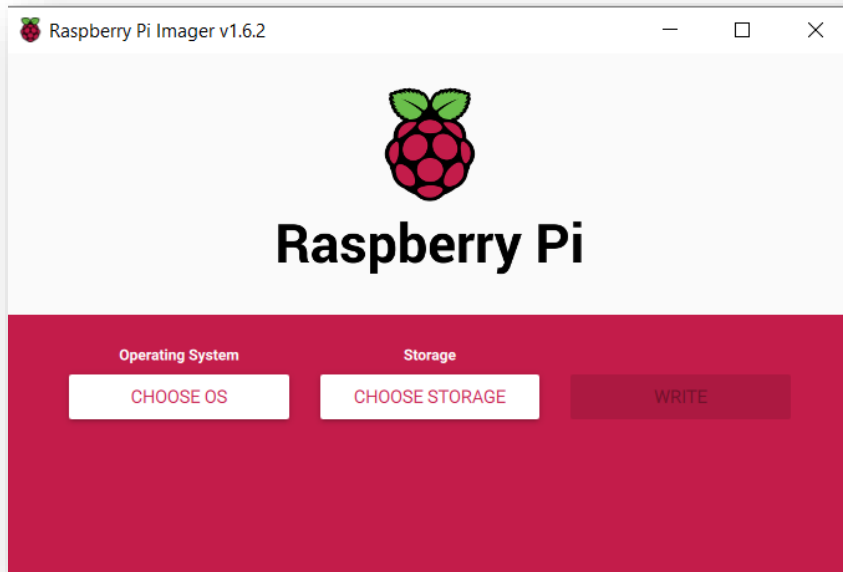


Guia d'instal·lació del nostre sistema NAS

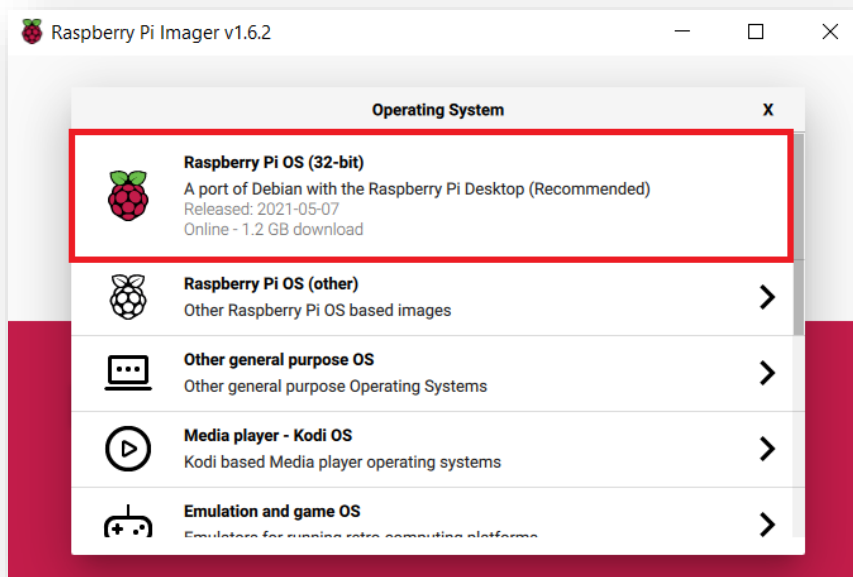
Instal·lar el sistema operatiu a la nostra RASPBERRY

Des de el nostre ordinador d'us particular instal·larem el sistema operatiu en una targeta micró sd que posteriorment utilitzarem en la nostra RASPBERRY

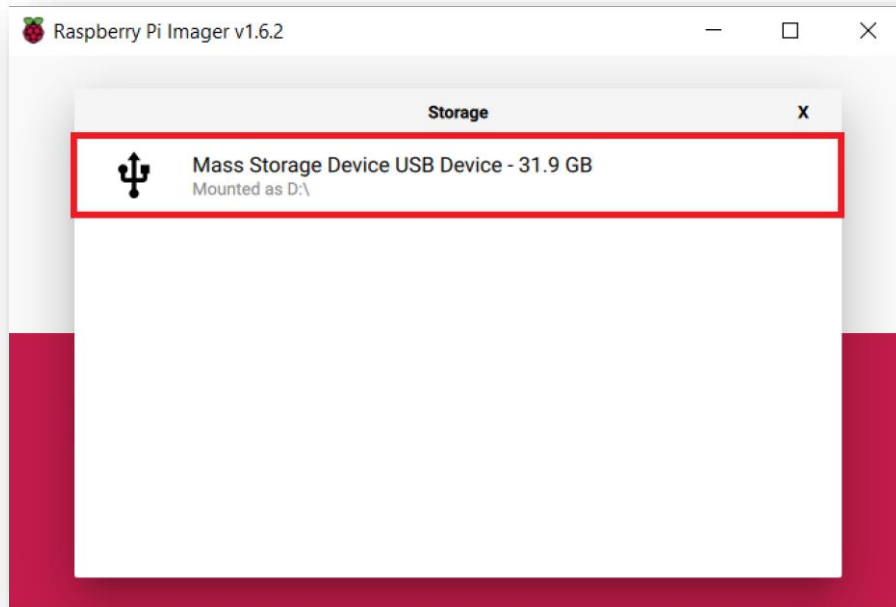
- 1r: descarregar i instal·lar *Raspberry Pi Imager* ([Descarrega](#))



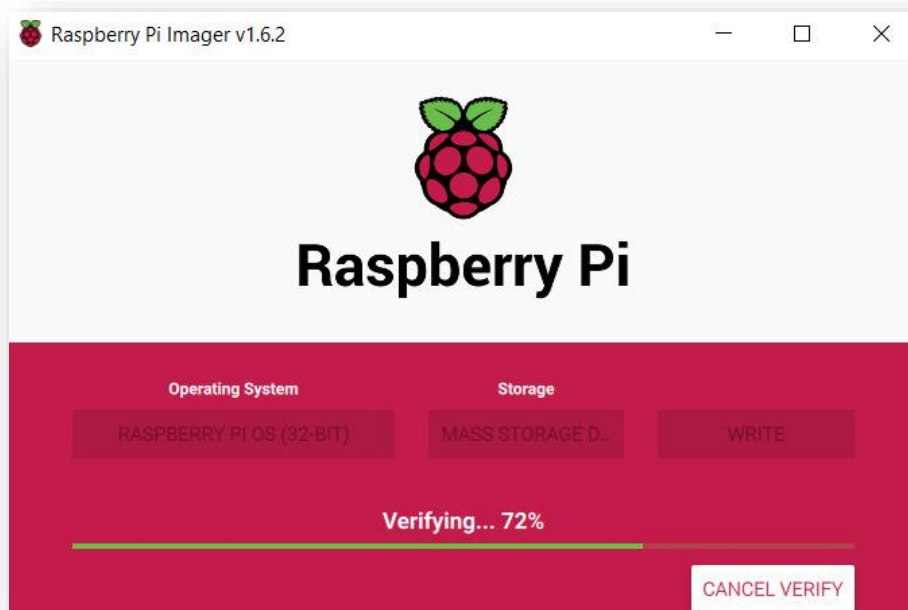
- 2n: al desplegable “Operating System” seleccionar la primera opció recomanada:



- 3r: al desplegable “Storage” seleccionar la targeta micró sd desitjada



- 4t: clicar a “Write” i esperar a que el procés finalitzi.



- 5è: Obrir el directori de la memòria sd (es possible que calgui extreure-la i tornar-la a introduir per poder fer-ho).

boot (D:)			
Nombre	Fecha de modificación	Tipo	Tamaño
overlays	07/05/2021 14:42	Carpeta de archivos	
bcm2708-rpi-b.dtb	03/03/2021 12:40	Archivo DTB	26 KB
bcm2708-rpi-b-plus.dtb	03/03/2021 12:40	Archivo DTB	26 KB
bcm2708-rpi-b-rev1.dtb	03/03/2021 12:40	Archivo DTB	25 KB
bcm2708-rpi-cm.dtb	03/03/2021 12:40	Archivo DTB	25 KB
bcm2708-rpi-zero.dtb	03/03/2021 12:40	Archivo DTB	25 KB
bcm2708-rpi-zero-w.dtb	03/03/2021 12:40	Archivo DTB	26 KB
bcm2709-rpi-2-b.dtb	03/03/2021 12:40	Archivo DTB	27 KB
bcm2710-rpi-2-b.dtb	03/03/2021 12:40	Archivo DTB	27 KB
bcm2710-rpi-3-b.dtb	03/03/2021 12:40	Archivo DTB	28 KB
bcm2710-rpi-3-b-plus.dtb	03/03/2021 12:40	Archivo DTB	29 KB
bcm2710-rpi-cm3.dtb	03/03/2021 12:40	Archivo DTB	27 KB
bcm2711-rpi-4-b.dtb	03/03/2021 12:40	Archivo DTB	48 KB
bcm2711-rpi-400.dtb	30/04/2021 14:01	Archivo DTB	48 KB
bcm2711-rpi-cm4.dtb	03/03/2021 12:40	Archivo DTB	49 KB

- 6è: crear un fitxer de text anomenat ssh.txt, esborrant del nom la terminació .txt (el fitxer ha de romandre buit)

fixup4db.dat	30/04/2021 14:01	Archivo DAT	9 KB
fixup4x.dat	30/04/2021 14:01	Archivo DAT	9 KB
issue.txt	07/05/2021 15:07	Documento de tex...	1 KB
kernel.img	30/04/2021 14:01	Archivo de image...	5.842 KB
kernel7.img	30/04/2021 14:01	Archivo de image...	6.173 KB
kernel7l.img	30/04/2021 14:01	Archivo de image...	6.538 KB
kernel8.img	30/04/2021 14:01	Archivo de image...	7.577 KB
LICENCE.broadcom	05/01/2021 6:30	Archivo BROADCO...	2 KB
start.elf	30/04/2021 14:01	Archivo ELF	2.884 KB
start_cd.elf	30/04/2021 14:01	Archivo ELF	775 KB
start_db.elf	30/04/2021 14:01	Archivo ELF	4.683 KB
start_x.elf	30/04/2021 14:01	Archivo ELF	3.618 KB
start4.elf	30/04/2021 14:01	Archivo ELF	2.177 KB
start4cd.elf	30/04/2021 14:01	Archivo ELF	775 KB
start4db.elf	30/04/2021 14:01	Archivo ELF	3.636 KB
start4x.elf	30/04/2021 14:01	Archivo ELF	2.912 KB
ssh.txt	10/06/2021 15:32	Documento de tex...	0 KB

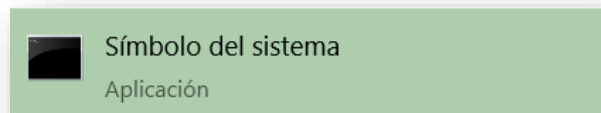
start4db.elf	30/04/2021 14:01	Archivo ELF	3.636 KB
start4x.elf	30/04/2021 14:01	Archivo ELF	2.912 KB
ssh	10/06/2021 15:32	Archivo	0 KB

- 7è: Retirar la memòria sd, i introduir-la a la Raspberry.

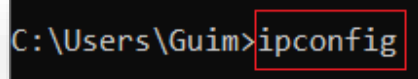
Bootejar la Raspberry via secure Shell (ssh)

Per a realitzar la preparació del sistema de fitxers, hem d'introduir una sèrie de comandes des de la consola de la pròpia Raspberry, per a poder fer-ho el més còmodament possible des de el nostre ordinador, utilitzarem programari per a connectar-nos-hi remotament.

- 1r: Instal·lar el programari necessari:
 - [MobaXterm](#) ([Descarrega](#))
 - [AngryIPScan](#) ([Descarrega](#))
- 2n: localitzar el rang de IP's on es troba la nostra xarxa domestica
 - Obrir el símbol del sistema introduint en el cercador de Windows “cmd”

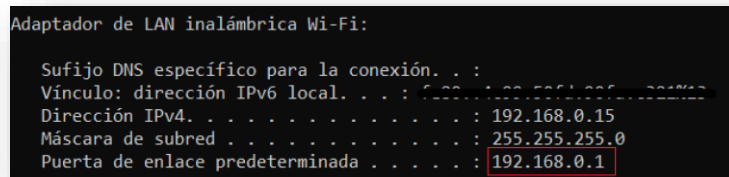


- Introduir la comanda “ipconfig”



```
C:\Users\Guim>ipconfig
```

- Identificar la direcció IP de la porta d'entrada d'entre les dades obtingudes



```
Adaptador de LAN inalámbrica Wi-Fi:

Sufijo DNS específico para la conexión. . . : 
Vínculo: dirección IPv6 local. . . : fe80::100:2001:2001:2001
Dirección IPv4. . . . . : 192.168.0.15
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . . : 192.168.0.1
```

- En aquest cas, el rang de la nostra xarxa interna es 192.168.0.0 – 192.168.0.255
- 3r: inspeccionar els dispositius que hi ha a la nostra xarxa utilitzant [AngryIPScan](#).
 - En el primer camp de “rango de ip” introduïrem el valor més baix del rang de la nostra xarxa

Rango de IP - Angry IP Scanner

Escanear Ir a Comandos Favoritos Herramientas Ayuda

Rango de IP: 192.168.0.0 a Rango de IP

Nombre de equipo: IPt Máscara de rec

IP	Ping	Nombre del equipo	Puertos [3+]
----	------	-------------------	--------------

- En el segon camp de “rango de ip” introduïrem el valor mes alt del rang de la nostra xarxa

Rango de IP - Angry IP Scanner

Escanear Ir a Comandos Favoritos Herramientas Ayuda

Rango de IP: 192.168.0.0 a 192.168.0.255 Rango de IP

Nombre de equipo: IPt Máscara de rec

IP	Ping	Nombre del equipo	Puertos [3+]
----	------	-------------------	--------------

- Realitzar l’escaneig amb les dades introduïdes de la nostra xarxa

Rango de IP - Angry IP Scanner

Escanear Ir a Comandos Favoritos Herramientas Ayuda

Rango de IP: 192.168.0.0 a 192.168.0.255 Rango de IP

Nombre de equipo: IPt Máscara de rec

IP	Ping	Nombre del equipo	Puertos [3+]
----	------	-------------------	--------------

- Identificar dins del camp “nombre del equipo” la nostra Raspberry i extreure’n la seva respectiva IP

Rango de IP - Angry IP Scanner

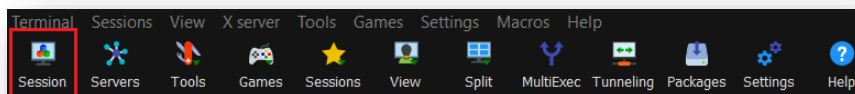
Escanear Ir a Comandos Favoritos Herramientas Ayuda

Rango de IP: 192.168.0.0 a 192.168.0.255 Rango de IP

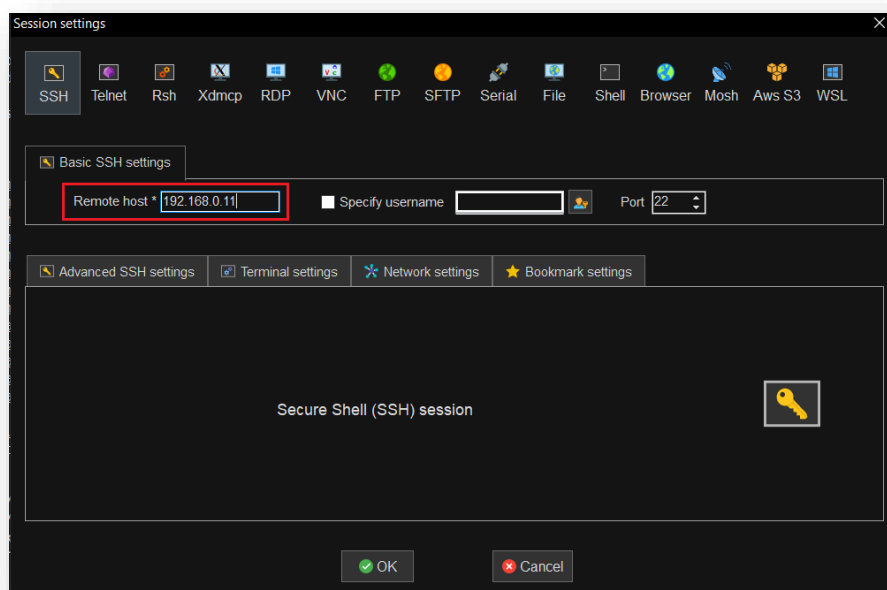
Nombre de equipo: IP Máscara de red Comenzar

IP	Ping	Nombre del equipo	Puertos [3+]
192.168.0.1	0 ms	www.adsl.vf	80,443
192.168.0.2	[n/a]	[n/s]	[n/s]
192.168.0.3	[n/a]	[n/s]	[n/s]
192.168.0.4	[n/a]	[n/s]	[n/s]
192.168.0.5	[n/a]	[n/s]	[n/s]
192.168.0.6	[n/a]	[n/s]	[n/s]
192.168.0.7	[n/a]	[n/s]	[n/s]
192.168.0.8	[n/a]	[n/s]	[n/s]
192.168.0.9	[n/a]	[n/s]	[n/s]
192.168.0.10	26 ms	Healthy Home Coach.local	[n/a]
192.168.0.11	0 ms	RASPBERRYPI	[n/a]
192.168.0.12	[n/a]	[n/s]	[n/s]
192.168.0.13	2 ms	RT-AX58U-41B8.local	80
192.168.0.14	[n/a]	[n/s]	[n/s]
192.168.0.15	0 ms	DESKTOP-LG84140	[n/a]

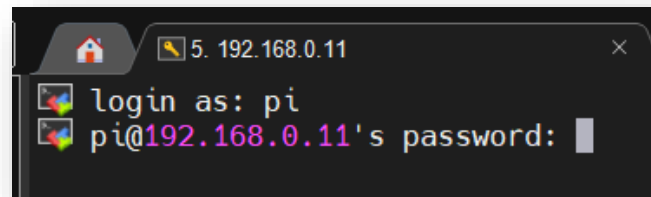
- 4t: establir la connexió amb la Raspberry utilitzant Mobaxtrem
 - crear una sessió ssh



- Introduir al camp "Remote host" la direcció IP corresponent a la Raspberry i escollir el port nº 22



- A la consola que s'ha obert, loguejar-se amb les credencials:
 - User: "pi"
 - Password: "Raspberry"



- Descarregar i instal·lar les actualitzacions disponibles utilitzant les comandes:
 - "Sudo apt-get Update"
 - "Sudo apt-get upgrade"

```
pi@raspberrypi:~ $ sudo apt-get update
Get:1 http://rasbian.raspberrypi.org/rasbian buster InRelease [15.0 kB]
Get:2 http://archive.raspberrypi.org/debian buster InRelease [32.6 kB]
Get:3 http://rasbian.raspberrypi.org/rasbian buster/main armhf Packages [13.0 MB]
Get:4 http://archive.raspberrypi.org/debian buster/main armhf Packages [376 kB]
Fetched 13.4 MB in 8s (1,739 kB/s)
Reading package lists... Done
pi@raspberrypi:~ $ sudo apt-get upgrade
```

Preparar les memòries i el sistema de fitxers

Un cop en funcionament, la Raspberry ja te la capacitat de treballar amb les diferents memòries que hi hagi connectades, ara tant sols cal preparar el sistema de fitxers per a poder accedir a aquestes memòries de forma remota

- 1r: localitzar i identificar les memòries dins dels directoris de la Raspberry.
 - Veure totes les memòries disponibles amb la comanda "lsblk"

```
pi@raspberrypi:~ $ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0 223.6G 0 disk
└─sda1       8:1    0 223.6G 0 part
sdb          8:16    1  57.3G 0 disk
└─sdb1       8:17    1  57.3G 0 part /USB
sdc          8:32    1  57.3G 0 disk
└─sdc1       8:33    1  57.3G 0 part
mmcblk0     179:0    0   29.7G 0 disk
├─mmcblk0p1 179:1    0   256M 0 part /boot
└─mmcblk0p2 179:2    0   29.5G 0 part /
```

(en aquesta imatge veiem les diferents memòries amb les seves respectives particions, en color vermell identifiquem la memòria USB que volem incorpora al sistema, i en color groc ressaltam el nom referent a aquesta memòria dins del directori dev, que es el directori on es troben tots els "device".)

- 2n: formateig de la memòria
 - **ADVERTENCIA: un cop es realitzi aquest 2n punt, es perdran totes les dades que hi hagi guardades a la memòria que estem formatejant!**
 - Utilitzant la comanda “**sudo fdisk /dev/sdc**” s’iniciarà el formateig de la memòria USB que volem introduir al sistema.

```
pi@raspberrypi:~ $ sudo fdisk /dev/sdc

Welcome to fdisk (util-linux 2.33.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help):
```

- Ara caldrà introduir un seguit de comandes per a seleccionar els diferents paràmetres que volguem per a la nostra memoria:
 - “o” (crear una nova etiqueta per a la memoria)

```
Command (m for help): o
Created a new DOS disklabel with disk identifier 0x554b2863.

Command (m for help):
```

- “n” (crear una partició dins de la memoria)

```
Command (m for help): n
Partition type
  p   primary (0 primary, 0 extended, 4 free)
  e   extended (container for logical partitions)
Select (default p):
```

- “p” (indicar que la partició creada sigui de tipus primari)

```
Select (default p): p
Partition number (1-4, default 1):
```

- “1” (indicar que la partició comenci al principi de la memòria, caldrà acceptar el “first sector” i el “last sector” clickant la tecla “enter”)


```
Partition number (1-4, default 1): 1
First sector (2048-120176639, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-120176639, default 120176639):
Created a new partition 1 of type 'Linux' and of size 57.3 GiB.
```

- Per finalitzar el formateig, introduir “w” per tal d’eliminar tot el que hi hagi a la memòria (si es necessari, respondre amb “y” a la pregunta que faci el sistema respecte a la eliminació de les dades)

```
Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

- 3r: crear el sistema de fitxers per a la nostra memòria usb
 - Utilitzarem la comanda “`sudo mkfs -t ext4 /dev/sdc1`” per crear el sistema de fitxers de format tipus ext4.

```
pi@raspberrypi:~$ sudo mkfs -t ext4 /dev/sdc1
mke2fs 1.44.5 (15-Dec-2018)
Creating filesystem with 15021824 4k blocks and 3760128 inodes
Filesystem UUID: 6fc02bb6-15e6-4f64-bc58-dc7578f52396
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624, 11239424

Allocating group tables: done
Writing inode tables: done
Creating journal (65536 blocks): done
Writing superblocks and filesystem accounting information: done
```

- 4t: crear el directori on s’ubicarà definitivament la nostra memòria i modificar els fitxers d’inicialització per a auto muntar la nostra memòria cada cop que s’encengui el dispositiu.
 - Utilitzant la comanda “`sudo mkdir /MemUSB`” crearem el directori (podem comprovar la creació del directori amb la comanda “`ls`”)

```
pi@raspberrypi:/ $ sudo mkdir /MemUSB
pi@raspberrypi:/ $ ls
bin  boot  dev  etc  home  lib  lost+found  media  MemUSB  mnt
```

- Amb la comanda “`ls -la /dev/disk/by-uuid`” aconseguirem una llista dels identificadors únics universals dels nostres dispositius (anotarem la identificació de la nostra memòria USB ja que la necessitarem en el pròxim pas)

```
pi@raspberrypi:~$ ls -la /dev/disk/by-uuid
total 0
drwxr-xr-x 2 root root 140 Jun 13 17:01 .
drwxr-xr-x 7 root root 140 Jun  4 12:17 ..
lrwxrwxrwx 1 root root 10 Jun  4 12:17 1af6eb5d-7220-4b4e-9dfe-870a9b782dcf -> ../../sdb1
lrwxrwxrwx 1 root root 10 Jun  4 12:17 2e571704-b475-4903-8f1f-d84c7383df74 -> ../../sda1
lrwxrwxrwx 1 root root 15 Jun  4 12:17 3857a514-b0f4-49ce-8430-34762068bb6f -> ../../mmcblk0p2
lrwxrwxrwx 1 root root 10 Jun 13 17:01 6fc02bb6-15e6-4f64-bc58-dc7578f52396 -> ../../sdc1
```

- Entrarem en el fitxer /etc/fstab amb la comanda “`sudo nano /etc/fstab`”, i haurem de crear una nova entrada que relacioni el directori que hem creat amb l’identificador de la nostra memoria per a que s’auto munti en aquella ubicació.

```
GNU nano 3.2
proc          /proc          proc          defaults      0          0
PARTUUID=ff1dfb7c-01 /boot          vfat          defaults      0          2
PARTUUID=ff1dfb7c-02 /              ext4          defaults,noatime 0          1
# a swapfile is not a swap partition, no line here
# use dphys-swapfile swap[on|off] for that
```

- Introduïm la nova entrada a la taula “`UUID=6fc02bb6-15e6-4f64-bc58-dc7578f52396 /MemUSB ext4 nofail,auto 0 0`” on cada valor correspon a: UUID / directori / format / opcions / backup / comprovacions. “Nofail” i “auto” ens serviran per impedir que el sistema envii un error quan el dispositiu no es trobi al iniciar, i per muntar el disc automàticament, respectivament.

```
GNU nano 3.2 /etc/fstab
proc          /proc          proc          defaults      0          0
PARTUUID=ff1dfb7c-01 /boot          vfat          defaults      0          2
PARTUUID=ff1dfb7c-02 /              ext4          defaults,noatime 0          1
# a swapfile is not a swap partition, no line here
# use dphys-swapfile swap[on|off] for that
UUID=6fc02bb6-15e6-4f64-bc58-dc7578f52396 /MemUSB ext4 nofail,auto 0 0
```

- Per a sortir de l’editor de text “nano” clicarem “ctrl + x” i respondrem “y” per guardar el fitxer i confirmarem el nom del fitxer amb “enter”

```
Save modified buffer? (Answering "No" will DISCARD changes.)
Y Yes
N No      ^C Cancel
```

- Per comprovar que aquest ultim pas ha estat realitzat correctament, sense haver de reiniciar el sistema, utilitzarem la comanda “`sudo mount -a`” i per poder visualitzar si la nostra memoria sha muntat en el directori especificat correctament o no, hauriem de ser capaços d’identificarla en la llista que obtindrem d’utilitzar la comanda “`df -h`”

```

pi@raspberrypi:/ $ sudo mount -a
pi@raspberrypi:/ $ df -h

```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/root	29G	3.4G	25G	13%	/
devtmpfs	1.8G	0	1.8G	0%	/dev
tmpfs	1.9G	0	1.9G	0%	/dev/shm
tmpfs	1.9G	9.7M	1.9G	1%	/run
tmpfs	5.0M	4.0K	5.0M	1%	/run/lock
tmpfs	1.9G	0	1.9G	0%	/sys/fs/cgroup
/dev/sdb1	57G	58M	54G	1%	/USB
/dev/mmcblk0p1	253M	48M	205M	19%	/boot
tmpfs	383M	0	383M	0%	/run/user/1000
/dev/sdc1	57G	53M	54G	1%	/MemUSB

- 5è: permisos i carpetes accessibles remotament.
 - Utilitzant la comanda “`cd /MemUSB`” ens situarem al directori on hem muntat la memòria

```

pi@raspberrypi:/ $ cd /MemUSB
pi@raspberrypi:/MemUSB $

```

- Crearem, utilitzant la comanda “`sudo mkdir nas`”, un nou directori dins de /MemUSB que serà on es guardin els fitxers quan accedim al sistema remotament.

```

pi@raspberrypi:/MemUSB $ sudo mkdir nas
pi@raspberrypi:/MemUSB $ ls
lost+found  nas

```

- Amb la comanda “`sudo adduser usuari`” afegirem un usuari nou al sistema, que posteriorment vincularem amb el directori creat per a que únicament ell i tinguí accés amb contrasenya
 - Caldrà introduir la contrasenya desitjada
 - Repetir la contrasenya
 - (Opcional) Introduir dades personals de l'usuari
 - Respondre amb “y” a la darrera pregunta per acabar la creació de l'usuari

```

pi@raspberrypi:/MemUSB $ sudo adduser usuari
Adding user `usuari' ...
Adding new group `usuari' (1003) ...
Adding new user `usuari' (1003) with group `usuari' ...
Creating home directory `/home/usuari' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for usuari
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y

```

- Modificarem la propietat del directori amb la comanda “`sudo chown usuari:usuari nas`” i ens assegurarem que tot hagi anat be amb la comanda “`ls -la`” que ens mostrara una llista on i podrem observar que l’usuari “usuari” es el propietari del directori “nas”

```

pi@raspberrypi:/MemUSB $ sudo chown usuari:usuari nas
pi@raspberrypi:/MemUSB $ ls -la
total 28
drwxr-xr-x  4 root    root      4096 Jun 13 20:03 .
drwxr-xr-x 21 root    root      4096 Jun 13 17:26 ..
drwx-----  2 root    root     16384 Jun 13 17:01 lost+found
drwxr-xr-x  2 usuari  usuari    4096 Jun 13 20:03 nas

```

- 6è: Instal·lació de “SAMBA” per a poder compartir el directori per la xarxa local
 - Amb la comanda “`sudo apt-get install samba`” instal·larem el programari que ens permetrà la comunicació amb els dispositius amb sistema operatiu Windows que estiguin a la mateixa xarxa on es situa la Raspberry

```

pi@raspberrypi:/MemUSB $ sudo apt-get install samba
Reading package lists... Done
Building dependency tree
Reading state information... Done
samba is already the newest version (2:4.9.5+dfsg-5+deb10u1+rpi1).
The following package was automatically installed and is no longer required:
  python-colorzero
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.

```

- Accedirem a un fitxer de configuració de samba mitjançant la comanda “`sudo nano /etc/samba/smb.conf`”
 - El primer pas serà inserir una nova entrada al final del fitxer amb aquestes característiques per a poder fer el nostre sistema funcional:


```

[MemUSB]
path = /MemUSB/nas
comment = creació de nas amb usb
browseable = yes

```

read only = no
writable = yes
valid users = usuari

```
printable = yes
guest ok = no
read only = yes
create mask = 0700

# Windows clients look for this share name as a source of downloadable
# printer drivers
[print$]
    comment = Printer Drivers
    path = /var/lib/samba/printers
    browseable = yes
    read only = yes
    guest ok = no
# Uncomment to allow remote administration of Windows print drivers.
# You may need to replace 'lpadmin' with the name of the group your
# admin users are members of.
# Please note that you also need to set appropriate Unix permissions
# to the drivers directory for these users to have write rights in it
; write list = root, @lpadmin

[MemUSB]
    path = /MemUSB/nas
    comment = creacio de nas amb usb
    browseable = yes
    read only = no
    writable = yes
    valid users = usuari
```

- El segon pas serà comentar (afegir “#” al davant) un conjunt de línies de l’apartat “share definitions” per tal de no causar problemes d’inconsistència.

```
===== Share Definitions =====

##[homes]
##    comment = Home Directories
##    browseable = no

# By default, the home directories are exported read-only. Change the
# next parameter to 'no' if you want to be able to write to them.
##    read only = yes

# File creation mask is set to 0700 for security reasons. If you want to
# create files with group=rw permissions, set next parameter to 0775.
##    create mask = 0700

# Directory creation mask is set to 0700 for security reasons. If you want to
# create dirs. with group=rw permissions, set next parameter to 0775.
##    directory mask = 0700

# By default, \\server\username shares can be connected to by anyone
# with access to the samba server.
# The following parameter makes sure that only "username" can connect
# to \\server\username
# This might need tweaking when using external authentication schemes
##    valid users = %S

# Un-comment the following and create the netlogon directory for Domain Logons
# (you need to configure Samba to act as a domain controller too.)
;[netlogon]
;    comment = Network Logon Service
;    path = /home/samba/netlogon
;    guest ok = yes
;    read only = yes
```

- Per tancar el fitxer clicarem “ctrl + x” i respondrem “y” per guardar el fitxer i confirmarem el nom del fitxer amb “enter”
- Afegirem l’usuari creat en els punts anteriors a SAMBA mitjançant la comanda “`sudo smbpasswd -a usuari`”
 - Haurem d’introduir novament la contrasenya abans creada

```
pi@raspberrypi:/MemUSB $ sudo smbpasswd -a usuari
New SMB password:
Retype new SMB password:
Added user usuari.
```

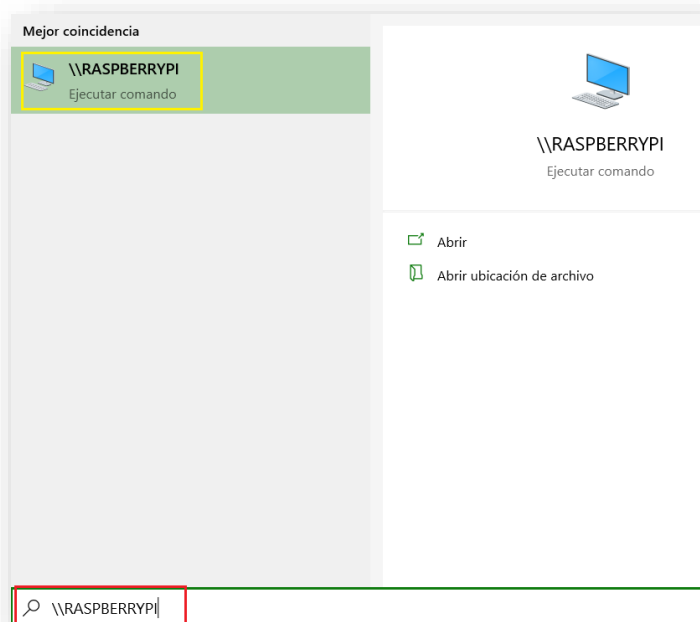
- Un cop realitzats totes les modificacions únicament caldrà reiniciar el servei de SAMBA amb la comanda “`sudo service smb restart`”

```
pi@raspberrypi:/MemUSB $ sudo service smb restart
pi@raspberrypi:/MemUSB $
```

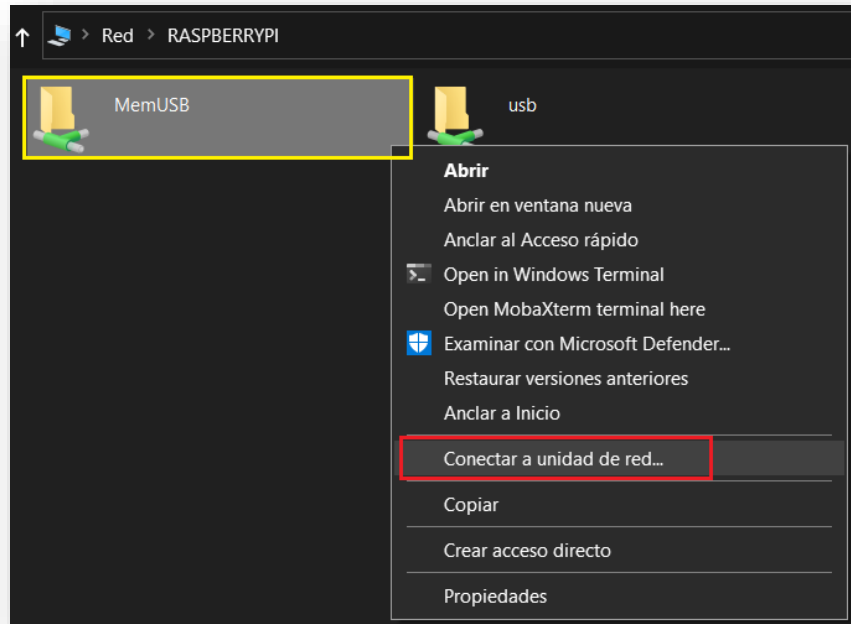
Accés remot

Després d’haver realitzat totes les accions dels punts anteriors, tant sols falta accedir a la carpeta des de el nostre sistema Windows, introduir l’usuari i la contrasenya i ja podrem començar a guardar fitxers en remot.

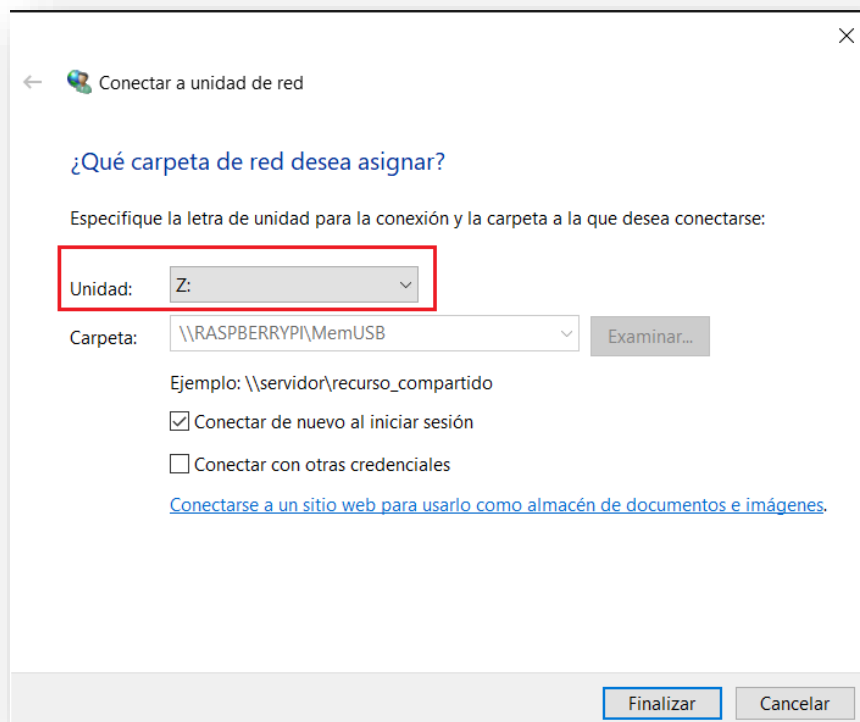
- 1r: connectar-se a la Raspberry mitjançant el cercador d’arxius de Windows utilitzant la comanda “`\\RASPBERRYPI`”



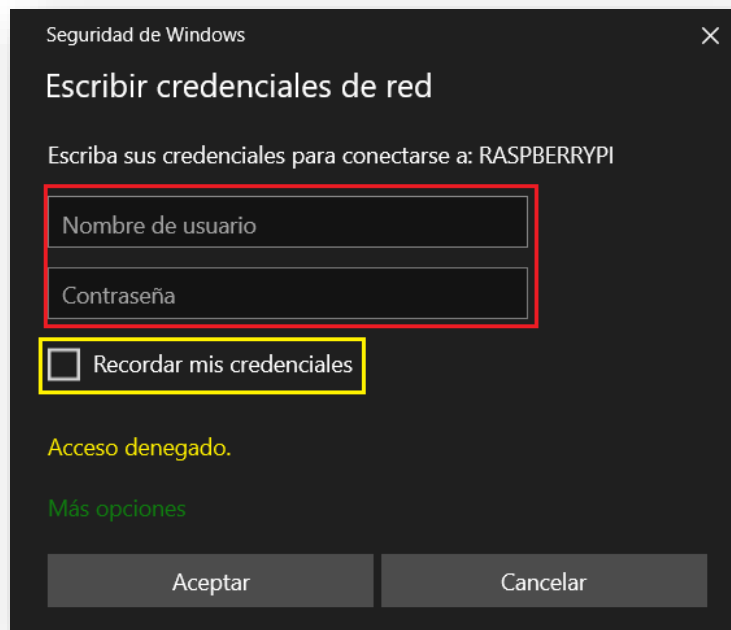
- 2n: Creació de la unitat de xarxa per a assignar una lletra de disc dur a la carpeta en remot.
 - Fent clic dret al ratolí sobre l'Icona de la carpeta compartida s'obrirà un desplegable, hem d'escollir la opció "conectar a unidad de red..."



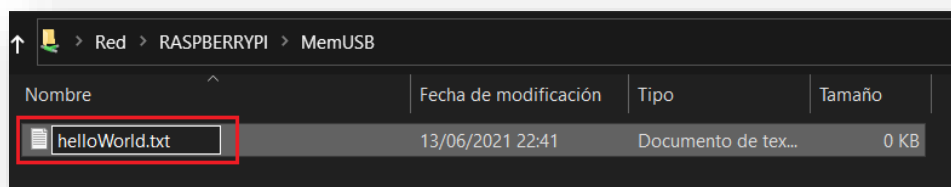
- Assignar la lletra de disc desitjada i finalitzar.



- Introduir les credencials que hem creat en els passos anteriors i decidir si volem que es recordin o si volem introduir-les cada vegada que entrem a la carpeta.



- Un cop dins de la carpeta, podem crear un fitxer de text, i comprovar des de mobaxtrem si s'ha creat correctament.



```
pi@raspberrypi:/MemUSB $ cd nas
pi@raspberrypi:/MemUSB/nas $ ls
helloWorld.txt
```

- Un cop realitzada la darrera comprovació, el sistema NAS queda totalment funcional
- Per a inserir noves unitats de disc, cal seguir el mateix procediment.

