



ACTIVIDADES PRESENCIALES

CicloFormativo: Administración de Sistemas Informáticos en Red

Móduloprofesional: Seguridad y Alta Disponibilidad

TÍTULO DEL TALLER	HORAS	RA, CE	PONDERACIÓN	CONTENIDOS
Clúster de alta disponibilidad para servicios web con apache y montado en docker	5	6.g	50%	Análisis de configuraciones de alta disponibilidad: Funcionamiento ininterrumpido. Integridad de datos y recuperación de servicio. Servidores redundantes. Sistemas de «clústeres». Balanceadores de carga. Instalación y configuración de soluciones de alta disponibilidad. Virtualización de sistemas. Posibilidades de la virtualización de sistemas. Alta disponibilidad y virtualización. Servicios reales con virtualización.
Instalación y configuración NAS Synology	4	6.b	50%	Funcionamiento ininterrumpido. Integridad de datos y recuperación del servicio.
FECHAS PROPUESTAS	<ul style="list-style-type: none"> • Para los alumnos de ASIR-SAD-Completa se recomienda realizar el taller entre el 01-02-25 y el 10-03-25 antes de realizar la FCT. • Para los alumnos de ASIR-SAD-Parcial Diferenciada hasta final de enero no se les abre la unidad 6 correspondiente a los RA y CE de este taller, por lo tanto, no pueden realizar el taller antes ya que no habrán adquirido los conocimientos necesarios. Para ellos se propone realizar el taller en cualquier semana del mes de abril o mayo. • Ambos se realizarían en 2 sesiones de 5 y 4 horas. 			

ACTIVIDADES PROPUESTAS:

TALLER1: Clúster de alta disponibilidad para servicios web con apache y montado en docker:

1. Hoy en día se usa una tecnología en la gran mayoría de empresas multinacionales llamada Docker. En este taller vamos a realizar un clúster de alta disponibilidad para servicios web pero montado en docker. Para ello debemos entender el uso y aplicación de docker Engine.
<https://docs.docker.com/engine/>
2. Entra en la página oficial de docker e instala docker Engine sobre 3 máquinas Ubuntu 22.04 con, al menos, 2 CPU, 8 GB RAM y 25Gb de disco duro.
<https://docs.docker.com/engine/install/ubuntu/>
Desde un terminal de linux desinstalamos antiguas versiones de docker, por si las hubie-

ra:

sudo apt-get remove docker docker-engine docker.io containerd runc

```
leonor@Ubuntu22Docker:~$ sudo apt-get remove docker docker-engine docker.io containerd runc
[sudo] contraseña para leonor:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
E: No se ha podido localizar el paquete docker-engine
leonor@Ubuntu22Docker:~$
```

Configuramos el repositorio:

sudo apt-get update

sudo apt-get install \
 ca-certificates \
 curl \
 gnupg \
 lsb-release

```
leonor@Ubuntu22Docker:~$ sudo apt-get update
Obj:1 http://es.archive.ubuntu.com/ubuntu jammy InRelease
Des:2 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease [114 kB]
Obj:3 http://security.ubuntu.com/ubuntu jammy-security InRelease
Obj:4 http://es.archive.ubuntu.com/ubuntu jammy-backports InRelease
Des:5 http://es.archive.ubuntu.com/ubuntu jammy/main Translation-es [332 kB]
Des:6 http://es.archive.ubuntu.com/ubuntu jammy/restricted Translation-es [964 B]
Des:7 http://es.archive.ubuntu.com/ubuntu jammy/universe Translation-es [1.356 kB]
Des:8 http://es.archive.ubuntu.com/ubuntu jammy/multiverse Translation-es [68,2 kB]
Descargados 1.872 kB en 3s (578 kB/s)
Leyendo lista de paquetes... Hecho
leonor@Ubuntu22Docker:~$ sudo apt-get install \
    ca-certificates \
    curl \
    gnupg \
    lsb-release
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
lsb-release ya está en su versión más reciente (11.1.0ubuntu4).
fijado lsb-release como instalado manualmente.
ca-certificates ya está en su versión más reciente (20211016ubuntu0.22.04.1).
fijado ca-certificates como instalado manualmente.
gnupg ya está en su versión más reciente (2.2.27-3ubuntu2.1).
fijado gnupg como instalado manualmente.
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
 libflashrom1 libftdi1-2
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes NUEVOS:
 curl
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 1 no actualizados.
Se necesita descargar 193 kB de archivos.
Se utilizarán 453 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
```

Añadimos la llave docker:

sudo mkdir -p /etc/apt/keyrings

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg

```
leonor@Ubuntu22Docker:~$ sudo mkdir -p /etc/apt/keyrings
leonor@Ubuntu22Docker:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
leonor@Ubuntu22Docker:~$
```

Configuramos el repositorio:

echo \

"deb [arch=\$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
 \$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

```
leonor@Ubuntu22Docker:~$ echo \
    "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
    $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
leonor@Ubuntu22Docker:~$
```

Actualizamos:

sudo apt-get update

Instalamos docker Engine:

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin



```
leonor@Ubuntu22Docker:~$ sudo apt-get update
Obj:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Obj:2 http://es.archive.ubuntu.com/ubuntu jammy InRelease
Des:3 https://download.docker.com/linux/ubuntu jammy InRelease [48,9 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease [114 kB]
Des:5 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [11,1 kB]
Obj:6 http://es.archive.ubuntu.com/ubuntu jammy-backports InRelease
Descargados 174 kB en 2s (105 kB/s)
Leyendo lista de paquetes... Hecho
leonor@Ubuntu22Docker:~$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  libflashrom1 libftdi1-2
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
  docker-ce-rootless-extras docker-scan-plugin git git-man liberror-perl libslirp0 pigz slirp4netns
Paquetes sugeridos:
  aufs-tools cgroupfs-mount | cgroup-lite git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb git-cvs
  git-mediawiki git-svn
Se instalarán los siguientes paquetes NUEVOS:
  containerd.io docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin docker-scan-plugin git git-man liberror-perl
  libslirp0 pigz slirp4netns
0 actualizados, 12 nuevos se instalarán, 0 para eliminar y 1 no actualizados.
Se necesita descargar 116 MB de archivos.
Se utilizarán 449 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
```

Después de un café y una vez instalado docker ejecutamos el mítico “Hello-world” para comprobar que se ha instalado bien:

sudo docker run hello-world

```
leonor@Ubuntu22Docker:~$ sudo docker run hello-world
[sudo] contraseña para leonor:
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:94ebc7edf3401f299cd3376a1669bc0a49aef92d6d2669005f9bc5ef028dc333
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

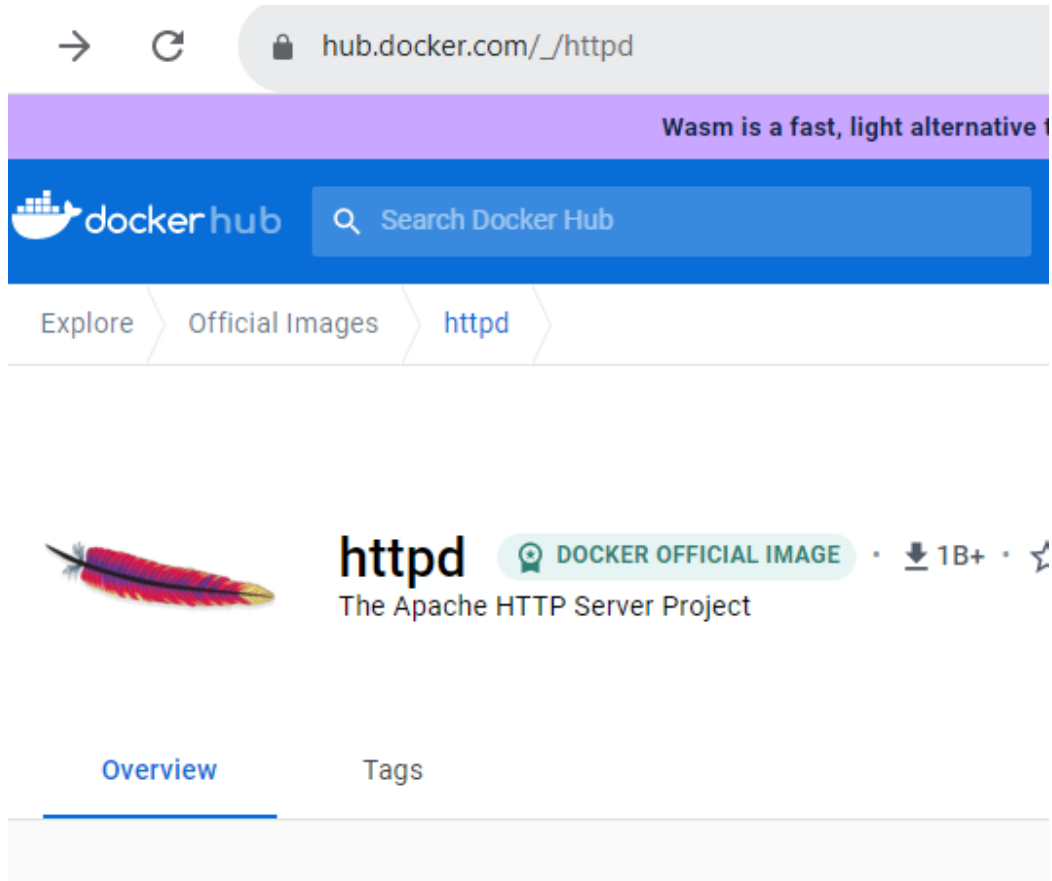
Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

leonor@Ubuntu22Docker:~$
```

3. Ahora debemos instalar en dos de las máquinas, la imagen de httpd de Apache.
<https://hub.docker.com/search?q=>

En el buscador buscamos httpd y pinchamos sobre ella, la imagen oficial de docker:



A la derecha, en la esquina superior tenemos el comando para instalar la imagen:

```
leonor@Ubuntu22Docker:~$ sudo docker pull httpd
[sudo] contraseña para leonor:
Using default tag: latest
latest: Pulling from library/httpd
3f4ca61aafcd: Pull complete
2e3d233b6299: Pull complete
6d859023da80: Pull complete
f856a04699cc: Pull complete
ec3bbe99d2b1: Pull complete
Digest: sha256:f8c7bdfa89fb4448c95856c6145359f67dd447134018247609e7a23e5c5ec03a
Status: Downloaded newer image for httpd:latest
docker.io/library/httpd:latest
leonor@Ubuntu22Docker:~$
```

En un terminal de la máquina escribimos:

```
sudo docker pull httpd
```

Para saber las imágenes de docker que tenemos:

```
sudo docker images
```

```
leonor@Ubuntu22Docker:~$ sudo docker images
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
httpd                latest         73c10eb9266e   2 weeks ago    145MB
hello-world         latest         feb5d9fea6a5   15 months ago  13.3kB
leonor@Ubuntu22Docker:~$
```

Vemos que por ahora tenemos la del hello-world y la de httpd.

Si queremos borrar una imagen, por ejemplo la de hello-world:

```
sudo docker rmi hello-world
```

Y si queremos borrar el contenedor de una imagen:

```
sudo docker rm CONTAINER_ID
```

Para ver los contenedores docker en ejecución o que se han ejecutado:

```
sudo docker ps -a
```

```
leonor@Ubuntu22Docker:~$ sudo docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
8d6bd7ecbc09  hello-world  "/hello"  34 minutes ago  Exited (0) 34 minutes ago  quirky_wescoff
leonor@Ubuntu22Docker:~$
```

Para hacer más eficiente la configuración vamos a crear una carpeta donde mapearemos el archivo de configuración de httpd, lo mismo es necesario cambiar los permisos de escritura y lectura del archivo de configuración para poder mapearlo, puede dar error de permiso denegado:

sudo mkdir apache

sudo docker run --rm httpd cat /usr/local/apache2/conf/httpd.conf > /apache/my-httpd.conf

```
leonor@Ubuntu22Docker:~/apache$ sudo docker run --rm httpd cat /usr/local/apache2/conf/httpd.conf > my-httpd.conf
leonor@Ubuntu22Docker:~/apache$ ll
total 36
drwxr-xr-x  2 root  root   4096 ene  7 18:55 ./
drwxr-x--  16 leonor leonor 4096 ene  7 18:26 ../
-rwxrwxrwx  1 root  root  20827 ene  7 18:58 my-httpd.conf*
-rw-r--r--  1 root  root    22 ene  7 18:42 servidor1
leonor@Ubuntu22Docker:~/apache$ sudo chmod 644 my-httpd.conf
leonor@Ubuntu22Docker:~/apache$ ll
total 36
drwxr-xr-x  2 root  root   4096 ene  7 18:55 ./
drwxr-x--  16 leonor leonor 4096 ene  7 18:26 ../
-rw-r--r--  1 root  root  20827 ene  7 18:58 my-httpd.conf
-rw-r--r--  1 root  root    22 ene  7 18:42 servidor1
leonor@Ubuntu22Docker:~/apache$ cat my-httpd.conf
#
# This is the main Apache HTTP server configuration file. It contains the
# configuration directives that give the server its instructions.
# See <URL:http://httpd.apache.org/docs/2.4/> for detailed information.
# In particular, see
# <URL:http://httpd.apache.org/docs/2.4/mod/directives.html>
```

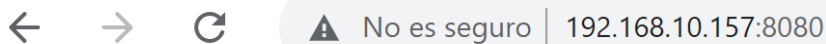
Para ejecutar un contenedor con la imagen de Apache, mapeando el puerto 80 del contenedor con el puerto 8080 de nuestra máquina y mapeando también la carpeta creada en /home/leonor/apache en nuestra máquina con la carpeta dentro del contenedor donde están los archivos de configuración de httpd. De esta manera podemos cambiar el fichero de configuración de Apache sin tener que entrar dentro del contenedor a modificar archivos, se gana en eficiencia:

sudo docker run -dit --name my-apache-app -p 8080:80 -v

"\$PWD"/apache:/usr/local/apache2/htdocs/ httpd

```
leonor@Ubuntu22Docker:~$ sudo mkdir apache
leonor@Ubuntu22Docker:~$ ls
apache  Descargas  Documentos  Escritorio  Imágenes  Música  Plantillas  Público  snap  test  Videos
leonor@Ubuntu22Docker:~$ sudo docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED              STATUS              PORTS              NAMES
8d6bd7ecbc09   hello-world    "/hello"                 About an hour ago    Exited (0) About an hour ago              quirky_wescoff
leonor@Ubuntu22Docker:~$ sudo docker run -dit --name my-apache-app -p 8080:80 -v "$PWD"/apache:/usr/local/apache2/htdocs/ httpd
0ba4bd685f5815541eb318f124855f9cda8bc032e6b96dcc2c95cc2318a5fa1
leonor@Ubuntu22Docker:~$ sudo docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED              STATUS              PORTS
0ba4bd685f58   httpd          "httpd-foreground"       3 seconds ago        Up 3 seconds        0.0.0.0:8080->80/tcp, :::8080->80/t
cp   my-apache-app
8d6bd7ecbc09   hello-world    "/hello"                 About an hour ago    Exited (0) About an hour ago
quirky_wescoff
leonor@Ubuntu22Docker:~$
```

Y en el navegador, si ponemos la IP:Puerto vemos ejecutándose apache:



Index of /

Si creamos un archivo dentro de la carpeta apache "servidor1" nos aparecerá en el navegador:

```
leonor@Ubuntu22Docker:~/apache$ ls
my-httpd.conf  servidor1
leonor@Ubuntu22Docker:~/apache$ cat servidor1
Este es el servidor 1
leonor@Ubuntu22Docker:~/apache$
```



No es seguro | 192.168.10.157:8080

Index of /

- [my-httpd.conf](#)
- [servidor1](#)

Si pinchamos sobre servidor1 nos aparece lo que hay en el archivo:



No es seguro | 192.168.10.157:8080/servidor1

Este es el servidor 1

De esta manera si en el servidor 2 ponemos un archivo que se llame "servidor2" podremos distinguir entre ambos servidores y saber cuál nos está dando el servicio a la hora de hacer comprobaciones.

Y si pinchamos sobre my-http.conf tenemos el archivo de configuración de Apache:



No es seguro | 192.168.10.157:8080/my-httpd.conf

```
#
# This is the main Apache HTTP server configuration file. It contains the
# configuration directives that give the server its instructions.
# See <URL:http://httpd.apache.org/docs/2.4/> for detailed information.
# In particular, see
# <URL:http://httpd.apache.org/docs/2.4/mod/directives.html>
# for a discussion of each configuration directive.
#
# Do NOT simply read the instructions in here without understanding
# what they do. They're here only as hints or reminders. If you are unsure
# consult the online docs. You have been warned.
#
# Configuration and logfile names: If the filenames you specify for many
# of the server's control files begin with "/" (or "drive:/" for Win32), the
# server will use that explicit path. If the filenames do *not* begin
# with "/", the value of ServerRoot is prepended -- so "logs/access_log"
# with ServerRoot set to "/usr/local/apache2" will be interpreted by the
# server as "/usr/local/apache2/logs/access_log", whereas "/logs/access_log"
# will be interpreted as '/logs/access_log'.
#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
# Do not add a slash at the end of the directory path. If you point
# ServerRoot at a non-local disk, be sure to specify a local disk on the
# Mutex directive, if file-based mutexes are used. If you wish to share the
# same ServerRoot for multiple httpd daemons, you will need to change at
# least PidFile.
#
ServerRoot "/usr/local/apache2"
```

En el caso en el que quisiéramos entrar en el contenedor docker:

```
docker exec -ti CONTAINER_ID /bin/bash
```

Aquí vemos que se ha hecho un ls dentro de la carpeta de configuración del contenedor y vemos que es igual a la que tenemos en nuestra máquina sin haber copiado nada den-

tro del contenedor:

```
leonor@ubuntu22docker:~$ sudo docker ps -a
[sudo] contraseña para leonor:
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAME
0ba4bd685f58  httpd         "httpd-foreground"     41 minutes ago Up 41 minutes  0.0.0.0:8080->80/tcp, :::8080->80/tcp  my-a
pache-app
8d6bd7ecbc09  hello-world   "/hello"                2 hours ago   Exited (0) 2 hours ago                quir
ky_wescoff
leonor@ubuntu22docker:~$ sudo docker exec -ti 0ba4bd685f58 /bin/bash
root@0ba4bd685f58:/usr/local/apache2# ls
bin  build  cgi-bin  conf  error  htdocs  icons
root@0ba4bd685f58:/usr/local/apache2# cd htdocs
root@0ba4bd685f58:/usr/local/apache2/htdocs# ls
my-httpd.conf  servidor1
root@0ba4bd685f58:/usr/local/apache2/htdocs#
```

Para parar la ejecución del contenedor:

sudo docker stop CONTAINER_ID

```
root@0ba4bd685f58:/usr/local/apache2/htdocs# ls
my-httpd.conf  servidor1
root@0ba4bd685f58:/usr/local/apache2/htdocs# exit
exit
leonor@ubuntu22docker:~$ sudo docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAME
0ba4bd685f58  httpd         "httpd-foreground"     45 minutes ago Up 45 minutes  0.0.0.0:8080->80/tcp, :::8080->80/tcp  my-a
pache-app
8d6bd7ecbc09  hello-world   "/hello"                2 hours ago   Exited (0) 2 hours ago                quir
ky_wescoff
leonor@ubuntu22docker:~$ sudo docker stop 0ba4bd685f58
0ba4bd685f58
leonor@ubuntu22docker:~$ sudo docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAMES
0ba4bd685f58  httpd         "httpd-foreground"     46 minutes ago Exited (0) 2 seconds ago            my-apache-app
8d6bd7ecbc09  hello-world   "/hello"                2 hours ago   Exited (0) 2 hours ago            quirky_wescoff
leonor@ubuntu22docker:~$
```

Para iniciar la ejecución otra vez del container:

sudo docker start CONTAINER_ID

```
leonor@ubuntu22docker:~$ sudo docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAMES
0ba4bd685f58  httpd         "httpd-foreground"     46 minutes ago Exited (0) 2 seconds ago            my-apache-app
8d6bd7ecbc09  hello-world   "/hello"                2 hours ago   Exited (0) 2 hours ago            quirky_wescoff
leonor@ubuntu22docker:~$ sudo docker start 0ba4bd685f58
0ba4bd685f58
leonor@ubuntu22docker:~$ sudo docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAME
0ba4bd685f58  httpd         "httpd-foreground"     49 minutes ago Up 1 second    0.0.0.0:8080->80/tcp, :::8080->80/tcp  my-a
pache-app
8d6bd7ecbc09  hello-world   "/hello"                2 hours ago   Exited (0) 2 hours ago                quir
ky_wescoff
leonor@ubuntu22docker:~$
```

Para ver los logs del container:

sudo docker logs CONTAINER_ID

```
leonor@ubuntu22docker:~$ sudo docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAME
0ba4bd685f58  httpd         "httpd-foreground"     49 minutes ago Up 1 second    0.0.0.0:8080->80/tcp, :::8080->80/tcp  my-a
pache-app
8d6bd7ecbc09  hello-world   "/hello"                2 hours ago   Exited (0) 2 hours ago                quir
ky_wescoff
leonor@ubuntu22docker:~$ sudo docker logs 0ba4bd685f58
AH00558: httpd: could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message
AH00558: httpd: could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message
[Sat Jan 07 17:33:50.468605 2023] [mpm_event:notice] [pid 1:tid 140683419217216] AH00489: Apache/2.4.54 (Unix) configured -- resuming no
rmal operations
[Sat Jan 07 17:33:50.468932 2023] [core:notice] [pid 1:tid 140683419217216] AH00094: Command line: 'httpd -D FOREGROUND'
192.168.10.114 - - [07/Jan/2023:17:33:58 +0000] "GET / HTTP/1.1" 200 161
192.168.10.114 - - [07/Jan/2023:17:39:16 +0000] "GET / HTTP/1.1" 200 161
192.168.10.114 - - [07/Jan/2023:17:39:16 +0000] "GET / HTTP/1.1" 200 161
192.168.10.114 - - [07/Jan/2023:17:39:16 +0000] "GET / HTTP/1.1" 200 161
192.168.10.114 - - [07/Jan/2023:17:39:17 +0000] "GET / HTTP/1.1" 200 161
192.168.10.114 - - [07/Jan/2023:17:43:04 +0000] "GET / HTTP/1.1" 200 205
192.168.10.114 - - [07/Jan/2023:17:43:05 +0000] "GET /servidor1 HTTP/1.1" 200 22
192.168.10.114 - - [07/Jan/2023:17:45:15 +0000] "GET /servidor1 HTTP/1.1" 304 -
192.168.10.114 - - [07/Jan/2023:17:59:57 +0000] "GET / HTTP/1.1" 200 257
192.168.10.114 - - [07/Jan/2023:17:59:58 +0000] "GET / HTTP/1.1" 200 257
192.168.10.114 - - [07/Jan/2023:17:59:58 +0000] "GET / HTTP/1.1" 200 257
192.168.10.114 - - [07/Jan/2023:18:05:36 +0000] "GET / HTTP/1.1" 200 257
192.168.10.114 - - [07/Jan/2023:18:05:37 +0000] "GET / HTTP/1.1" 200 257
192.168.10.114 - - [07/Jan/2023:18:05:38 +0000] "GET / HTTP/1.1" 200 257
192.168.10.114 - - [07/Jan/2023:18:05:39 +0000] "GET / HTTP/1.1" 200 257
192.168.10.114 - - [07/Jan/2023:18:05:39 +0000] "GET / HTTP/1.1" 200 257
192.168.10.114 - - [07/Jan/2023:18:05:40 +0000] "GET /my-httpd.conf HTTP/1.1" 200 20827
```

Ahora clonamos la máquina para tener 2 máquinas iguales con docker instalado y la imagen httpd en docker también.

- En la tercera máquina se instala la imagen docker de nginx.
<https://hub.docker.com/search?q=nginx>

Nos descargamos la imagen oficial docker de nginx:

hub.docker.com/search?q=nginx

Wasm is a fast, light alternative to Linux containers – try it out today with the [Docker+Wasm](#)

docker hub

1 - 25 of 10.000 results for **nginx**.

- ts
- images
- extensions
- plugins
- Content
- [Docker Official Image](#)
- [Verified Publisher](#)
- [Sponsored OSS](#)
- ing Systems
- .linux

nginx **DOCKER OFFICIAL IMAGE**
Updated 17 days ago
Official build of Nginx.
Linux x86-64 ARM ARM 64 386 mips64le PowerPC 64 LE IBM Z

bitnami/nginx **VERIFIED PUBLISHER**
By Bitnami • Updated 8 days ago
Bitnami nginx Docker Image
Linux x86-64

En el terminal de la máquina para instalarla escribimos:
sudo docker pull nginx

docker hub

Explore > Official Images > nginx

nginx **DOCKER OFFICIAL IMAGE** • 1B+ • 10K+
Official build of Nginx.

Overview Tags

Quick reference



```
leonor@Ubuntu22Docker:~$ sudo docker pull nginx
[sudo] contraseña para leonor:
Using default tag: latest
latest: Pulling from library/nginx
3f4ca61aafcd: Pull complete
50c68654b16f: Pull complete
3ed295c083ec: Pull complete
40b838968eea: Pull complete
88d3ab68332d: Pull complete
5f63362a3fa3: Pull complete
Digest: sha256:0047b729188a15da49380d9506d65959cce6d40291ccfb4e039f5dc7efd33286
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
leonor@Ubuntu22Docker:~$
```

Para ejecutar el contenedor de nginx:

`docker run --name SAD_nginx -d -p 7070:80 nginx`

```
leonor@Ubuntu22Docker:~$ sudo docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS    NAMES
8d6bd7ecbc09   hello-world   "/hello"                3 hours ago   Exited (0) 3 hours ago    quirkyy_wescoff
leonor@Ubuntu22Docker:~$ sudo docker run --name SAD_nginx -d -p 7070:80 nginx
83327a4ef0f5ee70d64a16eaadde3e962be2168ccc9b7ba440e5e7d072a1688
leonor@Ubuntu22Docker:~$ sudo docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS    NAMES
83327a4ef0f5   nginx    "/docker-entrypoint..." 2 seconds ago   Up 1 second    0.0.0.0:7070->80/tcp, :::7070->80/tcp   SAD_nginx
8d6bd7ecbc09   hello-world   "/hello"                3 hours ago   Exited (0) 3 hours ago    quirkyy_wescoff
leonor@Ubuntu22Docker:~$
```

En el navegador comprobamos que se está ejecutando nginx:

No es seguro | 192.168.10.193:7070

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Si queremos inspeccionar algún contenedor:

`sudo docker inspect CONTAINER_ID`

```

leonor@Ubuntu22Docker:~$ sudo docker inspect 83327a4ef0f5
[
  {
    "Id": "83327a4ef0f5eee70d64a16eaadde3e962be2168ccc9b7ba440e5e7d872a1688",
    "Created": "2023-01-07T19:57:53.039628679Z",
    "Path": "/docker-entrypoint.sh",
    "Args": [
      "nginx",
      "-g",
      "daemon off;"
    ],
    "State": {
      "Status": "running",
      "Running": true,
      "Paused": false,
      "Restarting": false,
      "OOMKilled": false,
      "Dead": false,
      "Pid": 3830,
      "ExitCode": 0,
      "Error": "",
      "StartedAt": "2023-01-07T19:57:53.45748658Z",
      "FinishedAt": "0001-01-01T00:00:00Z"
    },
    "Image": "sha256:1403e55ab369cd1c8039c34e6b4d47ca40bbde39c371254c7cba14756f...",
    "ResolvConfPath": "/var/lib/docker/containers/83327a4ef0f5eee70d64a16eaadde3e962be2168ccc9b7ba440e5e7d872a1688/resolv.conf",
    "HostnamePath": "/var/lib/docker/containers/83327a4ef0f5eee70d64a16eaadde3e962be2168ccc9b7ba440e5e7d872a1688/hostname",
    "HostsPath": "/var/lib/docker/containers/83327a4ef0f5eee70d64a16eaadde3e962be2168ccc9b7ba440e5e7d872a1688/hosts",
    "LogPath": "/var/lib/docker/containers/83327a4ef0f5eee70d64a16eaadde3e962be2168ccc9b7ba440e5e7d872a1688-json.log",
    "Name": "/SAD_nginx",
    "RestartCount": 0,
    "Driver": "overlay2",
    "Platform": "linux",
    "MountLabel": ""
  }
]

```

Mapeamos el archivo de configuración de nginx para poder modificarlo desde nuestra máquina, primero lo copiamos, lo modificamos y después lo mapeamos:
`sudo docker exec CONTAINER_ID cat /etc/nginx/conf.d/default.conf > default.conf`

```

leonor@Ubuntu22Docker:~$ sudo docker ps -a
[sudo] contraseña para leonor:
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS              PORTS                               NAMES
83327a4ef0f5   nginx    "/docker-entrypoint..." 27 minutes ago Up 27 minutes      0.0.0.0:7070->80/tcp, :::7070->80/tcp SAD_nginx
8d6bd7ecbc09   hello-world "/hello"                4 hours ago   Exited (0) 4 hours ago               quirky_w
scoff

leonor@Ubuntu22Docker:~$ ls
Descargas  Documentos  Escritorio  Imágenes  Música  Plantillas  Publico  snap  Videos
leonor@Ubuntu22Docker:~$ mkdir nginx
leonor@Ubuntu22Docker:~$ cd nginx/
leonor@Ubuntu22Docker:~/nginx$ ls
leonor@Ubuntu22Docker:~/nginx$ sudo docker exec CONTAINER_ID cat /etc/nginx/conf.d/default.conf > default.conf
Error: No such container: CONTAINER_ID
leonor@Ubuntu22Docker:~/nginx$ sudo docker exec 83327a4ef0f5 cat /etc/nginx/conf.d/default.conf > default.conf
leonor@Ubuntu22Docker:~/nginx$ ls
default.conf
leonor@Ubuntu22Docker:~/nginx$

```

`sudo docker run -dit --name SAD_nginx -p 7070:80 -v "$PWD"/nginx:/etc/nginx/conf.d/nginx`

Abrimos el archivo default.conf:



```
GNU nano 6.2 default.conf
server {
    listen 80;
    listen [::]:80;
    server_name localhost;

    #access_log /var/log/nginx/host.access.log main;

    location / {
        root /usr/share/nginx/html;
        index index.html index.htm;
    }

    #error_page 404 /404.html;

    # redirect server error pages to the static page /50x.html
    #
    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
        root /usr/share/nginx/html;
    }

    # proxy the PHP scripts to Apache listening on 127.0.0.1:80
    #
    #location ~ \.php$ {
    #    proxy_pass http://127.0.0.1;
    #}

    # pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000
    #
    #location ~ \.php$ {
    #    root html;
    #    fastcgi_pass 127.0.0.1:9000;
    #    fastcgi_index index.php;
    #    fastcgi_param SCRIPT_FILENAME /scripts$fastcgi_script_name;

```

Debemos borrar todo lo que hay y poner:

```
upstream backend {
    server IP_server1;
    server IP_server2;
}
server {
    listen 80;
    listen [::]:80;
    server_name SAD_nginx;
    location / {
        proxy_pass http://backend;
    }
}
```

```

GNU nano 6.2                                     default.conf
↑ upstream backend {
    server 192.168.10.157:8080;
    server 192.168.10.197:8080;
}
server {
    listen 80;
    listen [::]:80;
    server_name SAD_nginx;
    location / {
        proxy_pass http://backend;
    }
}

```

De esta manera hacemos el balanceo de carga.

Cuando tocamos el archivo de configuración de nginx, para saber que no lo hemos estropeado y sigue bien:

`sudo docker exec -ti CONTAINER_ID nginx -t`

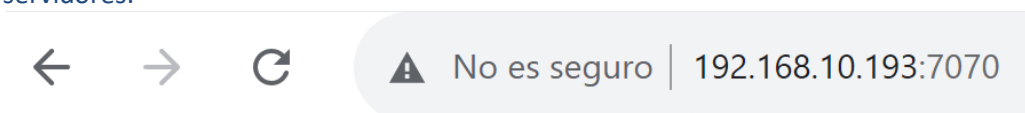
```

leonor@Ubuntu22Docker:~/nginx$ sudo docker exec -ti 83327a4ef0f5 nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
leonor@Ubuntu22Docker:~/nginx$ █

```

5. Ejecuta en cada máquina su contenedor docker y comprueba que se hace el balanceo de carga entre cada uno de los servidores. Comprueba también que si se cae uno de los servidores el otro sigue funcionando y dando servicio.

Si en el navegador ponemos la IP de nginx ahora nos carga una de las páginas de los servidores:



Index of /

- [my-httpd.conf](#)
- [servidor1](#)

Si recargamos la página podremos ver que nos sale también el otro servidor:



⚠ No es seguro | 192.168.10.193:7070

Index of /

- [my-httpd.conf](#)
- [servidor2](#)

Si paramos el servidor1 entonces siempre carga el servidor2 (y viceversa siempre cargaría el servidor1):



⚠ No es seguro | 192.168.10.193:7070

Index of /

- [my-httpd.conf](#)
- [servidor2](#)

Index of /



⚠ No es seguro | 192.168.10.197:8080



Index of /

- [my-httpd.conf](#)
- [servidor2](#)

192.168.10.157



ⓘ 192.168.10.157:8080



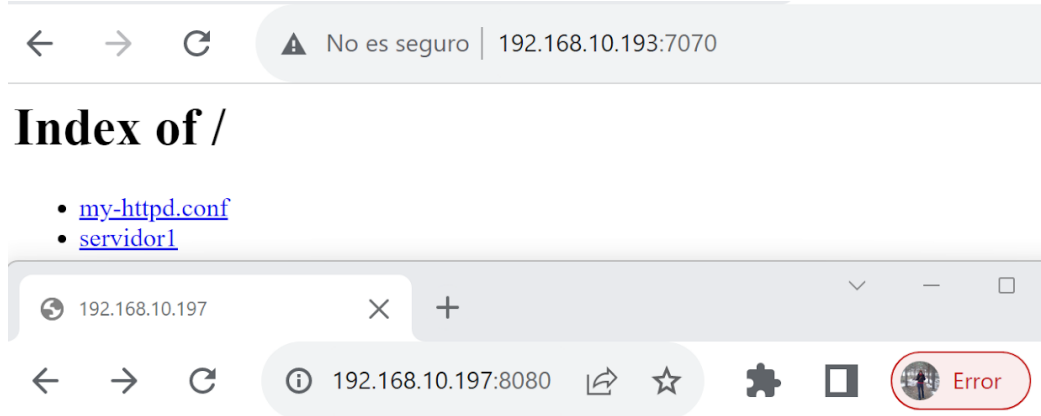
No se puede acceder a este sitio web

La página **192.168.10.157** ha rechazado la conexión.

Prueba a:

- Comprobar la conexión
- [Comprobar el proxy y el certificado](#)

En el caso de parar el servidor2:



No se puede acceder a este sitio web

La página **192.168.10.197** ha rechazado la conexión.

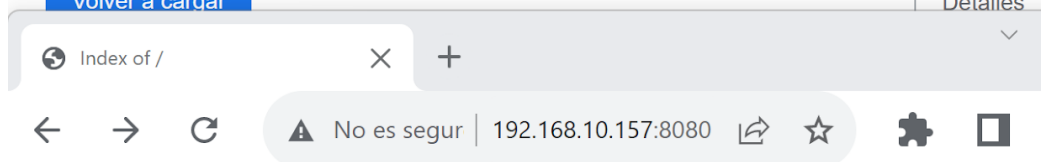
Prueba a:

- Comprobar la conexión
- [Comprobar el proxy y el cortafuegos](#)

ERR_CONNECTION_REFUSED

[Volver a cargar](#)

[Detalles](#)



Index of /

- [my-httpd.conf](#)
- [servidor1](#)

Hay que tener en cuenta que las páginas deben ser exactamente iguales, para hacer este ejercicio y comprobar el balanceo se ha puesto en cada servidor su número pero es sólo de prueba.

6. Si en vez de Nginx usamos HAProxy:

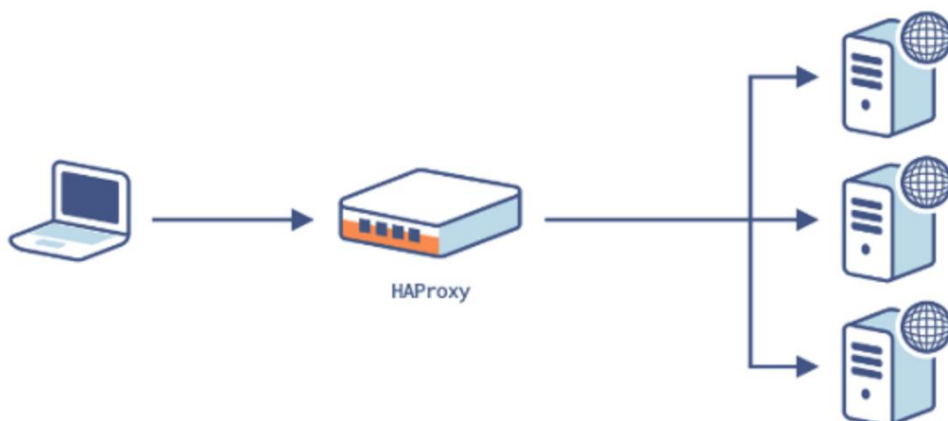
https://hub.docker.com/_/haproxy

<https://docs.haproxy.org/>

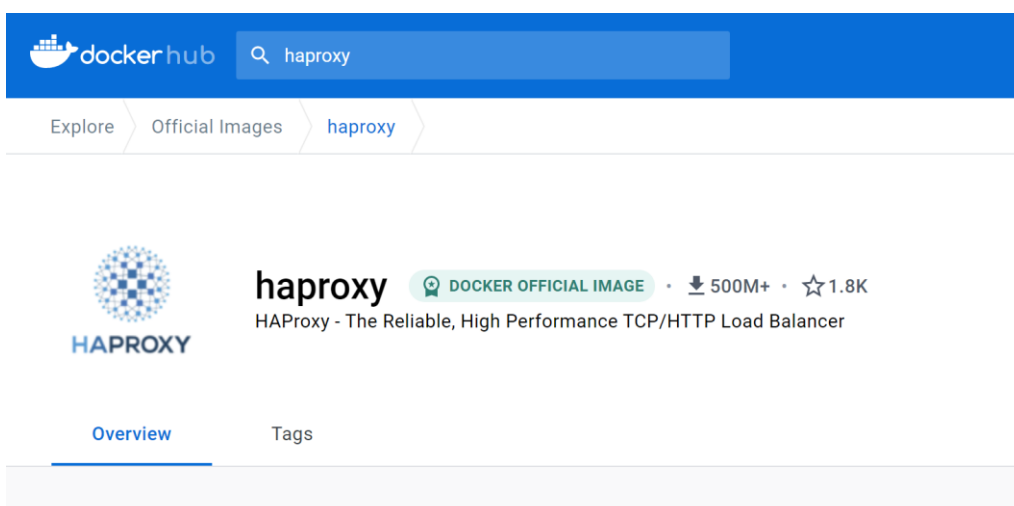
<https://www.it-civil.com/configurar-haproxy-docker/>

<https://vdzon.wordpress.com/2015/03/31/using-haproxy-for-loadbalancing-2-web-servers-in-docker/>

<https://www.youtube.com/watch?v=f9V-aHuMI2o&t=12s>



Buscamos la imagen docker de HAProxy en el docker hub:



Instalamos HAProxy con:

`sudo docker pull haproxy`

```
leonor@Ubuntu22Docker:~$ sudo docker pull haproxy
Using default tag: latest
latest: Pulling from library/haproxy
8740c948ffd4: Pull complete
388da74d1179: Pull complete
8cbaffeba70d: Pull complete
1a06058dcc4d: Pull complete
Digest: sha256:4f79e6112b2a2fba850e842a6c457bc80a2064ad573bfafafd1ed2df64caab30
Status: Downloaded newer image for haproxy:latest
docker.io/library/haproxy:latest
leonor@Ubuntu22Docker:~$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
haproxy       latest   f9a9dda01072   4 days ago    103MB
hello-world   latest   feb5d9fea6a5   16 months ago 13.3kB
```

Y comprobamos que tenemos la imagen con:

`sudo docker images`

Para arrancar el contenedor tenemos que emplear el comando:

`sudo docker run -itd -p 80:80 --name haproxy -v`

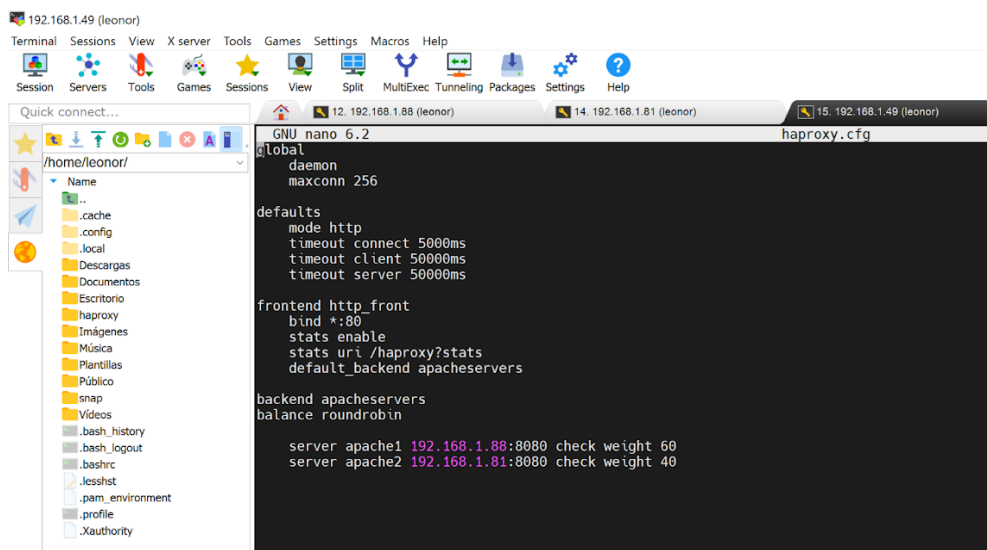
`"$PWD"/haproxy:/usr/local/etc/haproxy:ro --sysctl`

net.ipv4.ip_unprivileged_port_start=0 haproxy

La secuencia de parámetros es la siguiente:

Parámetro	Descripción
-p	Mapeo de puerto 80 de la PC física con el puerto 80 de la ip del contenedor que tiene el haproxy, para que al momento de ingresar la ip de la PC física desde cualquier IP de la misma red por medio de un browser mostrará el servicio del HAProxy.
-name	Nombre del contenedor
-v	Mapeo del directorio local de la máquina física donde está el archivo haproxy.cfg hacia el directorio del contenedor.
-sysctl	Parametrización de kernel para ipv4

Pero primero hay que crear el archivo de configuración de haproxy que estará en `/usr/local/etc/haproxy/` dentro del contenedor, por lo que mapearemos el archivo también para no tener que estar entrando dentro del contenedor. El archivo haproxy.cfg quedará tal que así:



```

GNU nano 6.2 haproxy.cfg
global
  daemon
  maxconn 256

defaults
  mode http
  timeout connect 5000ms
  timeout client 50000ms
  timeout server 50000ms

frontend http_front
  bind *:80
  stats enable
  stats uri /haproxy?stats
  default_backend apacheservers

backend apacheservers
  balance roundrobin

  server apache1 192.168.1.88:8080 check weight 60
  server apache2 192.168.1.81:8080 check weight 40
  
```

Con la sentencia `bind *:80` le decimos al servidor haproxy que escuche por el puerto 80. En el backend ponemos los servidores para el balanceo con su IP y el puerto. Además podemos añadir el porcentaje del balanceo con `weight` por si tenemos servidores más potentes que otros y queremos distribuir la carga de otra forma:

server apache1 192.168.1.88:8080 check weight 60
server apache2 192.168.1.81:8080 check weight 40

Con la sentencia stats uri /haproxy?stats podremos ver las estadísticas del servidor haproxy en el navegador.

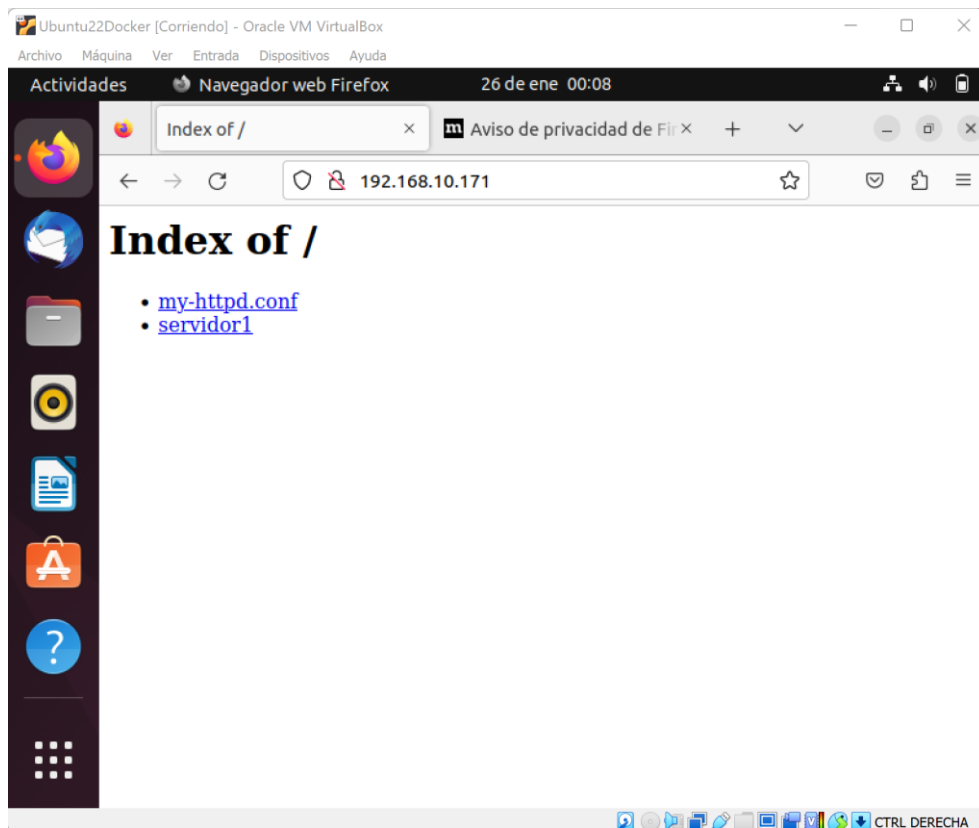
Y lo ubicamos dentro de una carpeta llamada haproxy en nuestra carpeta de usuario:

```
leonor@Ubuntu22Docker:~$ cd haproxy
leonor@Ubuntu22Docker:~/haproxy$ ls
haproxy.cfg
leonor@Ubuntu22Docker:~/haproxy$ pwd
/home/leonor/haproxy
leonor@Ubuntu22Docker:~/haproxy$
```

No olvidar que si desea editar el archivo haproxy.cfg para borrar, modificar o agregar parametros, puede utilizar la siguiente instrucción para reiniciar el servicio del contenedor.

`docker kill -s HUP srv_haproxy`

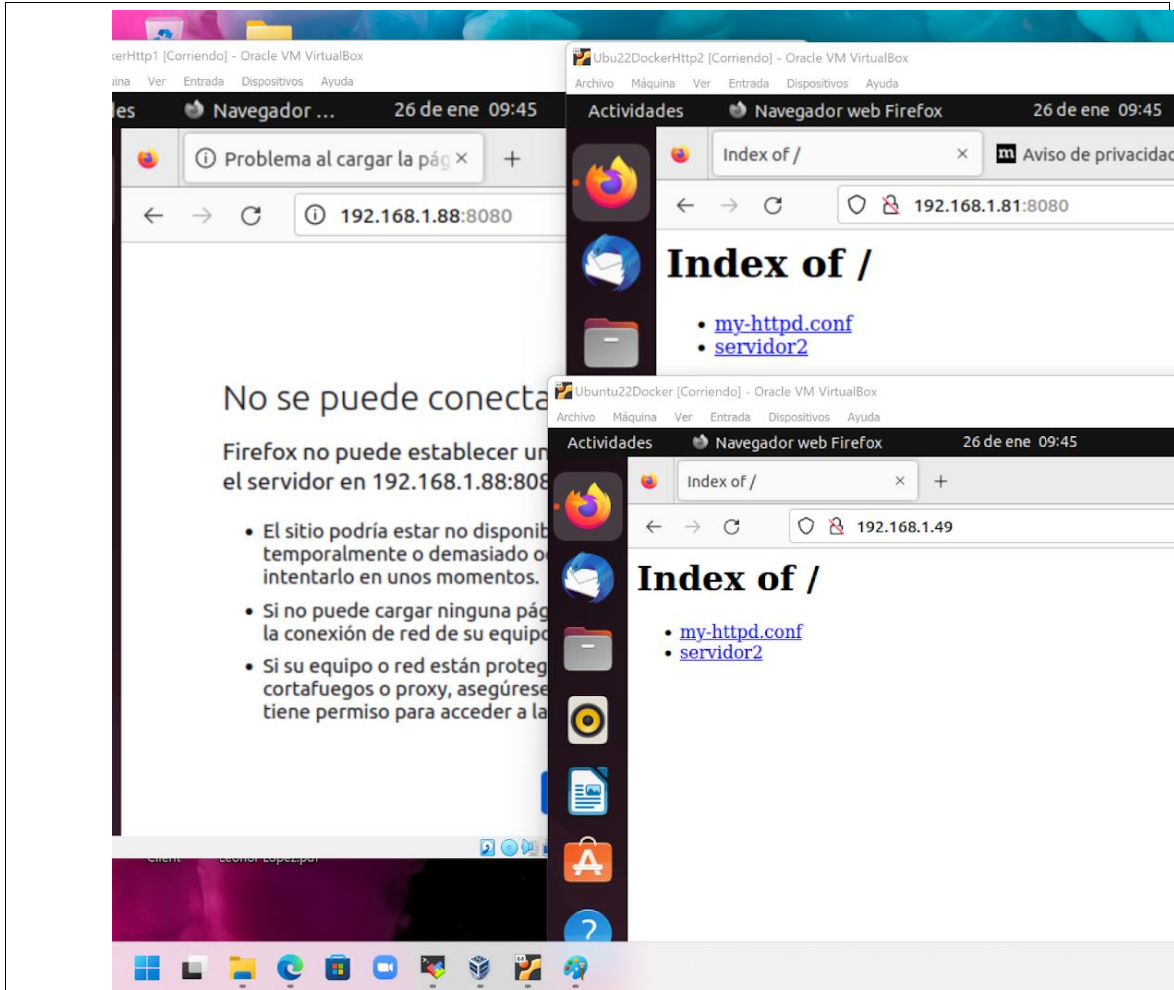
Si desde la máquina ejecutamos el navegador y ponemos la IP:80 entonces nos aparece uno de los servidores:



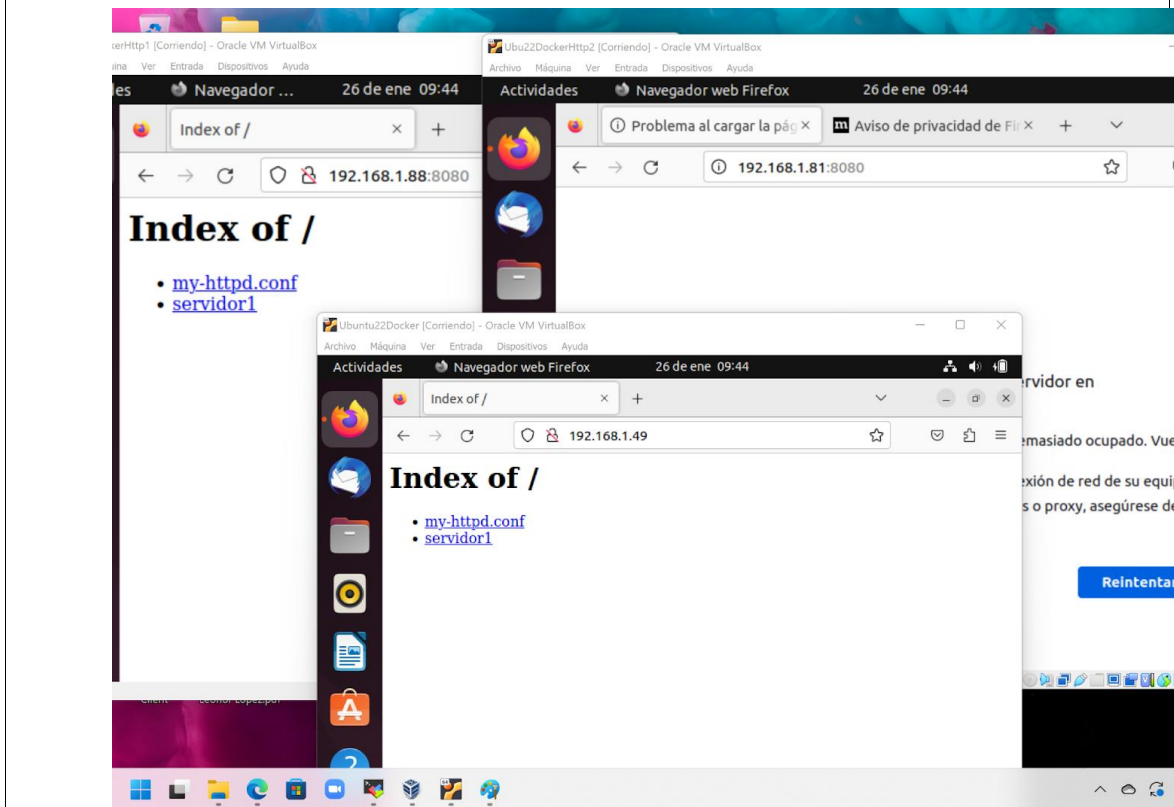
Al recargar la página nos aparece el otro:

Si paramos uno de los servidores entonces sólo funcionará el otro haciendo uso de la alta disponibilidad y del balanceo de carga.

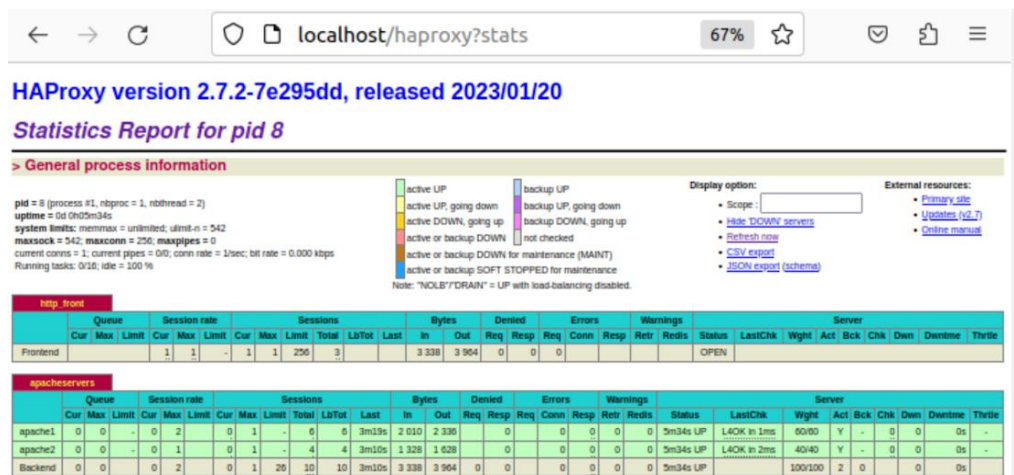
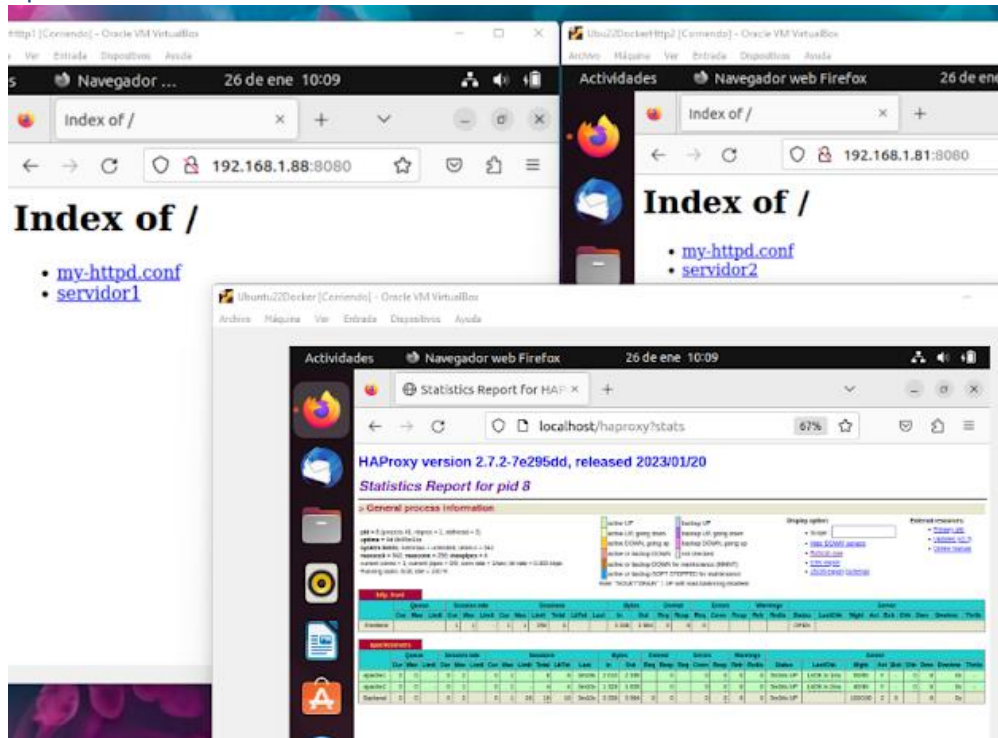
Si paramos el servidor apache 1:



Si paramos el servidor apache 2:



Para ver las estadísticas de los servidores, en el navegador escribimos:
<http://localhost/haproxy?stats>
Y podremos monitorear el estado de los servidores.



7. Realiza una memoria sobre la práctica, describiendo cada paso realizado en cada máquina con pantallazos de cada uno de los pasos y anotando los problemas encontrados. así como la forma en la que se han resuelto.

TALLER2: Instalación y configuración NAS Synology

- Un poco de lectura:
<https://www.xataka.com/basics/servidores-nas-que-como-funcionan-que-puedes-hacer-uno>
- Abre el NAS y conecta el disco duro. Seguimos las instrucciones dadas por la guía de instalación del fabricante <https://www.synology.com/es-es/support/download/DS120j?version=7.0#docs>

3. Conecta el NAS a la red y a la corriente y configura el dispositivo usando la página web de synology <https://finds.synology.com/>.

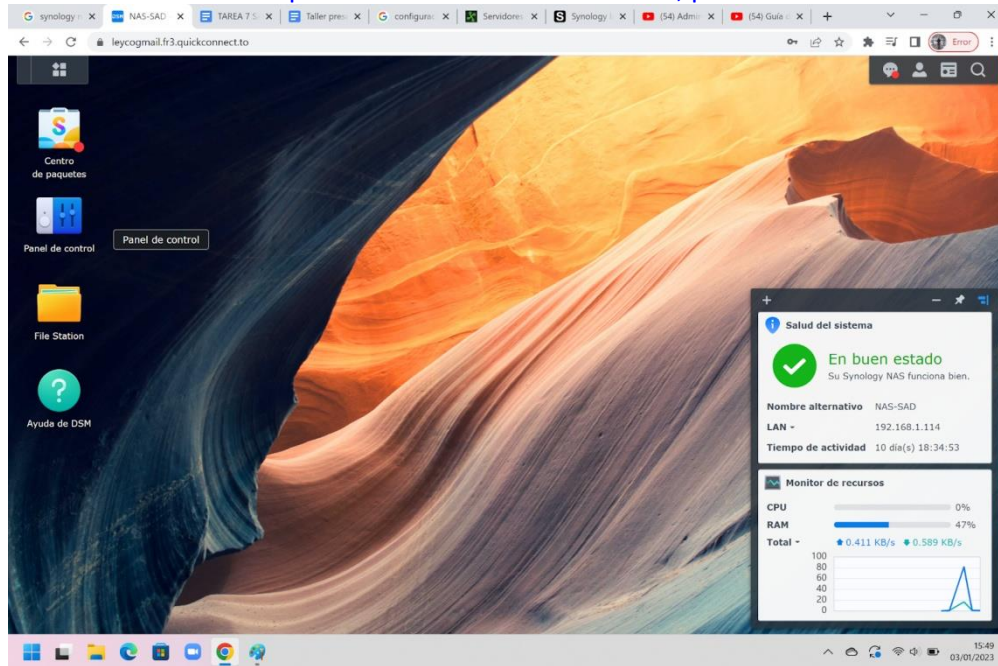
También puedes ayudarte de estos video tutoriales para la instalación del disco y configuración del NAS:

<https://www.youtube.com/watch?v=LqBU3Rh0uQg&t=2s>

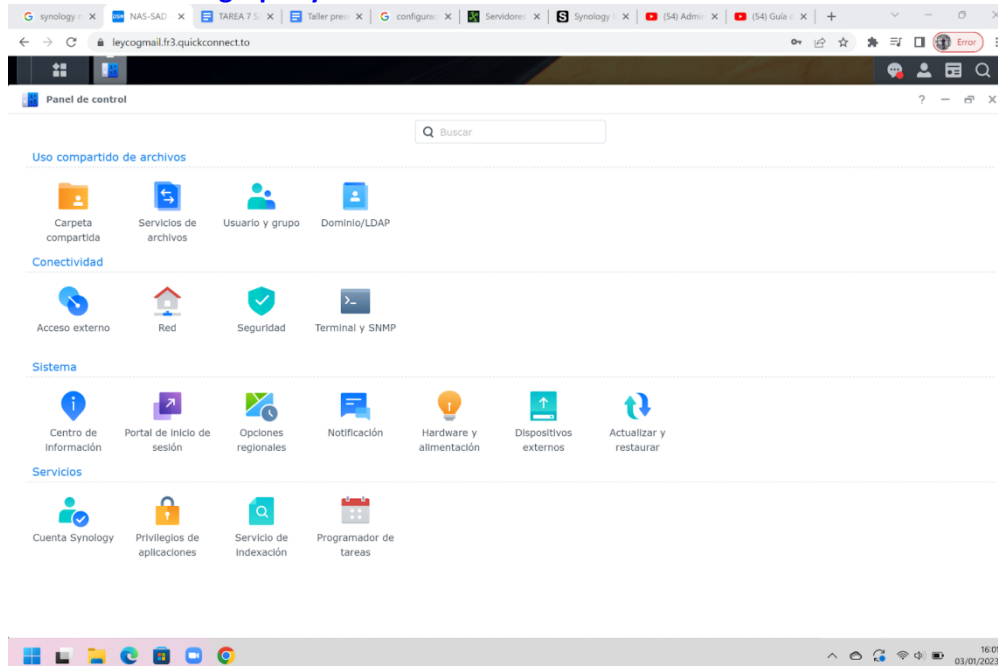
<https://www.youtube.com/watch?v=PrVAehzT8Yk>

4. Crea una o varias cuentas de usuario en el synology NAS.

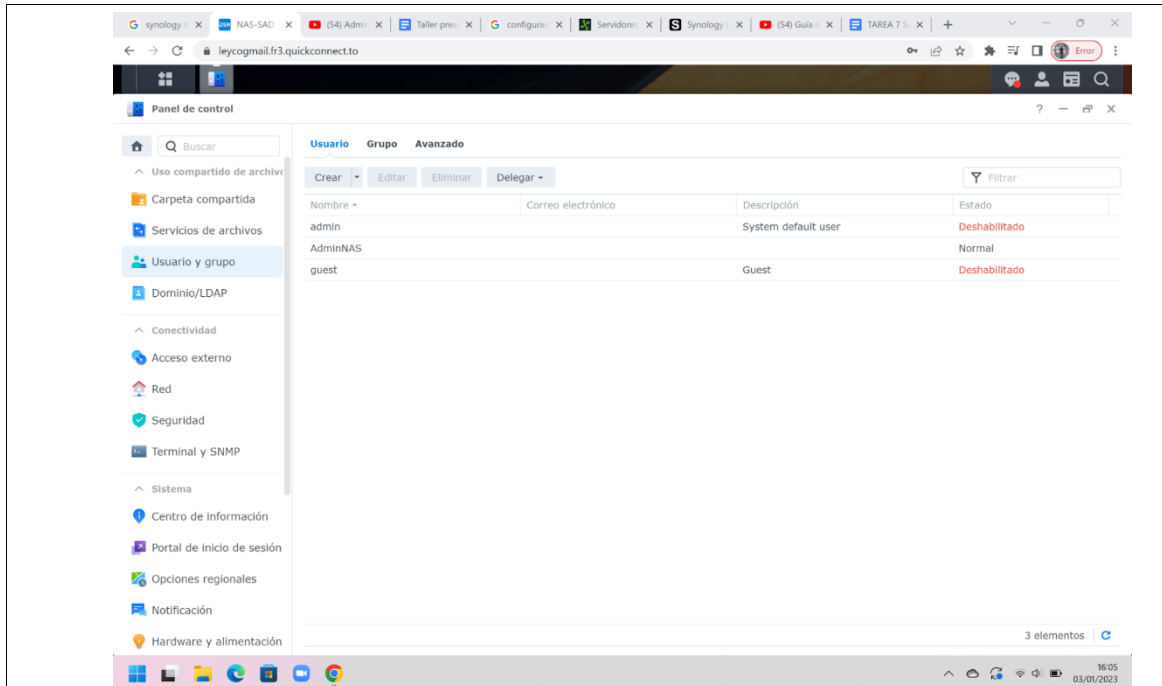
En el escritorio del NAS podemos ver el Panel de Control, pinchamos sobre él:



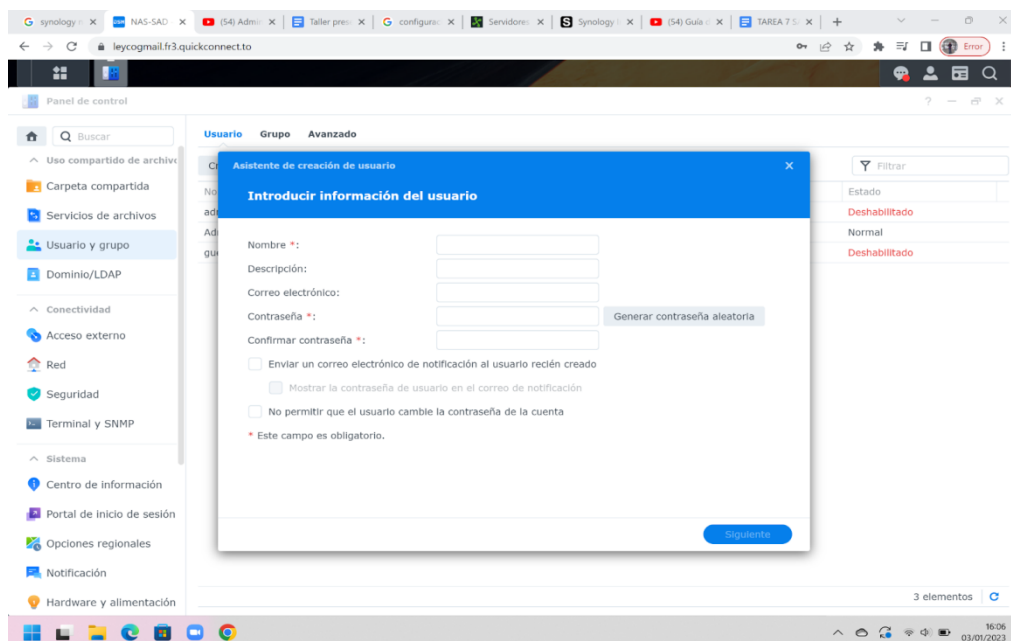
Pinchamos sobre grupos y usuarios:



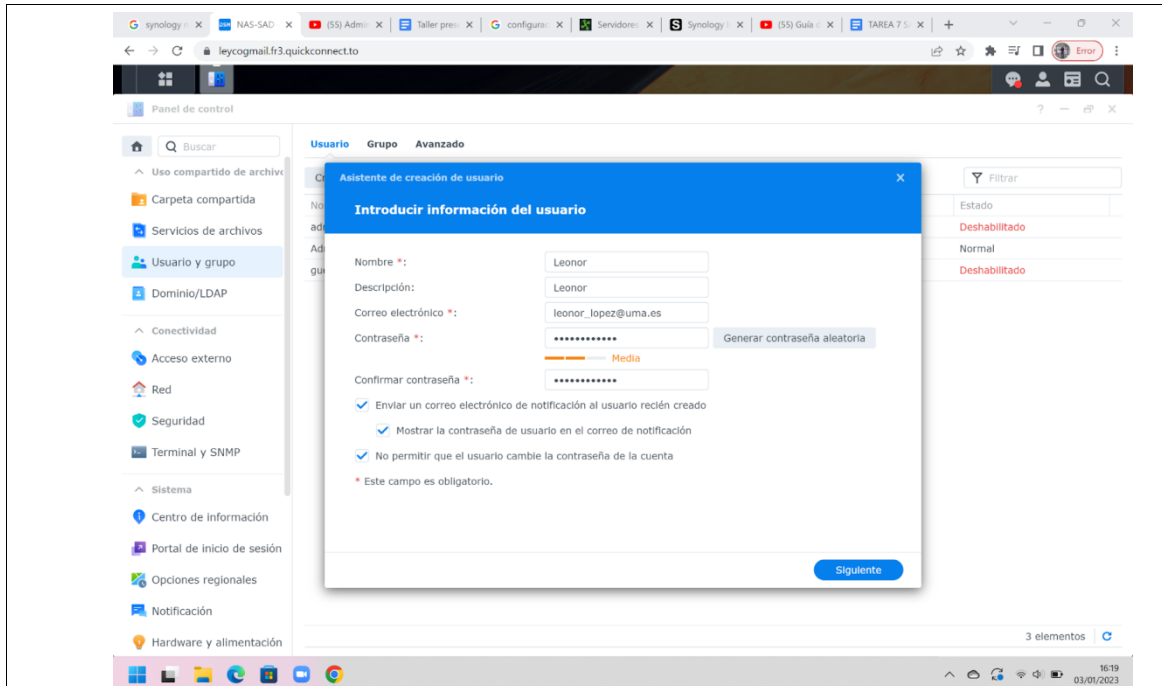
Se nos abrirá una ventana como ésta:



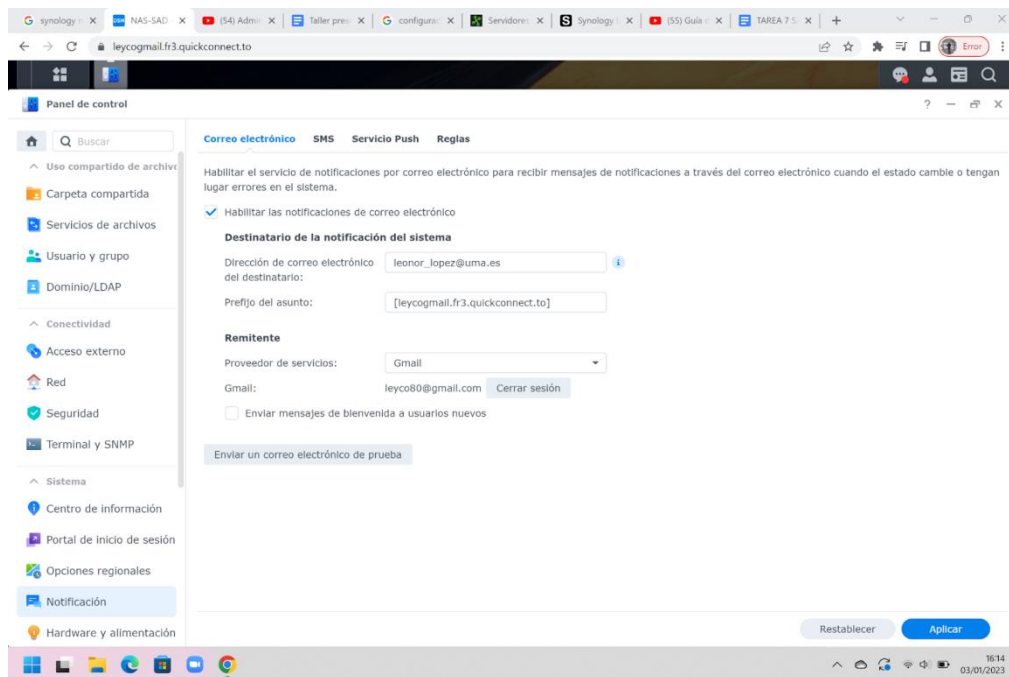
Pinchamos sobre “Crear”, debajo de Usuario y se nos abrirá otra ventana:



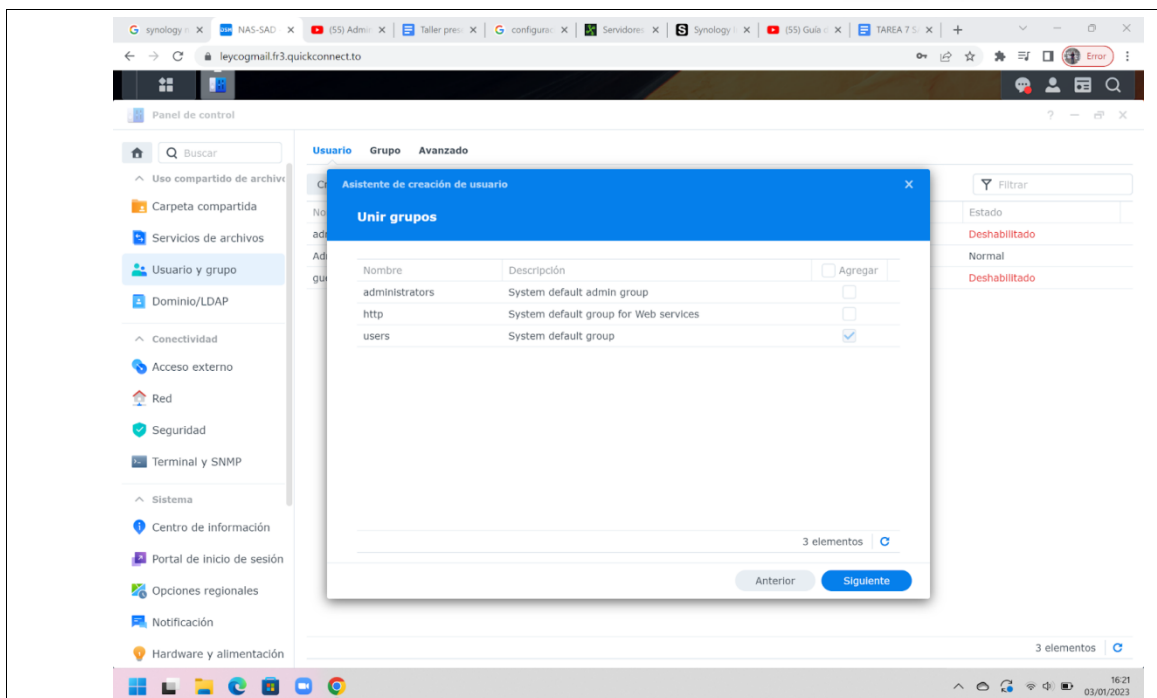
Re llena los campos que se piden para crear un nuevo usuario y pinchamos en siguiente:



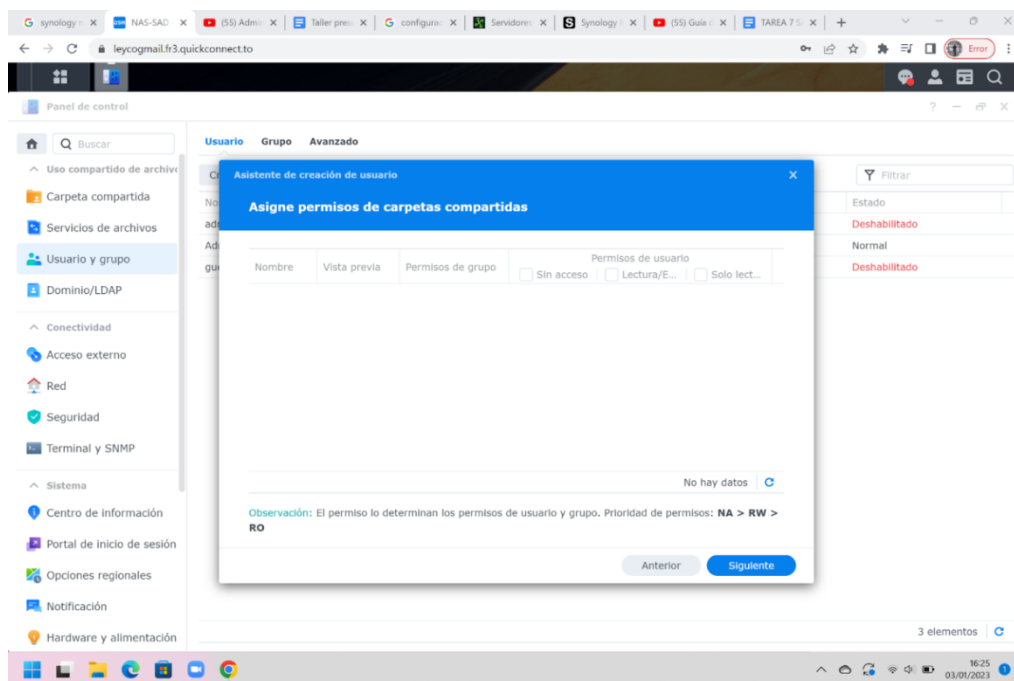
En notificaciones habilitamos las notificaciones de correo electrónico y rellenamos los campos pedidos para poder enviar notificaciones a los usuarios cuando se produzca algún cambio o error en el sistema, pinchamos en aplicar:



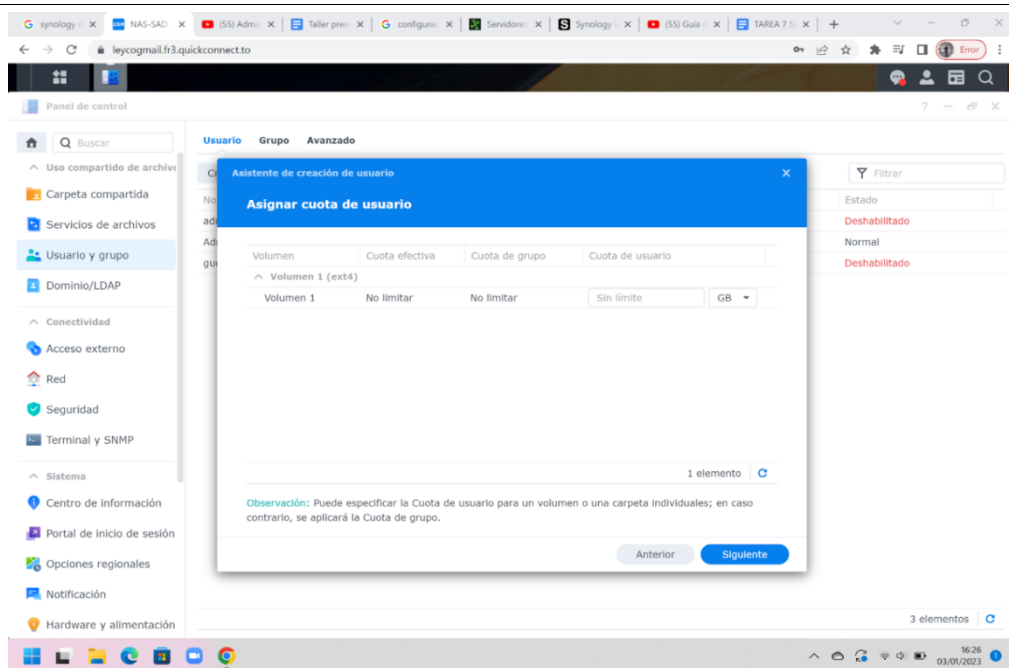
Dejamos al nuevo usuario en el grupo de "users" (system default group) y clicamos en siguiente:



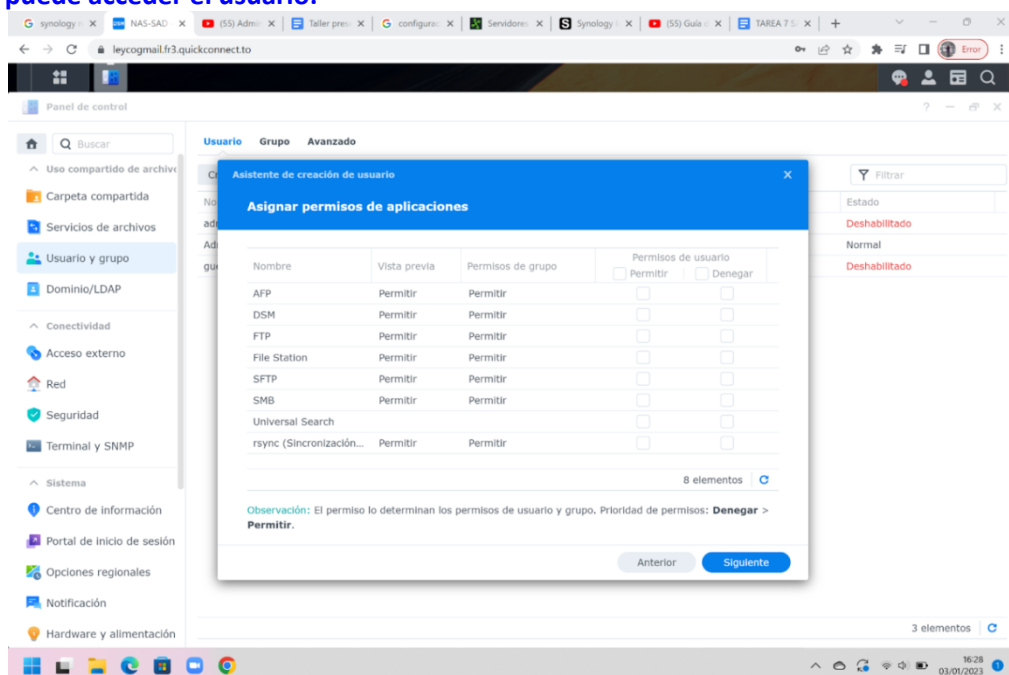
Asignamos permisos sobre las carpetas compartidas (si las hay):



Asignamos la cuota de espacio que va a tener el usuario en las carpetas:



Asignamos permisos sobre las aplicaciones, aquí podemos controlar a qué servicios puede acceder el usuario:



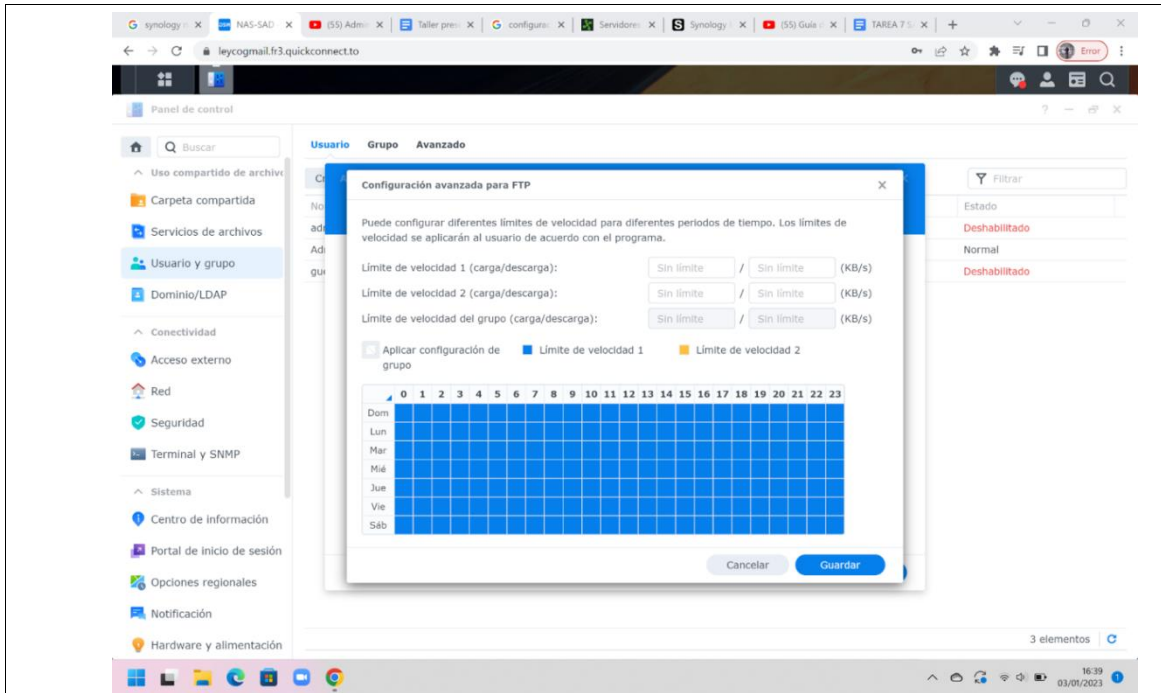
Establecemos el límite de velocidad del usuario para limitar el uso del ancho de banda del usuario cuando se transfieren archivos. Podemos aplicar la configuración que haya por defecto para el grupo al que pertenece el usuario, establecer una configuración de límite de subida y de bajada o usar una configuración avanzada donde podemos poner dos límites diferentes de velocidad a un mismo usuario:
Configuración de grupo:

Servicio	Resultado	Límite de velocidad	Límite de car...	Límite de des...
File Station	Sin límite / Sin...	Aplicar configuración de gr.▶	-	-
FTP	Sin límite / Sin...	Aplicar configuración de gr.▶	-	-
Rsync	Sin límite / Sin...	Aplicar configuración de gr.▶	-	-

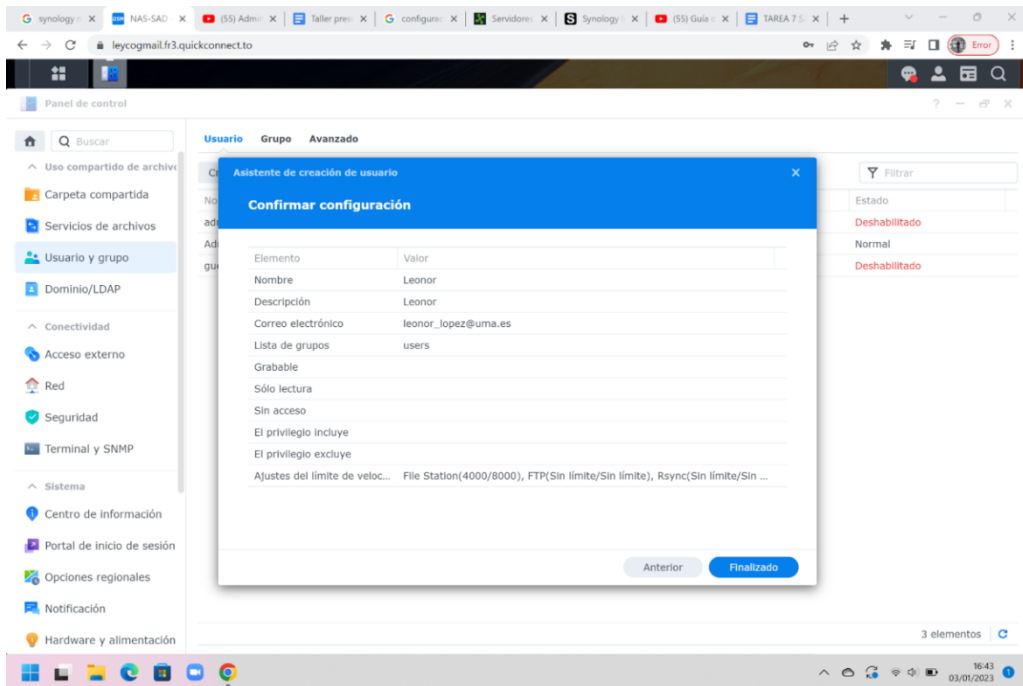
Configuración de límite de bajada y subida:

Servicio	Resultado	Límite de velocidad	Límite de car...	Límite de des...
File Station	4000 / 8000	Configurar el límite de velo.▶	4000	8000
FTP	Sin límite / Sin...	Aplicar configuración de gr.▶	-	-
Rsync	Sin límite / Sin...	Aplicar configuración de gr.▶	-	-

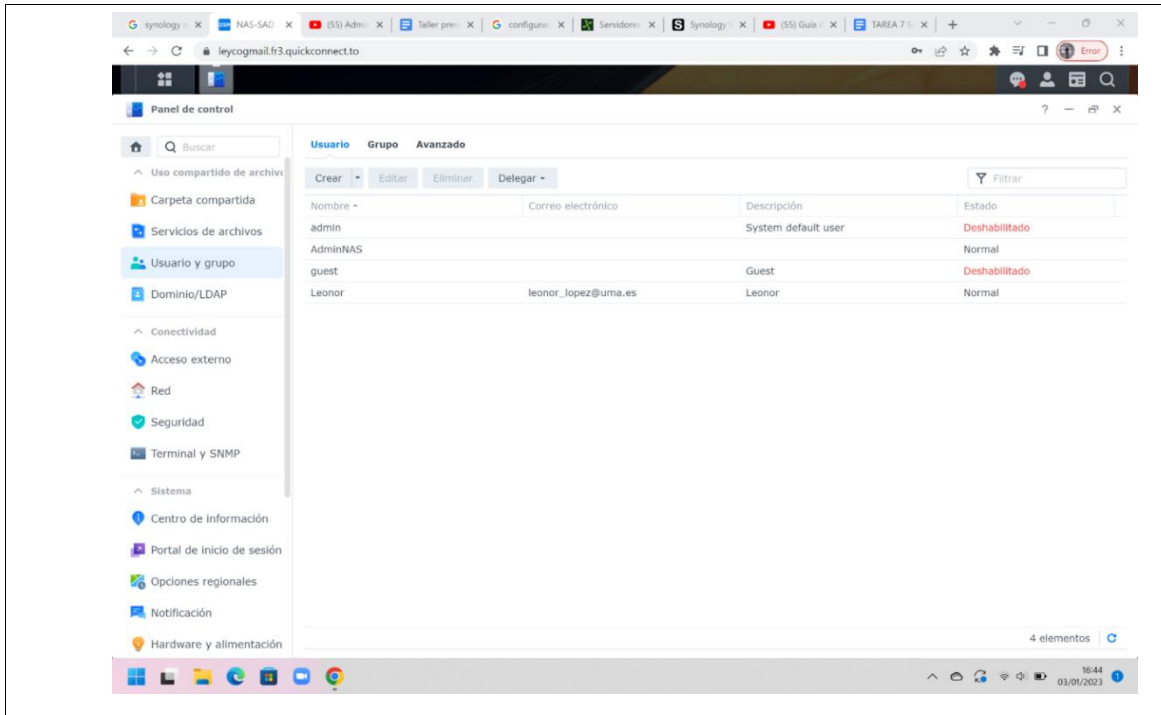
Configuración avanzada:



Pinchamos en aplicar para finalizar:

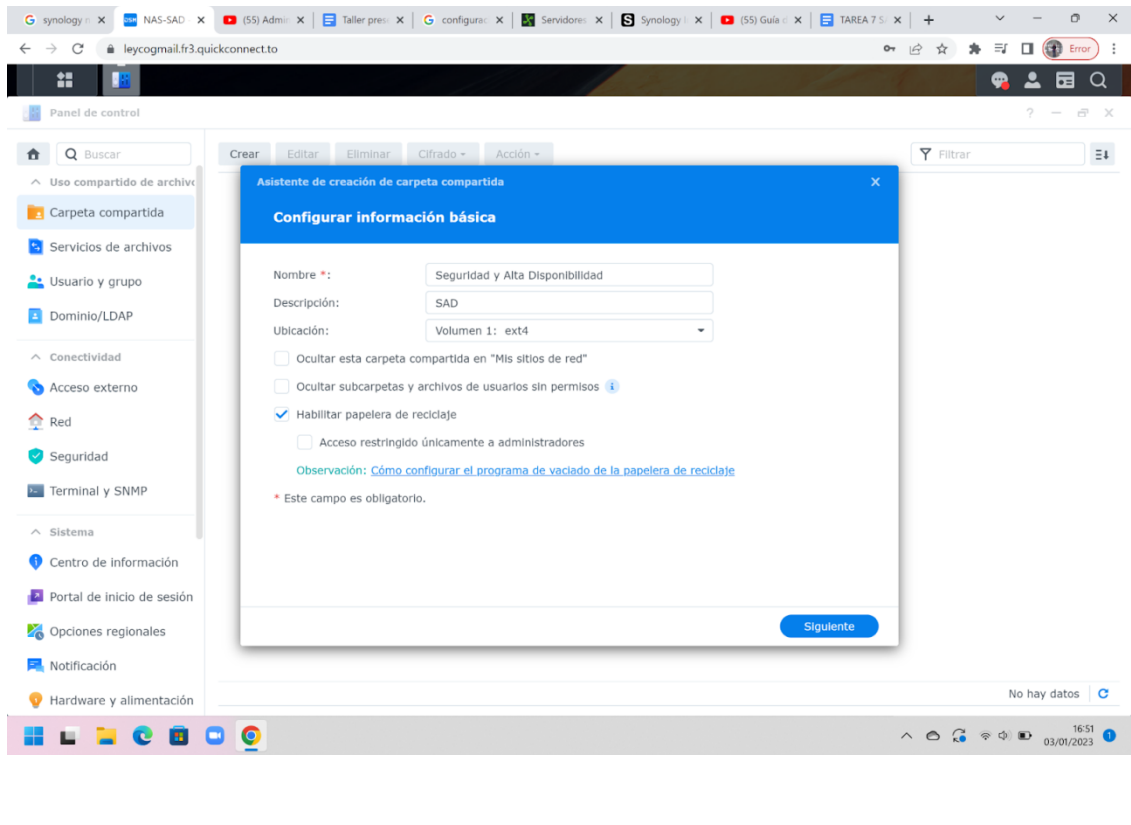


Y podemos ver el nuevo usuario en la lista de usuarios:



5. Crea una carpeta compartida para guardar tus documentos y cambia los permisos de acceso para que el nuevo usuario creado anteriormente pueda modificar esa carpeta y archivos.

Ve a Panel de control → Carpeta compartida y pincha en crear:



Panel de control

Crear Editar Eliminar Cifrado - Acción -

Asistente de creación de carpeta compartida

Confirmar configuración

Elemento	Valor
Nombre	Seguridad y Alta Disponibilidad
Descripción	SAD
Ubicación	Volumen 1: ext4
Visibilidad	
Papelera de reciclaje	Habilitada
Cifrado	

Anterior **Siguiente**

No hay datos

16:53
03/01/2023

Panel de control

Crear Editar Eliminar Cifrado - Acción -

Asistente de creación de carpeta compartida

Configurar permisos de usuario

Usuarios locales

Q- Buscar

Nombre	Vista previa	Permisos d...	Sin acceso	Lectura/Esc...	Solo lectura	Personali...
admin	Lectura/Escrit.	Lectura/Escrit.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
AdminNAS	Lectura/Escrit.	Lectura/Escrit.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
guest	Sin acceso	-	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Leonor	Lectura/Escrit.	-	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

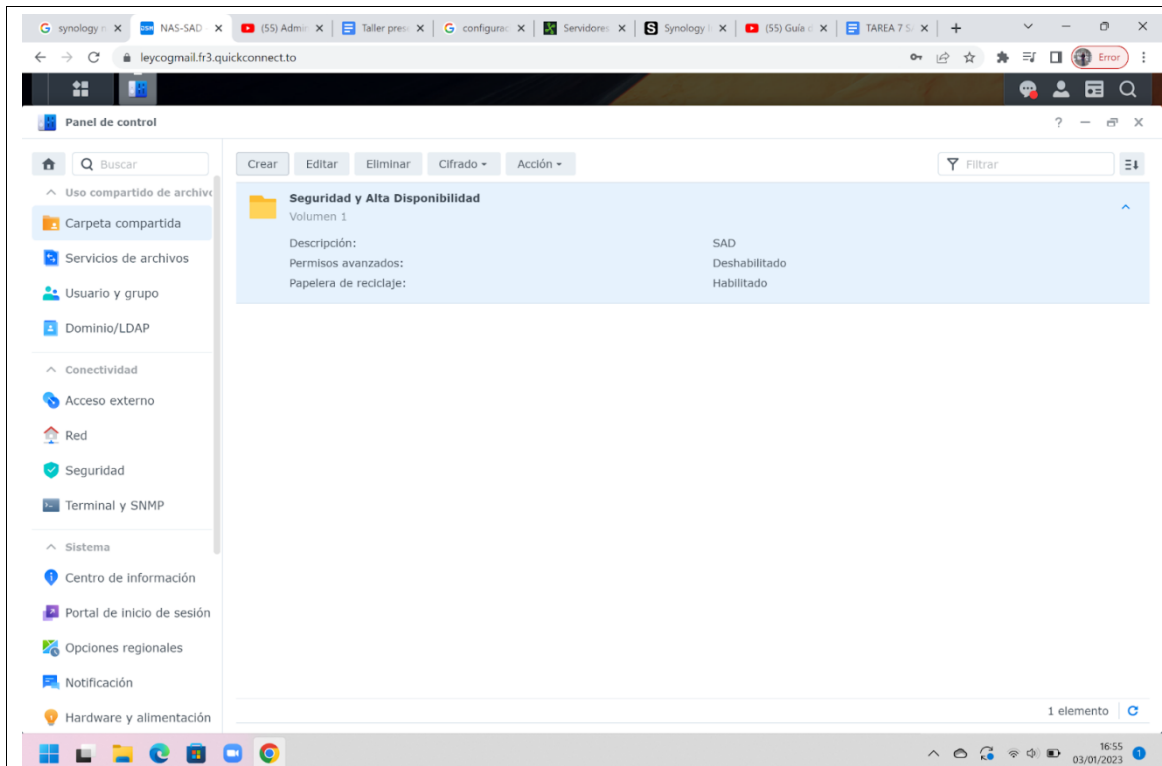
4 elementos

Aplicar

1 elemento

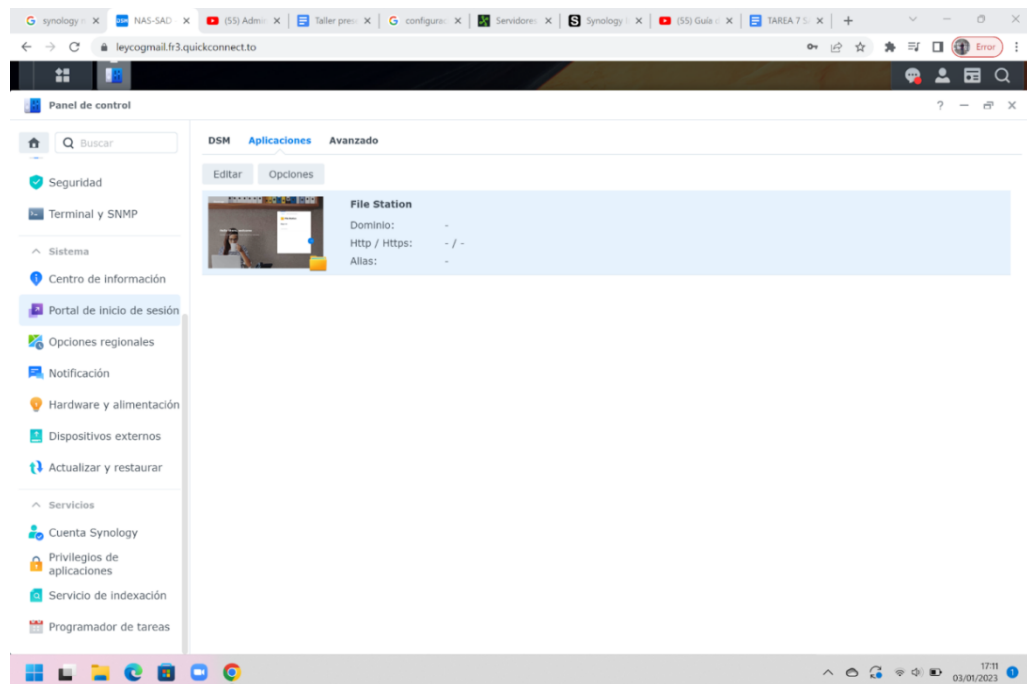
16:54
03/01/2023

Si pinchamos en "Editar → Permisos" se nos abre la ventana para poder configurar los permisos de usuario sobre la carpeta.

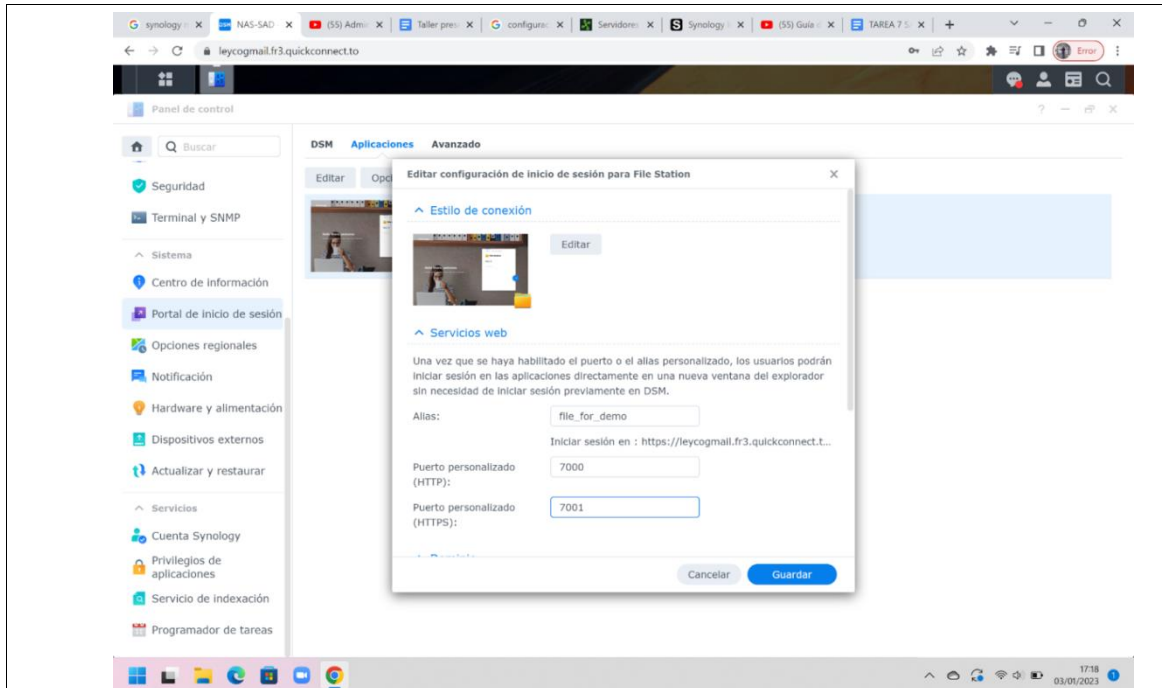


6. Da acceso y permiso al usuario para poder acceder a las aplicaciones sin tener que loguearse en el DSM (Disk Station Manager) con un portal de aplicación:

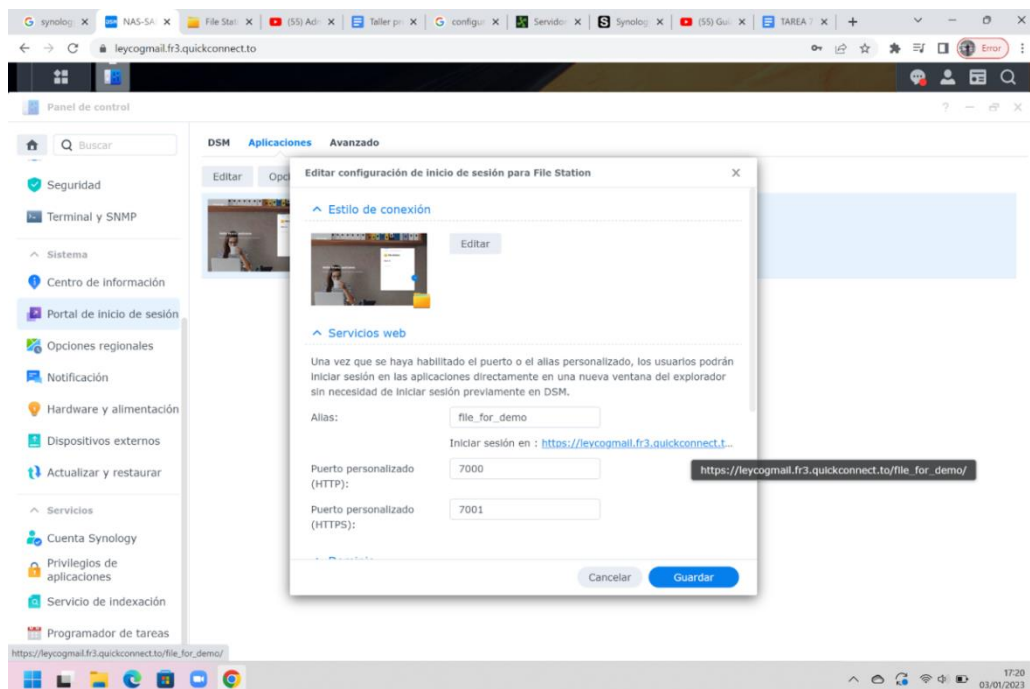
Pinchamos en Panel de control → Portal de inicio de sesión → Aplicaciones:



Pinchamos sobre File Station (o la aplicación que queremos) y clicamos sobre Editar:



Guardamos y volvemos a clicar en Editar:



Si pinchamos sobre el enlace generado podremos entrar en esa carpeta desde un navegador sin tener que identificarnos en el NAS completo:



Nombre	Tamaño	Tipo de archivo	Fecha de modificación
#recycle		Carpeta	03/01/2023 16:54:29
Taller Presencial		Carpeta	03/01/2023 17:57:42

Archivo	Tiempo restante	Velocidad	Progreso	Estado
Syno_HIG_DS120j...			100 %	Finalizado

Fuente (ejer 4,5,6): <https://www.youtube.com/watch?v=yecPU8ffEE>

7. Configura el NAS para acceder a los archivos a través del explorador de archivos de Windows.

Fuente: <https://www.youtube.com/watch?v=T0tihadMUgw>

Aprenderemos a usar el protocolo SMB para poder acceder a los archivos que tenemos en el NAS vía explorador de Windows.

Accedemos al Panel de control → Servicios de archivos → SMB:

Habilitamos el servicio SMB.

Se puede especificar el grupo de trabajo.

Pinchamos en Aplicar.

SMB AFP NFS FTP rsync Avanzado

^ SMB

Habilitar servicio SMB

Grupo de trabajo: WORKGROUP

Ocultar carpetas compartidas a los usuarios sin permiso

Habilitar registro de transferencia

Configuración del registro Ver registros

Configuración avanzada

Observación:

- Puede habilitar la papelera de reciclaje en la página de edición de **Carpeta compartida**.
- Una vez Indexadas las carpetas en **Lista de carpetas indexadas**, puede utilizar el Finder de Mac o el Explorador de archivos de Windows para buscar en los archivos y en su contenido.

Introduzca la dirección siguiente para acceder a carpetas compartidas utilizando un ordenador en su red local:

PC (Windows Explorer): \\NAS-SAD

Mac (Finder): smb://NAS-SAD

^ WS-Discovery

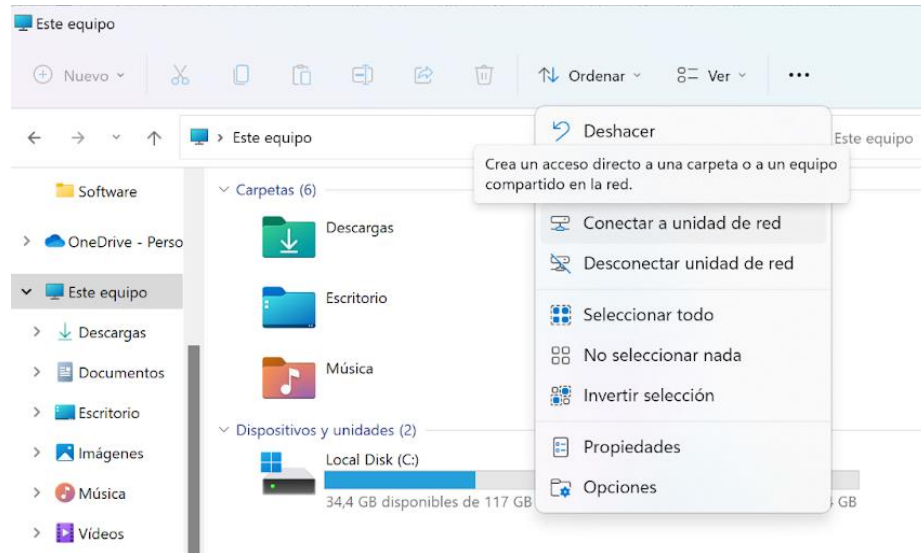
Los dispositivos de red local pueden utilizar la detección de redes de Windows para acceder a los archivos en su Synology NAS.

Activar la detección de redes de Windows para permitir el acceso de archivos a través de SMB

Restablecer Aplicar

Hay dos formas de configurar el NAS para acceder a los archivos a través del explorador:

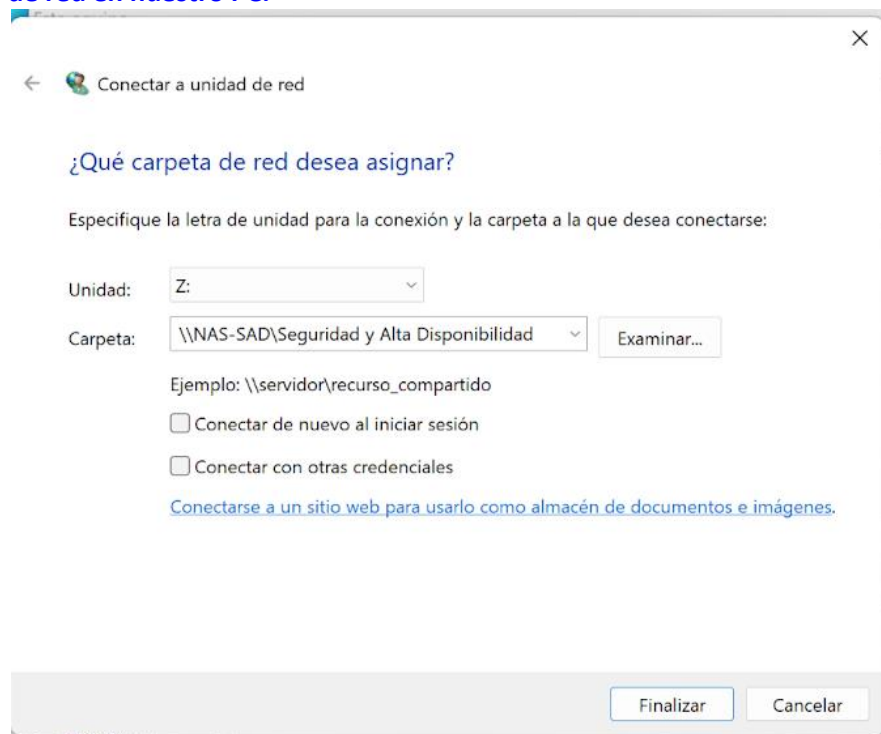
- **Abrimos el explorador de Windows y seleccionamos “Este equipo”**



Pinchamos en **Conectar a unidad de red**, seleccionamos la unidad y ponemos la dirección de la carpeta donde está la carpeta compartida del NAS que queremos tener disponible en el explorador de Windows, en este caso el NAS se llama **NAS-SAD** y la carpeta compartida se llama **Seguridad y Alta Disponibilidad**, por tanto, en la carpeta debe poner:

\\NAS-SAD\Seguridad y Alta Disponibilidad

De esta forma mapeamos la carpeta compartida en el NAS como una unidad de red en nuestro PC.



Nos saldrá que debemos meter las credenciales del NAS:

The screenshot shows a Windows error dialog titled "Conectar a unidad de red" with the message "Intentando conectar con \\NAS-SAD\Seguridad y Alta Disponibilidad...". Below it is a "Conectar a unidad de red" dialog box with the following fields and options:

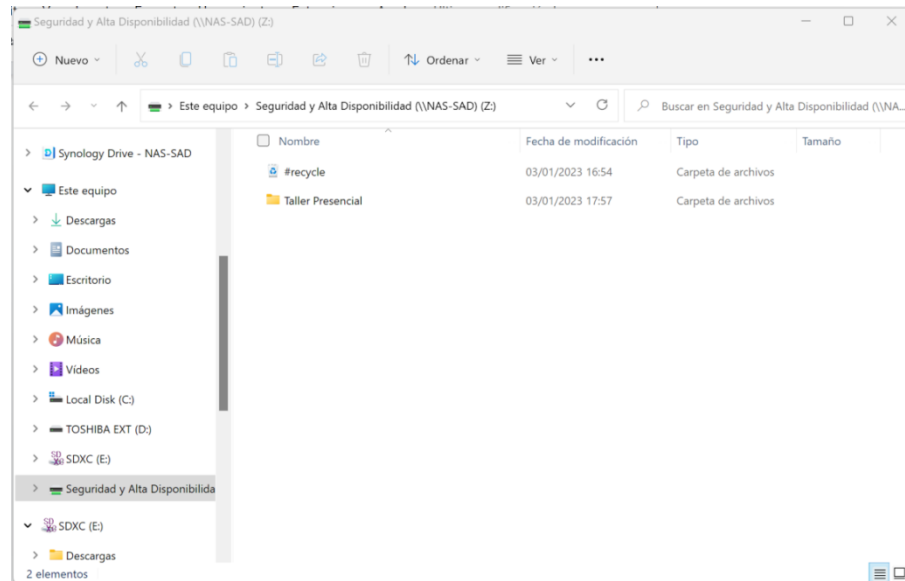
- Unidad: Z:
- Carpeta: \\NAS-SAD\Seguridad y Alta Disponibilidad
- Ejemplo: \\servidor\recurso_compartido
- Conectar de nuevo al iniciar sesión
- Conectar con otras credenciales
- [Conectarse a un sitio web para usarlo como almacén de documentos e imágenes.](#)

Below this is a "Seguridad de Windows" dialog titled "Escribir credenciales de red" with the text "Escriba sus credenciales para conectarse a: NAS-SAD". It contains input fields for "Nombre de usuario" and "Contraseña", a checkbox for "Recordar mis credenciales", and "Aceptar" and "Cancelar" buttons.

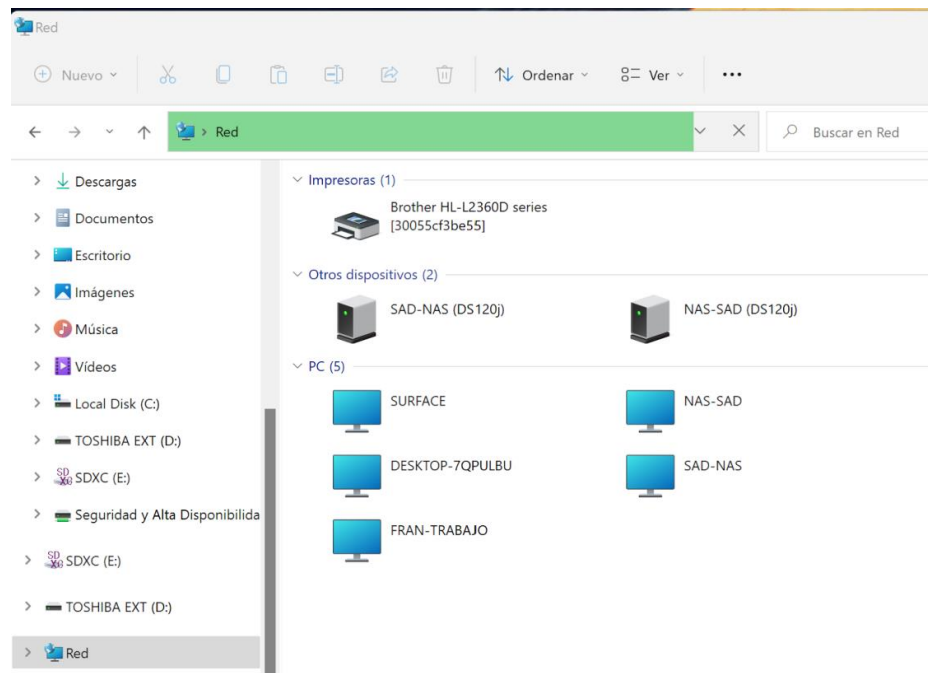
Metemos nuestras credenciales:

The screenshot shows the "Seguridad de Windows" dialog titled "Escribir credenciales de red" with the text "Escriba sus credenciales para conectarse a: NAS-SAD". The "Nombre de usuario" field is filled with "AdminNAS" and the "Contraseña" field is filled with 12 dots. The "Recordar mis credenciales" checkbox is checked. The "Aceptar" and "Cancelar" buttons are visible at the bottom.

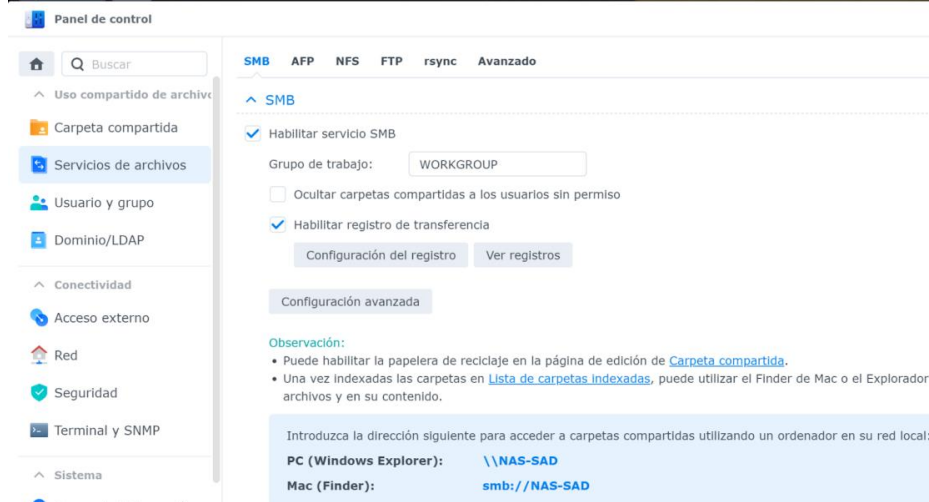
Y estaríamos dentro de la carpeta:



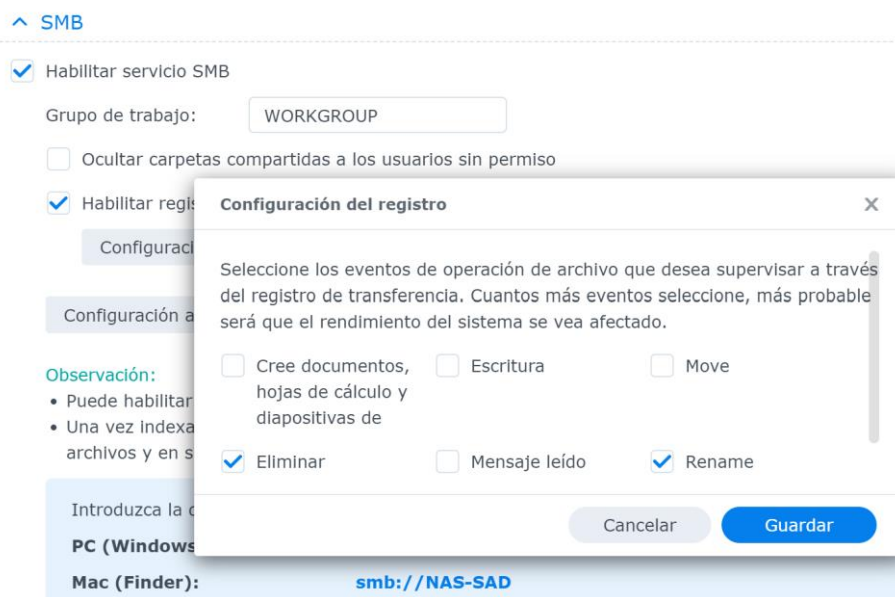
- **Otra forma de hacerlo sería haciendo doble clic en nuestro NAS en Red:**



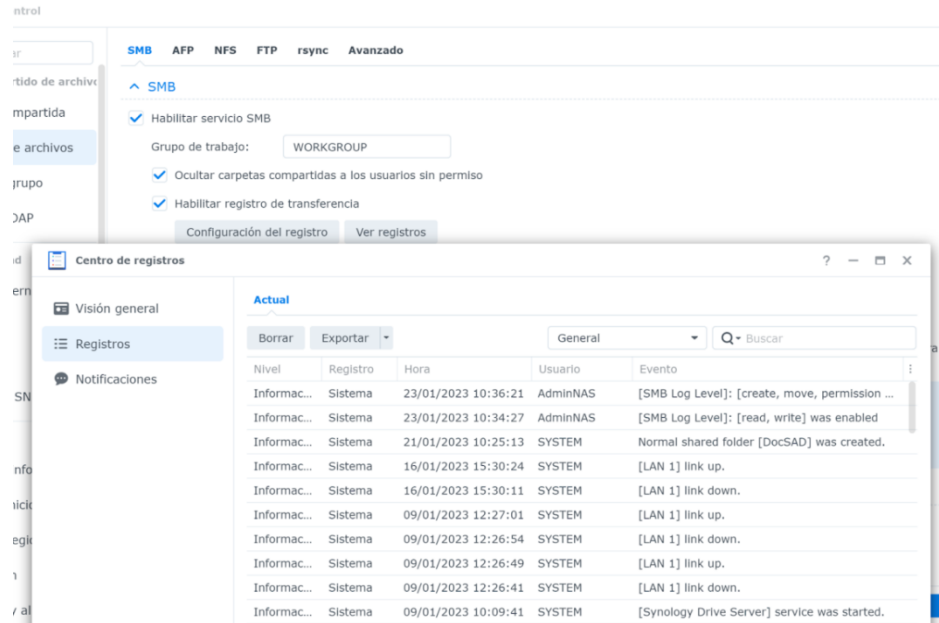
Una vez hecho esto volvemos al panel de control del NAS, servicios de archivos, SMB y habilitamos el registro de transferencia:



Esto nos permitirá guardar los registros de las operaciones que se realicen sobre los ficheros y que hayan sido ejecutados usando el protocolo SMB o CIFS. Seleccionamos los eventos que queremos registrar pinchando en configuración de registro:



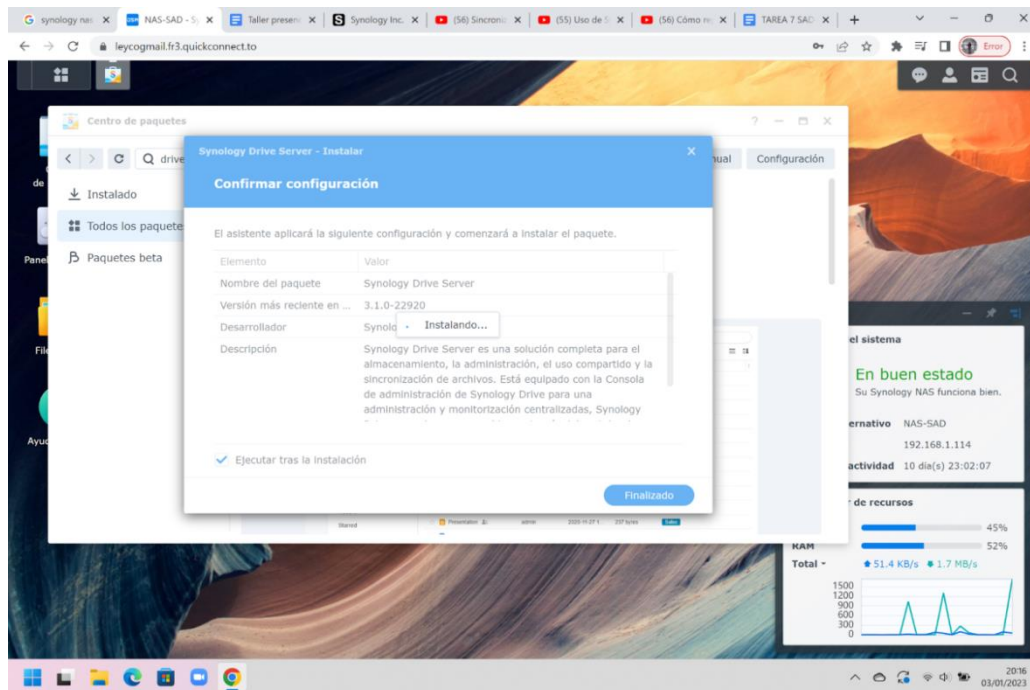
Si clicamos en ver registros podremos observar los logs creados:



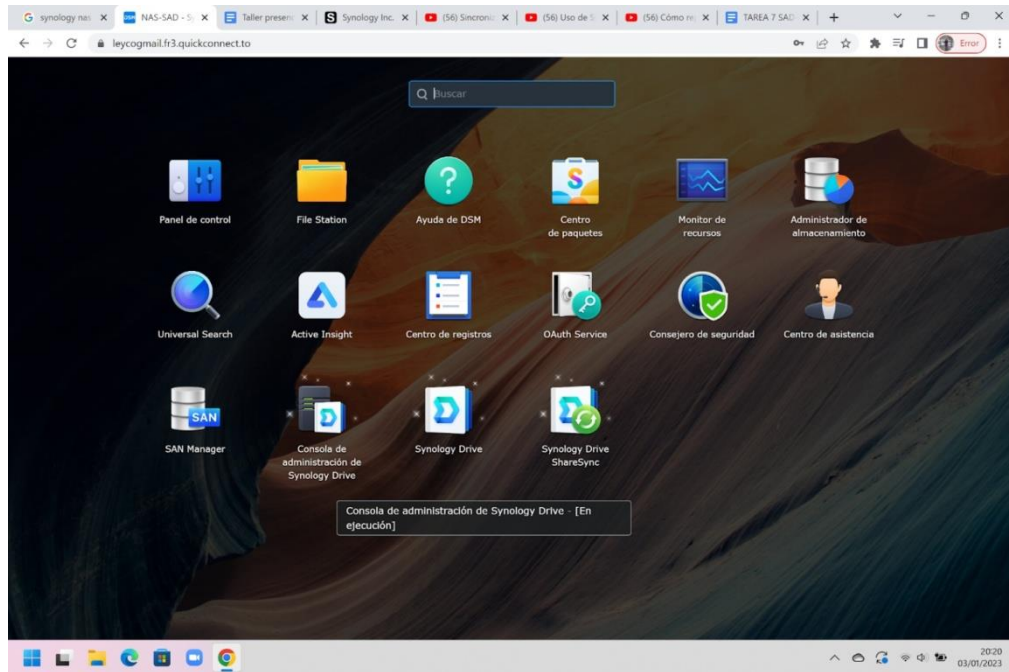
8. Sincroniza los archivos de tu PC con el NAS.

Fuente: <https://www.youtube.com/watch?v=vxQNjPpuLhM>

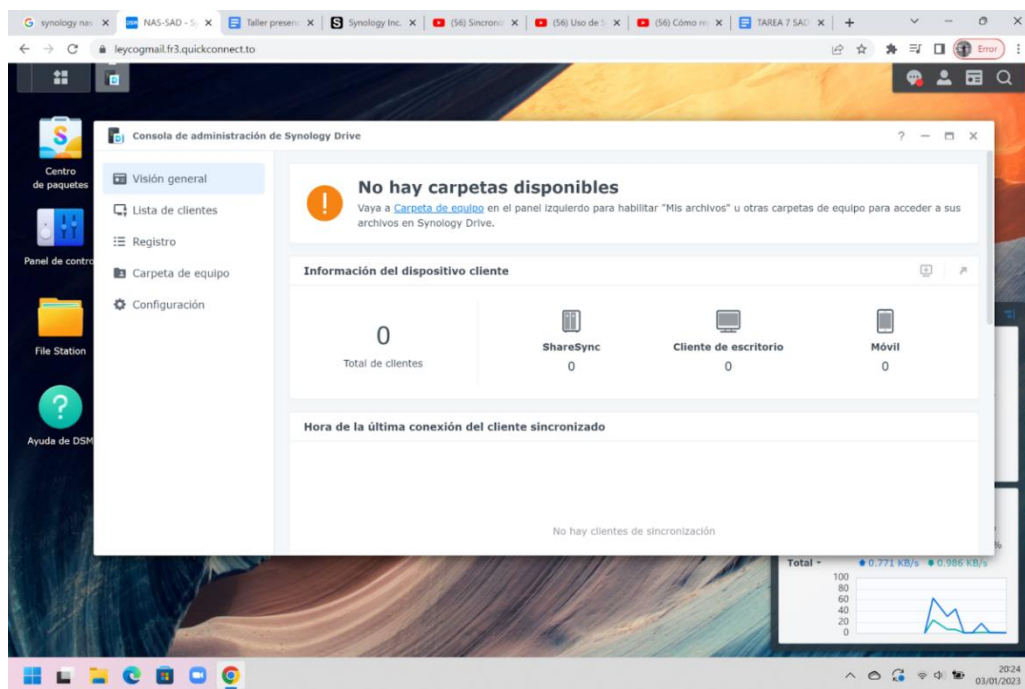
Instala Synology Drive Server en el NAS, para ello ve al centro de paquetes, busca el software y dale a instalar.



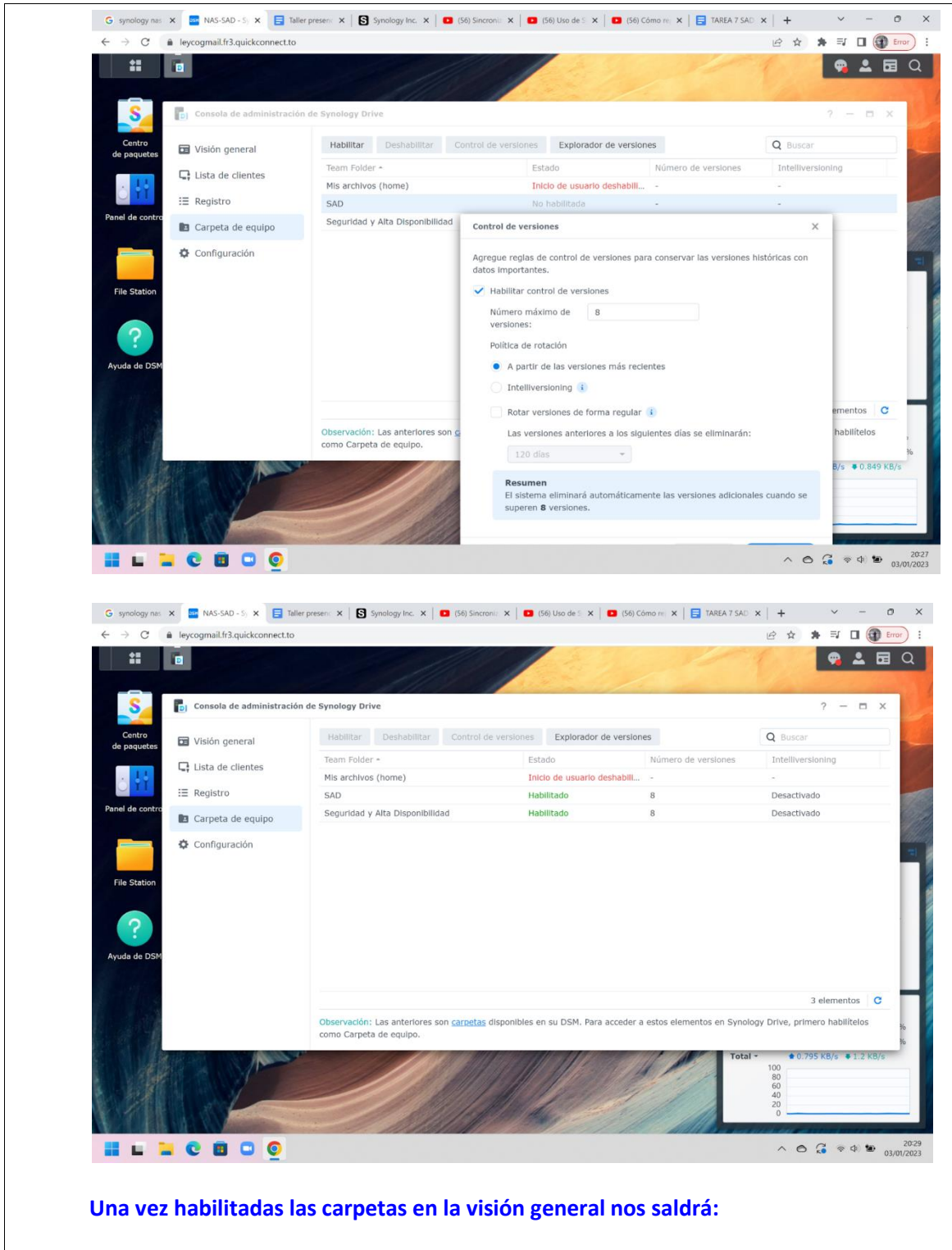
Una vez instalado, en el menú principal del NAS seleccionamos la consola de administración de Synology Drive:



Cuando la abrimos nos aparece:



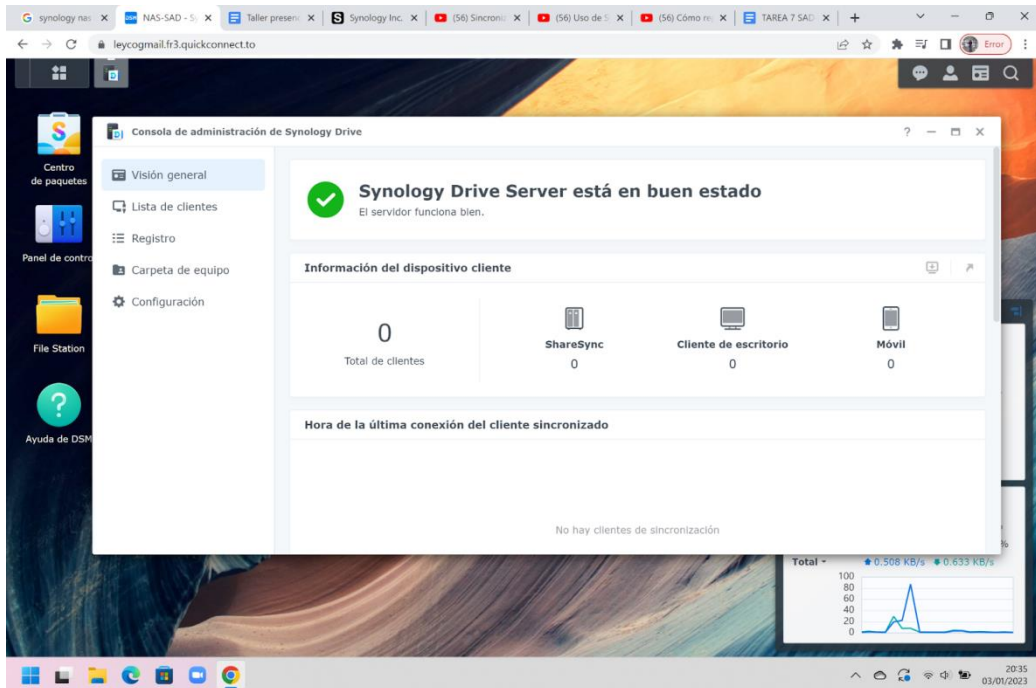
Debemos ir a la carpeta de equipo y seleccionar las carpetas que queremos que estén disponibles, pinchamos en habilitar y habilitamos el control de versiones:



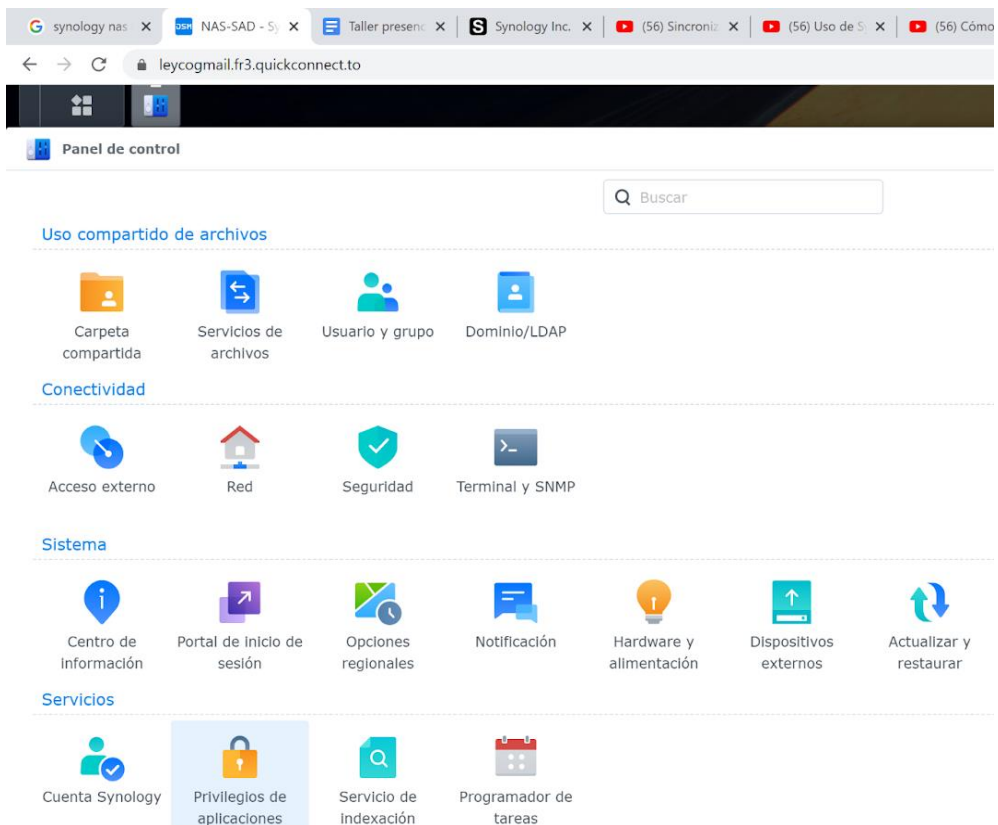
The image shows two screenshots of the Synology Drive administration console. The top screenshot shows the 'Control de versiones' (Version Control) dialog box for the 'Seguridad y Alta Disponibilidad' folder. The dialog is set to 'Habilitar control de versiones' (Enable version control) with a maximum of 8 versions. The rotation policy is set to 'A partir de las versiones más recientes' (From the most recent versions). A summary note states: 'El sistema eliminará automáticamente las versiones adicionales cuando se superen 8 versiones.' (The system will automatically delete additional versions when they exceed 8 versions.)

The bottom screenshot shows the 'Visión general' (Overview) view of the console. The 'Seguridad y Alta Disponibilidad' folder is now listed as 'Habilitado' (Enabled) with 8 versions. The 'Mis archivos (home)' folder is listed as 'Inicio de usuario deshabilitado' (User disabled).

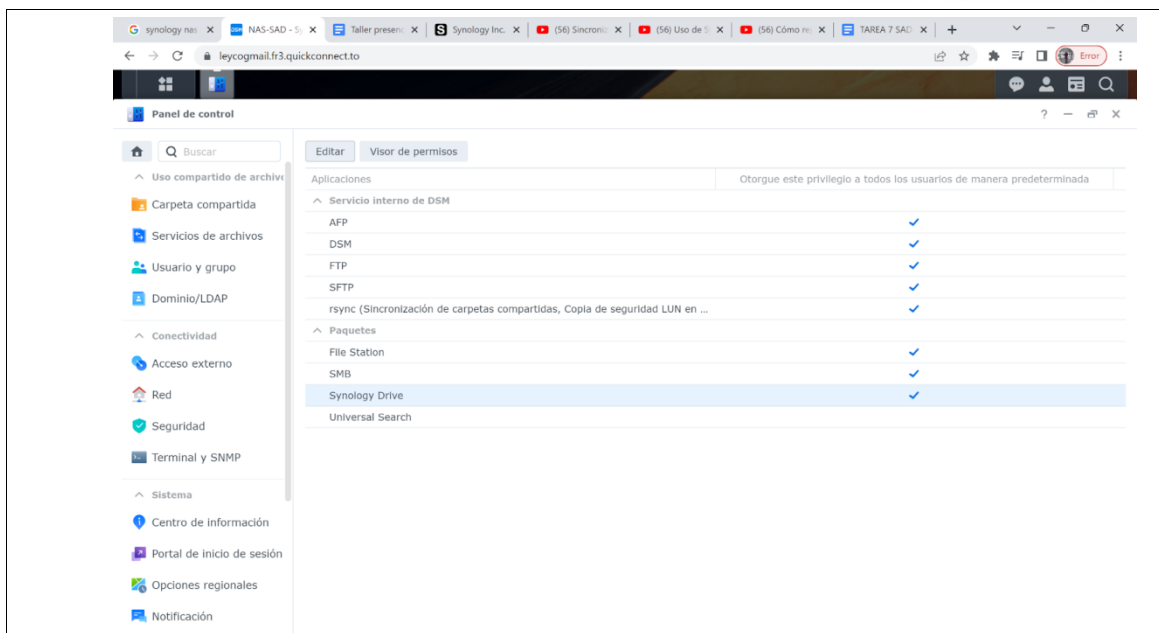
Una vez habilitadas las carpetas en la visión general nos saldrá:



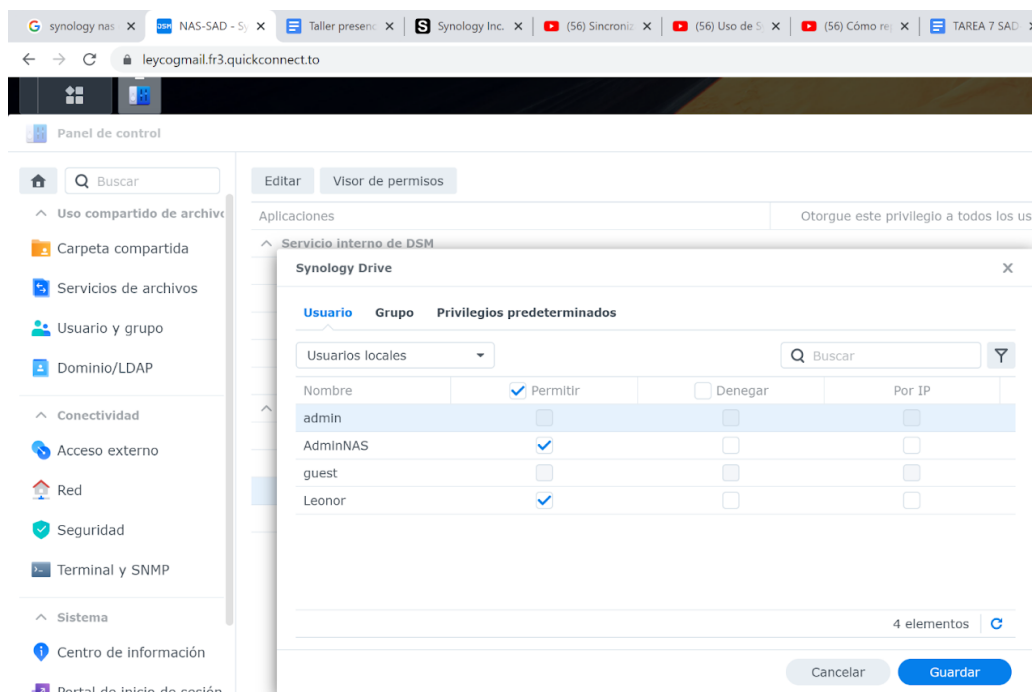
Accedemos después a Panel de control → Privilegios de aplicaciones



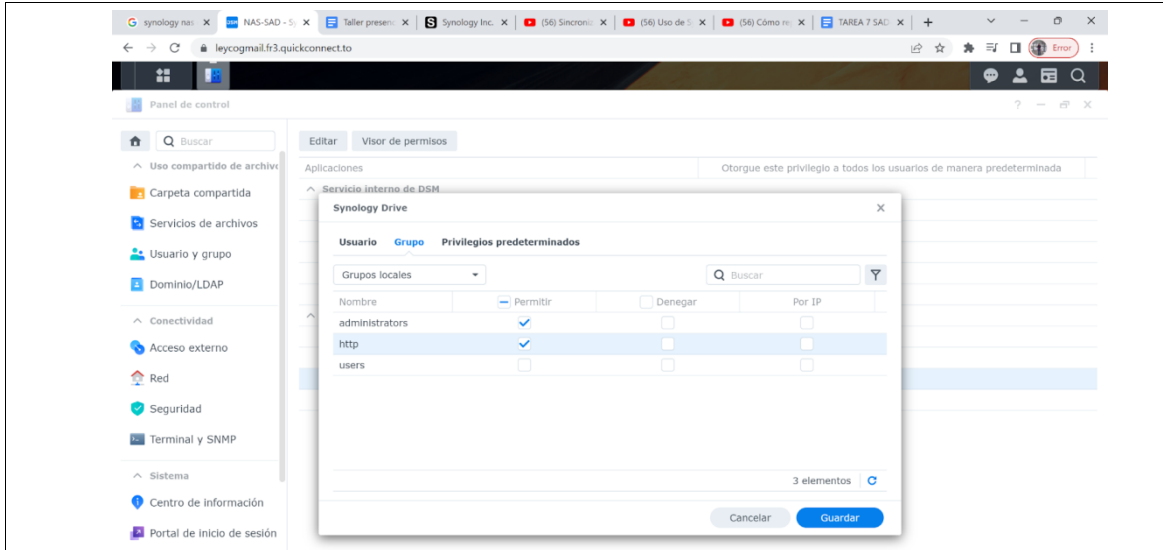
Seleccionamos Synology Drive → Editar:



Configuramos los usuarios locales:

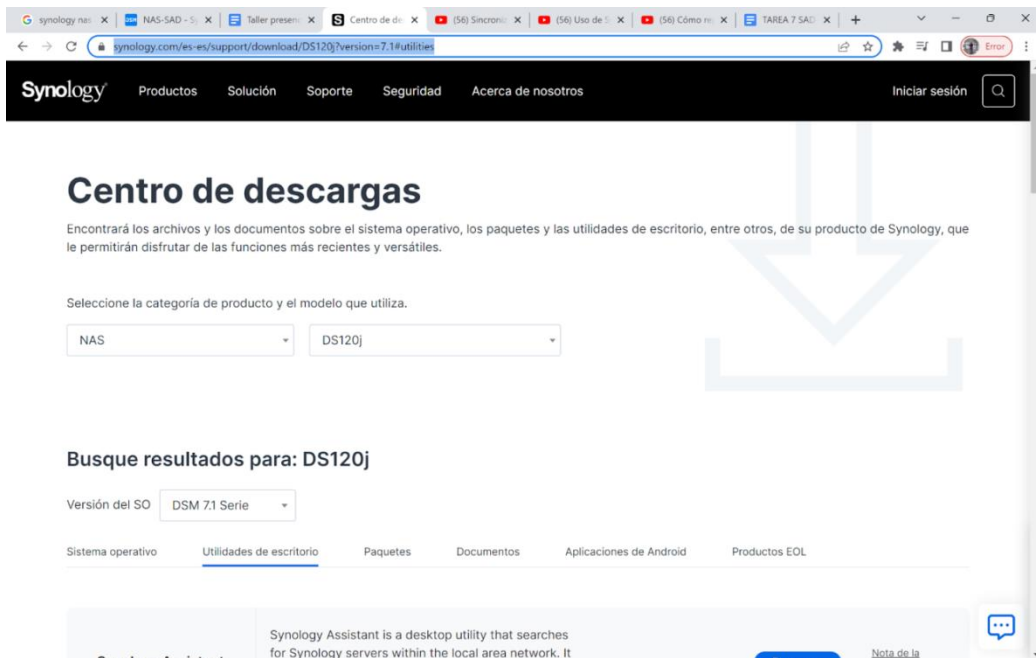


Y los grupos locales:

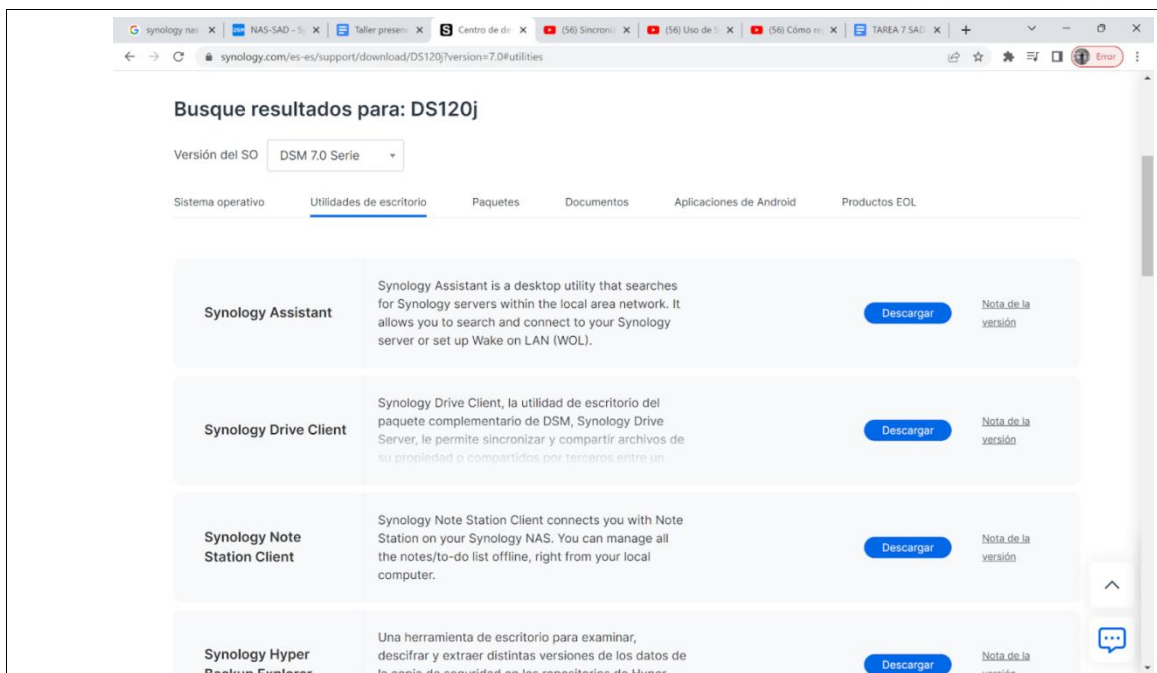


Ahora debemos hacer el set up de la conexión entre el NAS y nuestro PC. Para ello debemos instalar Synology Drive Client del centro de descargas de Synology:
<https://www.synology.com/es-es/support/download/DS120j?version=7.1#utilities>

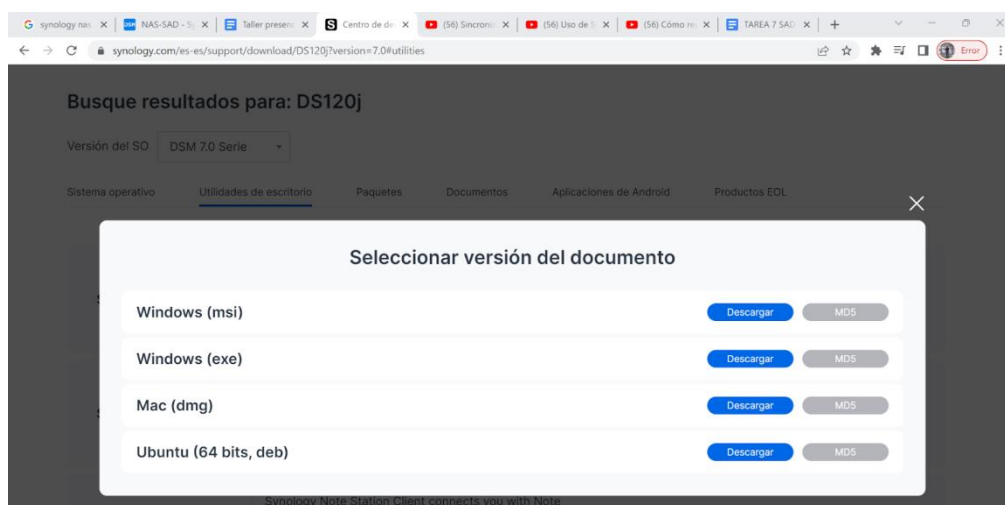
El centro de descargas está en Soporte → Centro de descargas.
Seleccionamos NAS y nuestro modelo DS120j y en Utilidades de escritorio encontramos el Synology Drive Client, previamente también debemos seleccionar la versión del SO que tiene nuestro NAS:



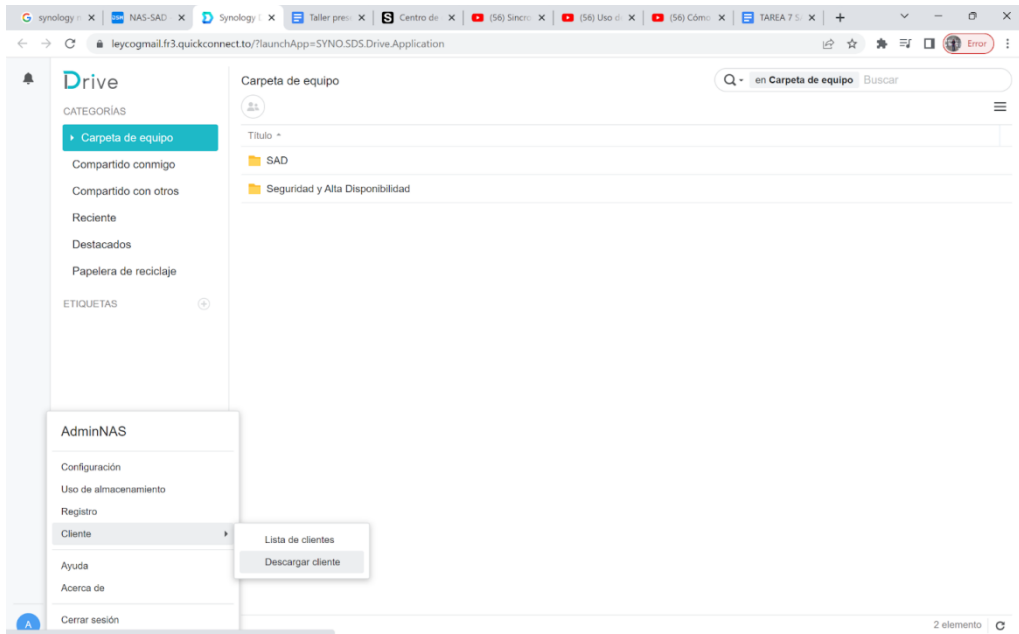
Descargamos el software Synology Drive Client:



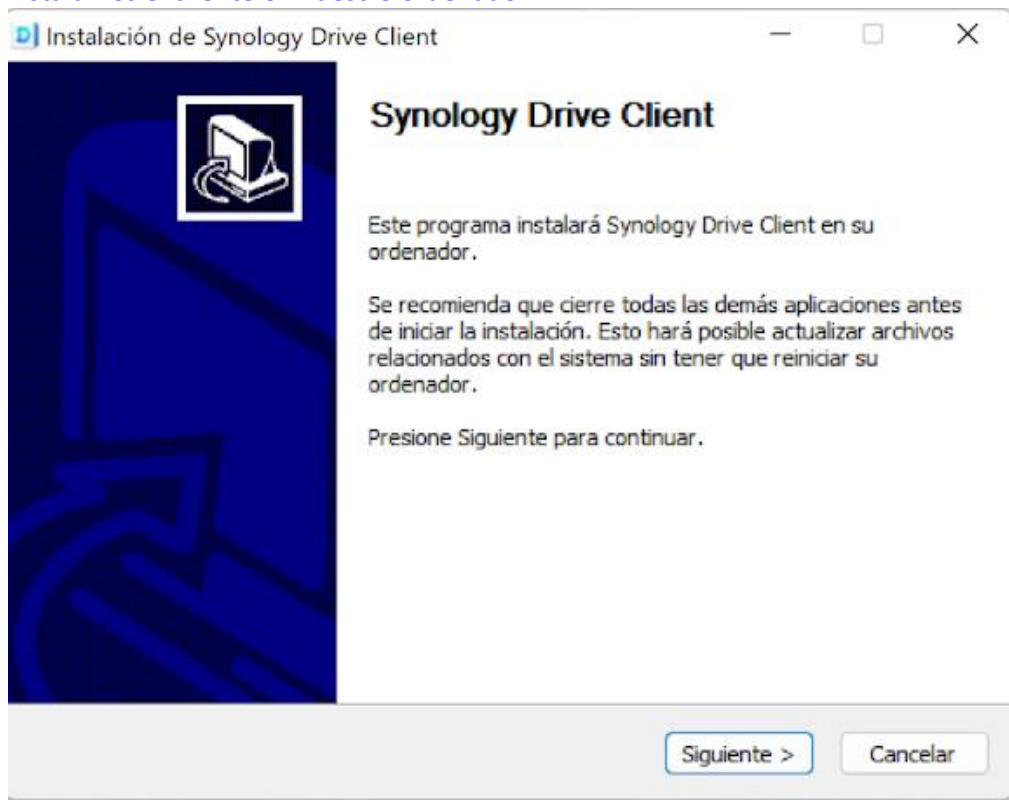
Seleccionamos el SO que tenemos en nuestro PC y se nos descarga:



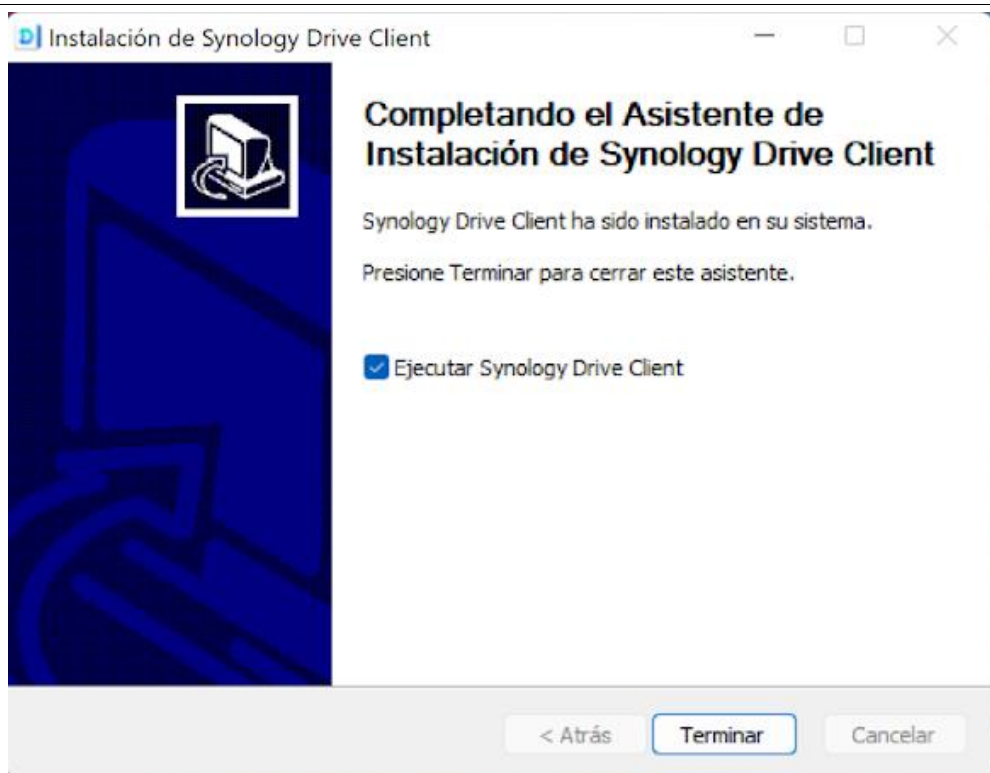
También se puede descargar accediendo a Menú principal → Synology Drive, en el icono del usuario de la esquina inferior izquierda:



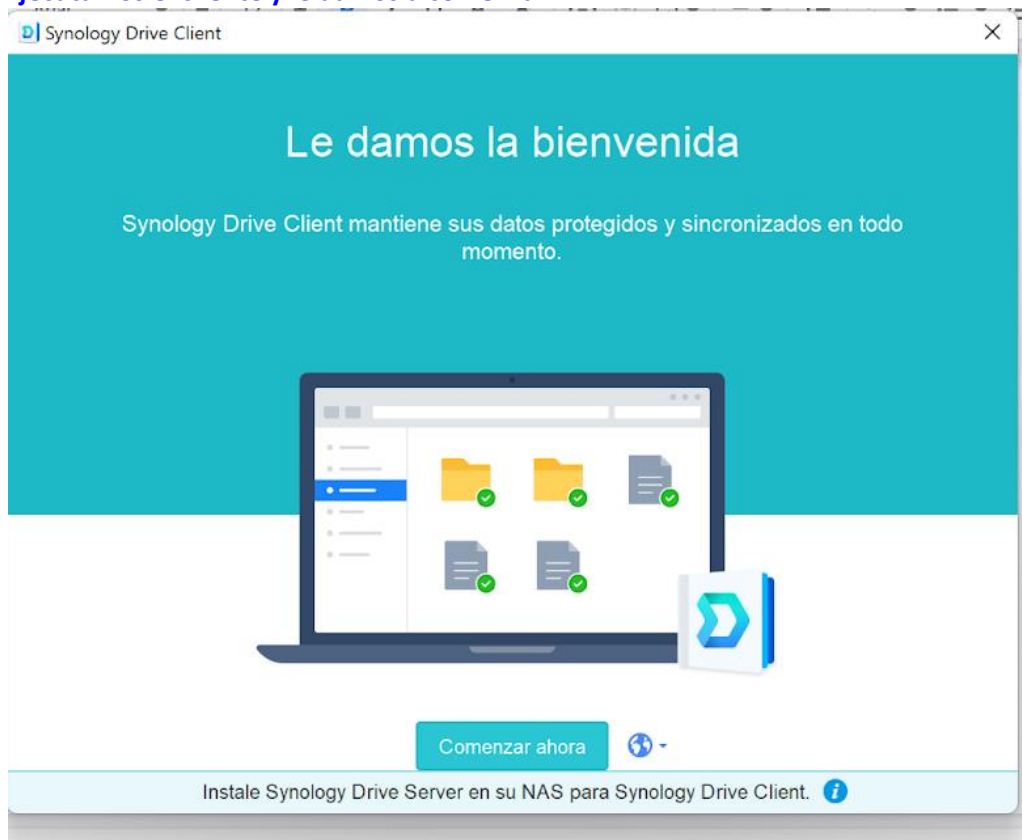
Instalamos el cliente en nuestro ordenador:



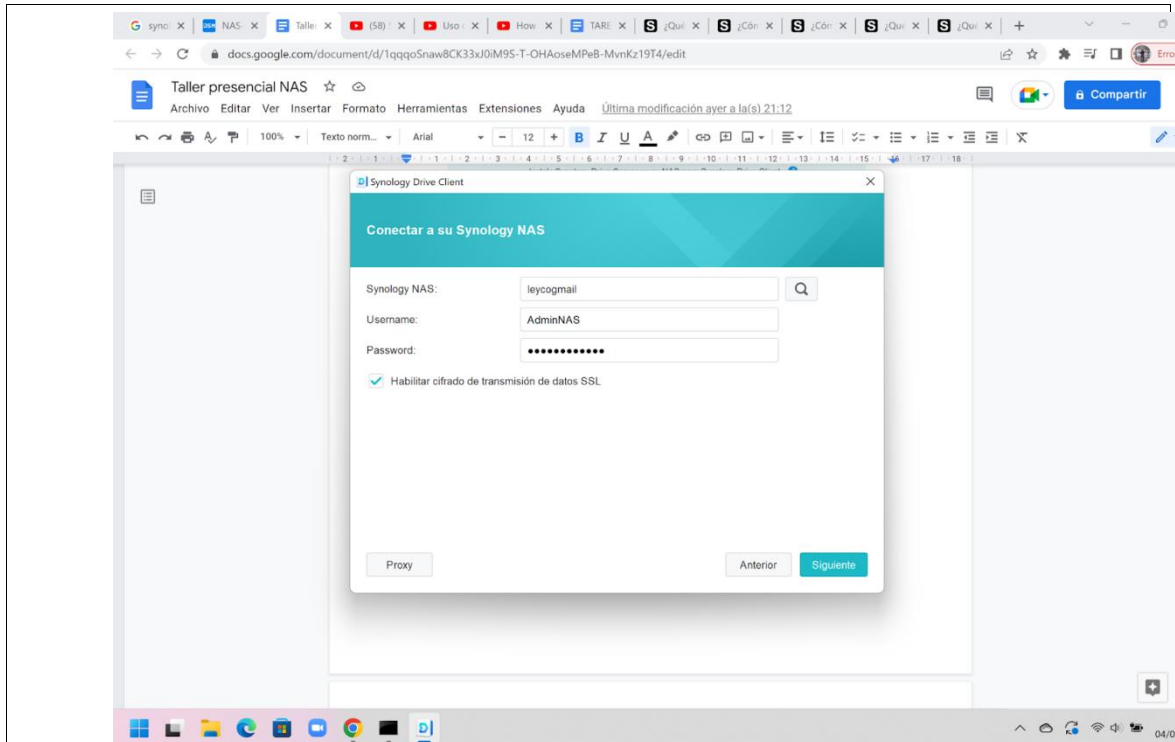
Aceptamos las condiciones y finaliza la instalación:



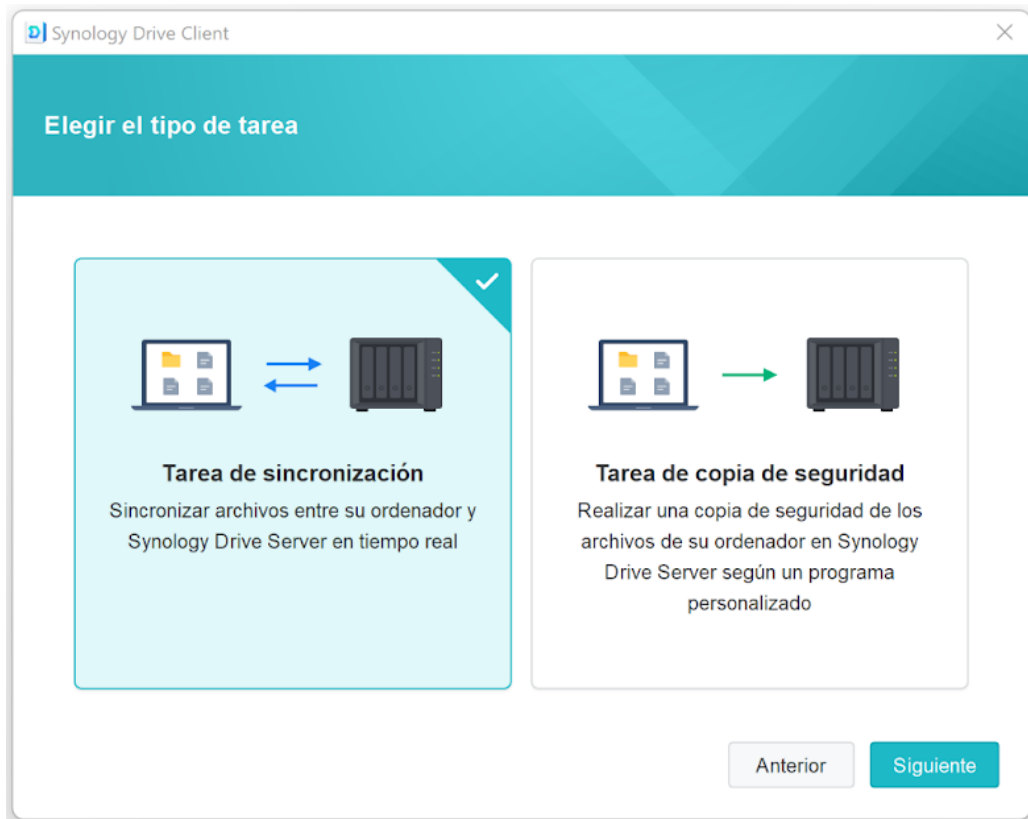
Ejecutamos el cliente y le damos a comenzar:



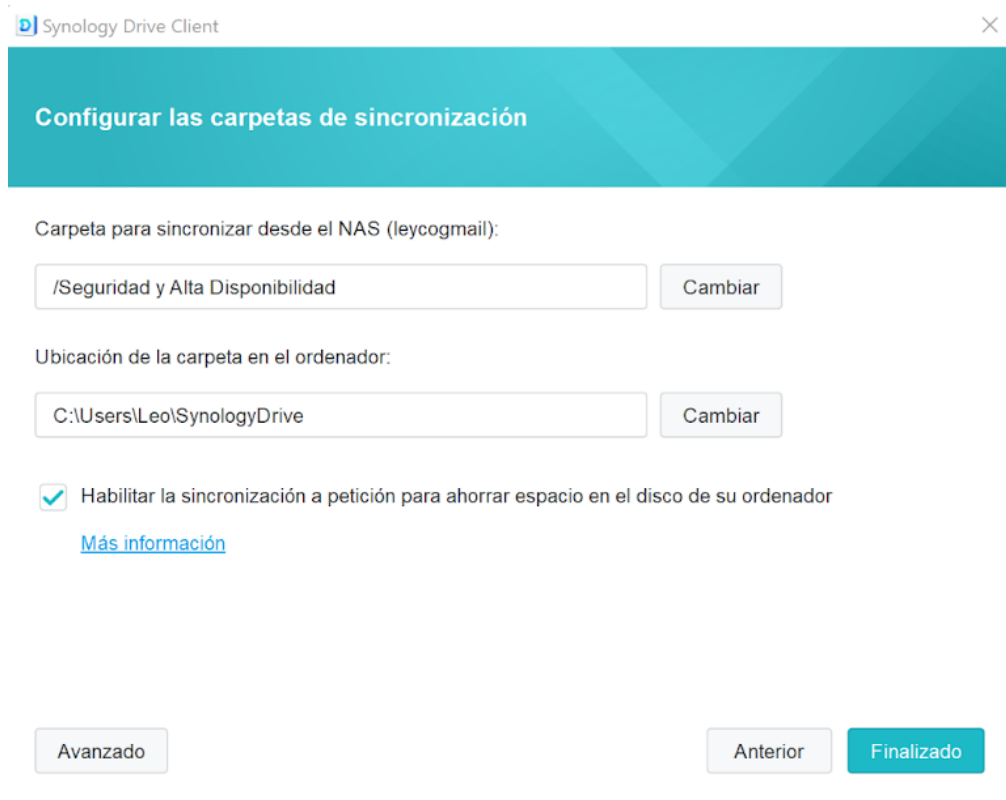
Introducimos en Synology NAS el dominio, la IP del NAS o el QuickConnect ID, nuestro usuario y contraseña:



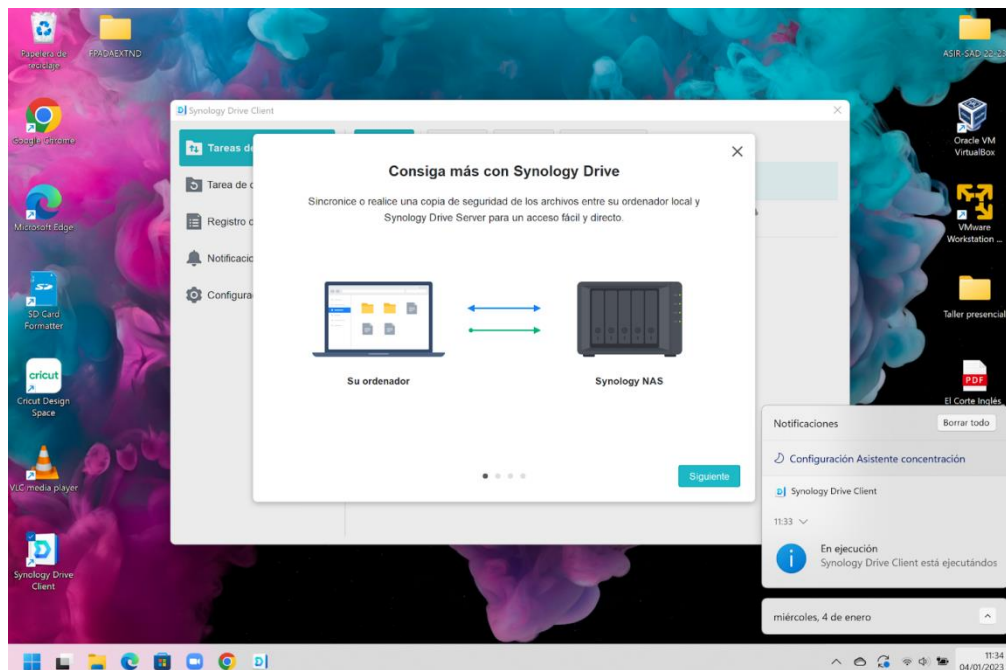
Elegimos la tarea que queremos hacer:



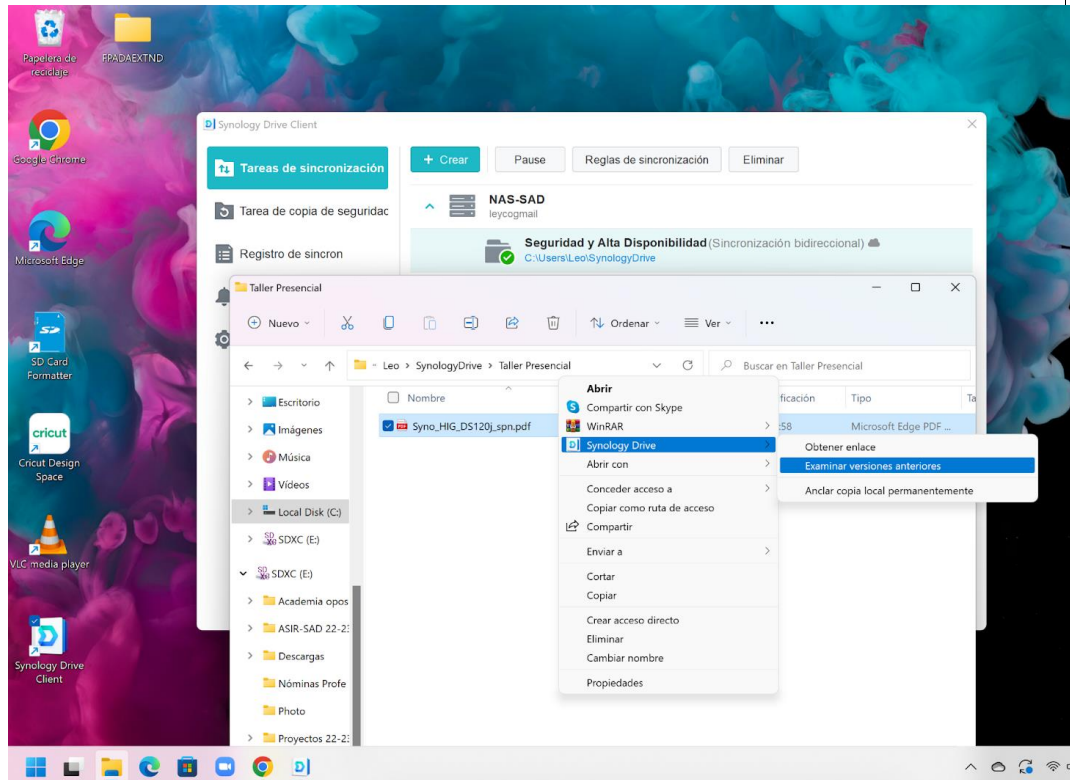
Seleccionamos tarea de sincronización y elegimos la carpeta a sincronizar dentro del NAS y la ubicación de dicha carpeta dentro de nuestro PC:



Una vez hemos terminado de seleccionar las carpetas le damos a finalizar y empezará la sincronización:



Si queremos examinar versiones anteriores del mismo archivo pinchamos botón derecho sobre el archivo y podremos seleccionar la versión que queremos descargar:



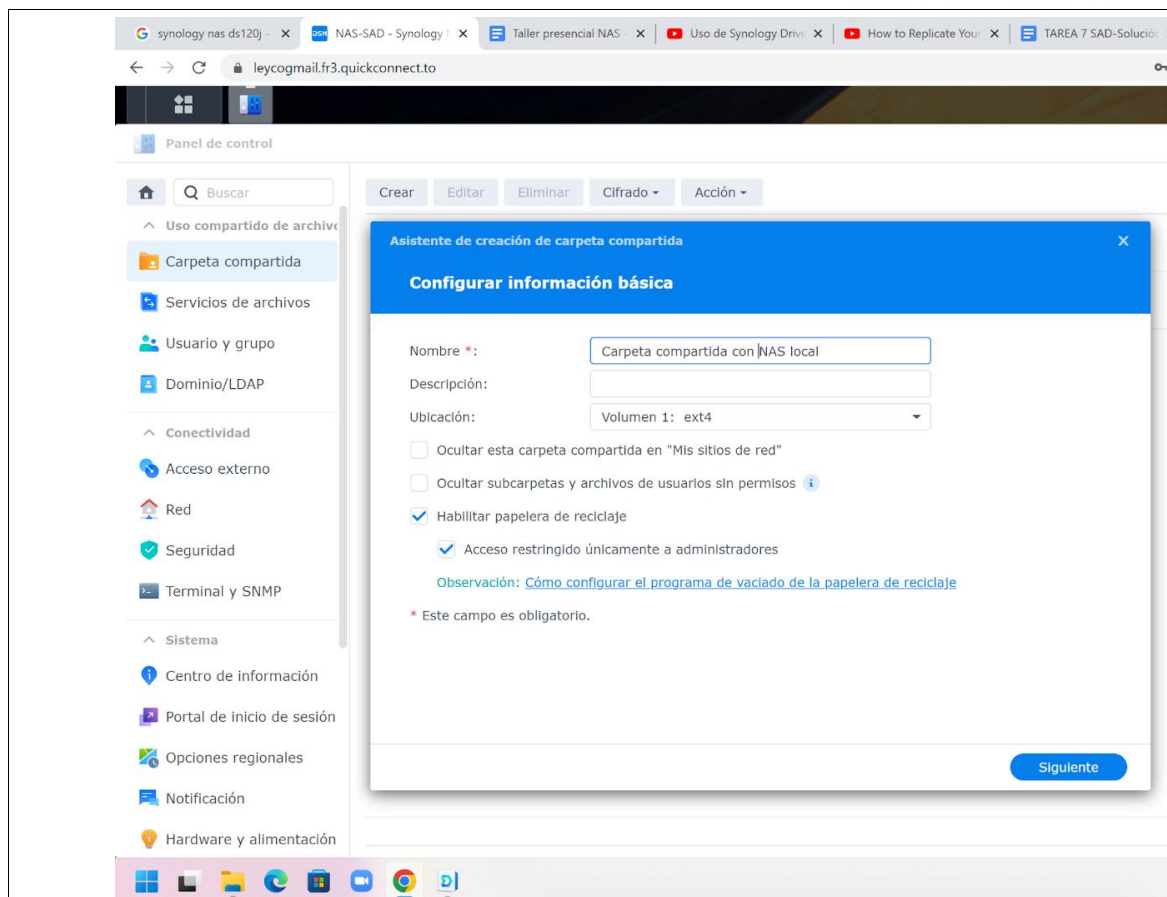
9. **Uso de Synology Drive Server para sincronizar datos entre diferentes servidores NAS.**
Fuente: <https://www.youtube.com/watch?v=klrZNCWCPO8>

En este ejercicio lo que se pretende es sincronizar los datos de un NAS “local” en otro NAS “remoto”.

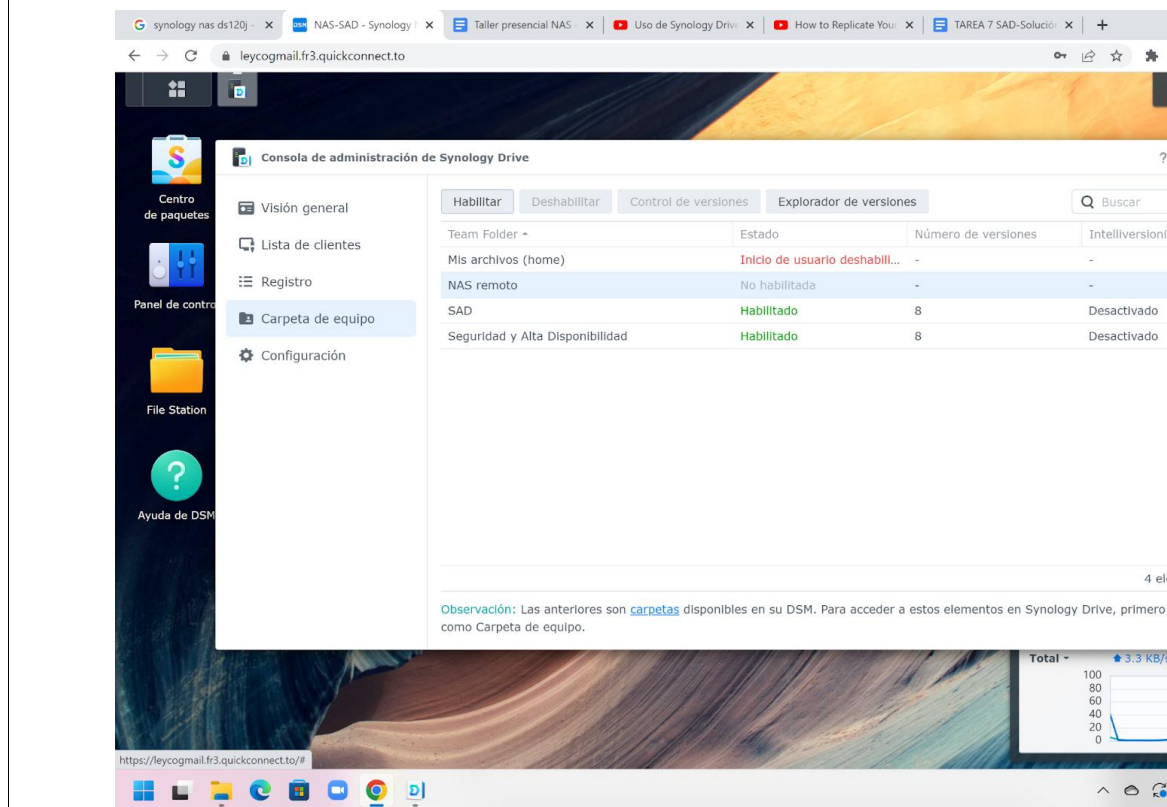
Lo primero es instalar Synology Drive Server en ambos NAS. Lo buscamos en el centro de paquetes (como hicimos anteriormente) y lo instalamos en el NAS remoto también.

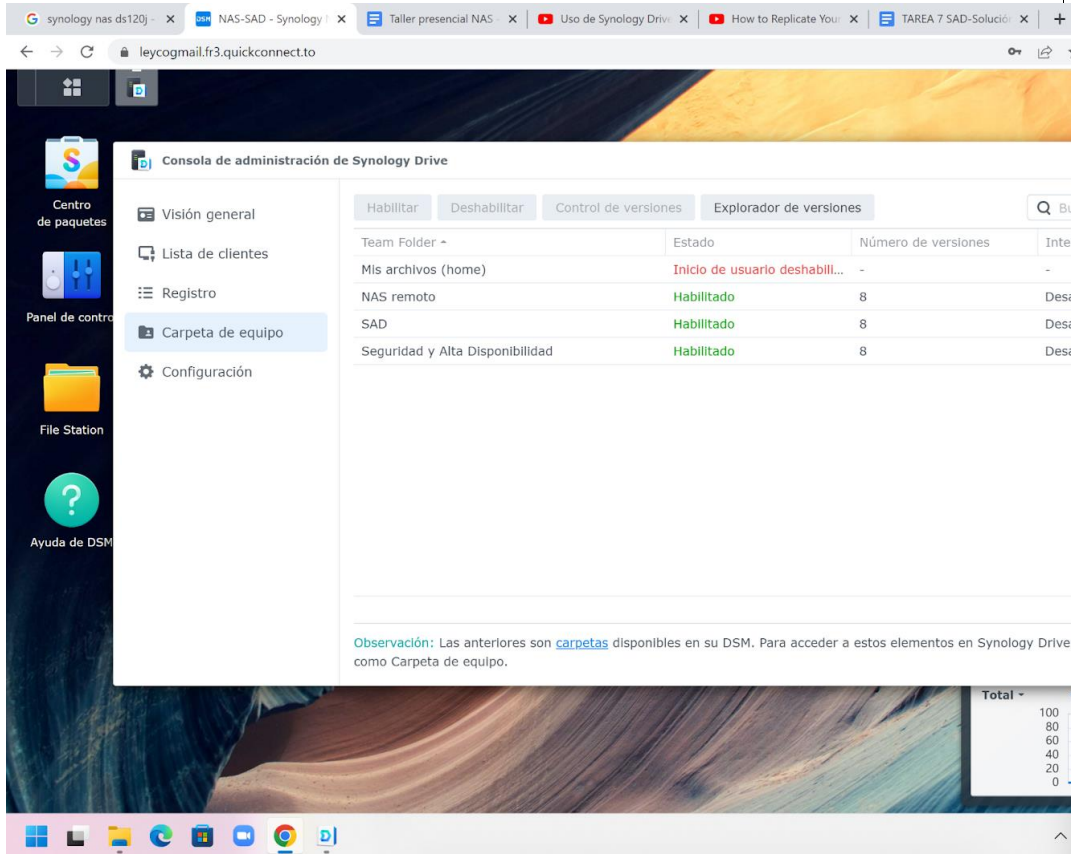
El NAS “local” será el que opere con Synology Drive ShareSync y el NAS “remoto” será el que queremos sincronizar con el NAS “local”.

Después habrá que configurar una carpeta compartida en el NAS “remoto” en Panel de control → Carpetas compartidas → Crear → Crear carpeta compartida:

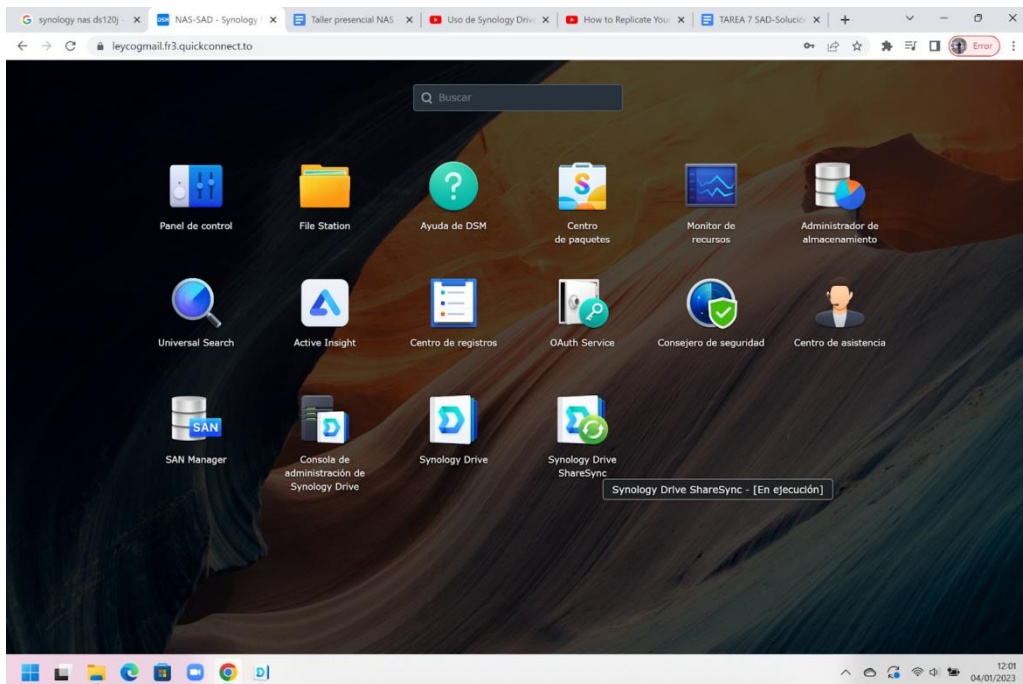


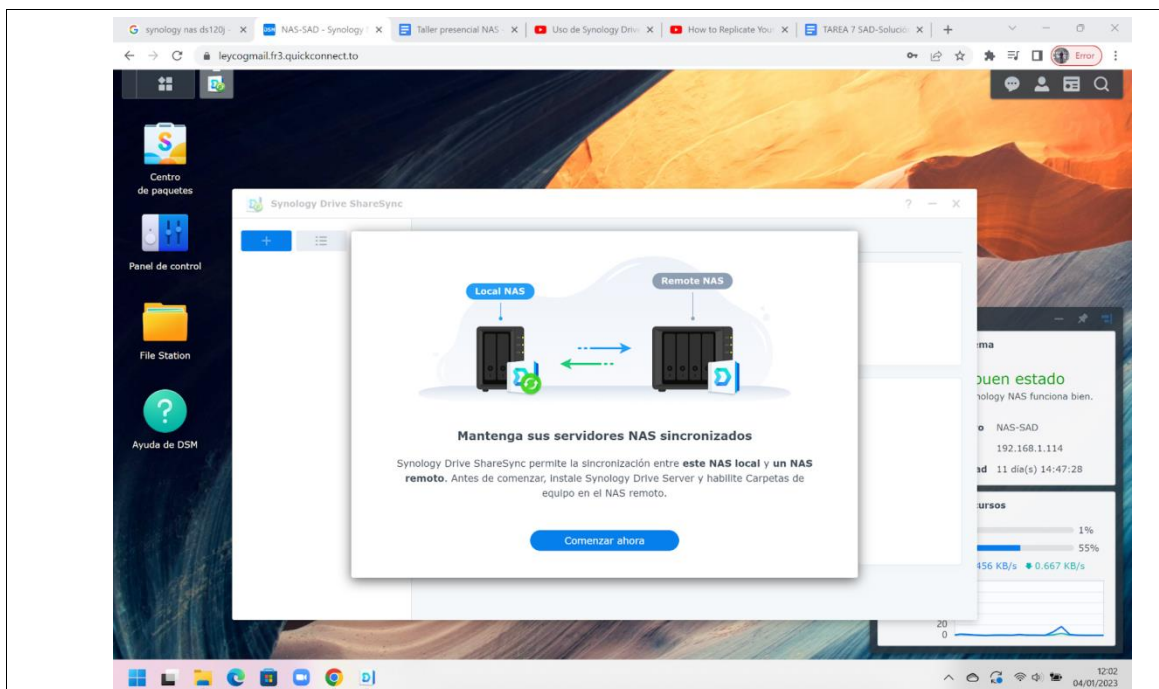
En el NAS “remoto” hay que habilitar las carpetas de equipo, en la Consola de administración de Synology Drive → Carpeta de equipo, seleccionas la carpeta que quieres sincronizar y seleccionas habilitar:



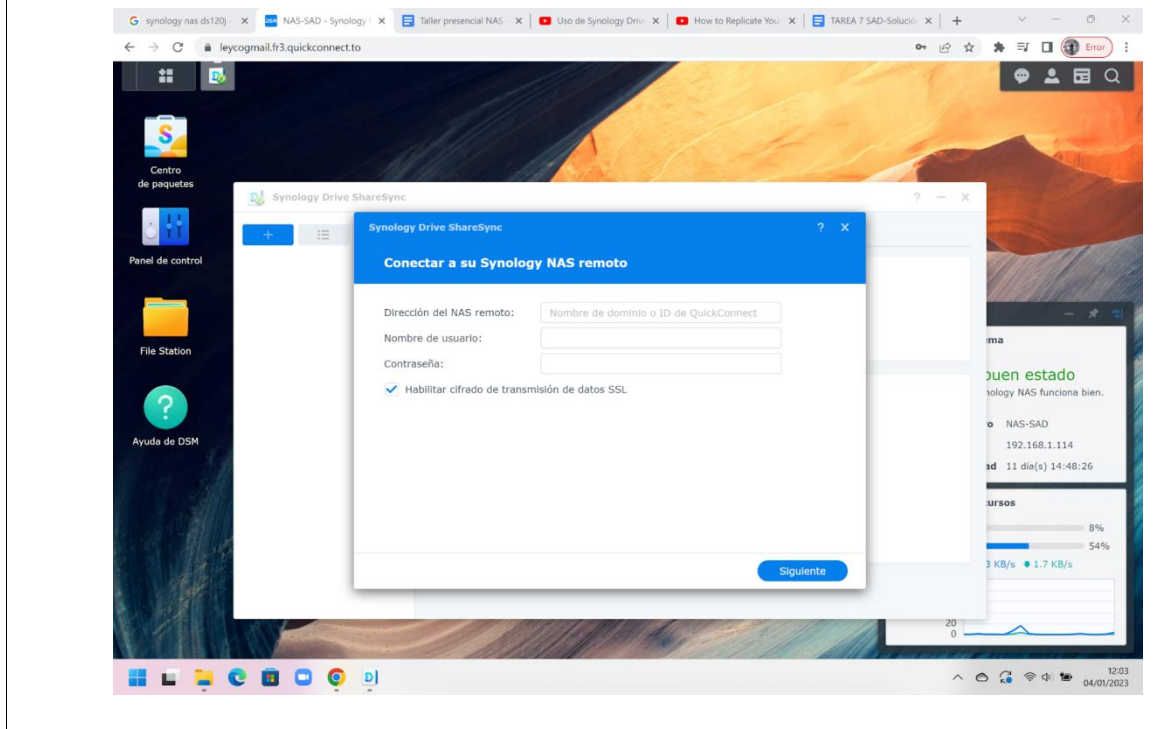


Ahora debemos establecer la conexión entre ambos NAS, el local y el remoto. Para ello en el NAS "local" en Panel de control → Synology Drive ShareSync:





Metemos las credenciales del NAS remoto:





Synology Drive ShareSync

? X

Conectar a su Synology NAS remoto

Dirección del NAS remoto:

Nombre de usuario:

Contraseña:

Habilitar cifrado de transmisión de datos SSL

Siguiente

Seleccionamos la carpeta que queremos sincronizar:

Synology Drive ShareSync

? X

Seleccionar carpetas remotas

Seleccione las carpetas remotas en **leyco80sadnas** que desea sincronizar con este NAS

<input checked="" type="checkbox"/> Habilita...	Carpeta compartida	Description
<input checked="" type="checkbox"/>	DocSAD	

1 elemento

Anterior

Siguiente

Podemos editar la configuración de la carpeta a sincronizar y elegir que sea bidireccional, así la sincronización será consistente:

Editar carpeta compartida DocSAD
✕

Carpeta
Filtro de archivos
Modo de sincronización

Modo de sincronización de archivos

Sincronizar todos los privilegios
 Sincronizar solo los privilegios de usuario del dominio
 Sincronizar archivos (excluir permisos y metadata)

Dirección de sincronización i

Sincronización bidireccional
 Descargar los datos solo del servidor de Synology Drive remoto
 Cargar los datos solo en el servidor de Synology Drive remoto

Mantener archivos eliminados localmente en el NAS remoto

Habilitar la comprobación de consistencia avanzada (se requieren más recursos)

Cancelar
OK

Una vez sincronizado ya hemos establecido la conexión entre ambos NAS:

Synology Drive ShareSync
? - ✕

+
☰
⚙️

Visión general

Carpetas sincronizadas

✓

Actualizado

Su Synology NAS se ha actualizado.

Pausar sincronización

Información de conexión

Dirección del NAS remoto: leyco80sadnas

Estado de la conexión: Conectar a través de LAN

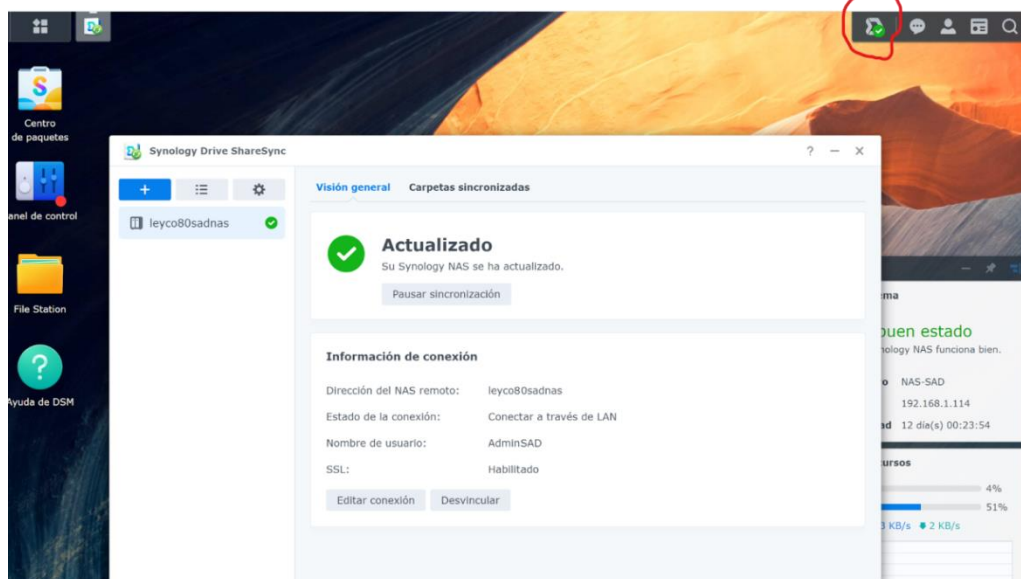
Nombre de usuario: AdminSAD

SSL: Habilitado

Editar conexión

Desvincular

Nos aparecerá en la esquina superior derecha del NAS local el icono de sincronización:

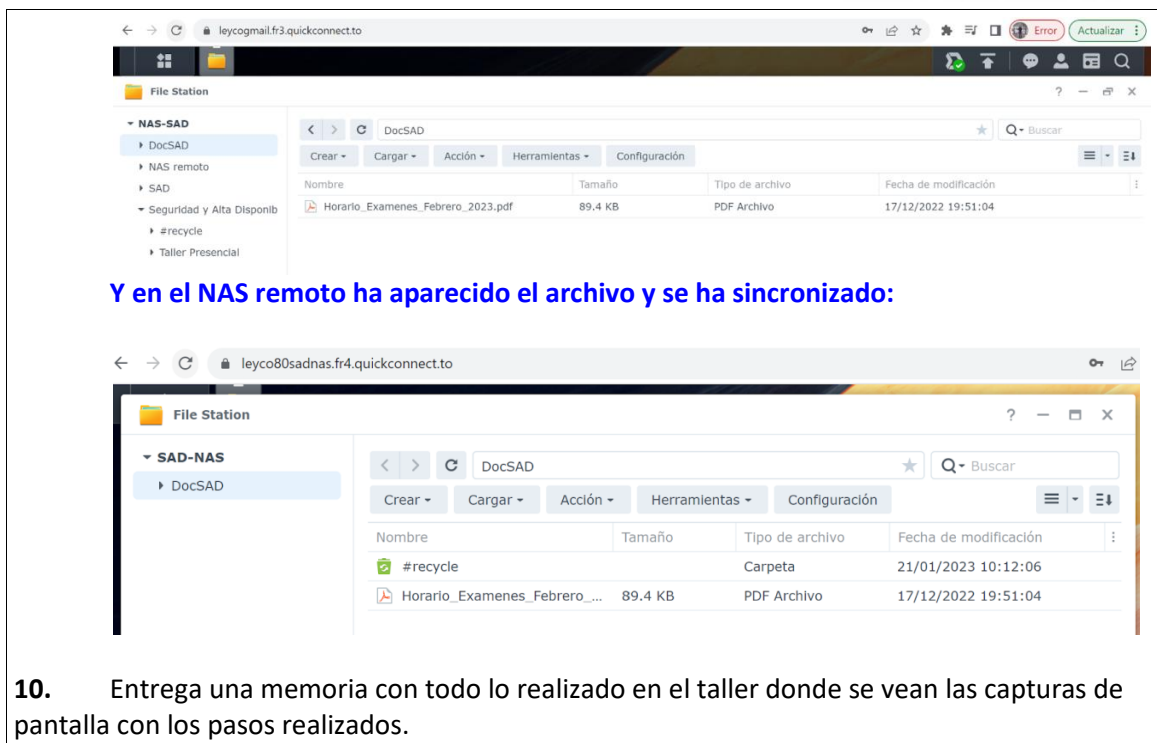


Y si pinchamos sobre el icono podremos ver el estado de la sincronización y las carpetas que se están sincronizando en la parte de “abrir en carpeta”:



Ahora si nos vamos a file manager y copiamos algún archivo en la carpeta que hemos sincronizado con el NAS “remoto” veremos que nos aparece en la carpeta correspondiente en el NAS “remoto”:

En el NAS local:



Y en el NAS remoto ha aparecido el archivo y se ha sincronizado:

10. Entrega una memoria con todo lo realizado en el taller donde se vean las capturas de pantalla con los pasos realizados.

() Añadir los talleres que se consideren oportunos, con sus correspondientes actividades propuestas*

RECURSOS Y ESPACIOS NECESARIOS:

- ✓ Aula con ordenadores, proyector y conexión a internet:
 - 2 NAS Synology DS120j por persona (**120,66 € x 2**) con un disco duro de 4TB (**102 € x 2**) cada uno. En caso de no disponer de NAS, éste se podrá virtualizar.
 - 3 PC por persona, 8 GB RAM mínimo, completos (monitor, ratón, teclado, torre, cable de red) (**840 € x 3**)

PRESUPUESTO POR PARTICIPANTE: 2965 €



Calificación:

Ítem evaluado (Uno o varios CEs)	Niveles de desempeño				
	Óptimo 1	Notable 0,75	Suficiente 0,5	Mejorable 0,25	No apto 0
6.g: Instala y configura apache en 2 máquinas					
6.g: Instala y configura nginx					
6.g: Instala y configura HAProxy					
6.g: Comprueba el balanceo de carga entre los 2 servidores con nginx					
6.g: Comprueba el balanceo de carga entre los 2 servidores y Haproxy					
6.b: Instala el disco duro en el NAS y lo configura					
6.b: Crea usuarios, carpetas compartidas y permite acceso sin loguearse en DSM en el NAS					
6.b: Configura el NAS para acceder a través del explorador de archivos					
6.b: Sincroniza archivos del PC con el NAS					
6.b: Sincroniza datos entre diferentes NAS					
TOTAL					

(*) Las filas corresponderán a uno o varios criterios de evaluación evaluados. El número de columnas podrán variar en función de niveles de logro que se definan en la rúbrica.