

Raspberry Pi : démarrage par réseau et montage NFS

■ ■ ■ Préparation du démarrage bootp, PXE

On va créer le répertoire RASPI, dédié au « *filesystem* » du Raspberry PI :

- le répertoire occupera autour de 3Go en utilisation :

```
xterm
$ mkdir RASPI
$ cd RASPI
```

- un répertoire *client* contenant l'intégralité du système Raspbian du Raspberry (répertoires */etc*, */home*, */bin*, *etc.*) qui sera accessible par le protocole NFS ;
- un répertoire *boot* contenant le noyau et les fichiers de « *bas-niveau* » pour le raspberry Pi lui-même, qui sera accessible par le protocole bootp ;

Vous téléchargerez la distribution « Raspbian lite » sur le site officiel du Raspberry PI et le mettrez dans votre répertoire RASPI.

```
xterm
$ wget https://downloads.raspberrypi.org/raspbian_lite_latest
$ unzip raspbian_lite_latest
```

Nous récupérerons le contenu des deux partitions de cette distribution pour remplir nos deux répertoires :

- ▷ le *filesystem* raspbian dans le répertoire *client* depuis la partition n°2 :

```
xterm
$ sudo losetup -P /dev/loop7 2019-09-26-raspbian-buster-lite.img
$ sudo mount /dev/loop7p2 /mnt
$ mkdir client
$ sudo rsync -xa --progress /mnt/ client/
$ sudo umount /mnt
```

- ▷ les fichiers de « *boot* » depuis la partition n°1 :

```
xterm
$ mkdir boot
$ sudo mount /dev/loop7p1 /mnt
$ cp -r /mnt/* boot/
```

Nous installerons le serveur NFS :

```
xterm
$ sudo apt install nfs-kernel-server
```

Configuration du partage NFS dans le fichier */etc/exports* :

```
xterm
pef@cube:/etc$ cat exports
# /etc/exports: the access control list for filesystems which may be exported
# to NFS clients. See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes hostname1(rw, sync, no_subtree_check)
# hostname2(ro, sync, no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4 gss/krb5i(rw, sync, fsid=0, crossmnt, no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw, sync, no_subtree_check)
#
/home/pef/RASPI/client *(rw, sync, no_subtree_check, no_root_squash)
/home/pef/RASPI/boot *(rw, sync, no_subtree_check, no_root_squash)
```

On active le service NFS et RPCBind :

```
xterm
$ sudo systemctl enable nfs-kernel-server
$ sudo systemctl enable rpcbind
```

Si vous modifiez la configuration d'un export, il faut redémarrer le service NFS :

```
xterm
$ sudo systemctl restart nfs-kernel-server
```

Pour voir les points de montage offert par un serveur NFS :

```
xterm
$ showmount -e 127.0.0.1
Export list for 127.0.0.1:
/home/pef/RASPI/boot *
/home/pef/RASPI/client *
```

■ ■ ■ Mise en service du serveur TFTP, DNS, DHCP

On va utiliser la commande dnsmasq :

```
# dongle ethernet gigabit
IF=enx000ec6885a80
#adaptateur USB C
#IF=enx4ce17347b17e

PREFIX=10.20.30
sudo sysctl -w net.ipv4.ip_forward=1
sudo ip link set dev $IF down
sudo ip link set dev $IF address aa:aa:aa:aa:aa:aa
sudo ip link set dev $IF up
sudo ip address add dev $IF $PREFIX.1/24
sudo iptables -t nat -A POSTROUTING -s $PREFIX.0/24 -j MASQUERADE

sudo dnsmasq -d -z -i $IF -F $PREFIX.100,$PREFIX.150,255.255.255.0,12h -O 3,$PREFIX.1
-O 6,8.8.8.8,8.8.4.4 --pxe
-service=0,"Raspberry Pi Boot" --enable-tftp --tftp-root=$HOME/RASPI/boot
```

Attention : `-service=0`, . . . fait partie des options de la commande `dnsmasq`

■ ■ ■ Montage de NFS sur le Raspberry Pi

On modifie le point de montage du Raspberry Pi pour son filesystem, en éditant le fichier

```
/RASPI/boot/cmdline.txt
dwc_otg.lpm_enable=0 console=serial0,115200 console=tty1 root=/dev/nfs
nfsroot=10.20.30.1:/home/pef/RASPI/client
,vers=3 rw ip=dhcp rootwait elevator=deadline
```

Attention : c'est une seule ligne.

On ajoute un point de montage qu'utilisera le Raspberry Pi après avoir booté en éditant le fichier

```
/RASPI/client/etc/fstab:
proc /proc proc defaults 0 0
10.20.30.1:/home/pef/RASPI/boot /boot nfs rsize=8192,wsiz=8192,timeo=14,in
tr,noauto,x-systemd.automount 0 0
```

■ ■ ■ Activation du service SSH sur le Raspberry PI

On passe par le point de montage NFS, c-à-d le répertoire local correspondant au filesystem NFS :

```
xterm
pef@cube:~/RASPI/client/lib/systemd/system$ cat sshswitch.service
[Unit]
Description=Turn on SSH if /boot/ssh is present
#ConditionPathExistsGlob=/boot/ssh{,.txt}
After=regenerate_ssh_host_keys.service

[Service]
Type=oneshot
ExecStart=/bin/sh -c "update-rc.d ssh enable && invoke-rc.d ssh start && rm -f
/boot/ssh ; rm -f /boot/ssh.txt"

[Install]
```

On mets en commentaire la ligne d'option `ConditionPathExistsGlob`.

Pour la dernière version de Raspbian buster

```
xterm
pi@raspberrypi:~ $ cat /etc/modprobe.d/blacklist.conf
blacklist bcm2835_codec
blacklist bcm2835_v4l2
```

■■■ Astuces

Vous pouvez récupérer le répertoire complet du système Raspbian Buster sur :

```
xterm
$ wget https://p-fb.net/fileadmin/TMC/2019-2020/RASPI_buster.tgz
$ sudo tar xvfz RASPI_buster.tgz
```

Si votre Raspberry Pi est connecté directement à votre ordinateur sans utiliser de switch, il peut être nécessaire de désactiver/réactiver votre interface réseau :

```
xterm
$ sudo ip l set enx00606e20053f down;sudo ip l set enx00606e20053f up
```

En effet, quand vous débranchez votre Raspverry Pi, son interface réseau n'est plus alimenté, ce qui peut amener votre ordinateur à faire « tomber » son interface réseau (sans que l'état indiqué par la commande `ip link` le montre)