

## ASR - Práctica 7

UO282440

### Servidores Web en Linux: Apache

#### 1. Instalación

- La máquina Linux debe llamarse linux.as.local (hostnamectl set-hostname linux.as.local y para comprobarlo uname -a). Comprueba con ping que tienes acceso a la red. En esta máquina vamos a montar un servidor web apache.

```
[uo282440@localhost ~]# hostnamectl set-hostname linux.as.local
[uo282440@localhost ~]# uname -a
Linux linux.as.local 5.14.0-583.11.1.el9_5.x86_64 #1 SMP PREEMPT_DYNAMIC Tue Nov 12 09:26:13 EST 2024 x86_64 x86_64 x86_64 GNU/Linux
[uo282440@localhost ~]#
[uo282440@localhost ~]# ping www.google.com
PING www.google.com (216.58.209.68) 56(84) bytes of data.
64 bytes from mad07s22-in-f4.1e100.net (216.58.209.68): icmp_seq=1 ttl=63 time=15.3 ms
64 bytes from waw02s06-in-f68.1e100.net (216.58.209.68): icmp_seq=2 ttl=63 time=46.3 ms
64 bytes from 216.58.209.68: icmp_seq=3 ttl=63 time=19.8 ms

--- www.google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 6243ms
rtt min/avg/max/mdev = 15.312/27.145/46.288/13.656 ms
[uo282440@localhost ~]# nslookup www.uniovi.es
Server:      10.0.2.3
Address:     10.0.2.3#53

Non-authoritative answer:
www.uniovi.es canonical name = crisb101.sic233.uniovi.es.
Name:   crisb101.sic233.uniovi.es
Address: 156.35.233.101
[uo282440@localhost ~]#
```

- Arranca WS2022 y comprueba que se resuelve la dirección linux.as.local desde las máquinas Linux y W10. Analiza y cambia lo necesario para que también se resuelva la dirección desde la propia WS2022.

Desde linux:

```
[uo282440@localhost ~]# nslookup linux.as.local
Server:      192.168.56.101
Address:     192.168.56.101#53

Name:   linux.as.local
Address: 192.168.56.100
[uo282440@localhost ~]#
```

Desde Windows 10:

```

C:\Users\uo282440>nslookup linux.as.local
Servidor:  UnKnown
Address:  192.168.56.101

Nombre:  linux.as.local
Address: 192.168.56.100

C:\Users\uo282440>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=1ms 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
Control-C
^C
C:\Users\uo282440>

```

Desde Windows Server:

```

Servidor local

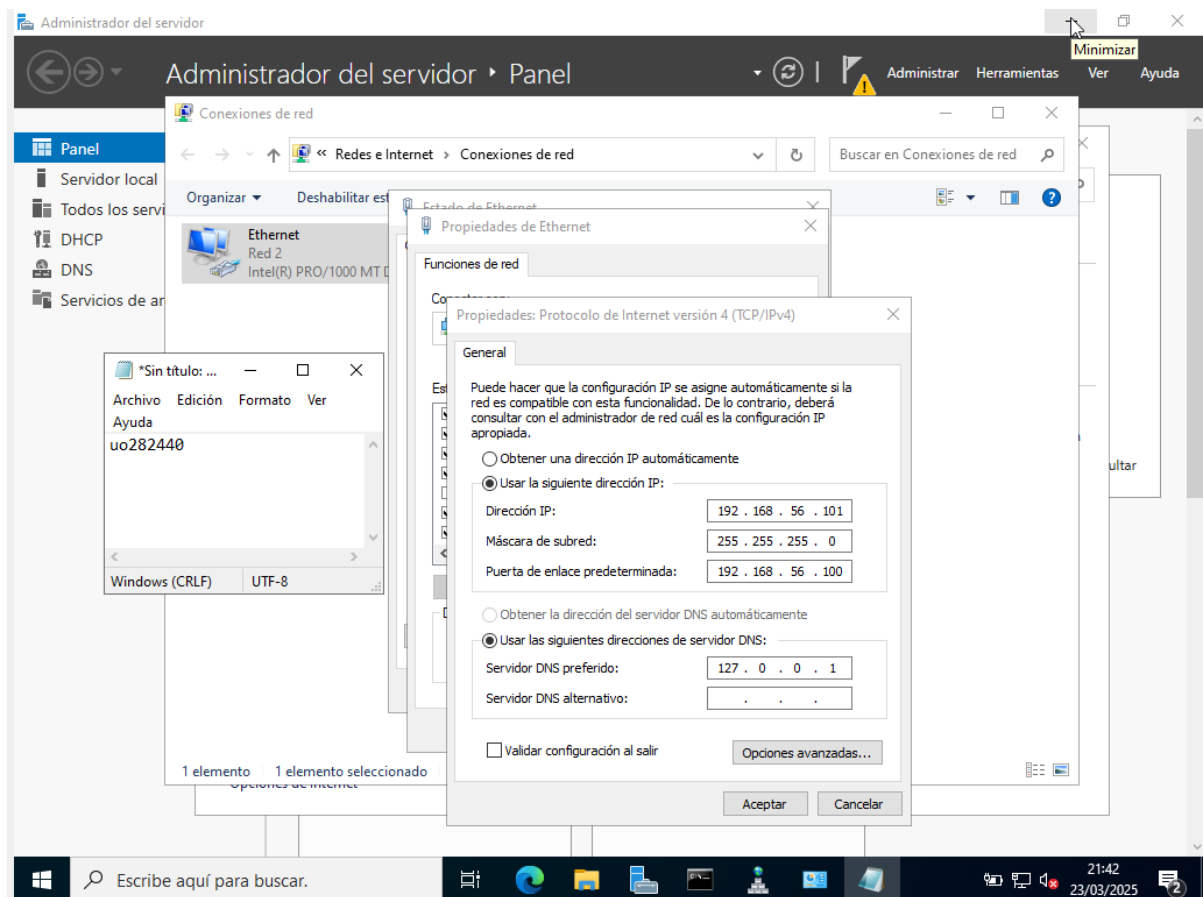
C:\Users\Administrador>nslookup linux.as.local
Servidor:  localhost
Address:  127.0.0.1

Nombre:  linux.as.local
Address: 192.168.56.100

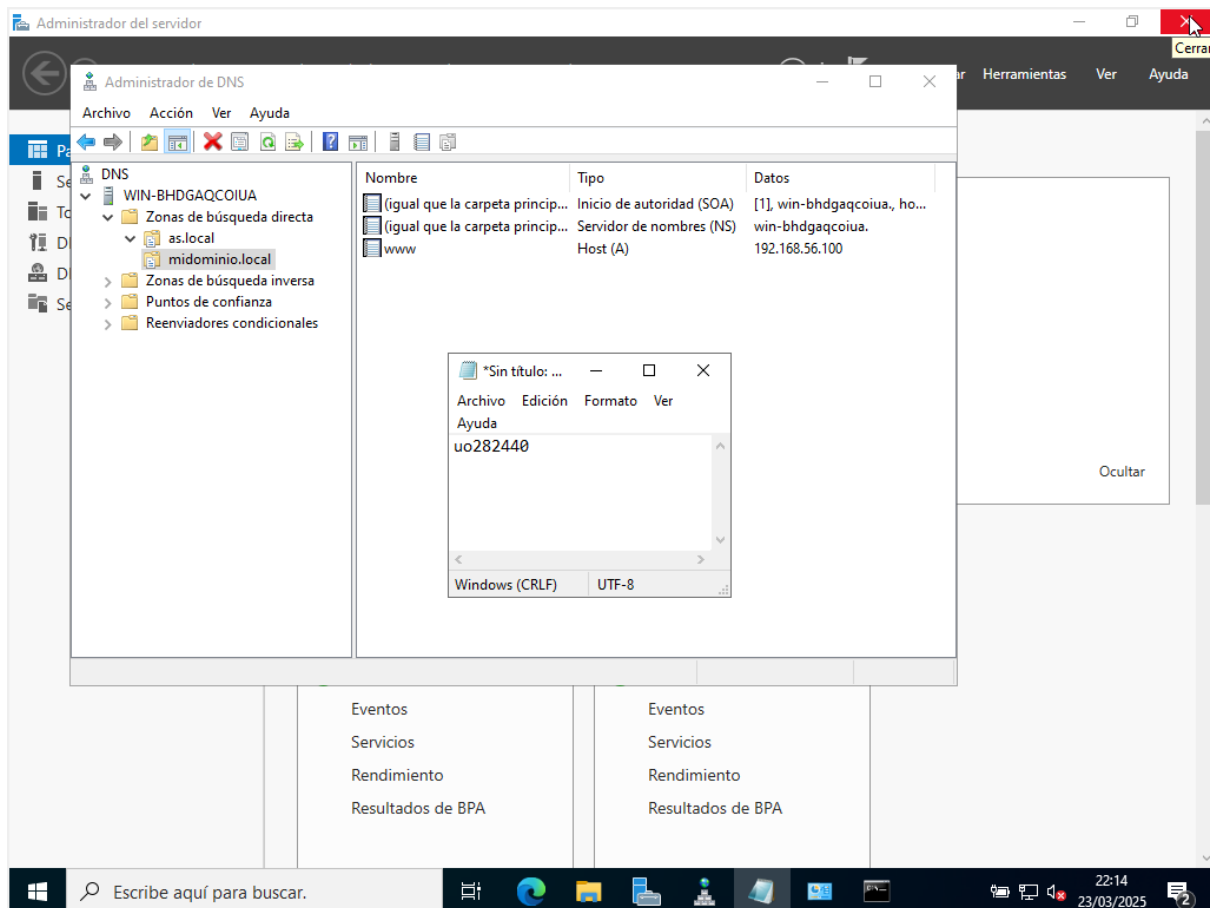
C:\Users\Administrador>

```

Para que funcionase hubo que modificar el servidor DNS para que apuntase a sí mismo:



- En la zona del servidor DNS añade `www.midominio.local` con la misma dirección IP que la máquina Linux (debes crear una nueva zona `midominio.local`). Comprueba que tanto la máquina Linux como W10 resuelven correctamente la dirección asignada a `www.midominio.local` (si toman del WS2022 la dirección del servidor DNS y está todo bien configurado, ya debería ser así).



Desde windows 10:

```
C:\> Símbolo del sistema

C:\Users\uo282440>nslookup www.midominio.local
Servidor:  UnKnown
Address:  192.168.56.101

Nombre:  www.midominio.local.as.local
Address:  192.168.56.100

C:\Users\uo282440>
```

Desde linux:

```
[luc282440@linux ~]# nslookup www.midominio.local
Server:      192.168.56.101
Address:     192.168.56.101#53

Name:   www.midominio.local
Address: 192.168.56.100

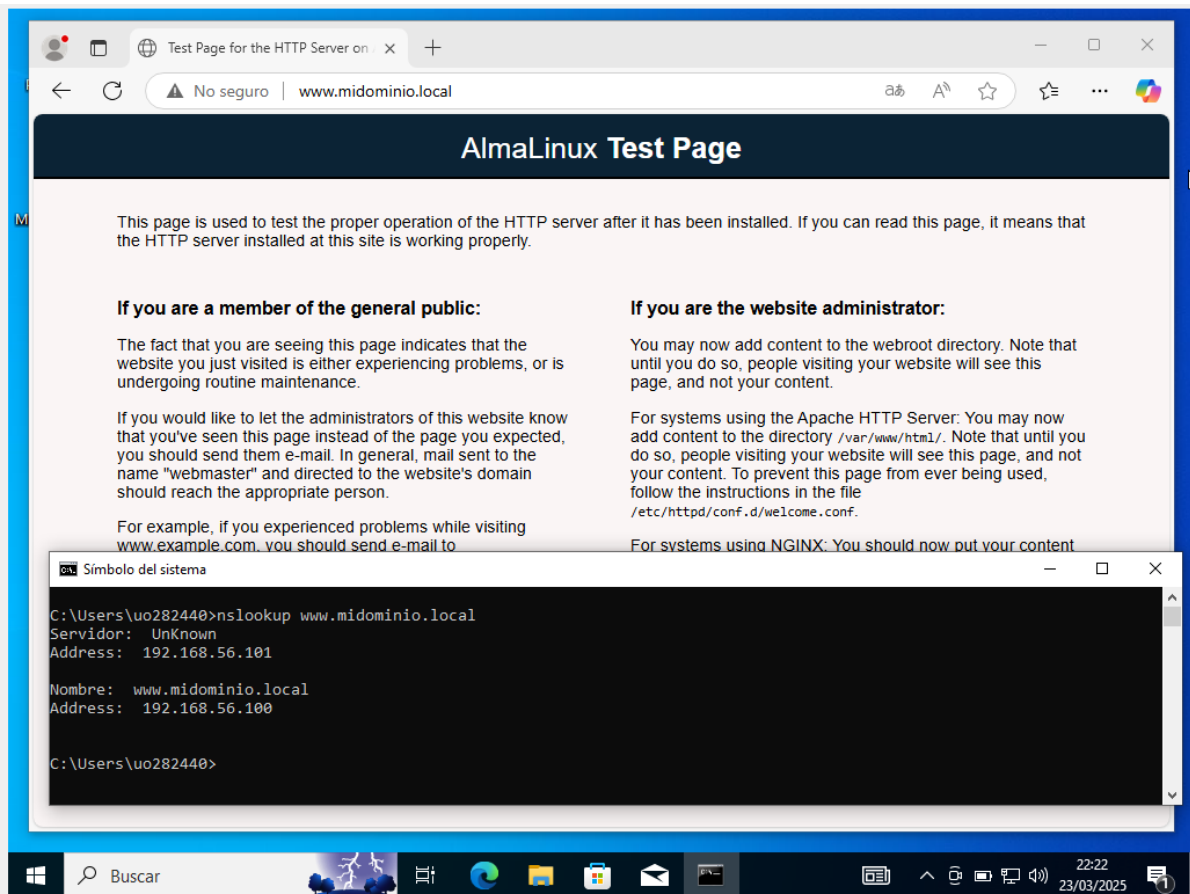
[luc282440@linux ~]# _
```

- Comprueba que el Apache está instalado en la máquina Linux, y si no es así instálalo con `dnf install httpd`. Arranca el servicio `httpd` en la máquina Linux. Añade una nueva regla al firewall para permitir las conexiones `http`:

```
[luc282440@linux ~]# systemctl start httpd
[luc282440@linux ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
   Active: active (running) since Sun 2025-03-23 22:19:03 CET; 10s ago
     Docs: man:httpd.service(8)
  Main PID: 1871 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
    Tasks: 177 (limit: 11073)
   Memory: 30.2M
      CPU: 193ms
   CGroup: /system.slice/httpd.service
           └─1871 /usr/sbin/httpd -DFOREGROUND
             └─1872 /usr/sbin/httpd -DFOREGROUND
               └─1873 /usr/sbin/httpd -DFOREGROUND
                 └─1874 /usr/sbin/httpd -DFOREGROUND
                   └─1875 /usr/sbin/httpd -DFOREGROUND

mar 23 22:19:03 linux.as.local systemd[1]: Starting The Apache HTTP Server...
mar 23 22:19:03 linux.as.local httpd[1871]: Server configured, listening on: port 80
mar 23 22:19:03 linux.as.local systemd[1]: Started The Apache HTTP Server.
[luc282440@linux ~]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service + /usr/lib/systemd/system/httpd.service.
[ 1299.019000] systemd-rc-local-generator[2071]: /etc/rc.d/rc.local is not marked executable, skipping.
[luc282440@linux ~]# firewall-cmd --zone=internal --permanent --add-service=http
success
[luc282440@linux ~]# firewall-cmd --reload
success
[luc282440@linux ~]# _
```

Lanza un navegador en WS2022 (o W10) y accede a `http://www.midominio.local`. ¿Ves la página de prueba? Si no es así, revisa los pasos anteriores hasta que lo consigas.



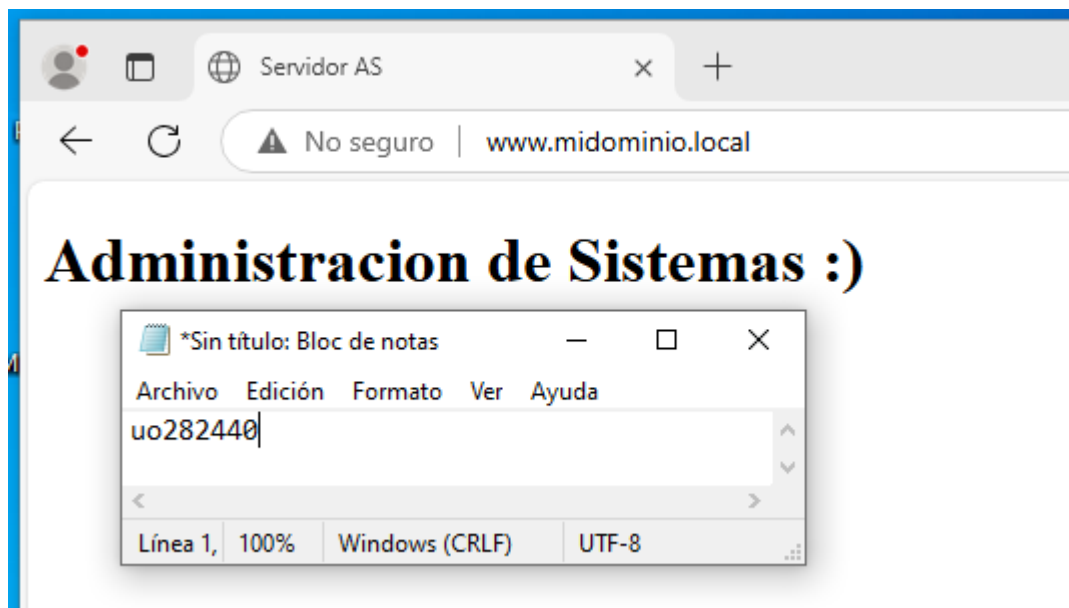
Crea dentro del directorio /var/www/html un archivo index.html con el siguiente contenido y comprueba que puedes visualizarlo correctamente

```
GNU nano 5.6.1
<!DOCTYPE html>

    <head>
        <meta charset="utf-8"/>
        <title>Servidor AS</title>
    </head>

    <body>
        <h1>Administracion de Sistemas :)</h1>
    </body>

</html>
```



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

- Crea, si no lo has hecho ya, el usuario asuser.
- Edita el fichero /etc/httpd/conf.d/userdir.conf o Comenta la línea (añade # al principio) 'UserDir disabled'. Quita el comentario (#) a la línea 'UserDir public\_html'

```

GNU nano 5.6.1 /etc/httpd/conf.d/userdir.conf
#
# UserDir: The name of the directory that is appended onto a user's home
# directory if a ~user request is received.
#
# The path to the end user account 'public_html' directory must be
# accessible to the webserver userid. This usually means that ~userid
# must have permissions of 711, ~userid/public_html must have permissions
# of 755, and documents contained therein must be world-readable.
# Otherwise, the client will only receive a "403 Forbidden" message.
#
<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>

#
# Control access to UserDir directories. The following is an example
# for a site where these directories are restricted to read-only.
#
<Directory "/home/*/public_html">
    AllowOverride FileInfo AuthConfig Limit Indexes
    Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
    Require method GET POST OPTIONS
</Directory>

```

- Aplica los permisos al directorio del usuario asuser # `chmod 711 /home/asuser`

```

[uo282440@linux ~]# chmod 711 /home/asuser
[uo282440@linux ~]# _

```

- Ejecuta el siguiente comando para permitir que Apache pueda leer contenidos localizados en los directorios de inicio de los usuarios locales

# `setsebool -P httpd_read_user_content on`

- Ejecuta el comando siguiente para habilitar el uso de los directorios ~/public\_html de los usuarios:

# `setsebool -P httpd_enable_homedirs on`



```

[uo282440@linux ~]# setsebool -P httpd_read_user_content on
[ 2191.263069] SELinux: Converting 375 SID table entries...
[ 2191.273765] SELinux: policy capability network_peer_controls=1
[ 2191.273803] SELinux: policy capability open_perms=1
[ 2191.273813] SELinux: policy capability extended_socket_class=1
[ 2191.273824] SELinux: policy capability always_check_network=0
[ 2191.273834] SELinux: policy capability cgroup_seclabel=1
[ 2191.273844] SELinux: policy capability nmp_nosuid_transition=1
[ 2191.273854] SELinux: policy capability genfs_seclabel_symlinks=1
[uo282440@linux ~]# setsebool -P httpd_eanble_homedirs on
Boolean httpd_eanble_homedirs is not defined
[uo282440@linux ~]# setsebool -P httpd_eanable_homedirs on
Boolean httpd_eanable_homedirs is not defined
[uo282440@linux ~]# setsebool -P httpd_enable_homedirs on
[ 2275.400869] SELinux: Converting 376 SID table entries...
[ 2275.410023] SELinux: policy capability network_peer_controls=1
[ 2275.410063] SELinux: policy capability open_perms=1
[ 2275.410073] SELinux: policy capability extended_socket_class=1
[ 2275.410084] SELinux: policy capability always_check_network=0
[ 2275.410094] SELinux: policy capability cgroup_seclabel=1
[ 2275.410104] SELinux: policy capability nmp_nosuid_transition=1
[ 2275.410113] SELinux: policy capability genfs_seclabel_symlinks=1
[uo282440@linux ~]# _

```

- Entra en una terminal como usuario asuser y crea en su directorio la carpeta public\_html y en ella un fichero básico index.html. Aplica los permisos de acceso adecuados a la carpeta public\_html # chmod 755 -R public\_html.

```

[uo282440@linux ~]# su asuser
[uo282440@linux root]# whoami
asuser
[uo282440@linux root]# ls
ls: no se puede abrir el directorio '.': Permiso denegado
[uo282440@linux root]# pwd
/root
[uo282440@linux root]# cd
[uo282440@linux ~]# pwd
/home/asuser
[uo282440@linux ~]# ls
prueba.txt
[uo282440@linux ~]# mkdir public_html
[uo282440@linux ~]# nano public_html/index.html_

```

```
GNU nano 5.6.1
<!DOCTYPE html>

    <head>
        <meta charset="utf-8" />
        <title>Asuser Page </title>
    </head>

    <body>
        <h1>Página de asuser</h1>
    </body>

</html>
```

```
[uo282440@linux ~]# ls
prueba.txt public_html
[uo282440@linux ~]# cat public_html/index.html
<!DOCTYPE html>

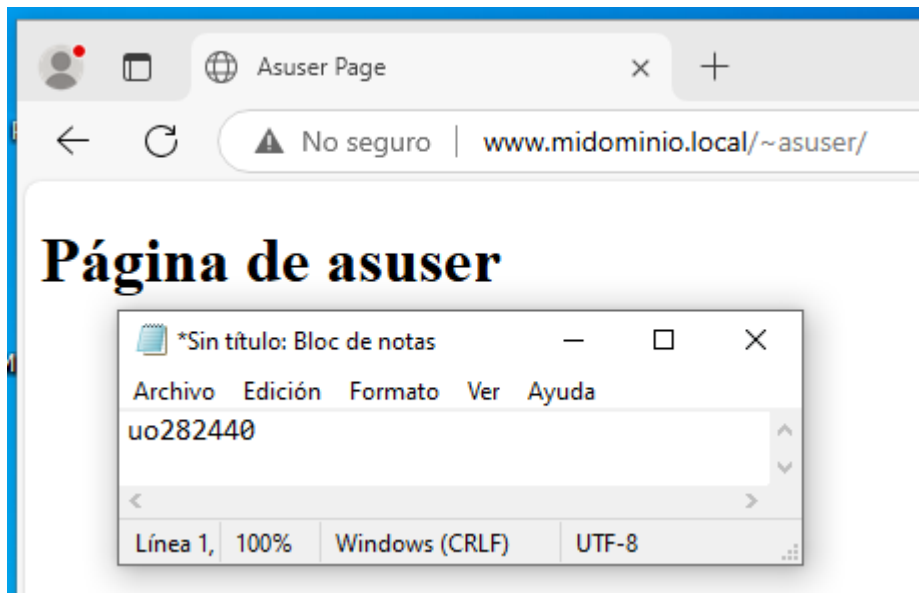
    <head>
        <meta charset="utf-8" />
        <title>Asuser Page </title>
    </head>

    <body>
        <h1>Página de asuser</h1>
    </body>

</html>
[uo282440@linux ~]# chmod 755 -R public_html/
[uo282440@linux ~]#
```

- Reinicia el servicio httpd y mira a ver si eres capaz de acceder desde la máquina W10 a <http://www.midominio.local/~asuser>. Ahora cada usuario de Linux puede disponer de su página personal, sin más que crear un directorio public\_html y colgar allí su contenido.

```
[sudo] password for asuser:
asuser is not in the sudoers file. This incident will be reported.
[uo282440@linux ~]# su root
Contraseña:
[uo282440@linux asuser]# sudo systemctl restart httpd
[uo282440@linux asuser]#
```



### 3. Configuración del servidor Apache

#### A) Ubicación

- Crearemos una nueva ubicación para la página web: crea la carpeta /as/web y copia en ella el fichero index.html. En el archivo de configuración que está en /etc/httpd/conf/httpd.conf modifica la directiva DocumentRoot para que busque los documentos en el nuevo directorio /as/web

```
[uo282440@linux web]# ls
[uo282440@linux web]# pwd
/as/web
[uo282440@linux web]# ls /home/asuser/public_html/
index.html
[uo282440@linux web]# cp /home/asuser/public_html/index.html .
[uo282440@linux web]# ls
index.html
[uo282440@linux web]# _
```

```
#
#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "/as/web_"
#
```

Sustituye la sección por la sección que sigue

```
# below.
#
#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "/as/web"
#
# Relax access to content within /var/www.
#
<Directory "/as/web">
    Options FollowSymLinks
    AllowOverride None
    Require all granted_
</Directory>
```

- Restaura el servicio httpd, asigna el contexto httpd\_sys\_content\_t a través de chcon y comprueba que la página [www.midominio.local](http://www.midominio.local) ha cambiado.

```
[uo282440@linux web]# sudo apachectl restart
[uo282440@linux web]# sudo systemctl restart httpd
[uo282440@linux web]# chcon -R -h -t httpd_sys_content_t /as/web
[uo282440@linux web]# _
```



B) ServerName

- Modifica las directivas ServerAdmin y ServerName de acuerdo con tu email y con el nombre [www.midominio.local](http://www.midominio.local).

```
#
# 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 uo282440@uniovi.es

#
# 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:80
```

- Haz que el servidor escuche en el puerto 9999. Para ello, o Cambia el puerto 80 a 9999 en la directiva Listen o Cambia el nombre del servidor a www.midominio.local:9999 en la directiva ServerName

```
#
#Listen 12.34.56.78:80
Listen 9999_
```

```
ServerAdmin uo282440@uniovi.es

#
# 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
```

```
[uo282440@linux web]#cat /etc/httpd/conf/httpd.conf | grep 9999
Listen 9999
ServerName www.midominio.local:9999
[uo282440@linux web]#
```

- Comprueba que ahora con http://www.midominio.local:9999 puedes visualizar la página. Añade una nueva regla al firewall si es necesario. Vuelve a dejar el servidor escuchando en el puerto 80.



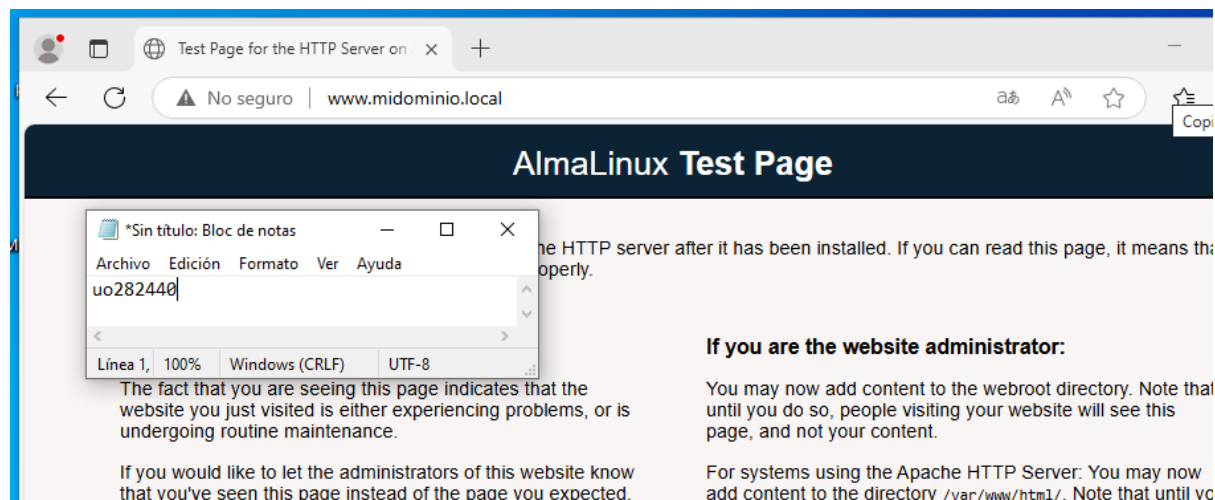
Lo volvemos a dejar en el 80:

```
[uo282440@linux web]# cat /etc/httpd/conf/httpd.conf | grep 80
#Listen 12.34.56.78:80
Listen 80
ServerName www.midominio.local:80
[uo282440@linux web]#
```

- Renombra el archivo index.html a índice.html. Recarga la página para que se borre la cache y consulta de nuevo la dirección www.midominio.local ¿Qué ocurre?

```
[uo282440@linux web]# mv index.html indice.html
[uo282440@linux web]# ls
indice.html
[uo282440@linux web]# _
```

Nos vuelve a salir la página de prueba:



- Edita el fichero `/etc/httpd/conf.d/welcome.conf` y comenta todas las líneas (añadiendo `#` al principio de todas ellas). Con ello desactivamos la presentación de la página de “Almalinux Test Page” para el caso que no encuentre el fichero `index.html`.  
Restaura el servicio `httpd` y vuelve a recargar la página. ¿Qué ocurre?

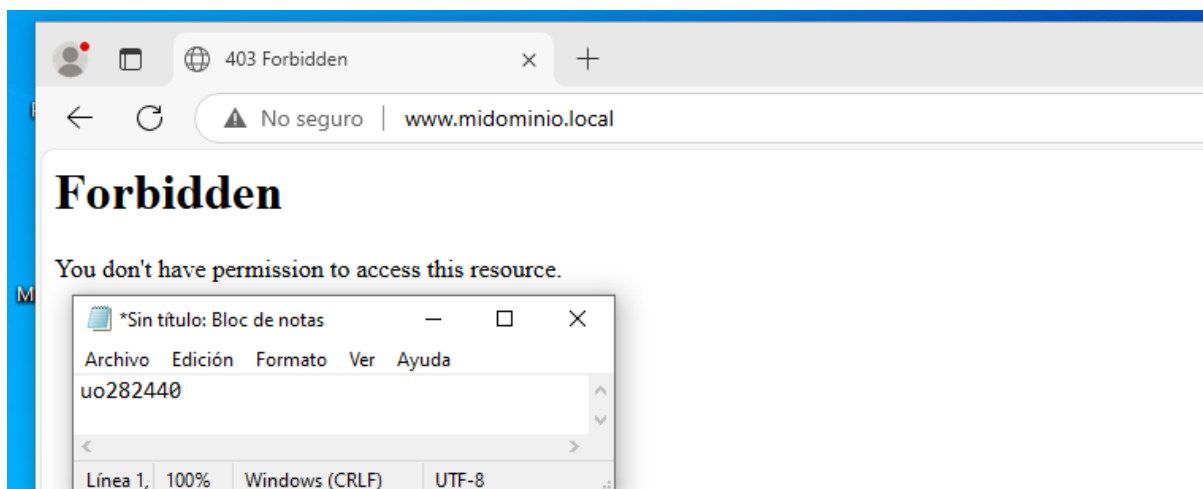
```
GNU nano 5.6.1 /etc/httpd/conf.d/welcome.conf
#
# This configuration file enables the default "Welcome" page if there
# is no default index page present for the root URL. To disable the
# Welcome page, comment out all the lines below.
#
# NOTE: if this file is removed, it will be restored on upgrades.
#
<LocationMatch "^/+?$">
#   Options -Indexes
#   ErrorDocument 403 /noindex.html
#</LocationMatch>

<Directory /usr/share/httpd/noindex>
#   AllowOverride None
#   Require all granted
#</Directory>

#Alias /noindex.html /usr/share/httpd/noindex/index.html
#Alias /poweredby.png /usr/share/httpd/icons/apache_pb3.png
#Alias /system_noindex_logo.png /usr/share/httpd/icons/system_noindex_logo.png

[uo282440@linux web]#sudo systemctl restart httpd
[uo282440@linux web]#
```

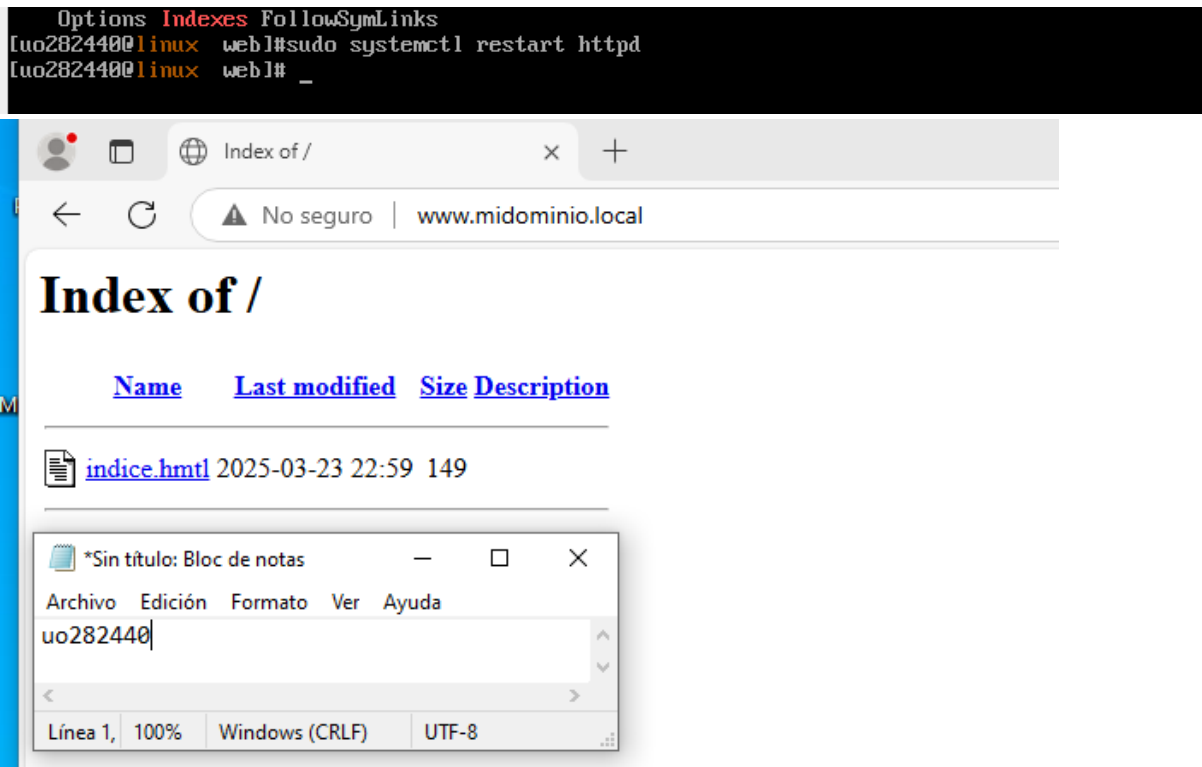
No tenemos permiso para acceder a este recurso:



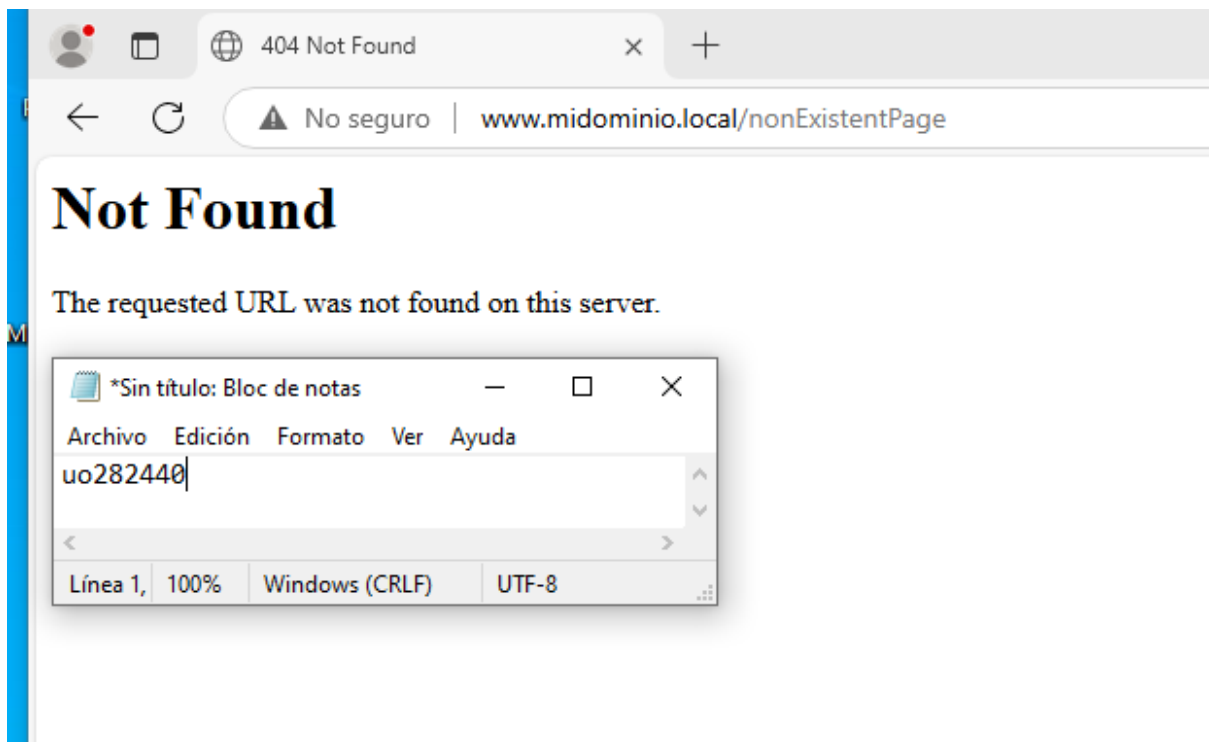
- Modifica la sección que creaste anteriormente para `as/web`:

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

- Vuelve a restaurar el servicio, recarga la página y comprueba que ahora ya puedes visualizar el contenido de los directorios.



- Haz un acceso a una página no existente y comprueba qué ocurre en /var/log/httpd/access\_log.





```
[tuo282448@linux web]# cat /var/log/httpd/access_log | grep nonExistentPage
192.168.56.112 - - [23/Mar/2025:23:37:26 +0100] "GET /nonExistentPage HTTP/1.1" 404 196 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/537.36 Edg/134.0.0.0"
[tuo282448@linux web]#
```

Se realizó una petición GET a las 23:37 pm por parte de 192.168.56.112 (mi windows 10) de la página “nonExistentPage”, la cual no existe, incluye también más información acerca del navegador utilizado o el sistema operativo de la máquina que realiza la petición.