISS/403 506 GET http://www.midominio.local/ - HIER_DIRECT/192.168.56.100 text/html
1680445970.532         0 192.168.56.110 TCP_MISS/403 506 GET http://www.midominio.local/favicon.ico - HIER_DIRECT/192.168.56.100 text/html
```