

Ciberseguridad en Entornos de las Tecnologías de la Información

Módulo 5021 – Incidentes de Ciberseguridad

Ejercicio – Implementación de un NAS Securizado

Pliego de Descargo

- *Los ejercicios y conocimientos contenidos en el Módulo 5021, Incidentes de Ciberseguridad, tienen un propósito exclusivamente formativo, por lo que **nunca se deberán utilizar con fines maliciosos o delictivos.***
- *Ni el Ministerio de Educación y Formación Profesional como organismo oficial, ni el CIDEAD como área integrada en el mismo, serán responsables en ningún caso de los daños directos o indirectos que pudieran derivarse del uso inadecuado de las herramientas de hacking ético utilizadas en dichos ejercicios.*



Introducción

- En el ámbito Open Source existen muchas alternativas SW para implementar un NAS (*Network-Attached Storage*), no obstante, aunque se trata de buenas aplicaciones cuyo código abierto se puede modificar derivándolo mediante un branch, pensamos que no aportan demasiado en el plano formativo.
- Así pues, en este ejercicio montaremos un NAS completo integrando las unidades de disco con la CPU mediante los correspondientes comandos de bajo nivel. Esto nos permitirá tener control total sobre lo que estamos haciendo en cada momento, pues uno de los principales problemas de las aplicaciones NAS es la falta de visión sobre las tareas en máquina y sobre la forma de manejar nuestra información almacenada, tan valiosa para nosotros.
- Este mismo criterio aplica al capítulo de la seguridad de la información. Al montar nosotros el NAS completo, gestionaremos las cuentas de usuario, cifraremos el disco duro y no perderemos el acceso accidentalmente a la información almacenada.



Índice de contenidos

1. Cifrado del Disco Duro
2. Montaje del Disco Duro
3. Exportación por Samba
4. Conexión al Disco desde Windows



1. Cifrado del Disco Duro

Conexión del Disco Duro

- Conectamos un disco duro a un puerto USB de una Raspberry Pi y chequeamos las unidades de almacenamiento disponibles.

```
pi@cloe: ~  
login as: pi  
pi@192.168.1.76's password:  
Linux cloe 5.10.17-v7+ #1403 SMP Mon Feb 22 11:29:51 GMT 2021 armv7l  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Fri Apr 16 10:34:59 2021 from 192.168.1.101  
pi@cloe:~ $ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
sda           8:0    0  1,8T  0 disk  
└─sda1         8:1    0  1,8T  0 part /media/pi/Elements  
mmcblk0      179:0    0  59,5G  0 disk  
├─mmcblk0p1   179:1    0   256M  0 part /boot  
└─mmcblk0p2   179:2    0  59,2G  0 part /  
pi@cloe:~ $
```

- El disco duro en cuestión se sobrescribirá, por lo que deberá estar vacío, con información ya respaldada, o contener datos no relevantes.
- En caso de que alguno de los programas usados en este ejercicio aún no esté instalado en la Raspberry Pi, bastará con instalarlo ad hoc:

`sudo apt-get install [programa]`

Conexión del Disco Duro

- En la captura de pantalla adyacente se muestra la situación antes y después de la conexión del disco duro.
- Si el disco está formateado para Linux ó UNIX, o como NTFS, la Raspberry lo automontará inmediatamente.
- En caso de que esto ocurra, antes de continuar será necesario desmontarlo mediante el comando siguiente:

```
sudo umount /media/pi/[nombre disco]
```

```
pi@cloe: ~  
pi@cloe:~ $ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
mmcblk0     179:0    0  59,5G  0 disk  
└─mmcblk0p1 179:1    0   256M  0 part /boot  
└─mmcblk0p2 179:2    0  59,2G  0 part /  
pi@cloe:~ $ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
sdc         8:32     0   1,8T  0 disk  
└─sdc1       8:33     0   1,8T  0 part /media/pi/Nuevo vol  
mmcblk0     179:0    0  59,5G  0 disk  
└─mmcblk0p1 179:1    0   256M  0 part /boot  
└─mmcblk0p2 179:2    0  59,2G  0 part /  
pi@cloe:~ $
```

Conexión del Disco Duro

- Como el disco aún no tiene MOUNTPOINT, estableceremos su estructura de particiones como GPT. En este momento **se perderá toda la información** existente en el disco.

```
pi@cloe: ~  
pi@cloe:~ $ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
sdc          8:32   0  1,8T  0 disk  
└─sdc1       8:33   0  1,8T  0 part  
mmcblk0     179:0   0 59,5G  0 disk  
├─mmcblk0p1 179:1   0  256M  0 part /boot  
└─mmcblk0p2 179:2   0 59,2G  0 part /  
pi@cloe:~ $ sudo parted /dev/sdc mklabel gpt  
Warning: The existing disk label on /dev/sdc will be destroyed and all data on this disk will be lost. Do you want to continue?  
Yes/No? y  
Information: You may need to update /etc/fstab.  
  
pi@cloe:~ $ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
sdc          8:32   0  1,8T  0 disk  
mmcblk0     179:0   0 59,5G  0 disk  
├─mmcblk0p1 179:1   0  256M  0 part /boot  
└─mmcblk0p2 179:2   0 59,2G  0 part /  
pi@cloe:~ $
```


Cifrado Disco Duro

- Procederemos al cifrado del disco duro, proporcionando una contraseña de seguridad que custodiaremos celosamente.

```
pi@cloe: ~  
pi@cloe:~ $ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
sdc          8:32   0  1,8T  0 disk  
mmcblk0     179:0    0  59,5G  0 disk  
└─mmcblk0p1 179:1    0   256M  0 part /boot  
└─mmcblk0p2 179:2    0  59,2G  0 part /  
pi@cloe:~ $ sudo cryptsetup -c aes -h sha256 -y -s 256 luksFormat /dev/sdc  
AVISO: El dispositivo /dev/sdc ya contiene una firma de partición 'gpt'.  
  
WARNING!  
=====  
Esto sobrescribirá los datos en /dev/sdc de forma irrevocable.  
  
Are you sure? (Type uppercase yes): YES  
Introduzca la frase contraseña de /dev/sdc:  
Verifique la frase contraseña:  
pi@cloe:~ $ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
sdc          8:32   0  1,8T  0 disk  
mmcblk0     179:0    0  59,5G  0 disk  
└─mmcblk0p1 179:1    0   256M  0 part /boot  
└─mmcblk0p2 179:2    0  59,2G  0 part /  
pi@cloe:~ $
```

Apertura del Cifrado del Disco Duro

- Una vez cifrado el disco duro, para poder seguir trabajando sobre él será necesario abrir el cifrado con el comando indicado en la captura, y proporcionando la contraseña.
- Hecho esto, aparecerá la nueva partición con el nombre que indiquemos y con tipo cifrado (TYPE=crypt).

```
pi@cloe: ~  
pi@cloe:~ $ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
sdc          8:32   0  1,8T  0 disk  
mmcblk0     179:0    0  59,5G  0 disk  
└─mmcblk0p1  179:1    0   256M  0 part /boot  
└─mmcblk0p2  179:2    0  59,2G  0 part /  
pi@cloe:~ $ sudo cryptsetup luksOpen /dev/sdc hdd  
Introduzca la frase contraseña de /dev/sdc:  
pi@cloe:~ $ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
sdc          8:32   0  1,8T  0 disk  
└─hdd        254:2    0  1,8T  0 crypt  
mmcblk0     179:0    0  59,5G  0 disk  
└─mmcblk0p1  179:1    0   256M  0 part /boot  
└─mmcblk0p2  179:2    0  59,2G  0 part /  
pi@cloe:~ $
```

Verificación de la Creación del Dispositivo

- Verificamos con fdisk la creación del correspondiente dispositivo y su registro en la tabla de dispositivos.
- Aparecerá precedido de la ruta “/dev/mapper”.

```
pi@cloe: ~  
pi@cloe:~ $ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT  
sdc          8:32   0  1,8T  0 disk  
└─hdd        254:2   0  1,8T  0 crypt  
mmcblk0     179:0   0 59,5G  0 disk  
├─mmcblk0p1 179:1   0  256M  0 part  /boot  
└─mmcblk0p2 179:2   0 59,2G  0 part  /  
pi@cloe:~ $ sudo fdisk -l|grep hdd  
Disk /dev/mapper/hdd: 1,8 TiB, 2000348512256 bytes, 3906930688 sectors  
pi@cloe:~ $
```

Formateo de la Partición

- En este punto ya se puede proceder a formatear la partición con formato ext4, que funciona muy bien en discos NAS para Linux y UNIX.
- Es importante capturar la pantalla tras ejecutar este comando, pues indicará los bloques de respaldo del Superblock, información muy útil en caso de recuperación de datos.

```
pi@cloe:~ $ sudo mkfs.ext4 -L 'almacen' /dev/mapper/hdd
mke2fs 1.44.5 (15-Dec-2018)
Found a atari partition table in /dev/mapper/hdd
Proceed anyway? (y,N) y
Creating filesystem with 488366336 4k blocks and 122093568 inodes
Filesystem UUID: ac3a16cf-1f9c-452f-9377-b55c1a25ac85
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,
    102400000, 214990848

Allocating group tables: done
Writing inode tables: done
Creating journal (262144 blocks): done
Writing superblocks and filesystem accounting information: done

pi@cloe:~ $ sudo lsblk -fs
NAME        FSTYPE     LABEL     UUID                                         FSAVAIL FSUSE% MOUNTPOINT
mmcblk0p1   vfat       boot      F4F1-BC2C                                   204,4M   19% /boot
└─mmcblk0
mmcblk0p2   ext4       rootfs    163660a6-ad17-44fc-99c5-5c75e78ad815       51G      8% /
└─mmcblk0
hd
hdl
hdd         ext4       almacen   ac3a16cf-1f9c-452f-9377-b55c1a25ac85
└─sdc       crypto_LUKS a0d59f2f-ce6c-4856-8716-502513219f12
pi@cloe:~ $
```

2. Montaje del Disco Duro

Punto de Montaje

- Creamos un directorio que servirá como punto de montaje para el disco duro, asignándole cualquier nombre.
- En este caso hemos asignado el mismo nombre que hemos utilizado para nombrar la partición, “hdd”, pero este detalle no es obligatorio.

```
pi@cloe: ~  
pi@cloe:~ $ sudo fdisk -l|grep hdd  
Disk /dev/mapper/hdd: 1,8 TiB, 2000348512256 bytes, 3906930688 sectors  
pi@cloe:~ $ pwd  
/home/pi  
pi@cloe:~ $ mkdir hdd  
pi@cloe:~ $ ls -l  
total 60  
drwxr-xr-x 3 pi pi 4096 abr  5 17:53 AES  
-rwxr-xr-x 1 pi pi  22 abr 15 11:20 ARRANCAR_DB  
drwxr-xr-x 2 pi pi 4096 ene 11 14:01 Bookshelf  
drwxr-xr-x 5 pi pi 4096 abr 14 19:09 data-masking-utils  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Desktop  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Documents  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Downloads  
drwxr-xr-x 2 pi pi 4096 abr 17 16:57 hdd  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Music  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Pictures  
drwxr-xr-x 9 pi pi 4096 abr  5 17:56 pip  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Public  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Templates  
-r-xr-xr-x 1 pi pi  497 abr 12 22:07 vector.py  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Videos  
pi@cloe:~ $
```

Montaje del Disco Duro

- Ejecutamos el comando de montaje y comprobamos que éste se ha efectuado correctamente.

```
pi@cloe: ~  
pi@cloe:~ $ sudo fdisk -l|grep hdd  
Disk /dev/mapper/hdd: 1,8 TiB, 2000348512256 bytes, 3906930688 sectors  
pi@cloe:~ $ sudo mount /dev/mapper/hdd /home/pi/hdd  
pi@cloe:~ $ df -h  
S.ficheros      Tamaño Usados  Disp Uso% Montado en  
/dev/root       59G    4,9G   52G   9% /  
devtmpfs        430M     0   430M   0% /dev  
tmpfs           463M     0   463M   0% /dev/shm  
tmpfs           463M    13M   451M   3% /run  
tmpfs           5,0M    4,0K   5,0M   1% /run/lock  
tmpfs           463M     0   463M   0% /sys/fs/cgroup  
/dev/mmcblk0p1  253M    48M   205M  19% /boot  
tmpfs           93M    36K    93M   1% /run/user/1000  
/dev/mapper/hdd 1,8T    77M   1,7T   1% /home/pi/hdd  
pi@cloe:~ $
```

Cambio de Usuario y Grupo

- Después de efectuar el montaje, el usuario y el grupo del directorio serán root, por lo que los cambiaremos para que el usuario corriente pueda trabajar con la información sin ningún problema.

```
pi@cloe: ~  
pi@cloe:~ $ pwd  
/home/pi  
pi@cloe:~ $ ls -l  
total 60  
drwxr-xr-x 3 pi pi 4096 abr 5 17:53 AES  
-rwxr-xr-x 1 pi pi 22 abr 15 11:20 ARRANCAR_DB  
drwxr-xr-x 2 pi pi 4096 ene 11 14:01 Bookshelf  
drwxr-xr-x 5 pi pi 4096 abr 14 19:09 data-masking-utils  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Desktop  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Documents  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Downloads  
drwxr-xr-x 3 root root 4096 abr 17 16:45 hdd  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Music  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Pictures  
drwxr-xr-x 9 pi pi 4096 abr 5 17:56 pip  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Public  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Templates  
-r-xr-xr-x 1 pi pi 497 abr 12 22:07 vector.py  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Videos  
pi@cloe:~ $ sudo chown -R pi:pi /home/pi/hdd  
pi@cloe:~ $ ls -l  
total 60  
drwxr-xr-x 3 pi pi 4096 abr 5 17:53 AES  
-rwxr-xr-x 1 pi pi 22 abr 15 11:20 ARRANCAR_DB  
drwxr-xr-x 2 pi pi 4096 ene 11 14:01 Bookshelf  
drwxr-xr-x 5 pi pi 4096 abr 14 19:09 data-masking-utils  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Desktop  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Documents  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Downloads  
drwxr-xr-x 3 pi pi 4096 abr 17 16:45 hdd  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Music  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Pictures  
drwxr-xr-x 9 pi pi 4096 abr 5 17:56 pip  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Public  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Templates  
-r-xr-xr-x 1 pi pi 497 abr 12 22:07 vector.py  
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Videos  
pi@cloe:~ $
```


Grabación de Datos

- Grabamos algo de información en el disco recién montado, que monitorizaremos durante el resto del ejercicio.
- En este caso nos descargaremos de GitHub el Editor Gutenberg de Wordpress, que ocupa 300 MB aproximadamente.

```
pi@cloe: ~/hdd
pi@cloe:~ $ pwd
/home/pi
pi@cloe:~ $ cd hdd
pi@cloe:~/hdd $ ls -l
total 16
drwx----- 2 pi pi 16384 abr 17 16:45 lost+found
pi@cloe:~/hdd $ git clone https://github.com/WordPress/gutenberg.git
Clonando en 'gutenberg'...
remote: Enumerating objects: 264623, done.
remote: Counting objects: 100% (2118/2118), done.
remote: Compressing objects: 100% (1154/1154), done.
remote: Total 264623 (delta 1231), reused 1567 (delta 948), pack-reused 262505
Recibiendo objetos: 100% (264623/264623), 254.97 MiB | 5.71 MiB/s, listo.
Resolviendo deltas: 100% (192152/192152), listo.
Revisando archivos: 100% (6493/6493), listo.
pi@cloe:~/hdd $ ls -l
total 20
drwxr-xr-x 15 pi pi 4096 abr 17 17:05 gutenberg
drwx----- 2 pi pi 16384 abr 17 16:45 lost+found
pi@cloe:~/hdd $ du -sh gutenberg
310M    gutenberg
pi@cloe:~/hdd $ df -h
S.ficheros      Tamaño Usados  Disp Uso% Montado en
/dev/root        59G    4,9G   52G   9% /
devtmpfs         430M      0   430M   0% /dev
tmpfs            463M      0   463M   0% /dev/shm
tmpfs            463M    13M   451M   3% /run
tmpfs            5,0M    4,0K   5,0M   1% /run/lock
tmpfs            463M      0   463M   0% /sys/fs/cgroup
/dev/mmcblk0p1   253M    48M   205M  19% /boot
tmpfs            93M     36K    93M   1% /run/user/1000
/dev/mapper/hdd  1,8T   386M   1,7T   1% /home/pi/hdd
pi@cloe:~/hdd $
```

Reinicio de la Máquina

- Reiniciamos la máquina para estudiar la forma de proceder si se produce un rearranque.

```
pi@cloe: ~/hdd
pi@cloe:~/hdd $ ls -l
total 20
drwxr-xr-x 15 pi pi 4096 abr 17 17:05 gutenber
drwx----- 2 pi pi 16384 abr 17 16:45 lost+found
pi@cloe:~/hdd $ df -h
S.ficheros      Tamaño Usados  Disp Uso% Montado en
/dev/root        59G   4,9G   52G   9% /
devtmpfs         430M     0   430M   0% /dev
tmpfs            463M     0   463M   0% /dev/shm
tmpfs            463M   13M   451M   3% /run
tmpfs            5,0M   4,0K   5,0M   1% /run/lock
tmpfs            463M     0   463M   0% /sys/fs/cgroup
/dev/mmcblk0p1   253M   48M   205M  19% /boot
tmpfs            93M    36K    93M   1% /run/user/1000
/dev/mapper/hdd  1,8T  386M  1,7T   1% /home/pi/hdd
pi@cloe:~/hdd $ sudo reboot
```

Reinicio de la Máquina

- Tras el reinicio, se comprueba que la partición cifrada está desmontada y cerrada, lo cual es correcto.

```
pi@cloe: ~  
pi@cloe:~ $ sudo lsblk -fs  
NAME        FSTYPE     LABEL     UUID                                 FSAVAIL FSUSE% MOUNTPOINT  
sda          crypto_LUKS a0d59f2f-ce6c-4856-8716-502513219f12  
mmcblk0p1   vfat       boot      F4F1-BC2C                          204,4M   19% /boot  
└─mmcblk0  
mmcblk0p2   ext4       rootfs    163660a6-ad17-44fc-99c5-5c75e78ad815 51G      8% /  
└─mmcblk0  
pi@cloe:~ $ df -h  
S.ficheros    Tamaño Usados  Disp  Uso% Montado en  
/dev/root      59G   4,9G   52G   9% /  
devtmpfs       430M     0   430M   0% /dev  
tmpfs          463M     0   463M   0% /dev/shm  
tmpfs          463M   13M   451M   3% /run  
tmpfs          5,0M   4,0K   5,0M   1% /run/lock  
tmpfs          463M     0   463M   0% /sys/fs/cgroup  
/dev/mmcblk0p1 253M   48M   205M  19% /boot  
tmpfs          93M    36K   93M    1% /run/user/1000  
pi@cloe:~ $
```

Reapertura de la Partición Cifrada

- Se vuelve a abrir la partición cifrada, suministrando la contraseña de cifrado.
- Una vez abierta la partición, se repite la operación de montaje del disco.
- Hecho esto, se comprueba que el montaje se ha efectuado correctamente sobre el punto de montaje.

```
pi@cloe: ~  
pi@cloe:~ $ sudo lsblk -fs  
NAME        FSTYPE     LABEL      UUID                                 FSAVAIL FSUSE% MOUNTPOINT  
sda         crypto_LUKS a0d59f2f-ce6c-4856-8716-502513219f12  
mmcblk0p1   vfat        boot       F4F1-BC2C                          204,4M   19% /boot  
└─mmcblk0  
mmcblk0p2   ext4        rootfs     163660a6-ad17-44fc-99c5-5c75e78ad815 51G      8% /  
└─mmcblk0  
pi@cloe:~ $ sudo cryptsetup luksOpen /dev/sda hdd  
Introduzca la frase contraseña de /dev/sda:  
pi@cloe:~ $ sudo mount /dev/mapper/hdd /home/pi/hdd  
pi@cloe:~ $ df -h  
S.ficheros      Tamaño Usados  Disp Uso% Montado en  
/dev/root        59G   4,9G   52G   9% /  
devtmpfs         430M     0   430M   0% /dev  
tmpfs            463M     0   463M   0% /dev/shm  
tmpfs            463M   13M   451M   3% /run  
tmpfs            5,0M   4,0K   5,0M   1% /run/lock  
tmpfs            463M     0   463M   0% /sys/fs/cgroup  
/dev/mmcblk0p1   253M   48M   205M  19% /boot  
tmpfs            93M    36K    93M   1% /run/user/1000  
/dev/mapper/hdd  1,8T  386M  1,7T   1% /home/pi/hdd
```

Comprobación de Datos

- Tras la reapertura y el montaje de la partición cifrada, se comprueba que los datos están intactos y ubicados en su lugar correspondiente.

```
pi@cloe: ~/hdd
pi@cloe:~/hdd $ cd
pi@cloe:~ $ ls -l
total 60
drwxr-xr-x 3 pi pi 4096 abr  5 17:53 AES
-rwxr-xr-x 1 pi pi  22 abr 15 11:20 ARRANCAR_DB
drwxr-xr-x 2 pi pi 4096 ene 11 14:01 Bookshelf
drwxr-xr-x 5 pi pi 4096 abr 14 19:09 data-masking-utils
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Desktop
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Documents
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Downloads
drwxr-xr-x 4 pi pi 4096 abr 17 17:03 hdd
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Music
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Pictures
drwxr-xr-x 9 pi pi 4096 abr  5 17:56 pip
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Public
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Templates
-r-xr-xr-x 1 pi pi  497 abr 12 22:07 vector.py
drwxr-xr-x 2 pi pi 4096 mar 25 13:15 Videos
pi@cloe:~ $ cd hdd
pi@cloe:~/hdd $ ls -l
total 20
drwxr-xr-x 15 pi pi  4096 abr 17 17:05 gutenbergr
drwx----- 2 pi pi 16384 abr 17 16:45 lost+found
pi@cloe:~/hdd $
```

Creación del Script de Montaje

- Creamos un script de montaje con estos comandos, para ejecutarlo tras un rearranque, **SIEMPRE manualmente** e introduciendo la clave de apertura del cifrado (labor del administrador del host).

```
pi@cloe: ~  
pi@cloe:~ $ nano MONTAR_ALMACEN
```

```
pi@cloe: ~  
GNU nano 3.2  
  
Script de Montaje de Disco NAS Cifrado - Almacen  
# Montaje SIEMPRE manual, solicita clave al usuario  
  
sudo cryptsetup luksOpen /dev/sda hdd  
sudo mount /dev/mapper/hdd /home/pi/hdd
```

```
pi@cloe: ~  
pi@cloe:~ $ df -h  
S.ficheros      Tamaño Usados  Disp Uso% Montado en  
/dev/root        59G   4,9G   52G   9% /  
devtmpfs         430M     0   430M   0% /dev  
tmpfs            463M     0   463M   0% /dev/shm  
tmpfs            463M   6,3M   456M   2% /run  
tmpfs            5,0M   4,0K   5,0M   1% /run/lock  
tmpfs            463M     0   463M   0% /sys/fs/cgroup  
/dev/mmcblk0p1   253M   48M   205M  19% /boot  
tmpfs            93M    36K    93M   1% /run/user/1000  
pi@cloe:~ $ ./MONTAR_ALMACEN  
Introduzca la frase contraseña de /dev/sda:  
pi@cloe:~ $ df -h  
S.ficheros      Tamaño Usados  Disp Uso% Montado en  
/dev/root        59G   4,9G   52G   9% /  
devtmpfs         430M     0   430M   0% /dev  
tmpfs            463M     0   463M   0% /dev/shm  
tmpfs            463M   6,3M   456M   2% /run  
tmpfs            5,0M   4,0K   5,0M   1% /run/lock  
tmpfs            463M     0   463M   0% /sys/fs/cgroup  
/dev/mmcblk0p1   253M   48M   205M  19% /boot  
tmpfs            93M    36K    93M   1% /run/user/1000  
/dev/mapper/hdd  1,8T   386M   1,7T   1% /home/pi/hdd  
pi@cloe:~ $
```



3. Exportación por Samba

Instalación de Samba

- Instalamos el protocolo Samba en el host para que los usuarios de Windows puedan conectarse al almacén NAS.

```
pi@cloe: ~  
pi@cloe:~ $ sudo apt-get -y install samba samba-common-bin  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.  
ant ant-optional antlr groovy ivy java-wrappers junit4 libantlr-java libapache-pom-java libasm-java libbsf-java libcommons-cli-java libcommons-lang3-java  
libcommons-logging-java libcommons-parent-java libel-api-java libelasticsearch1.7-java libgpars-groovy-java libhamcrest-java libhawtjni-runtime-java  
libhyperic-sigar-java libjansi-java libjansi-native-java libjaxen-java libjcommander-java libjcsp-java libjdom2-java libjline2-java libjna-java libjna-jni  
libjson-simple-java libjsp-api-java libjsr166y-java libjts-java liblog4j-extras1.2-java liblog4j1.2-java liblucene4.10-java libmultiverse-core-java libqdox-java  
libregexp-java libservlet-api-java libservlet3.1-java libspatial4j-0.4-java libwebsocket-api-java libxpp3-java libxstream-java
```


Configuración de Samba

- Añadimos un epígrafe descriptivo del disco NAS al final del fichero de configuración de Samba, según se indica en las capturas de pantalla adjuntas.

```
pi@cloe: ~  
pi@cloe:~ $ sudo nano /etc/samba/smb.conf  
  
# Exportar Samba - Almacen Cifrado  
[ALMACEN]  
comment = ALMACEN  
path = /home/pi/hdd  
browseable = yes  
writeable = yes  
only guest = no  
create mask = 0777  
directory mask = 0777  
public = no
```

Creación de Usuario en Samba

- Creamos las credenciales de usuario Samba para el usuario que montará el disco NAS desde Windows.
- Finalizada la configuración y la creación de usuarios, reiniciamos el servicio Samba.

```
pi@cloe: ~  
pi@cloe:~ $ sudo smbpasswd -a pi  
New SMB password:  
Retype new SMB password:  
Added user pi.  
pi@cloe:~ $
```

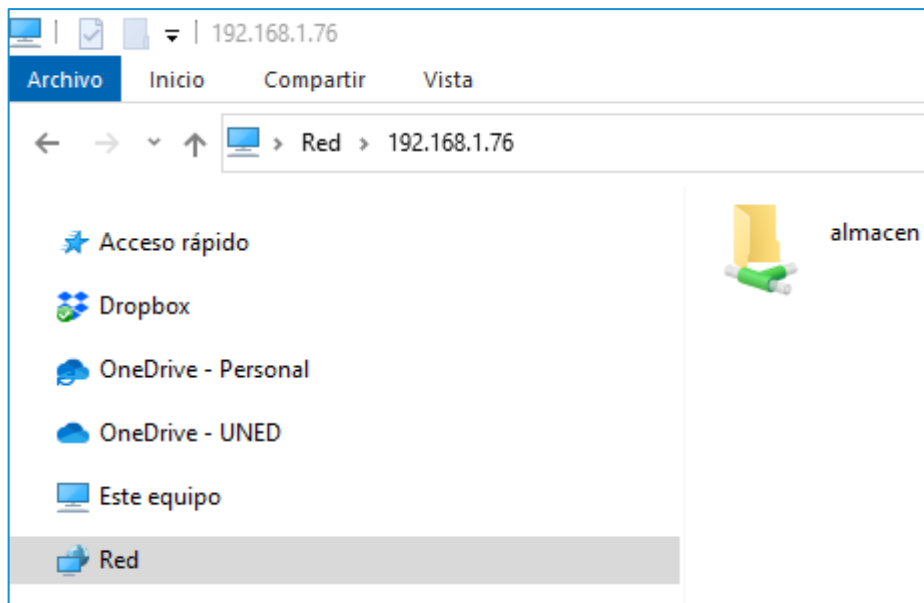
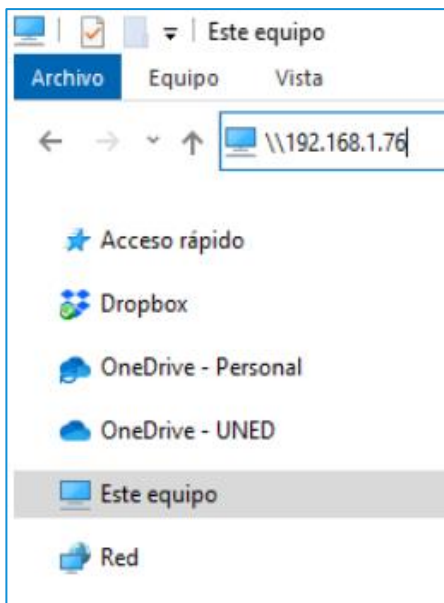
```
pi@cloe: ~  
pi@cloe:~ $ sudo /etc/init.d/smbd restart  
[ ok ] Restarting smbd (via systemctl): smbd.service.  
pi@cloe:~ $
```



4. Conexión al Disco desde Windows

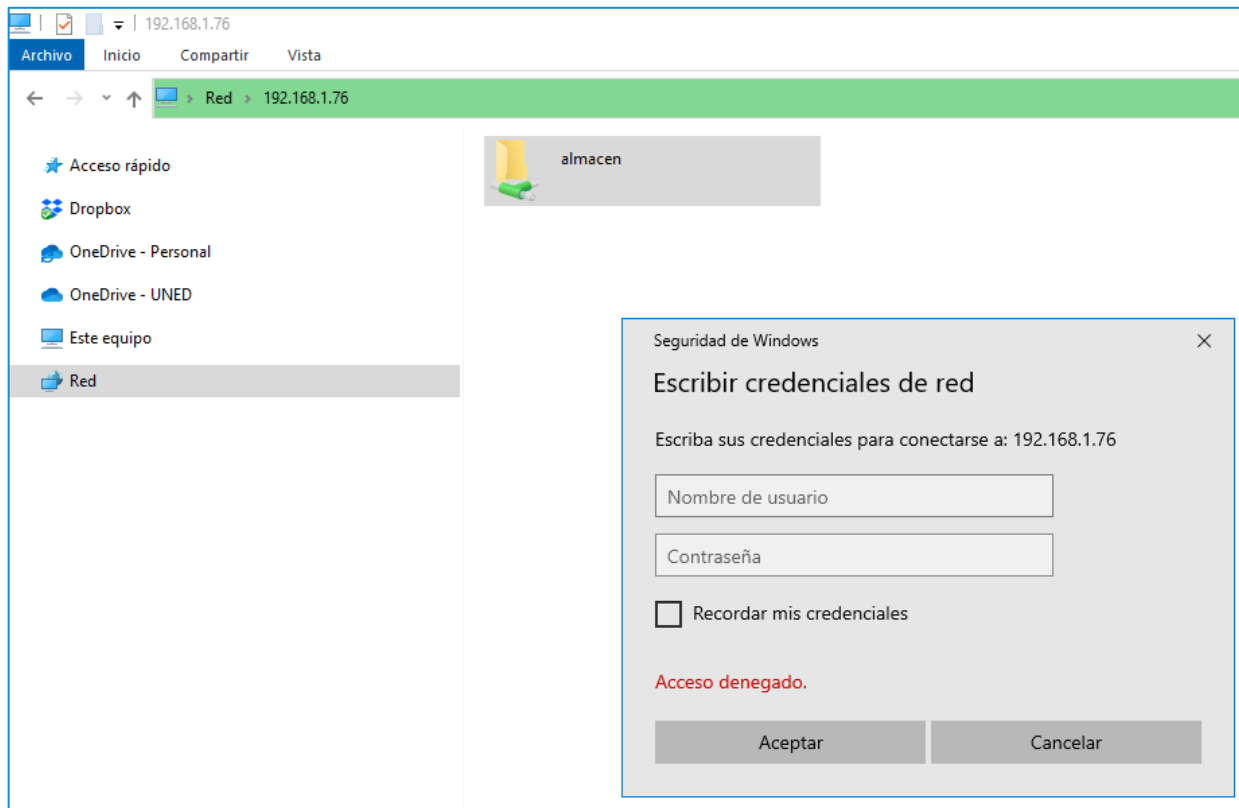
Localización del NAS desde Windows

- Abrimos un Explorador de Archivos e introducimos la dirección del host, precedida por dos barras invertidas. Hecho esto, se visualizará el directorio del almacén NAS.



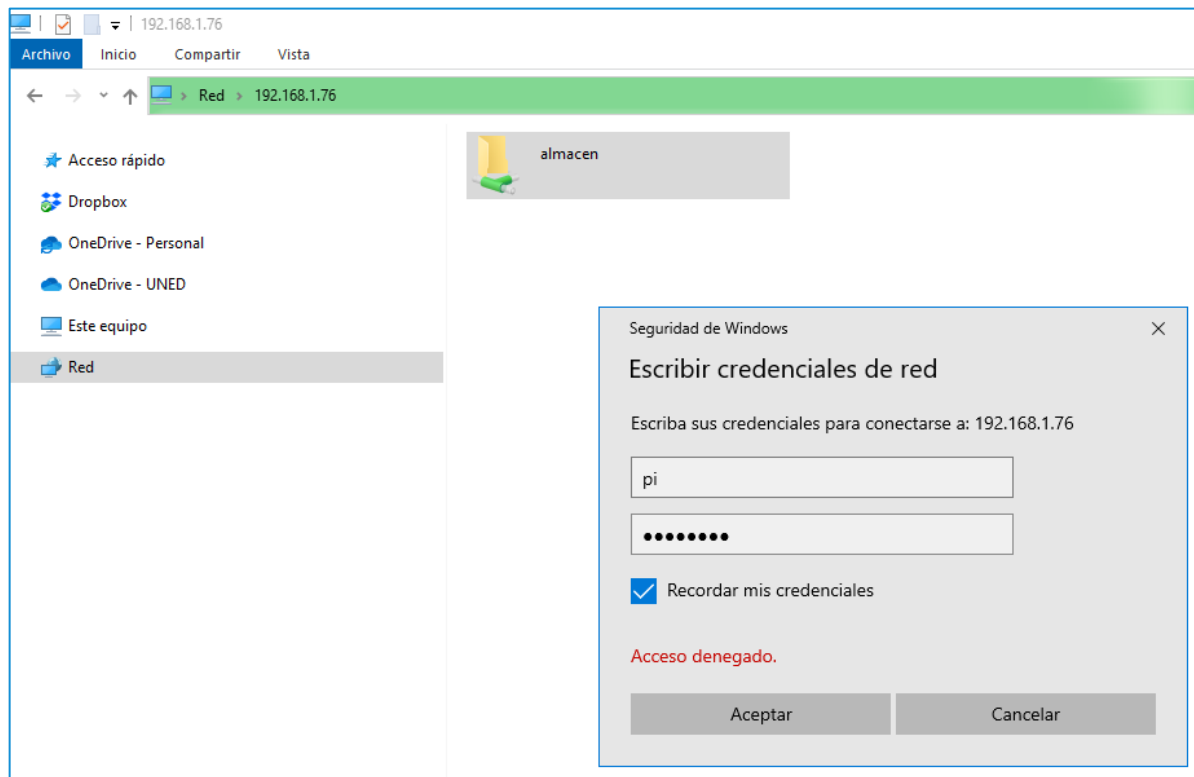
Apertura Almacén

- Al intentar abrir el almacén, el host nos solicitará las credenciales de acceso.



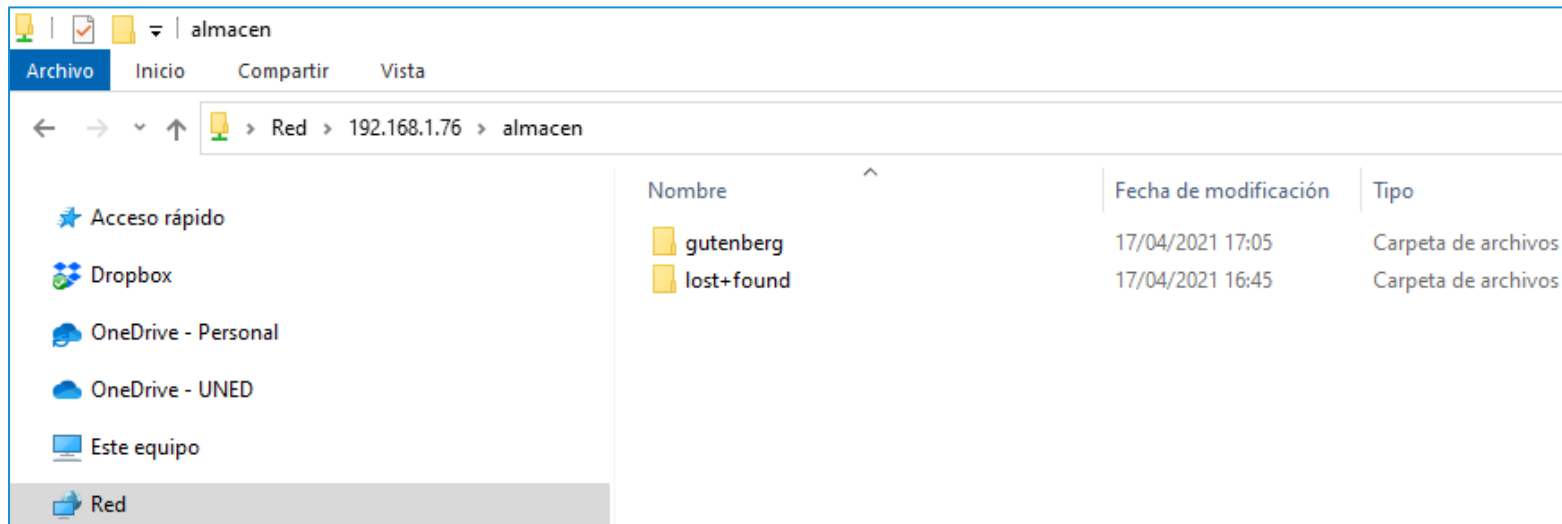
Credenciales Samba

- Introduciremos las credenciales que hemos dado de alta en la configuración Samba del host.



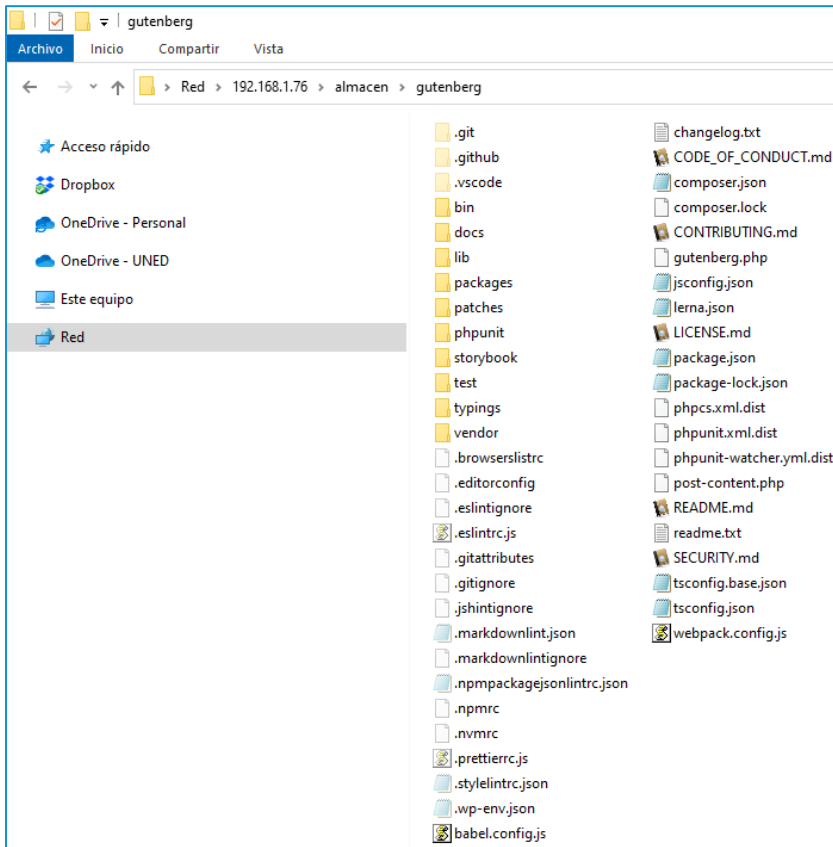
Apertura del Directorio del NAS

- Una vez introducidas las credenciales, se abrirá el almacén NAS, mostrando los directorios: lost+found (habitual en cualquier medio Linux/UNIX) y el directorio de datos creado por nosotros.



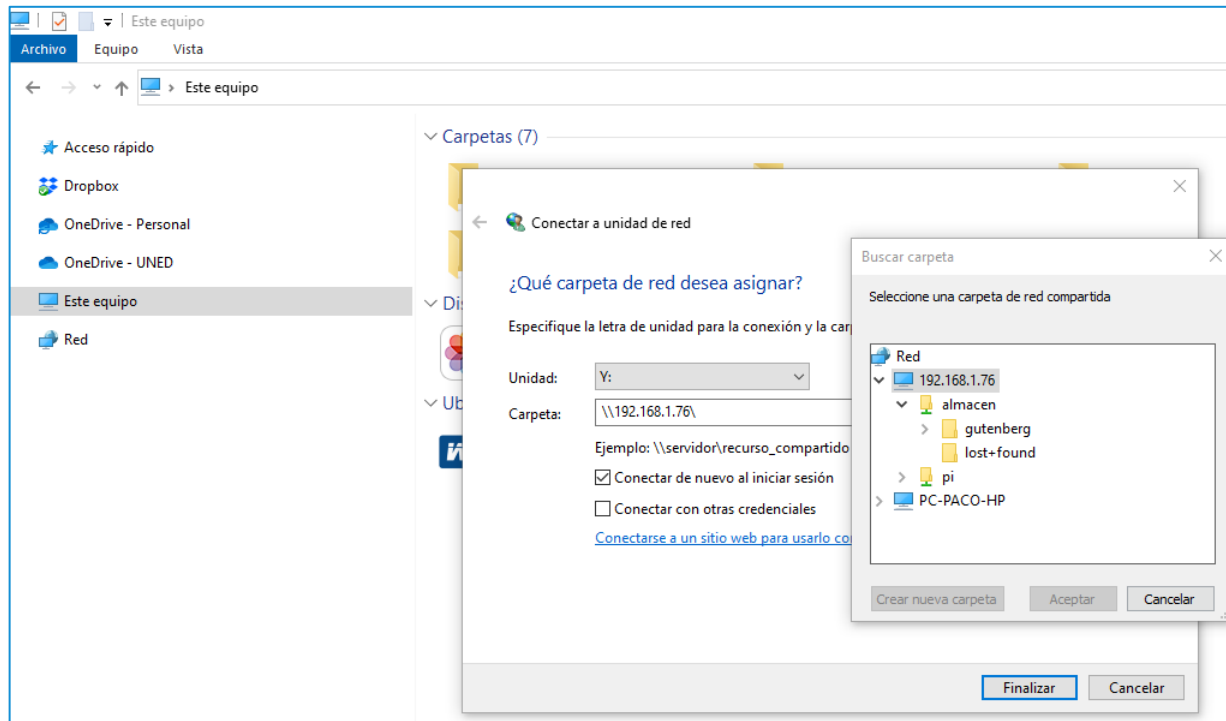
Verificación de Datos

- Una vez abierto el directorio del NAS, podremos comprobar la integridad de la información que habíamos grabado antes de exportar y montar el almacén.



Verificación de Datos y Fin de la Práctica

- Una vez comprobado el correcto acceso al NAS desde Windows, consolidamos la conexión mediante la opción “Conectar a Unidad de Red”.
- Para ello se introducirá la dirección del servidor precedida por dos barras invertidas, se pulsará “Examinar” y se elegirá la carpeta correspondiente.



Bibliografía

- <https://gitlab.com/cryptsetup/cryptsetup/>
- <https://www.samba.org/>
- <https://support.microsoft.com/es-es/windows/asignar-una-unidad-de-red-en-windows-10-29ce55d1-34e3-a7e2-4801-131475f9557d>