

Practica 07-ASR Lab

ANA FERNÁNDEZ OSTIO-UO275780

Contenido

Instalación 2

Configuración de páginas web de los usuarios 5

Configuración del servidor Apache 6

Hosts virtuales 8

Autenticación 10

Servidor Proxy 12

Instalación

Comprobamos que la maquina de Linux tiene el nombre de linux.as.local y también si tiene conexión al exterior

```
[uo275780@linux ~]$ uname -a
Linux linux.as.local 4.18.0-348.el8.x86_64 #1 SMP Tue Nov 9 06:28:28 EST 2021 x86_64 x86_64 x86_64 GNU/Linux
[uo275780@linux ~]$
```

```
[uo275780@linux ~]$ ping www.google.es
PING www.google.es (142.250.200.99) 56(84) bytes of data.
64 bytes from mad41s13-in-f3.1e100.net (142.250.200.99): icmp_seq=1 ttl=115 time=71.0 ms
64 bytes from mad41s13-in-f3.1e100.net (142.250.200.99): icmp_seq=2 ttl=115 time=63.1 ms
64 bytes from mad41s13-in-f3.1e100.net (142.250.200.99): icmp_seq=3 ttl=115 time=66.2 ms
64 bytes from mad41s13-in-f3.1e100.net (142.250.200.99): icmp_seq=4 ttl=115 time=86.2 ms
64 bytes from mad41s13-in-f3.1e100.net (142.250.200.99): icmp_seq=5 ttl=115 time=75.9 ms
^C
--- www.google.es ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 63.133/72.477/86.184/8.105 ms
[uo275780@linux ~]$
```

Comprobamos si desde la maquina de Windows 10 y de Windows Server tenemos acceso a las maquina de Linux

```
C:\Users\uo275780>ping linux

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=1ms TTL=64
Respuesta desde 192.168.56.100: bytes=32 tiempo<1m TTL=64

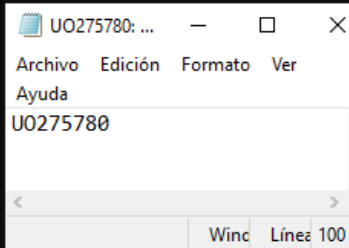
Estadísticas de ping para 192.168.56.100:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 1ms, Media = 0ms

C:\Users\uo275780>
```

```
C:\Users\Administrador>ping linux

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

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 = 0ms, Media = 0ms
Control-C
^C
C:\Users\Administrador>
```

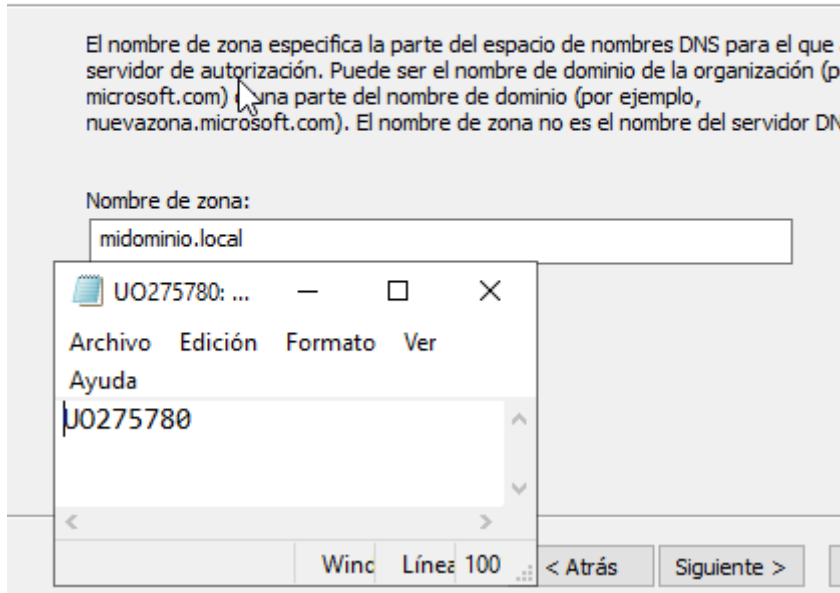


Tras haber comprobado esto, pasamos a configurar una nueva zona, midominio.local

Asistente para nueva zona

Nombre de zona

¿Qué nombre tiene la zona nueva?



Y de paso comprobamos que win10 puede resolver esa dirección y que coincide con la dirección de Linux:

```
C:\Users\uo275780>nslookup www.midominio.local
Servidor:  ws2019.as.local
Address:  192.168.56.101

Nombre:  www.midominio.local
Address:  192.168.56.100

C:\Users\uo275780>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\uo275780>
```

Comprobamos si Linux tiene apache instalado, en este caso no lo tiene por lo que tenemos que instalarlo

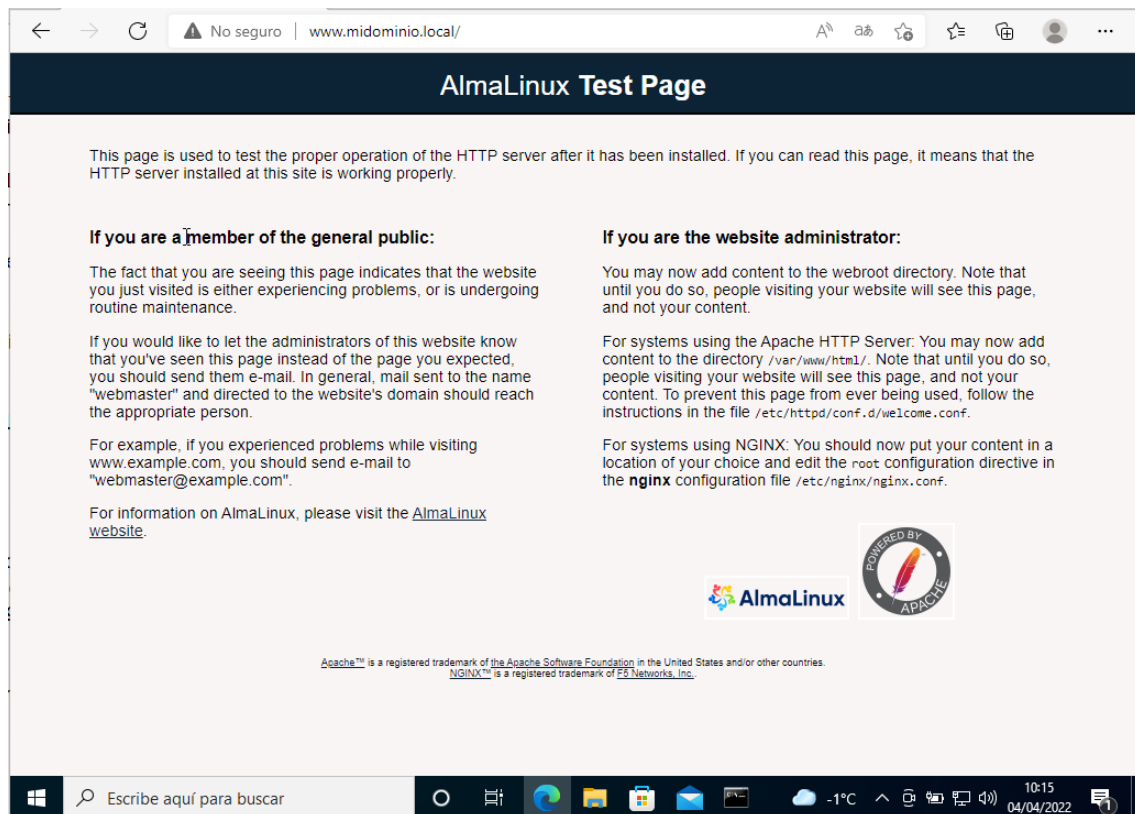
```
[uo275780@linux ~]$ su -
Contraseña:
[root@linux ~]# dnf install httpd
Última comprobación de caducidad de metadatos hecha hace 1 day, 20:05:19, el sáb
02 abr 2022 13:57:39 CEST.
Dependencias resueltas.
=====
Paquete      Arq.  Versión                      Repositorio      Tam.
=====
Instalando:
  httpd      x86_64 2.4.37-43.module_el8.5.0+2631+6f259f31.3.alma
                                         appstream 1.4 M
Instalando dependencias:
  almalinux-logos-httpd
```

Arrancamos el servidor de apache y configuramos de nuevo el firewall:

```
[root@linux ~]# /sbin/service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@linux ~]#
```

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

Comprobamos desde alguna de las máquinas de Linux que a través del nombre con el protocolo http funciona, en este caso se ha escogido la máquina de Windows 10 ya que contiene un mejor navegador que la de ws2019

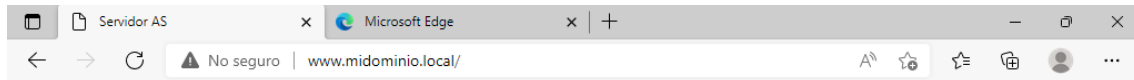


Procedemos a crear la pagina web para ver si así también se puede ver desde la maquina de Windows:

```

<!DOCTYPE html>
<head>
  <meta charset = "utf8" />
  <title>Servidor AS</title>
</head>
<body>
  <h1>Administración de Sistemas</h1>
  <h2>Ana Fernandez Ostio, UO275780</h2>
</body>
</html>

```



Administración de Sistemas

Ana Fernandez Ostio, UO275780



Configuración de páginas web de los usuarios

Configuramos los ficheros para los usuarios

```

# The path to the end user account 'public_html' directory must be
# accessible to the webserver userid. This usually means that the
# user 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
  # existence of a username on the system (depending on home directory
  # permissions).
  #
  #   UserDir disabled
  #
  #
  # To enable requests to /~user/ to serve the user's public
  # directory, remove the "UserDir disabled" line above, and
  # the following line instead:
  #
  # UserDir public_html
</IfModule>

```

Añadimos permisos al usuario

```
[root@linux home]# cd /etc/httpd/conf.d
[root@linux conf.d]# chmod 711 /home/asuser
[root@linux conf.d]# setsebool -P httpd_read_user_content on
[root@linux conf.d]#
```

Seguimos los pasos tal y como se nos indica y añadimos una nueva página desde asuser quedando de la siguiente forma:



Configuración del servidor Apache

Para comenzar creamos la nueva carpeta donde copiaremos el fichero index.html.

```
[root@linux ~]# cd as
-bash: cd: as: No existe el fichero o el directorio
[root@linux ~]# mkdir -p /as/web
[root@linux ~]# ls
anaconda-ks.cfg  initial-setup-ks.cfg
[root@linux ~]# cd /as/web
[root@linux web]# cp /var/www/html/index.html /as/web/
[root@linux web]# ls
index.html
[root@linux web]#
```

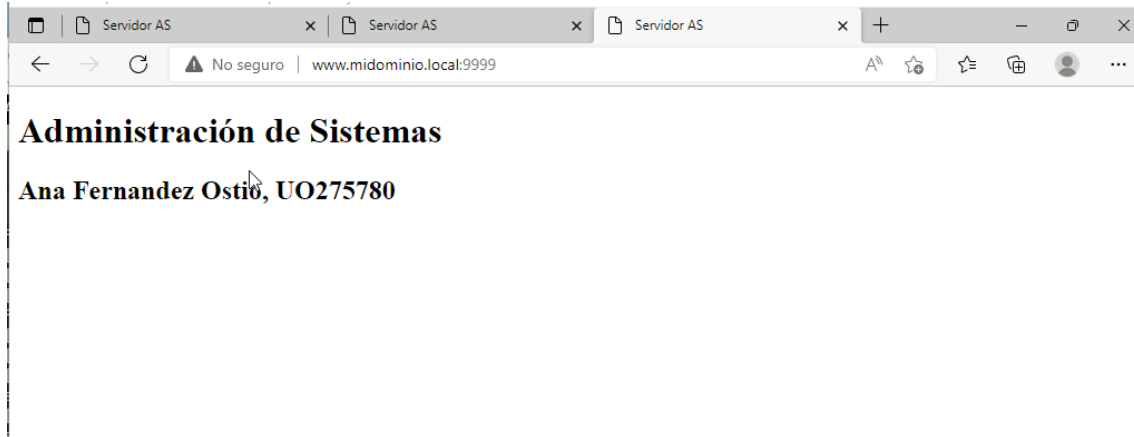
Actualizamos de donde tiene que sacar la información

```
#
# 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">
    Option FollowSymLinks
    AllowOverride None
    # Allow open access:
    Require all granted
</Directory>
```

Y restauramos el servicio.

Por lo que ahora toca actualizar son las directivas de admin, por mi correo corporativo uo275780@uniovi.es y por el nombre del server, www.midominio.local. Tras haber modificado estas dos directivas, hay que añadir una nueva regla al firewall, en este caso abrir de forma interna el puerto 9999. Tras ello comprobamos que si accedemos a la pagina desde www.midominio.local:9999

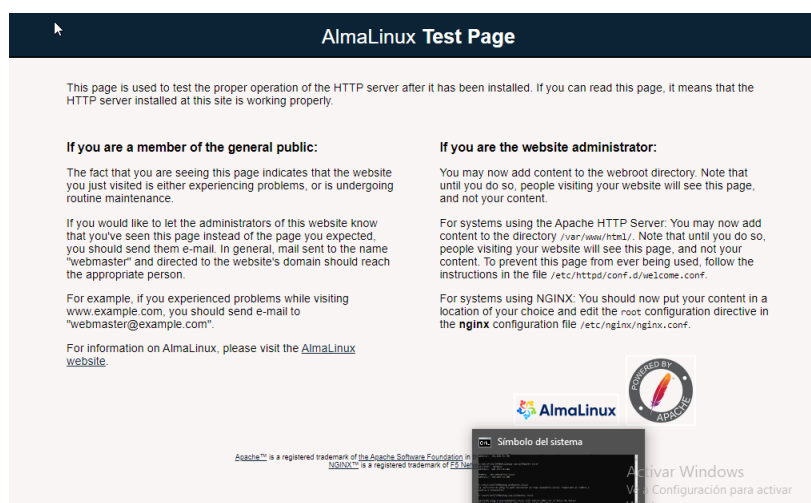


Tras comprobar que esto funciona de forma correcta, procedemos a volver a poner todos los valores que se habían cambiado por un 80. Se ha comprobado si todo sigue funcionando de forma correcta y así es.

Para los repositorios, comenzamos renombrando el archivo a índice.html, en este caso se ha hecho mediante el comando "mv index.html índice.html" y se comprueba que ha sido efectivo el cambio. Se ha evitado colocar el acento sobre la primera i, ya que a lo mejor puede provocar diferentes errores.

```
[root@localhost web]# mv index.html índice.html
[root@localhost web]# ls
índice.html
[root@localhost web]#
```

Lo que ocurre tras realizar esto, es que la pagina nos vuelve a salir la de inicio, es decir, como si no existiera, ya que el sistema va a buscar aquel index.html que exista, no un índice ya que no se le ha indicado que busque con este nuevo nombre.



Comentamos todos las líneas que se encuentran en el fichero que se nos comenta. Tras comentar estas líneas, se nos dice que no hay permisos para acceder a este. Por lo que ahora hay que volver a modificar aquello que se nos comentó.

Recargamos las páginas y nos funciona correctamente



Intentamos acceder a un pagina que no existe en nuestro repositorio, en este caso se intentó acceder a www.midominio.local/noExiste miramos en el log de httpd a donde se ha intentado acceder, en este caso, como se ve en la captura superior, se nos dice la día y hora de cuando se ha intentado acceder y a la parte de la web. Además de darnos una posible respuesta de que tipo de SSOO lo ha intentado, así como el posible navegador.

Tras haber realizado esto, la practica obligatoria, se habría acabado

Hosts virtuales

Creamos un nuevo dominio. Una vez este creado este host nuevo, se procede a la creación de la carpeta.

Host nuevo

Nombre (si se deja en blanco, se usa el nombre del dominio primario):
www.otraempresa

Nombre de dominio completo (FQDN):
www.otraempresa.midominio.local.

Dirección IP:
192.168.56.100

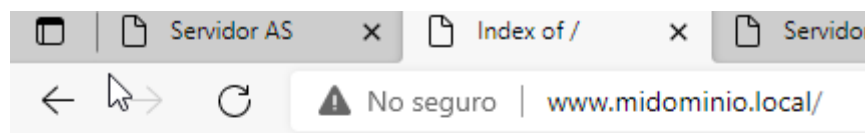
☒ Crear registro del puntero (PTR) asociado

Indicamos en el índice de otra Empresa que nos estamos refiriendo a este .html para que cuando recarguemos la página no produzca fallos.



```
[root@localhost otraempresa]# cat indice.html
<!DOCTYPE html>
<head>
  <meta charset = "utf-8" />
  <title>Servidor AS</title>
</head>

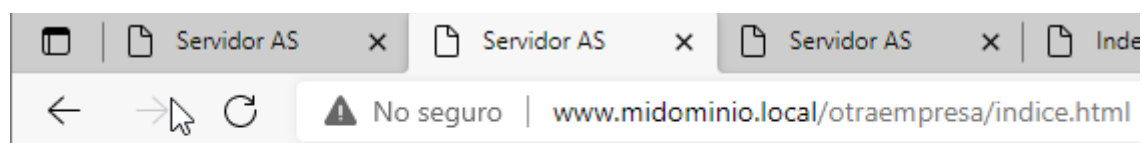
<body>
  <h1>Administracion de Sistemas</h1>
  <h2>Ana Fernandez Ostio</h2>
  <h3>Página de otra empresa</h3>
</body>
</html>
[root@localhost otraempresa]#
```

Tras añadir un nuevo directory sobre el archivo httpd conf, restauramos el servicio y comprobamos que sale. En este caso vemos como se nos ha creado un nueva carpeta donde dentro de este existe el archivo indice con los nuevo valores.



Index of /

Name	Last modified	Size	Description
 indice.html	2022-04-09 14:40	178	
 otraempresa/	2022-04-09 15:16	-	

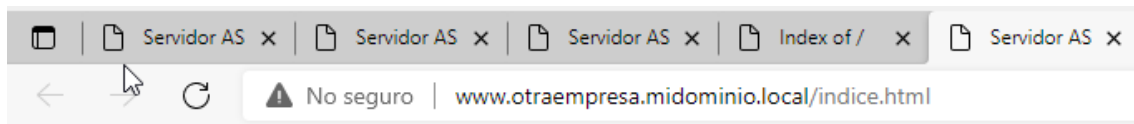


Administracion de Sistemas

Ana Fernandez Ostio

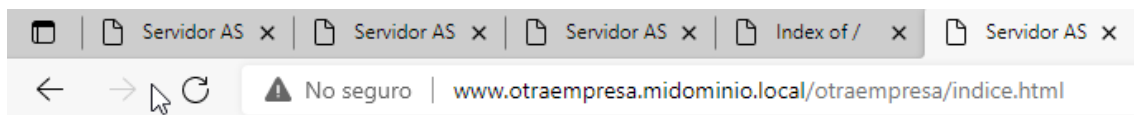
Página de otra empresa

Luego recargamos los servicios y comprobamos que funciona todo correctamente.



Administracion de Sistemas

Ana Fernandez Ostio



Administracion de Sistemas

Ana Fernandez Ostio

Página de otra empresa

Autenticación

Cambiamos el archivo httpd config añadiendo una nueva regla, la que se nos dice en el archivo de practicas

```
<Directory "/as/web/otraempresa">
    Options Indexes FollowSymLinks
    AllowOverride AuthConfig
    Require all granted
</Directory>
```

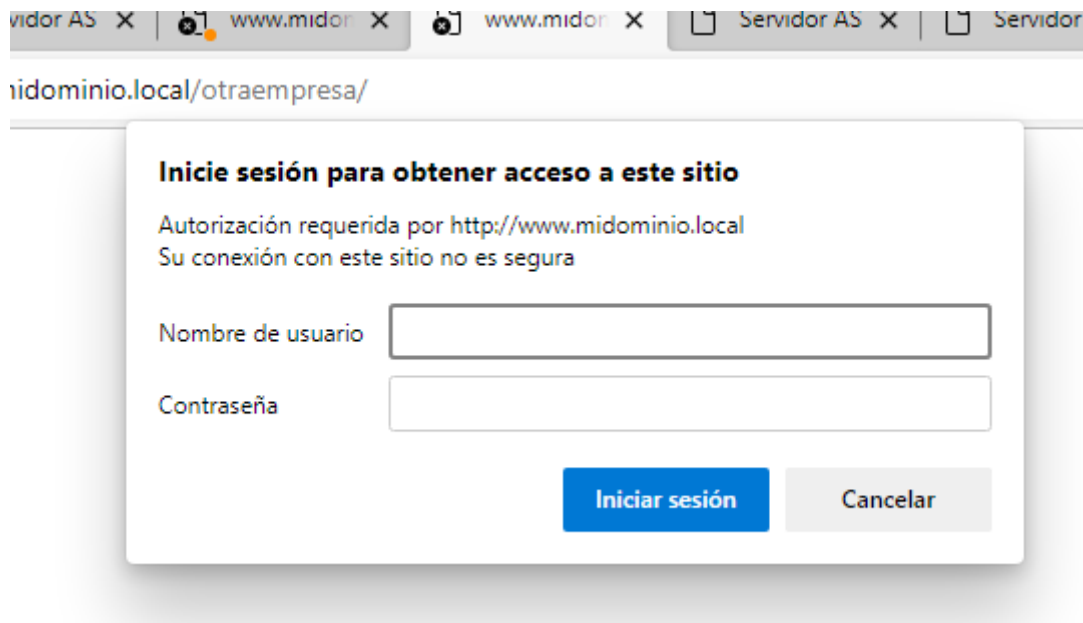
Añadimos sobre la carpeta /as/web/otraempresa un nuevo archivo privado donde indicaremos a donde puede coger los datos de acceso.

```
[root@localhost otraempresa]# cat .htaccess
AuthType Basic
AuthName "Area Restringida"
AuthUserFile /etc/httpd/password.file
AuthGroupFile /dev/null
Require valid-user
[root@localhost otraempresa]#
```

Añadimos al archivo los nuevos usuarios

```
[root@localhost ~]# htpasswd -c /etc/httpd/password.file usuario1-uo275780
New password:
Re-type new password:
Adding password for user usuario1-uo275780
[root@localhost ~]# htpasswd /etc/httpd/password.file usuario2-uo275780
New password:
Re-type new password:
Adding password for user usuario2-uo275780
[root@localhost ~]#
```

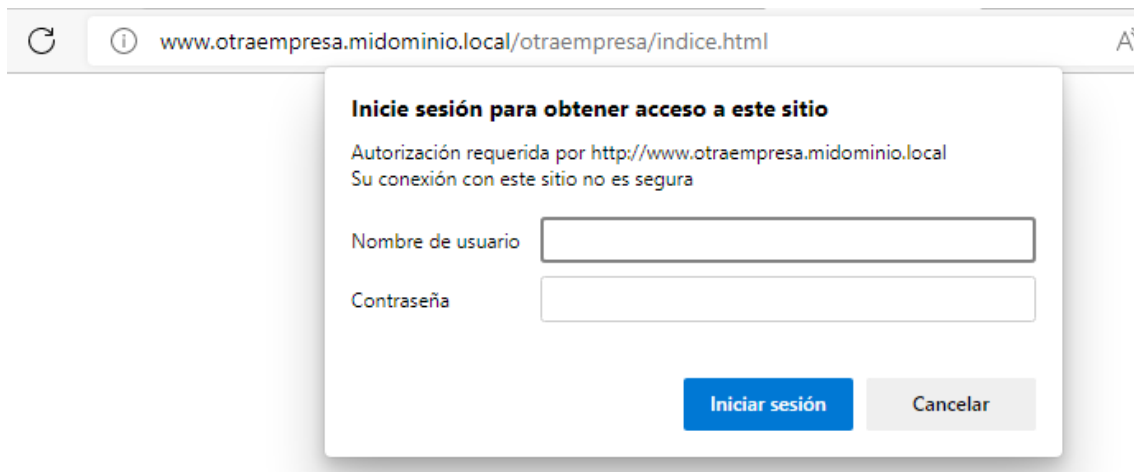
Se ha añadido con mi uo para marcar diferencia. Vemos que se nos pide una nueva conexión mediante la contraseña



Una vez que introducimos los datos, se ve que funciona



Ahora comprobamos que si accedemos desde el dominio de otra empres también nos pide la contraseña.



Servidor Proxy

Para poder instalar el proxy, lo primero que debemos hacer es instalar el software libre, en este caso llamado squid. Tras haberlo instalado, lo iniciamos y lo ponemos en activo.

```
Verificando : squid-7:4.15-1.module_el8.5.0+2548+24c223d9.x86_64 3/3
Instalado:
libecap-1.0.1-2.module_el8.5.0+2548+24c223d9.x86_64
perl-Digest-SHA-1:6.02-1.el8.x86_64
squid-7:4.15-1.module_el8.5.0+2548+24c223d9.x86_64

¡Listo!
[root@localhost ~]# systemctl start squid
[root@localhost ~]# systemctl enable squid
Created symlink /etc/systemd/system/multi-user.target.wants/squid.service → /usr
/lib/systemd/system/squid.service.
[root@localhost ~]#
```

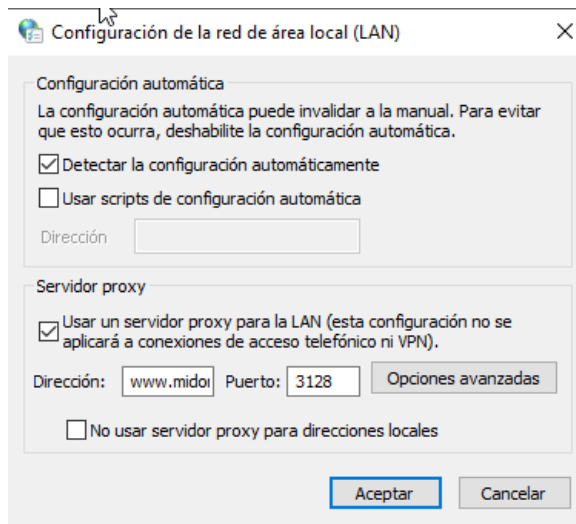
Comprobamos el rango de direcciones y vemos que es correcto, ya que consigue que un número mayor de redes pueda acceder.

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

Ahora lo que toca es añadir una nueva regla al firewall.

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

Lo que toca a continuación es desde la máquina de Windows configurar el LAN.



Comprobamos el registro para ver que nos sale

```
success
[root@localhost ~]# tail -f /var/log/squid/access.log
1649514970.389      37 192.168.56.111 TCP_MISS/503 4751 GET http://www.midominio.local/
- HIER_NONE/- text/html
1649514971.283      0 192.168.56.111 TCP_HIT/200 13098 GET http://localhost.localdomain
:3128/squid-internal-static/icons/SN.png - HIER_NONE/- image/png
1649514972.663      0 192.168.56.111 NONE/000 0 NONE error:transaction-end-before-head
rs - HIER_NONE/- -
1649514972.664     4493 192.168.56.111 TCP_TUNNEL/200 1257 CONNECT aefd.nelreports.net:44
3 - HIER_DIRECT/77.26.255.203 -
1649514972.664     4578 192.168.56.111 TCP_TUNNEL/200 7948 CONNECT www.bing.com:443 - HIE
R_DIRECT/13.107.21.200 -
```

Como se puede observar no se puede acceder a la pagina.