


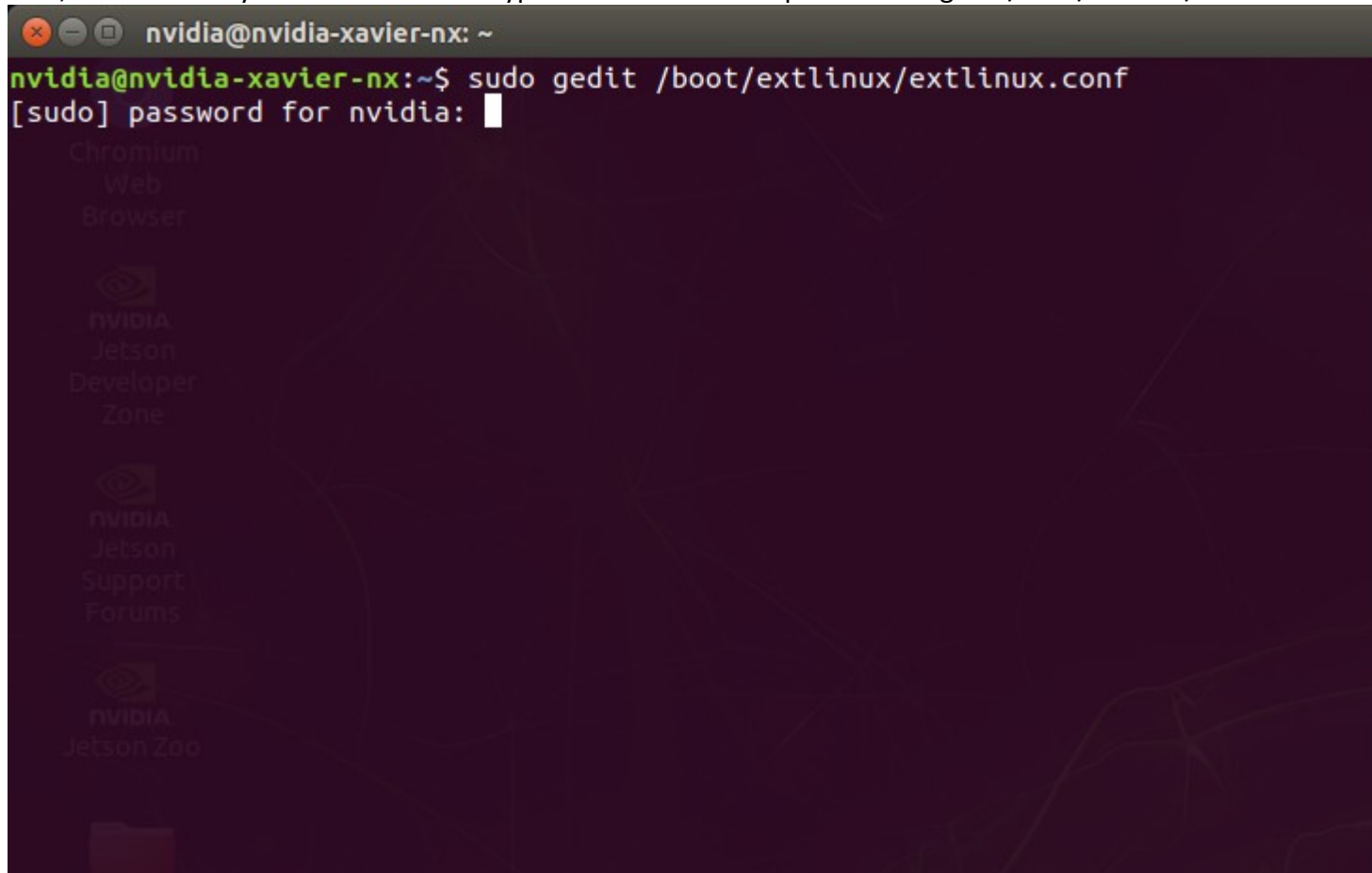
1-) Open a terminal on your Jetson. Then, type "df -h"

As you can see that our SSD connected on "/" (as Filesystem Root)

```
nvidia@nvidia-xavier-nx: ~  
nvidia@nvidia-xavier-nx:~$ df -h  
Filesystem      Size  Used Avail Use% Mounted on  
/dev/nvme0n1p1  229G   12G  206G   6% /  
devtmpfs        3.5G     0  3.5G   0% /dev  
tmpfs           3.8G   40K  3.8G   1% /dev/shm  
tmpfs           3.8G   21M  3.8G   1% /run  
tmpfs           5.0M   4.0K  5.0M   1% /run/lock  
tmpfs           3.8G     0  3.8G   0% /sys/fs/cgroup  
tmpfs           777M  148K  777M   1% /run/user/1000  
/dev/mmcblk0p1   14G   12G   1.8G  87% /media/nvidia/5169909d-5f5f-4b5e-963d-dce9260d93d7  
nvidia@nvidia-xavier-nx:~$  
nvidia@nvidia-xavier-nx:~$
```

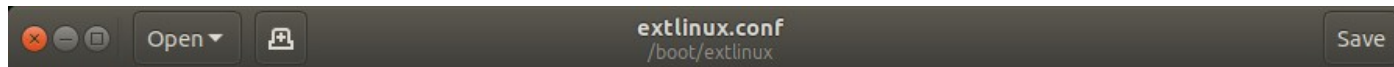


2-) At this point we need to change two extlinux.conf files which located on boot/extlinux folders both eMMC & SSD. First, we will modify the file at the SSD. Type this command to open it "sudo gedit /boot/extlinux/extlinux.conf"



```
nvidia@nvidia-xavier-nx: ~  
nvidia@nvidia-xavier-nx:~$ sudo gedit /boot/extlinux/extlinux.conf  
[sudo] password for nvidia: 
```

The image shows a terminal window with a dark background and a light-colored border. The window title is "nvidia@nvidia-xavier-nx: ~". The command "sudo gedit /boot/extlinux/extlinux.conf" has been entered, and the prompt "[sudo] password for nvidia:" is displayed with a cursor. On the left side of the terminal, there is a vertical sidebar with several icons and links: "Chromium Web Browser", "nvidia Jetson Developer Zone", "nvidia Jetson Support Forums", and "nvidia Jetson Zoo".



```
TIMEOUT 30
DEFAULT primary

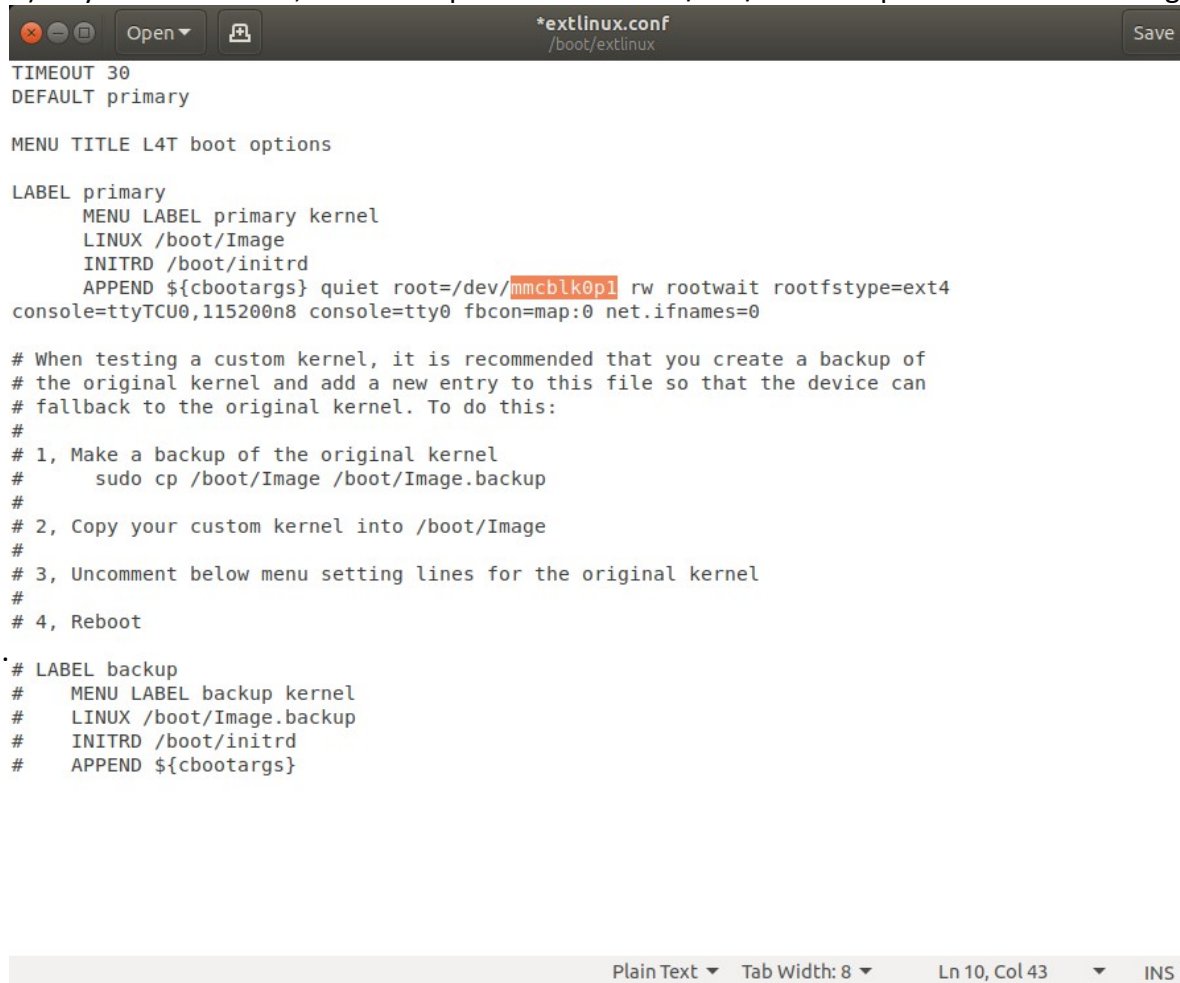
MENU TITLE L4T boot options

LABEL primary
    MENU LABEL primary kernel
    LINUX /boot/Image
    INITRD /boot/initrd
    APPEND ${cbootargs} quiet root=/dev/nvme0n1p1 rw rootwait rootfstype=ext4
console=ttyTCU0,115200n8 console=tty0 fbcon=map:0 net.ifnames=0

# When testing a custom kernel, it is recommended that you create a backup of
# the original kernel and add a new entry to this file so that the device can
# fallback to the original kernel. To do this:
#
# 1, Make a backup of the original kernel
#     sudo cp /boot/Image /boot/Image.backup
#
# 2, Copy your custom kernel into /boot/Image
#
# 3, Uncomment below menu setting lines for the original kernel
#
# 4, Reboot

# LABEL backup
#     MENU LABEL backup kernel
#     LINUX /boot/Image.backup
#     INITRD /boot/initrd
#     APPEND ${cbootargs}
```

3-) As you can see that, the "root" parameter set as `"/dev/nvme0n1p1"`. You need to change it as `"/dev/mmcblk0p1"`



```
*extlinux.conf
/boot/extlinux

TIMEOUT 30
DEFAULT primary

MENU TITLE L4T boot options

LABEL primary
    MENU LABEL primary kernel
    LINUX /boot/Image
    INITRD /boot/initrd
    APPEND ${cbootargs} quiet root=/dev/mmcblk0p1 rw rootwait rootfstype=ext4
console=ttyTCU0,115200n8 console=tty0 fbcon=map:0 net.ifnames=0

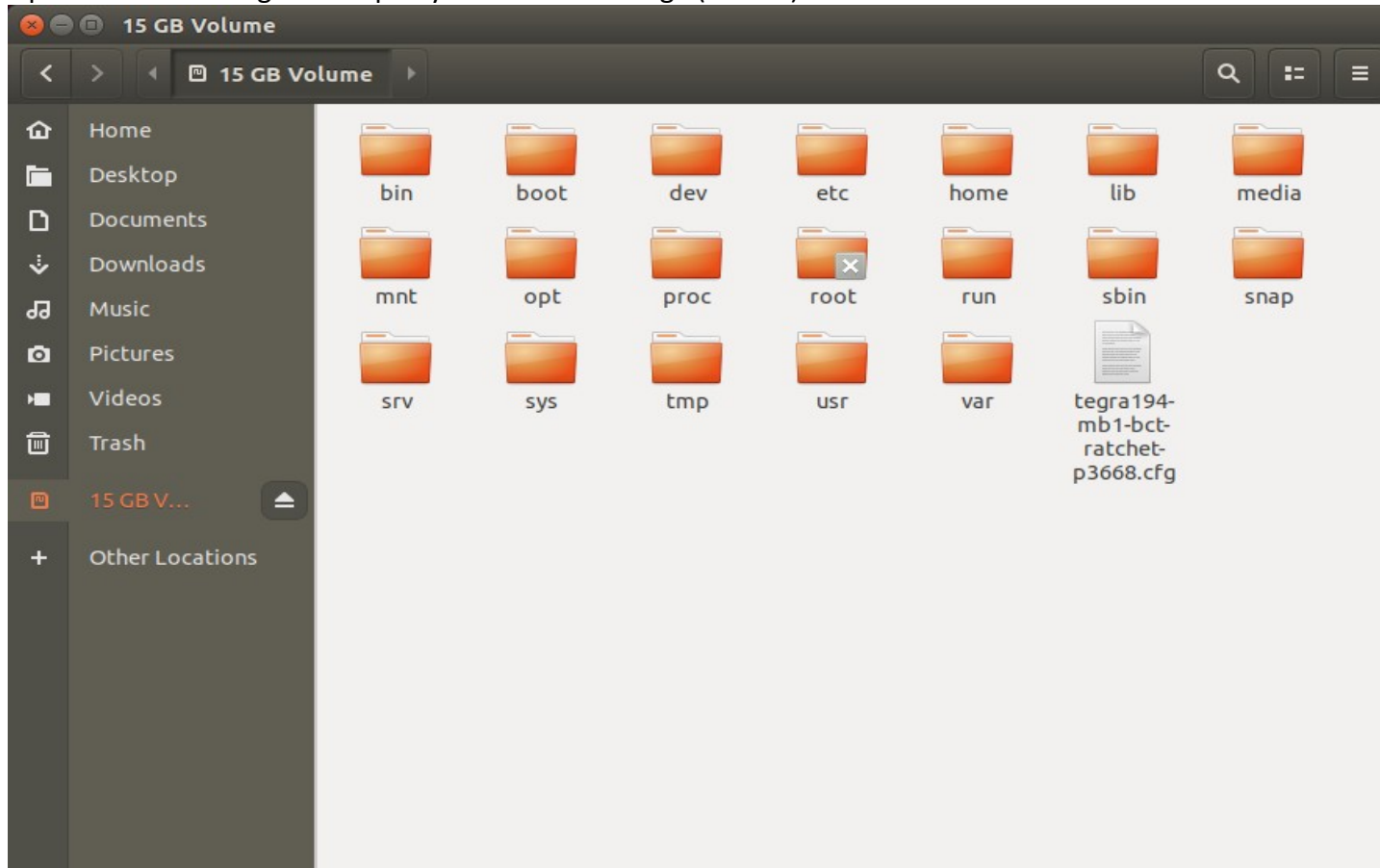
# When testing a custom kernel, it is recommended that you create a backup of
# the original kernel and add a new entry to this file so that the device can
# fallback to the original kernel. To do this:
#
# 1, Make a backup of the original kernel
#     sudo cp /boot/Image /boot/Image.backup
#
# 2, Copy your custom kernel into /boot/Image
#
# 3, Uncomment below menu setting lines for the original kernel
#
# 4, Reboot

# LABEL backup
#     MENU LABEL backup kernel
#     LINUX /boot/Image.backup
#     INITRD /boot/initrd
#     APPEND ${cbootargs}
```

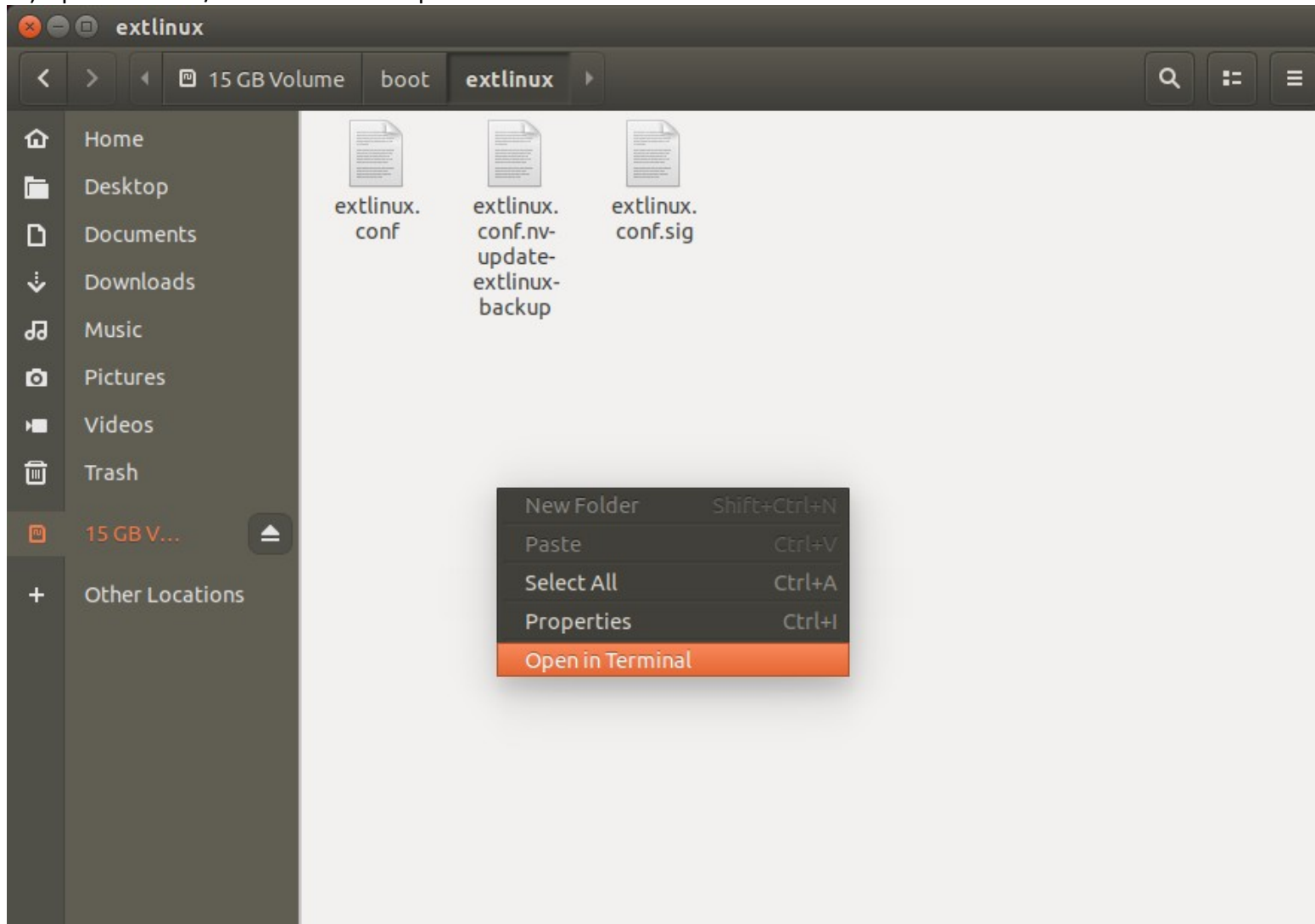
Plain Text ▾ Tab Width: 8 ▾ Ln 10, Col 43 ▾ INS

4-)Then, save it and close.

Open the File MANager and open your internal storage (eMMC).



5-) Open the boot/extlinux folder & open a terminal from here.



6-) Type this command below to modify the second extlinux.conf file:

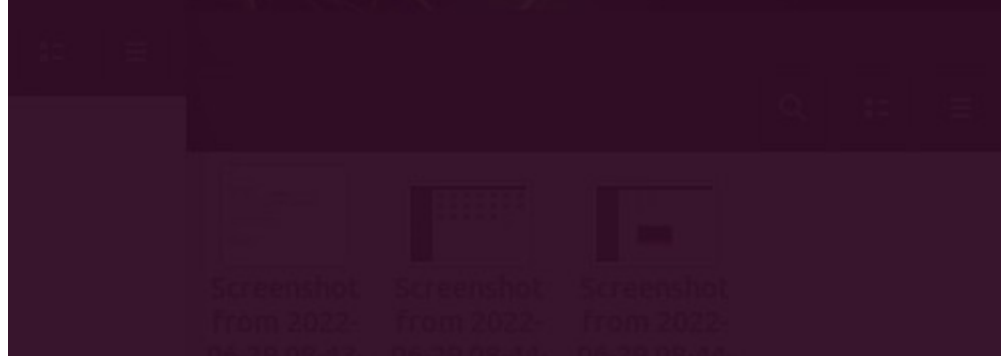
"sudo gedit extlinux.conf"

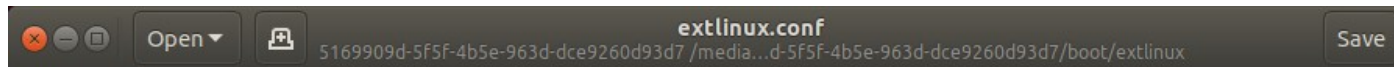
nvidia@nvidia-xavier-nx: /media/nvidia/5169909d-5f5f-4b5e-963d-dce9260d93d7/boot/extlinu

nvidia@nvidia-xavier-nx:/media/nvidia/5169909d-5f5f-4b5e-963d-dce9260d93d7/boot/

extlinux\$ sudo gedit extlinux.conf

[sudo] password for nvidia:





```
TIMEOUT 30
DEFAULT primary

MENU TITLE L4T boot options

LABEL primary
    MENU LABEL primary kernel
    LINUX /boot/Image
    INITRD /boot/initrd
    APPEND ${cbootargs} quiet root=/dev/nvme0n1p1 rw rootwait rootfstype=ext4
console=ttyTCU0,115200n8 console=tty0 fbcon=map:0 net.ifnames=0

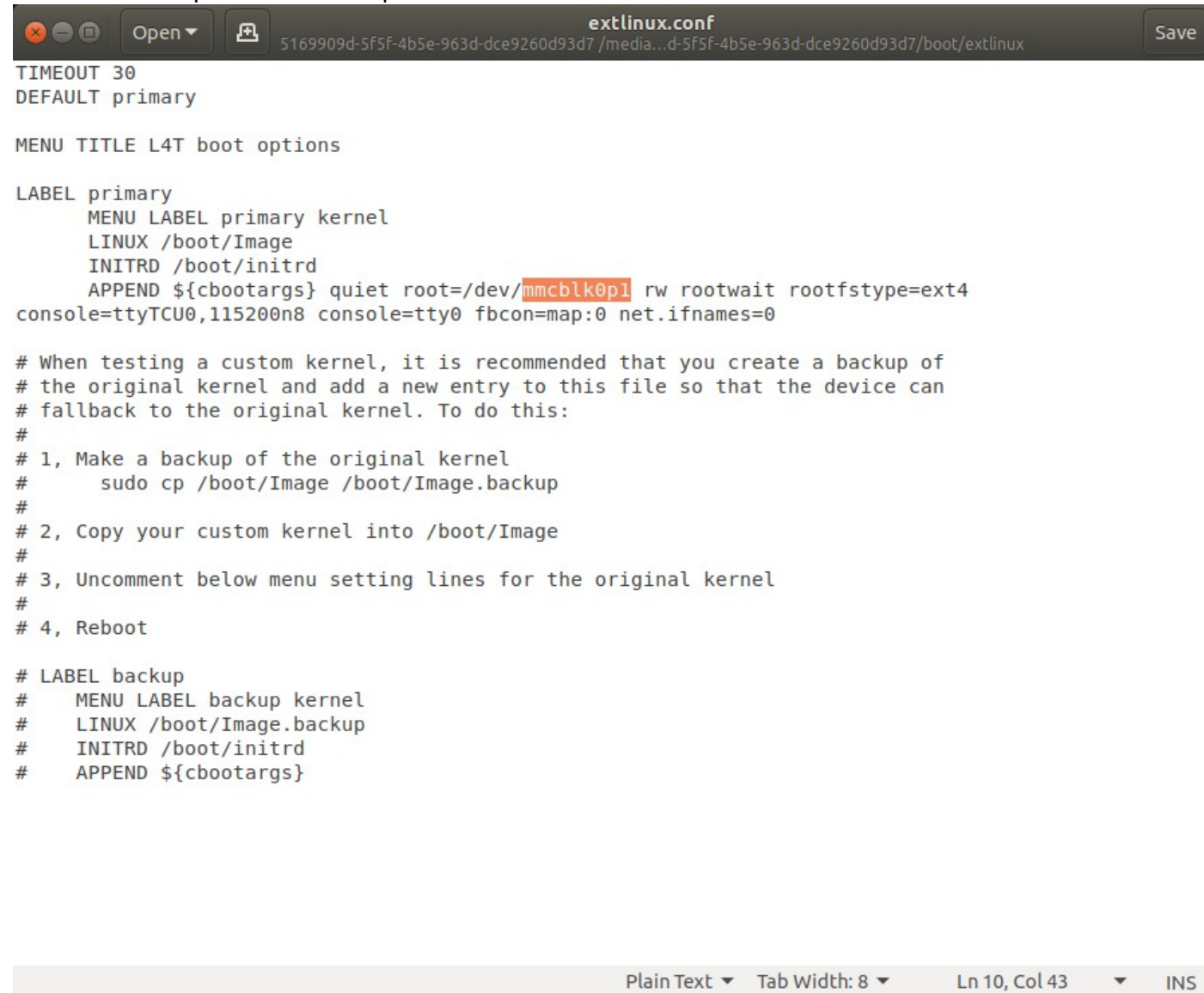
# When testing a custom kernel, it is recommended that you create a backup of
# the original kernel and add a new entry to this file so that the device can
# fallback to the original kernel. To do this:
#
# 1, Make a backup of the original kernel
#     sudo cp /boot/Image /boot/Image.backup
#
# 2, Copy your custom kernel into /boot/Image
#
# 3, Uncomment below menu setting lines for the original kernel
#
# 4, Reboot

# LABEL backup
#     MENU LABEL backup kernel
#     LINUX /boot/Image.backup
#     INITRD /boot/initrd
#     APPEND ${cbootargs}
```



7-) You need to do that the same update on this file as well.

From "nvme0n1p1" to "mmcblk0p1"



```
TIMEOUT 30
DEFAULT primary

MENU TITLE L4T boot options

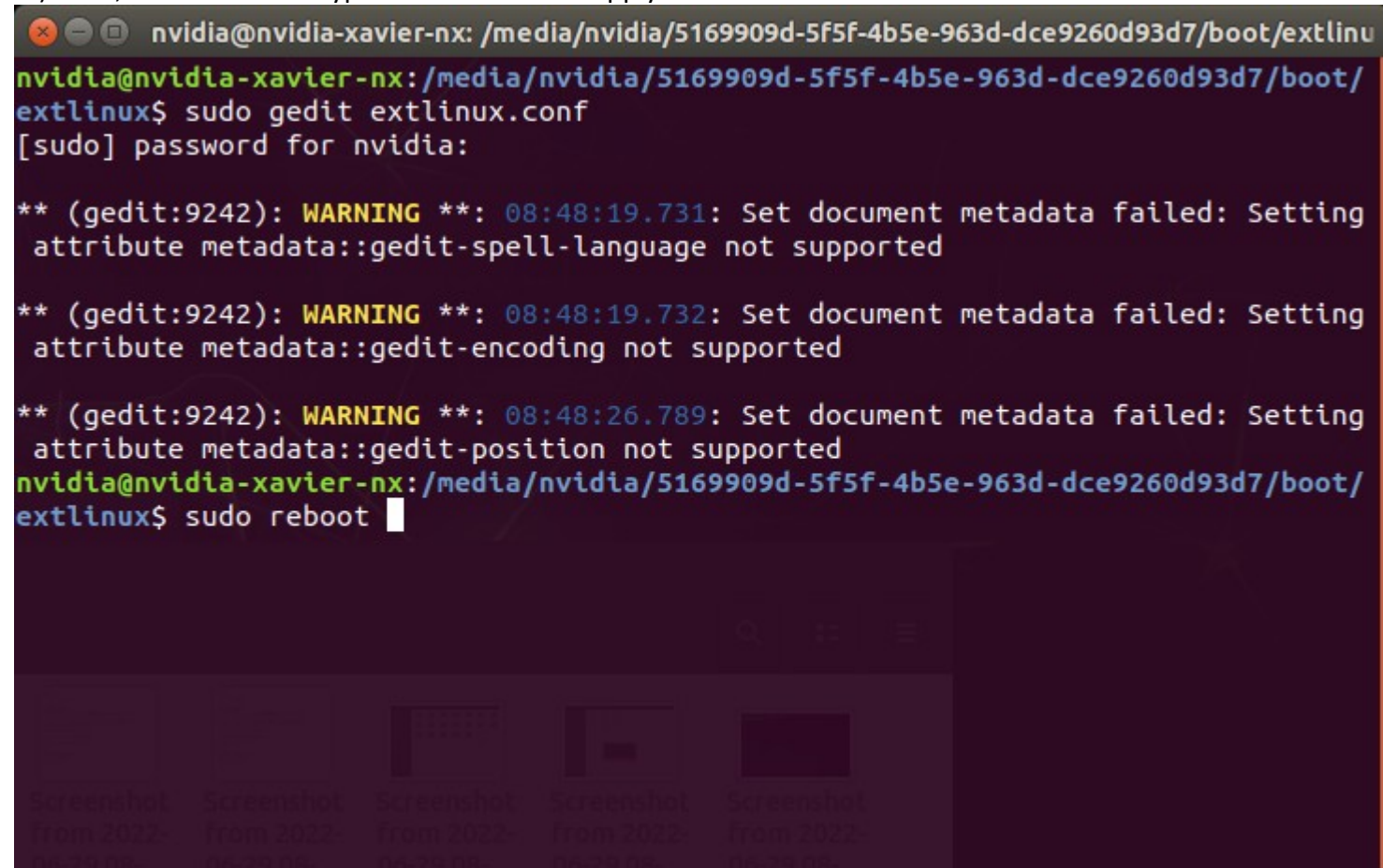
LABEL primary
    MENU LABEL primary kernel
    LINUX /boot/Image
    INITRD /boot/initrd
    APPEND ${cbootargs} quiet root=/dev/mmcblk0p1 rw rootwait rootfstype=ext4
console=ttyTCU0,115200n8 console=tty0 fbcon=map:0 net.ifnames=0

# When testing a custom kernel, it is recommended that you create a backup of
# the original kernel and add a new entry to this file so that the device can
# fallback to the original kernel. To do this:
#
# 1, Make a backup of the original kernel
#     sudo cp /boot/Image /boot/Image.backup
#
# 2, Copy your custom kernel into /boot/Image
#
# 3, Uncomment below menu setting lines for the original kernel
#
# 4, Reboot

# LABEL backup
#     MENU LABEL backup kernel
#     LINUX /boot/Image.backup
#     INITRD /boot/initrd
#     APPEND ${cbootargs}
```

Plain Text ▾ Tab Width: 8 ▾ Ln 10, Col 43 ▾ INS

8-) Then, save it & close. Type "sudo reboot" to apply the differences



```
nvidia@nvidia-xavier-nx: /media/nvidia/5169909d-5f5f-4b5e-963d-dce9260d93d7/boot/extlinu
nvidia@nvidia-xavier-nx:/media/nvidia/5169909d-5f5f-4b5e-963d-dce9260d93d7/boot/
extlinux$ sudo gedit extlinux.conf
[sudo] password for nvidia:

** (gedit:9242): WARNING **: 08:48:19.731: Set document metadata failed: Setting
attribute metadata::gedit-spell-language not supported

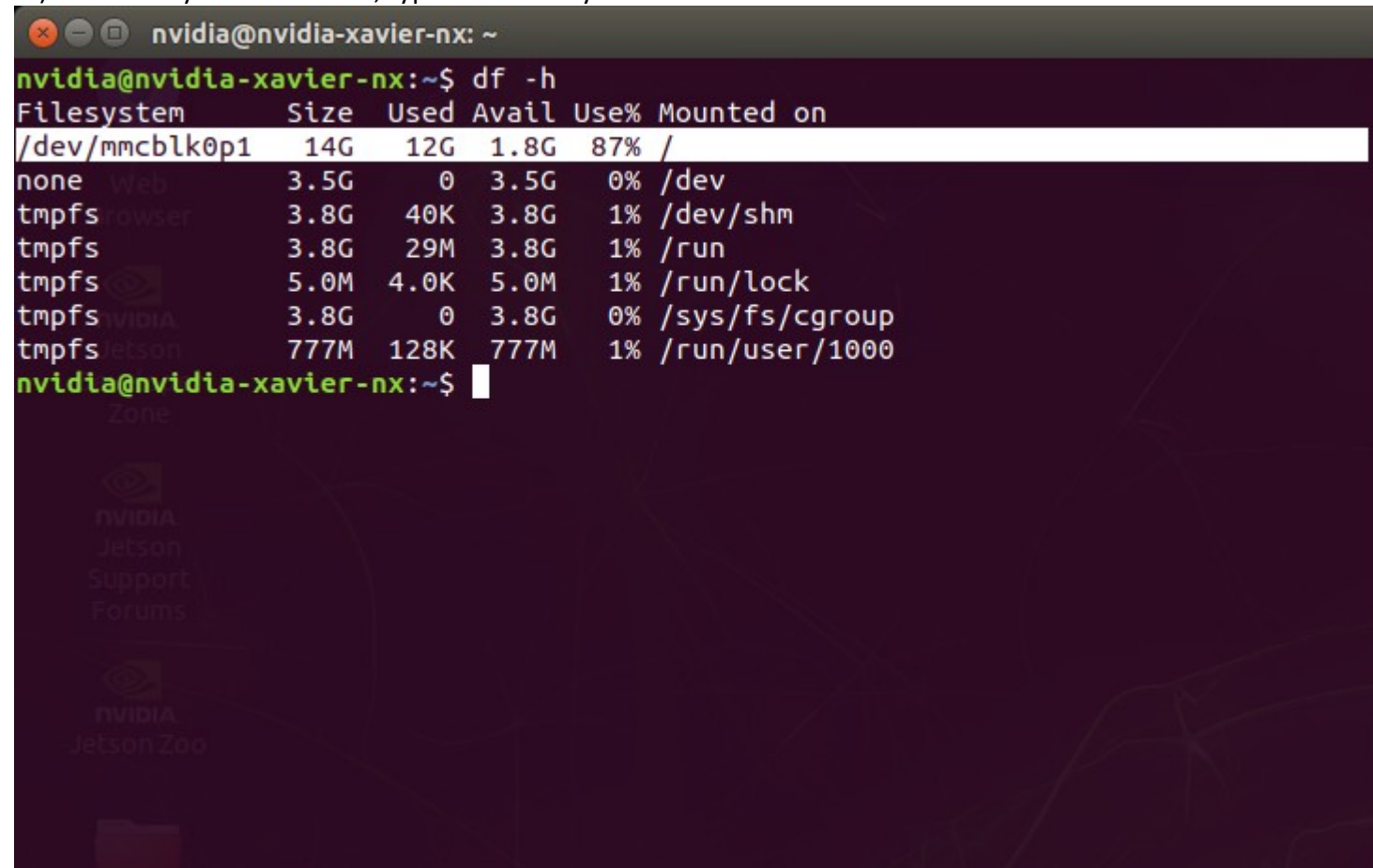
** (gedit:9242): WARNING **: 08:48:19.732: Set document metadata failed: Setting
attribute metadata::gedit-encoding not supported

** (gedit:9242): WARNING **: 08:48:26.789: Set document metadata failed: Setting
attribute metadata::gedit-position not supported
nvidia@nvidia-xavier-nx:/media/nvidia/5169909d-5f5f-4b5e-963d-dce9260d93d7/boot/
extlinux$ sudo reboot
```

The image shows a terminal window with a dark background. The prompt is 'nvidia@nvidia-xavier-nx: /media/nvidia/5169909d-5f5f-4b5e-963d-dce9260d93d7/boot/extlinu'. The user enters 'nvidia@nvidia-xavier-nx:/media/nvidia/5169909d-5f5f-4b5e-963d-dce9260d93d7/boot/extlinux\$ sudo gedit extlinux.conf'. A password prompt '[sudo] password for nvidia:' is shown. Three warning messages from gedit are displayed: 'WARNING \*\*: 08:48:19.731: Set document metadata failed: Setting attribute metadata::gedit-spell-language not supported', 'WARNING \*\*: 08:48:19.732: Set document metadata failed: Setting attribute metadata::gedit-encoding not supported', and 'WARNING \*\*: 08:48:26.789: Set document metadata failed: Setting attribute metadata::gedit-position not supported'. Finally, the user enters 'nvidia@nvidia-xavier-nx:/media/nvidia/5169909d-5f5f-4b5e-963d-dce9260d93d7/boot/extlinux\$ sudo reboot' and the prompt returns. Below the terminal window, a dock is visible with five 'Screenshot from 2022-06-29 08:48:26' icons.

9-) After the system rebooted, type "df -h". As you can see the eMMC connected on root.

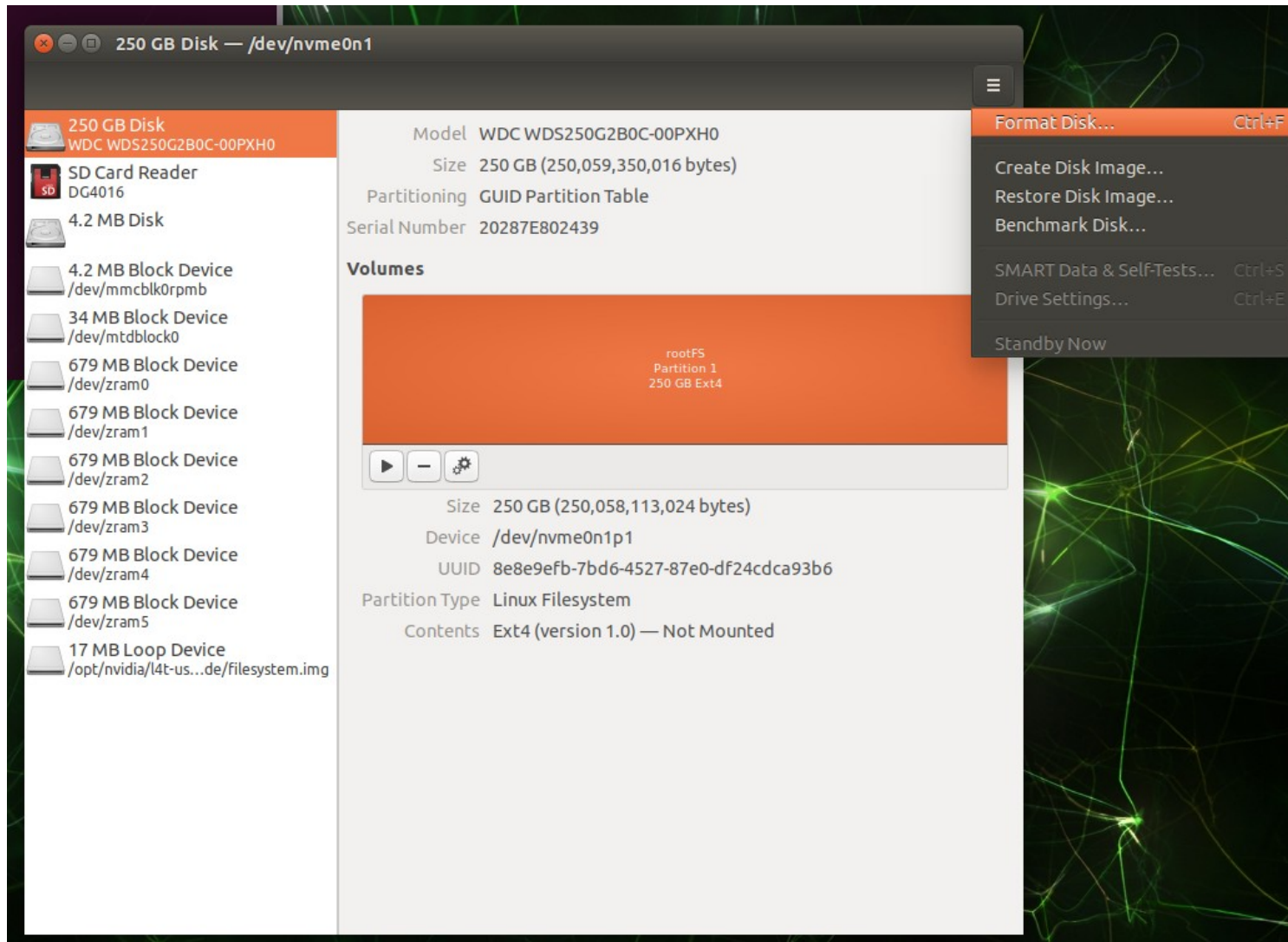
```
nvidia@nvidia-xavier-nx: ~  
nvidia@nvidia-xavier-nx:~$ df -h  
Filesystem      Size  Used Avail Use% Mounted on  
/dev/mmcblk0p1  14G   12G   1.8G  87% /  
none            3.5G    0    3.5G   0% /dev  
tmpfs           3.8G   40K   3.8G   1% /dev/shm  
tmpfs           3.8G   29M   3.8G   1% /run  
tmpfs           5.0M   4.0K   5.0M   1% /run/lock  
tmpfs           3.8G    0    3.8G   0% /sys/fs/cgroup  
tmpfs           777M  128K   777M   1% /run/user/1000  
nvidia@nvidia-xavier-nx:~$
```

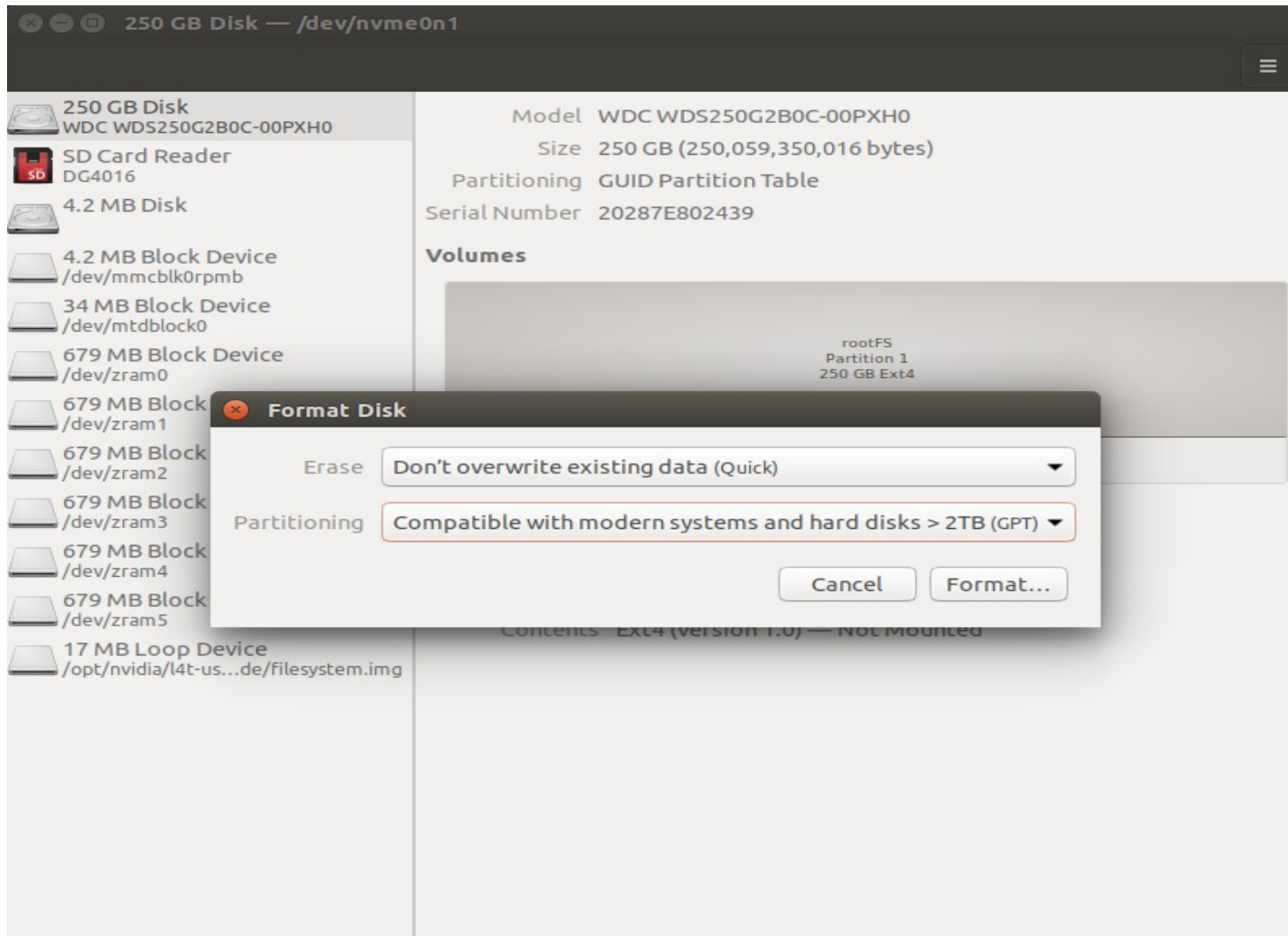
The image shows a terminal window on an NVIDIA Jetson device. The terminal output of the 'df -h' command is as follows:

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/mmcblk0p1	14G	12G	1.8G	87%	/
none	3.5G	0	3.5G	0%	/dev
tmpfs	3.8G	40K	3.8G	1%	/dev/shm
tmpfs	3.8G	29M	3.8G	1%	/run
tmpfs	5.0M	4.0K	5.0M	1%	/run/lock
tmpfs	3.8G	0	3.8G	0%	/sys/fs/cgroup
tmpfs	777M	128K	777M	1%	/run/user/1000

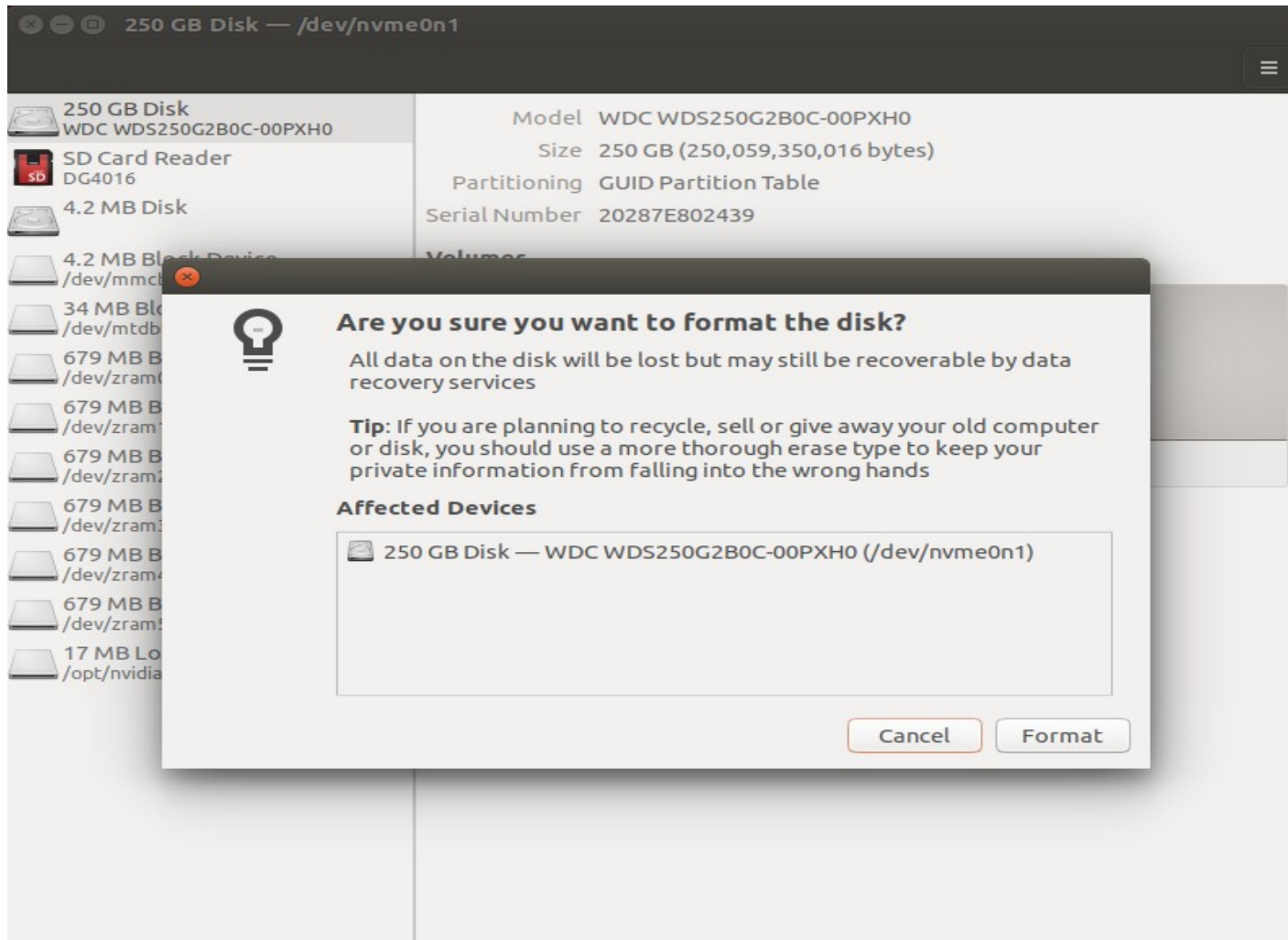
The terminal window is titled 'nvidia@nvidia-xavier-nx: ~'. Below the terminal output, the desktop background is visible, featuring a dark purple theme with a faint geometric pattern. In the bottom-left corner, there are two links: 'NVIDIA Jetson Support Forums' and 'NVIDIA Jetson Zoo', each with the NVIDIA logo and a magnifying glass icon.

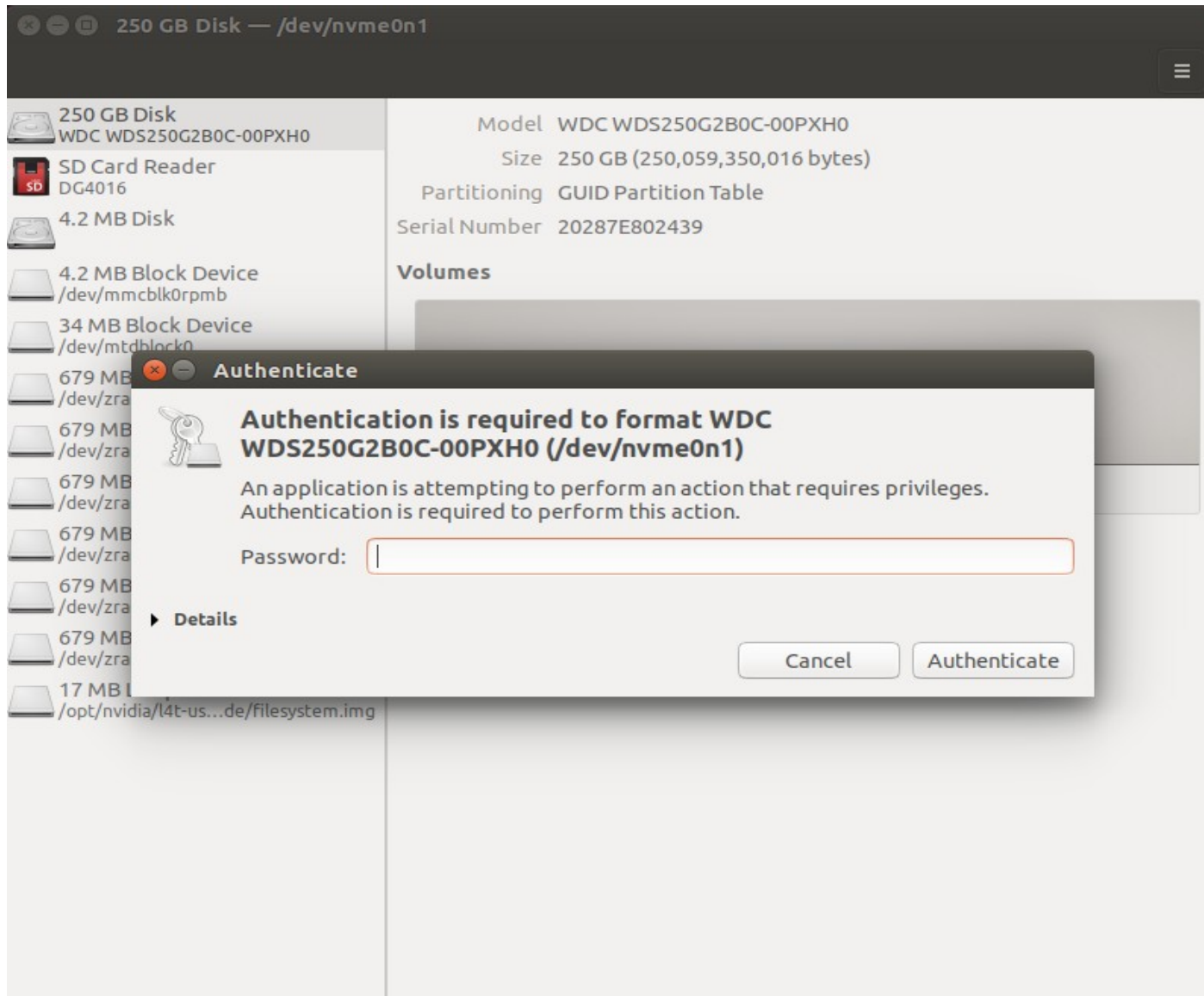
10-) Open the "Disks" application (or type "gnome-disks" from terminal).  
Select the SSD drive & format it

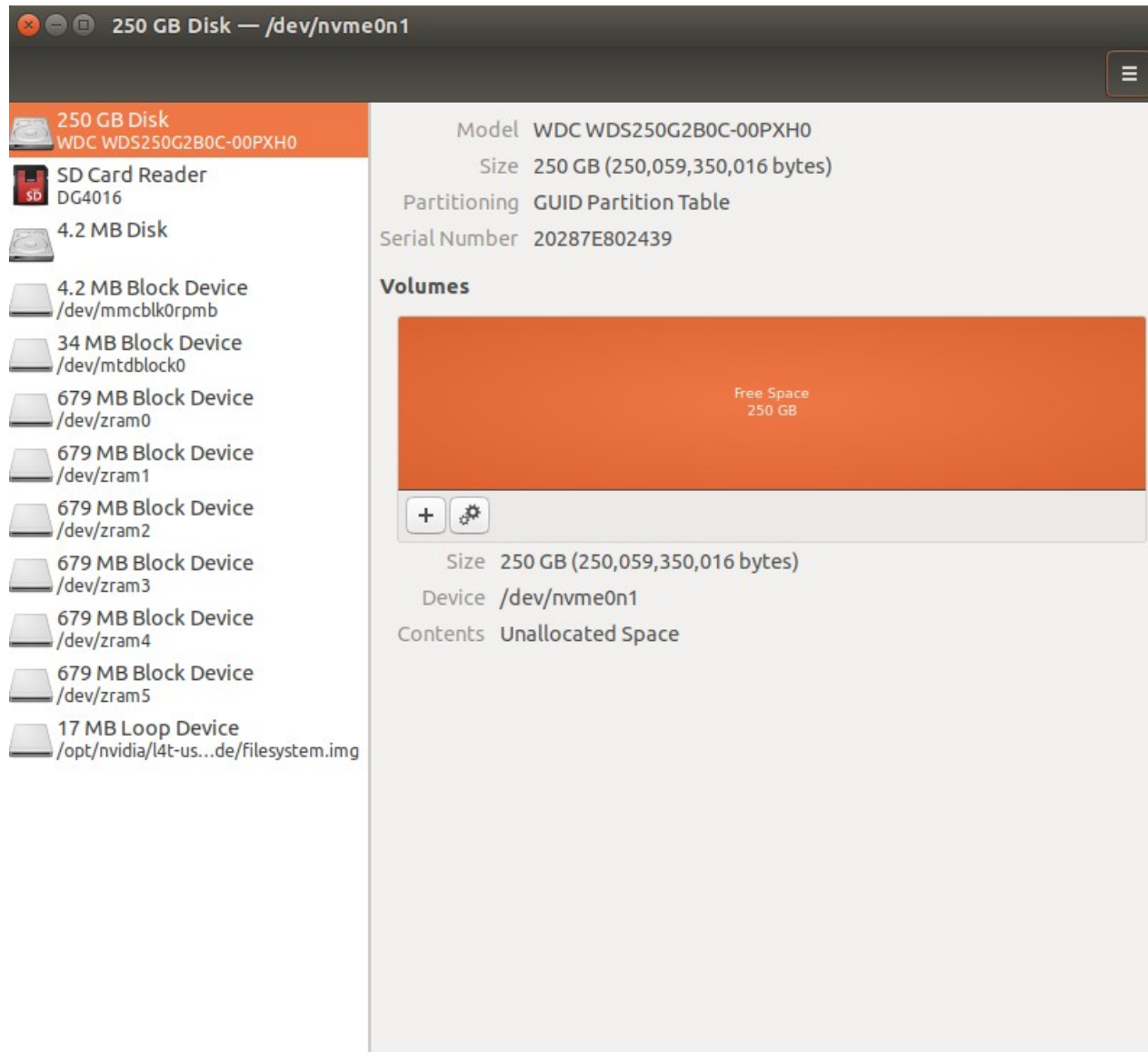















11-) Afterwards, format it as ext4.





250 GB Disk — /dev/nvme0n1


 250 GB Disk  
WDC WDS250G2B0C-00PXH0


 SD Card Reader  
DG4016


 4.2 MB Disk


 4.2 MB Block Device  
/dev/mmcblk0rpm


 34 MB Block Device  
/dev/mtdblock0


 679 MB Block Device  
/dev/zram0


 679 MB Block Device  
/dev/zram1

 679 MB Block Device  
/dev/zram2

 679 MB Block Device  
/dev/zram3

 679 MB Block Device  
/dev/zram4

 679 MB Block Device  
/dev/zram5

 17 MB Loop Device  
/opt/nvidia/l4t-us...de/filesystem.img

Model WDC WDS250G2B0C-00PXH0



Size 250 GB (250,059,350,016 bytes)

Partitioning GUID Partition Table

Serial Number 20287E802439

Volumes

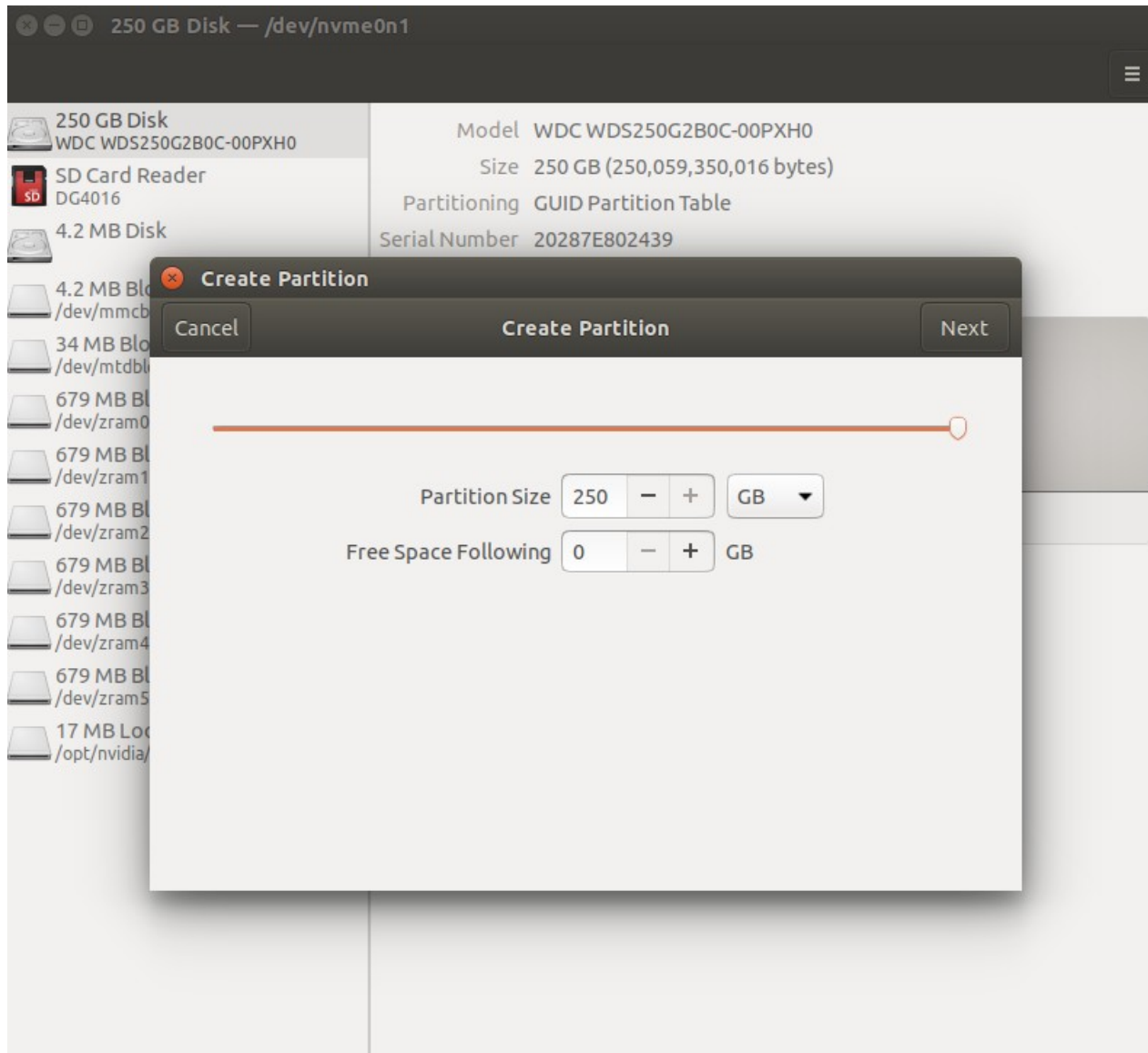
Free Space  
250 GB

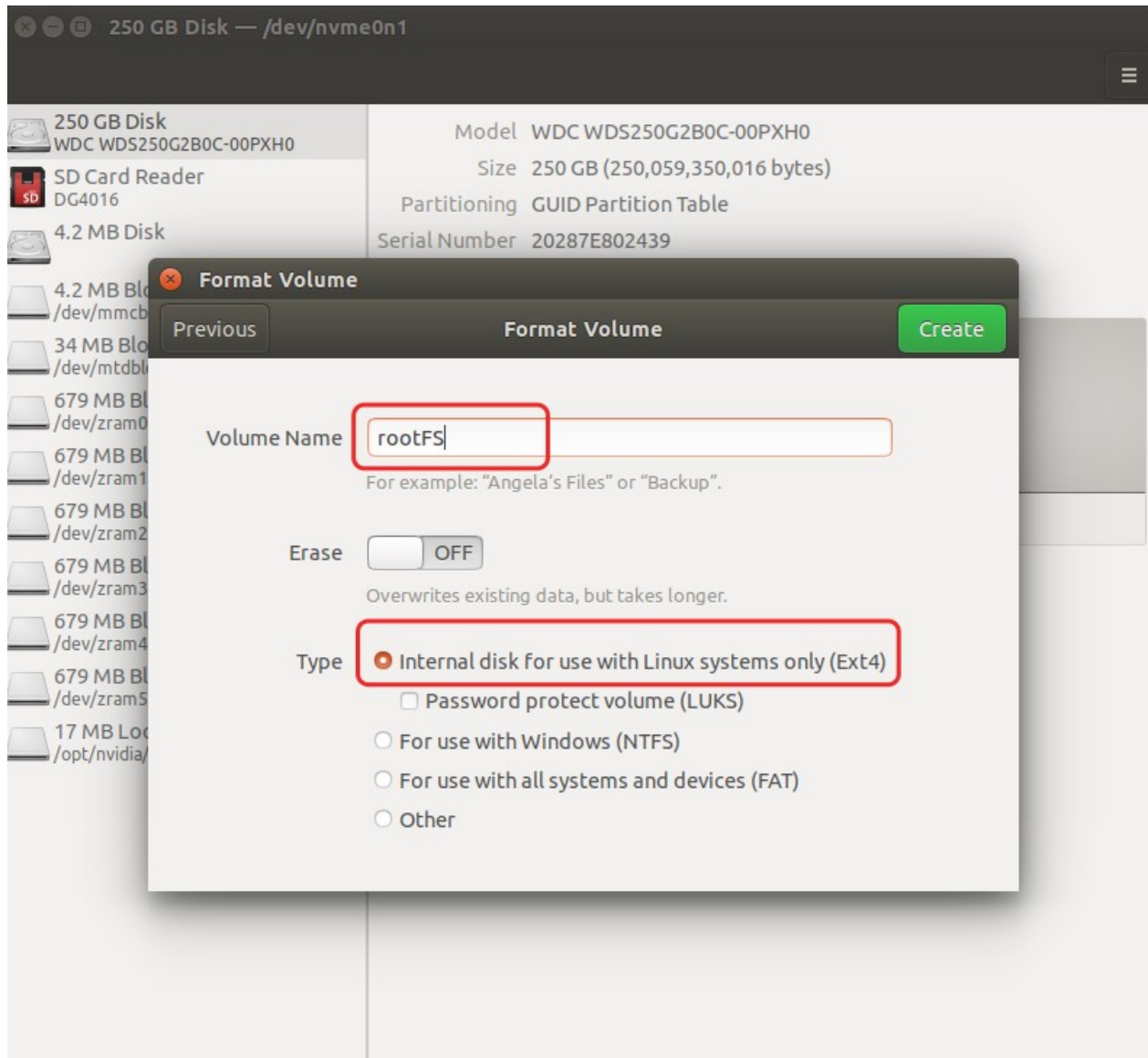


Create partition in unallocated space (250,059,350,016 bytes)

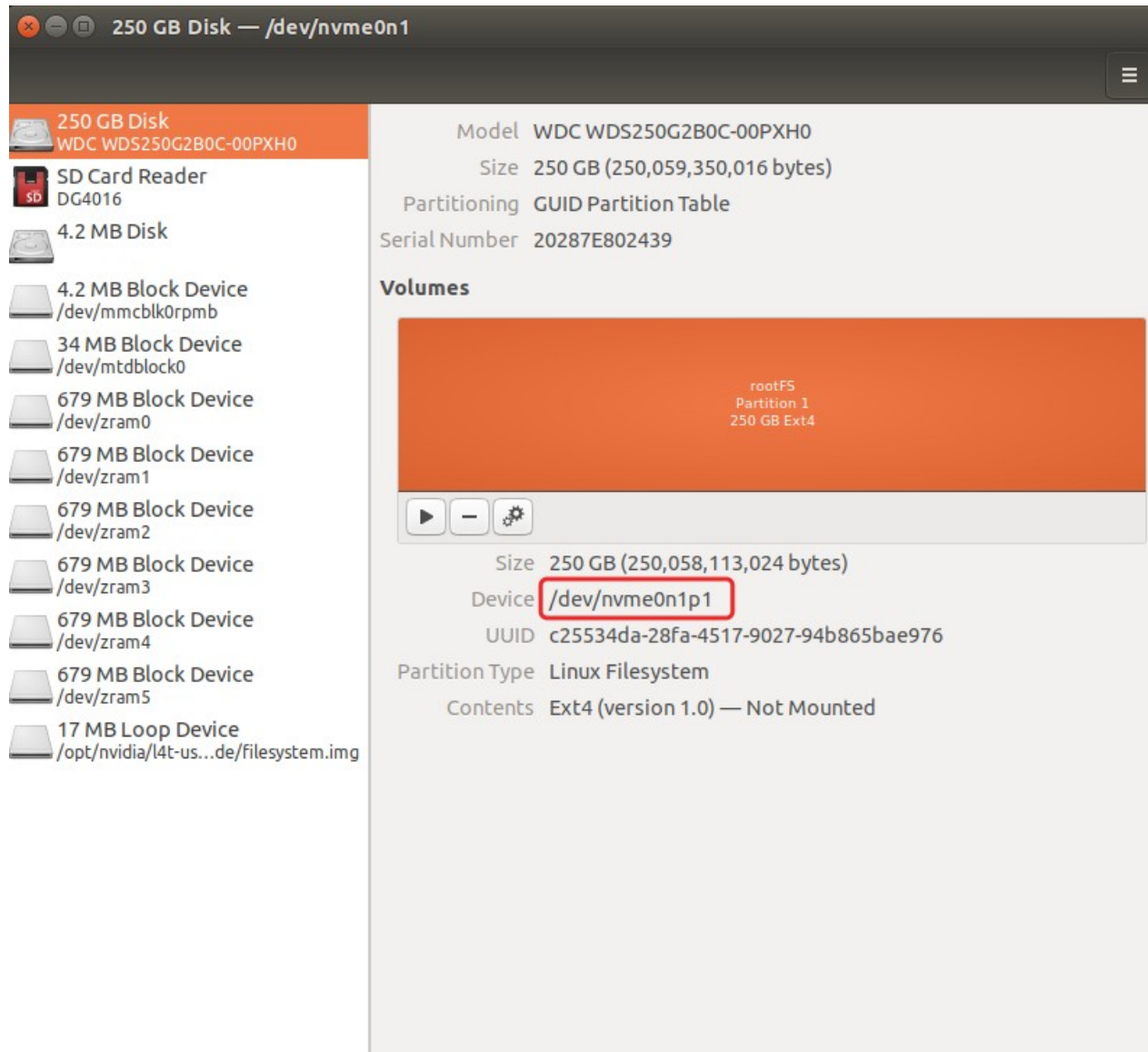
Device /dev/nvme0n1

Contents Unallocated Space



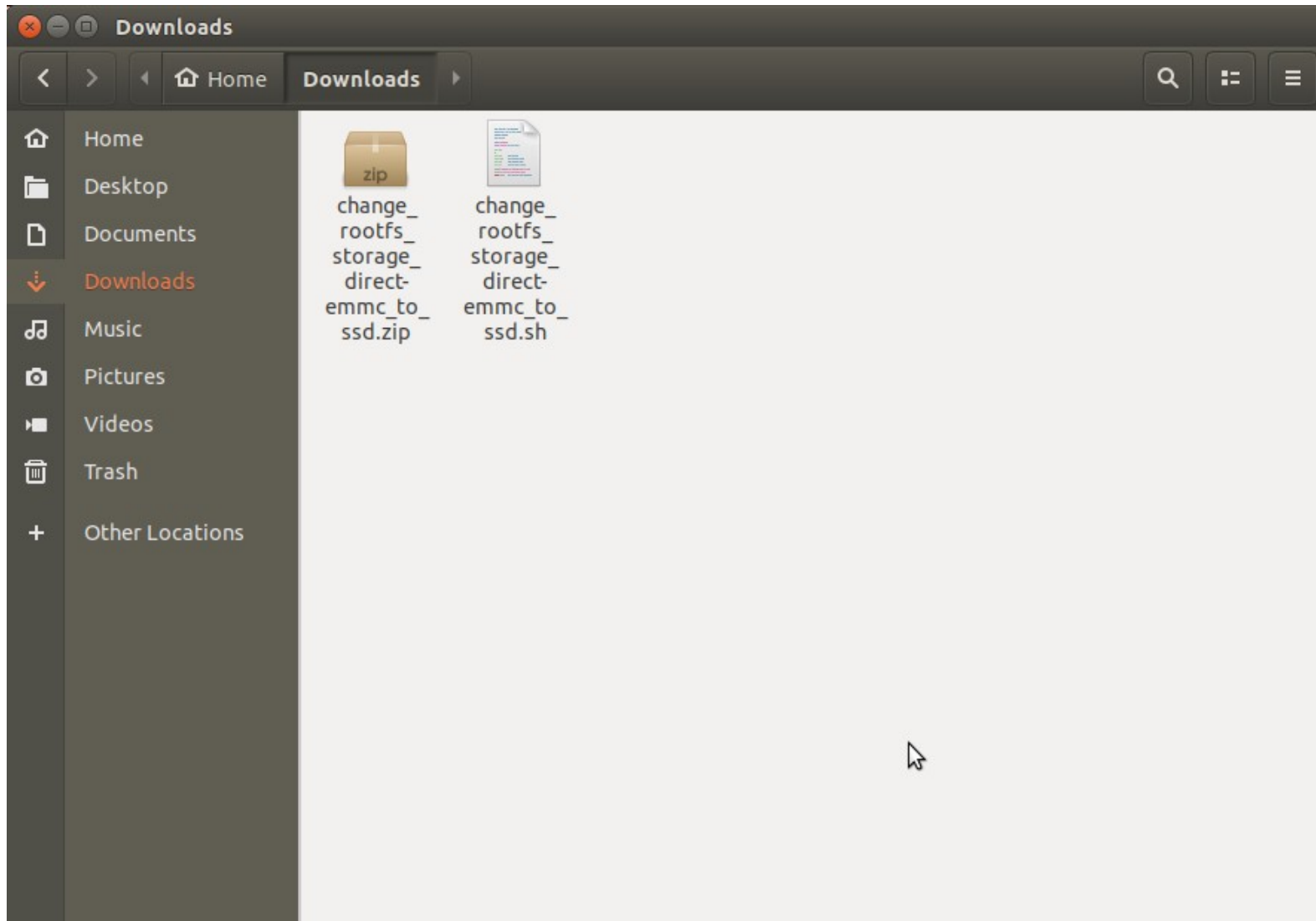


Hint: Note that the SSD Drive's mounting point. We will use it on transferring step.

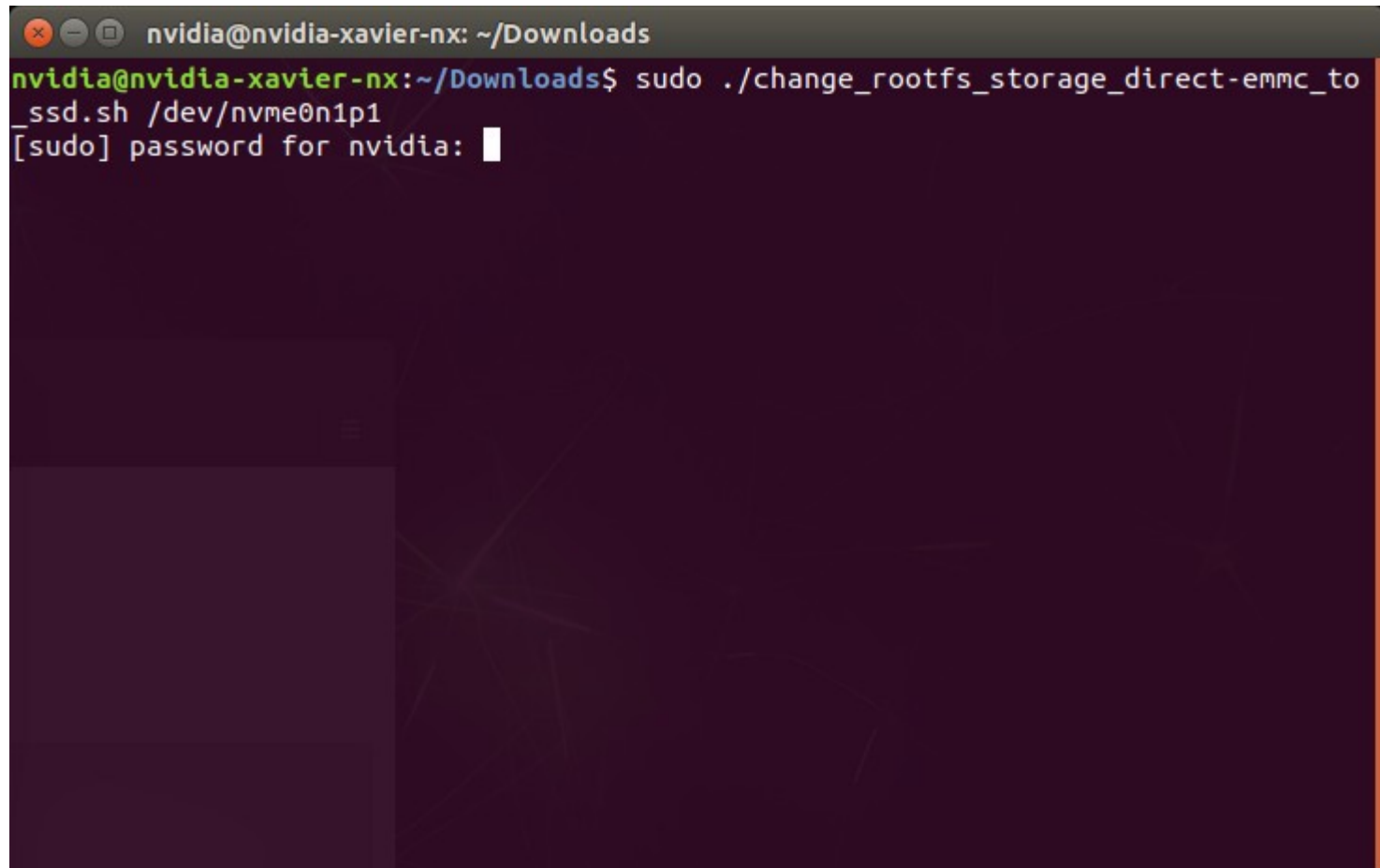


12-) Download the root file-system transferring script file from here & extract it:

[https://github.com/mistelektronik/forecr\\_blog\\_files/raw/master/change\\_rootfs\\_storage\\_direct-emmc\\_to\\_ssd.zip](https://github.com/mistelektronik/forecr_blog_files/raw/master/change_rootfs_storage_direct-emmc_to_ssd.zip)



13-) Open a terminal from this folder and type this command below (our SSD mounted on /dev/nvme0n1p1)  
"sudo ./change\_rootfs\_storage\_direct-emmc\_to\_ssd.sh /dev/nvme0n1p1"

A terminal window with a dark background and light-colored text. The window title bar shows standard Linux window controls (close, minimize, maximize) and the text 'nvidia@nvidia-xavier-nx: ~/Downloads'. The terminal content shows a user prompt 'nvidia@nvidia-xavier-nx:~/Downloads\$' followed by the command 'sudo ./change\_rootfs\_storage\_direct-emmc\_to\_ssd.sh /dev/nvme0n1p1'. Below the command, a prompt '[sudo] password for nvidia:' is visible with a white cursor character. The rest of the terminal area is empty.

```
nvidia@nvidia-xavier-nx: ~/Downloads
nvidia@nvidia-xavier-nx:~/Downloads$ sudo ./change_rootfs_storage_direct-emmc_to
_ssd.sh /dev/nvme0n1p1
[sudo] password for nvidia: 
```

14-) After a few minutes later, it completed the steps.

```
nvidia@nvidia-xavier-nx: ~/Downloads
nvidia@nvidia-xavier-nx:~/Downloads$ sudo ./change_rootfs_storage_direct-emmc_to_ssd.sh /dev/nvme0n1p1
[sudo] password for nvidia:
Sorry, try again.
[sudo] password for nvidia:
 11,595,407,399  95%  65.46MB/s   0:02:48 (xfr#130603, to-chk=0/190018)
The rootfs have copied to SSD.
Before extlinux.conf:      APPEND ${cbootargs} quiet root=/dev/mmcblk0p1 rw rootwait rootfstype=ext4 console=ttyTCU0,115200n8 console=tty0 fbcon=map:0 net.ifnames=0
After extlinux.conf:      APPEND ${cbootargs} quiet root=/dev/nvme0n1p1 rw rootwait rootfstype=ext4 console=ttyTCU0,115200n8 console=tty0 fbcon=map:0 net.ifnames=0
extlinux.conf file updated.
Reboot for changes to take effect.
nvidia@nvidia-xavier-nx:~/Downloads$
```

15-) Then, type "sudo reboot"

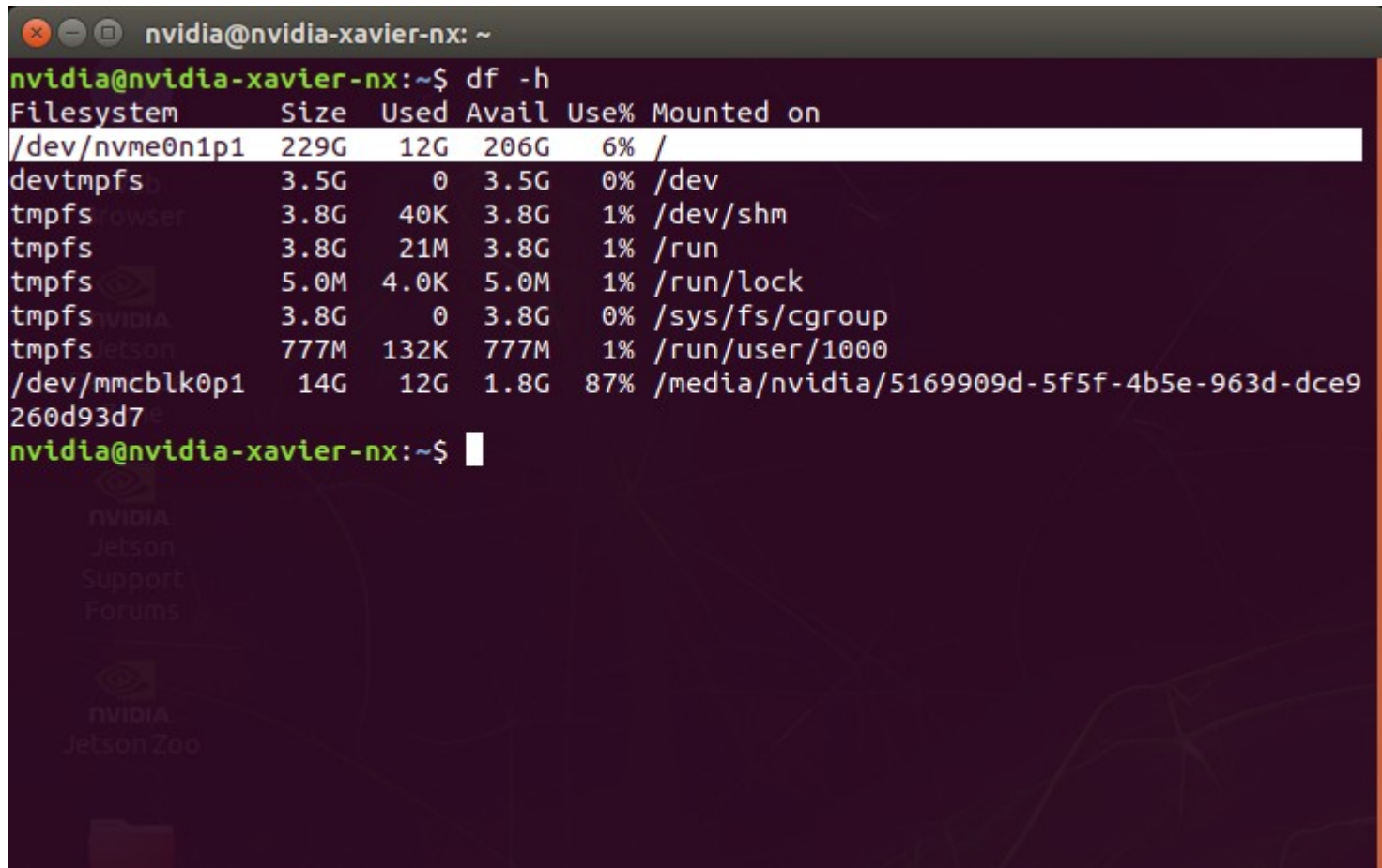


```
nvidia@nvidia-xavier-nx: ~/Downloads
nvidia@nvidia-xavier-nx:~/Downloads$ sudo ./change_rootfs_storage_direct-emmc_to_ssd.sh /dev/nvme0n1p1
[sudo] password for nvidia:
Sorry, try again.
[sudo] password for nvidia:
 11,595,407,399 95% 65.46MB/s 0:02:48 (xfr#130603, to-chk=0/190018)
The rootfs have copied to SSD.
Before extlinux.conf: APPEND ${cbootargs} quiet root=/dev/mmcblk0p1 rw rootwait rootfstype=ext4 console=ttyTCU0,115200n8 console=tty0 fbcon=map:0 net.ifnames=0
After extlinux.conf: APPEND ${cbootargs} quiet root=/dev/nvme0n1p1 rw rootwait rootfstype=ext4 console=ttyTCU0,115200n8 console=tty0 fbcon=map:0 net.ifnames=0
extlinux.conf file updated.
Reboot for changes to take effect.
nvidia@nvidia-xavier-nx:~/Downloads$ sudo reboot
```

16-) After the JEtson opened successfully, type "df -h" again to check the SSD mounted as root file-system



```
nvidia@nvidia-xavier-nx: ~  
nvidia@nvidia-xavier-nx:~$ df -h  
Filesystem      Size  Used Avail Use% Mounted on  
/dev/nvme0n1p1  229G   12G  206G   6% /  
devtmpfs        3.5G     0  3.5G   0% /dev  
tmpfs            3.8G   40K   3.8G   1% /dev/shm  
tmpfs            3.8G   21M   3.8G   1% /run  
tmpfs            5.0M   4.0K   5.0M   1% /run/lock  
tmpfs            3.8G     0  3.8G   0% /sys/fs/cgroup  
tmpfs            777M  132K   777M   1% /run/user/1000  
/dev/mmcblk0p1   14G   12G   1.8G  87% /media/nvidia/5169909d-5f5f-4b5e-963d-dce9260d93d7  
nvidia@nvidia-xavier-nx:~$
```

The image shows a terminal window on an NVIDIA Jetson device. The terminal displays the output of the 'df -h' command, which shows disk usage for various filesystems. The root filesystem (/dev/nvme0n1p1) is 229G in size with 12G used (6% usage). Other filesystems include devtmpfs, tmpfs for /dev/shm, /run, /run/lock, /sys/fs/cgroup, and /run/user/1000, all with very low usage. The /dev/mmcblk0p1 partition is 14G in size with 12G used (87% usage). The terminal window has a dark background with a faint NVIDIA Jetson logo and text in the bottom left corner, including 'nvidia Jetson Support Forums' and 'nvidia Jetson Zoo'.