**DRBL + NFS**

### Objectif du Projet

Ce projet a pour but de configurer un serveur **DRBL (Diskless Remote Boot in Linux)** avec un serveur de stockage **NFS (Network File System)** pour faciliter la gestion des déploiements d'images système à grande échelle. Le serveur DRBL permet aux clients de démarrer en **PXE** et d'accéder à **Clonezilla** pour cloner ou restaurer des disques à partir du serveur. Les images des disques sont ensuite stockées sur un serveur NFS, qui centralise la gestion des sauvegardes et des restaurations.

### Composants Clés

* **DRBL** : Fournit un environnement sans disque où les machines clientes démarrent via le réseau en utilisant PXE, avec la possibilité d'accéder à Clonezilla pour cloner ou restaurer des disques.
* **NFS** : Assure le stockage centralisé des images de disques, accessible par les clients via le réseau.
* **Clonezilla** : Utilisé pour cloner les disques des clients, les sauvegarder sur le serveur NFS, et les restaurer ultérieurement.

### Avantages du Projet

* **Déploiement à grande échelle** : DRBL permet de cloner plusieurs machines simultanément, ce qui est particulièrement utile pour les environnements avec de nombreuses machines comme les salles de formation, les entreprises, ou les écoles.
* **Centralisation des images** : Le serveur NFS stocke de manière centralisée toutes les images de disque, ce qui facilite leur gestion et leur restauration à tout moment.
* **Automatisation** : Grâce à une configuration automatisée de PXE et NFS, les machines clientes peuvent démarrer et restaurer des images sans intervention manuelle.

### Configurations Réseau

* **ens33** : Connecté à Internet via DHCP pour assurer la connectivité externe.
* **ens37** : Utilisé pour le réseau interne avec une IP statique 192.168.79.5 pour gérer les communications PXE et NFS entre le serveur et les clients.

**Source :** [**https://www.ubuntuupdates.org/package/core/noble/universe/base/drbl**](https://www.ubuntuupdates.org/package/core/noble/universe/base/drbl)

**Step 1: Network Configuration + TFTP**

1. Open the **Netplan** configuration file to set up your network interfaces:

sudo nano /etc/netplan/01-netcfg.yaml

1. Modify the file to include the following configuration:

|  |
| --- |
| Config for one interface  network:  version: 2  ethernets:  ens33:  dhcp4: no  addresses:  - 192.168.102.139/24 # Interface pour le réseau local et l'accès internet  gateway4: 192.168.102.2 # Passerelle par défaut (à adapter selon votre réseau)  nameservers:  addresses:  - 8.8.8.8  - 8.8.4.4 |

|  |  |
| --- | --- |
| Config for two interface  network:  version: 2  ethernets:  ens33:  dhcp4: yes # This interface will obtain an IP from DHCP for internet connectivity  ens34:  dhcp4: no  addresses:  - 192.168.79.5/24 # This interface is for the local DRBL network |  |

sudo netplan apply

**sudo nano /etc/default/tftpd-hpa**

**Add🡪**

|  |
| --- |
| **TFTP\_USERNAME="tftp"**  **TFTP\_DIRECTORY="/tftpboot/nbi\_img"**  **TFTP\_ADDRESS="0.0.0.0:69"**  **TFTP\_OPTIONS="--secure --ipv4"** |

**Step 2: System Update and Package Installation**

1. **Update and upgrade** your system:

sudo apt-get update

sudo apt-get upgrade -y

sudo add-apt-repository universe

sudo apt update

**Install the required packages** for DRBL, NFS, TFTP, DHCP, and Syslinux:

sudo apt-get install drbl nfs-kernel-server tftpd-hpa isc-dhcp-server syslinux-common unzip wget -y

**Step 3: Add DRBL GPG Key and Repository**

1. **Add the DRBL GPG Key**:

wget -O - https://drbl.org/GPG-KEY-DRBL | sudo gpg --dearmor -o /etc/apt/trusted.gpg.d/drbl.gpg

1. **Edit the sources list** to add the DRBL repository:

sudo nano /etc/apt/sources.list

1. **Add the following lines** to the sources list file:

deb http://archive.ubuntu.com/ubuntu noble main restricted universe multiverse

deb http://free.nchc.org.tw/drbl-core drbl stable

**Step 4: Install and Configure DRBL**

1. **Update the package list**:

sudo apt update

1. **Install DRBL**:

sudo apt install drbl -y

1. **Start DRBL server installation**:

sudo drblsrv -i

1. **Push the DRBL configuration to the clients**:
2. sudo drblpush -i

**Step 5: Configure DHCP for PXE Boot**

1. **Edit the DHCP configuration file**:

sudo nano /etc/dhcp/dhcpd.conf

1. **Add the following configuration** for the PXE boot network:

subnet 192.168.79.0 netmask 255.255.255.0 {

option subnet-mask 255.255.255.0;

option routers 192.168.79.5;

option domain-name-servers 8.8.8.8, 8.8.4.4;

option rpi-tftp-server "192.168.79.5"; # Serveur TFTP

next-server 192.168.79.5; # Serveur PXE

pool {

range 192.168.79.1 192.168.79.13;

allow unknown-clients;

}

filename "pxelinux.0"; # Fichier de démarrage PXE pour Clonezilla

}

**Step 6: Set Up PXE Boot with Clonezilla**

1. **Ensure PXELINUX files are available**:

sudo cp /usr/lib/syslinux/modules/bios/pxelinux.0 /tftpboot/nbi\_img/

sudo cp /usr/lib/syslinux/modules/bios/ldlinux.c32 /tftpboot/nbi\_img/

sudo cp /usr/lib/syslinux/modules/bios/menu.c32 /tftpboot/nbi\_img/

**Step 7: Download Clonezilla Live**

1. **Download Clonezilla Live ISO**:

**wget https://downloads.sourceforge.net/clonezilla/clonezilla-live-3.1.3-16-amd64.iso**

1. **Mount the ISO and copy the necessary files**:

sudo mkdir /mnt/clonezilla

sudo mount -o loop clonezilla-live-3.1.3-16-amd64.iso /mnt/clonezilla

sudo mkdir -p /tftpboot/nbi\_img/clonezilla

sudo cp /mnt/clonezilla/live/vmlinuz /tftpboot/nbi\_img/clonezilla/

sudo cp /mnt/clonezilla/live/initrd.img /tftpboot/nbi\_img/clonezilla/

sudo cp /mnt/clonezilla/live/filesystem.squashfs /tftpboot/nbi\_img/clonezilla/

sudo umount /mnt/clonezilla

**Step 8: Configuration NFS**

**sudo nano /etc/exports**

**ADD🡪**

**/tftpboot/node\_root/clonezilla-live 192.168.79.0/24(rw,sync,no\_subtree\_check,no\_root\_squash)**

**/home/partimag 192.168.79.0/24(rw,sync,no\_subtree\_check,no\_root\_squash)**

**sudo exportfs -ra**

**sudo systemctl restart nfs-kernel-server**

**sudo chmod -R 777 /tftpboot/node\_root/clonezilla-live**

**sudo chmod -R 777 /home/partimag**

**sudo systemctl restart tftpd-hpa isc-dhcp-server nfs-kernel-server**

**Pour le montage automatique du disque de partage NFS :**

**sudo nano /etc/fstab**

**192.168.79.5:/home/partimag /mnt/partimag nfs defaults,\_netdev 0 0**

**sudo mkdir -p /mnt/partimag**

**sudo mount -a**

**df -h**

**autorisation :**

**sudo chmod -R 777 /home/partimag**

**sudo chown -R nobody:nogroup /home/partimag**

|  |
| --- |
| **( IGNORER CETTE ETAPES )**  sudo nano /etc/exports  add🡪/tftpboot/node\_root/clonezilla-live 192.168.79.0/24(rw,sync,no\_subtree\_check,no\_root\_squash)  sudo exportfs -ra  sudo chmod -R 777 /tftpboot/node\_root/clonezilla-live  sudo systemctl restart nfs-kernel-server  Optional: sudo ufw allow from 192.168.79.0/24 to any port nfs  ls -ld /home/partimag  sudo chmod -R 777 /home/partimag  sudo nano /etc/exports  Add🡪/home/partimag 192.168.79.0/24(rw,sync,no\_subtree\_check,no\_root\_squash)  sudo exportfs -ra  sudo systemctl restart nfs-kernel-server |

Fichier responsible du boot a corrigier pour que DRBL enregistre les images directement sur synology

|  |
| --- |
| **Configuration du montage NFS**   1. **Montage du partage NFS Synology sur le serveur DRBL** :    * Ajoutez les lignes suivantes dans votre fichier **/etc/fstab** pour monter le partage NFS du Synology sur le serveur DRBL et lier le répertoire **/home/partimag** à ce montage.   bash  Copier le code  192.168.10.33:/volume1/drbl-images /mnt/synology\_partimag nfs defaults,\_netdev 0 0  /mnt/synology\_partimag /home/partimag none bind 0 0   1. **Montage automatique du NFS** :    * Après avoir ajouté ces lignes dans **/etc/fstab**, exécutez la commande suivante pour monter automatiquement le partage NFS :   bash  Copier le code  sudo mount -a  Vous pouvez vérifier que le montage fonctionne correctement avec :  bash  Copier le code  df -h /home/partimag  Cela devrait montrer que **/home/partimag** est monté depuis le partage NFS sur le Synology.  **Configuration PXE de Clonezilla avec montage NFS**  Voici la configuration complète du fichier **PXE** pour Clonezilla, avec l'option **ocs\_prerun** ajustée pour s'assurer que le partage NFS est monté avant le démarrage de Clonezilla :  bash  Copier le code  label Clonezilla-live  MENU LABEL Clonezilla: choose save or restore later  KERNEL Clonezilla-live-vmlinuz  APPEND initrd=Clonezilla-live-initrd.img boot=live union=overlay username=user hostname=noble config loglevel=0 noswap edd=on nomodeset enforcing=0 locales=en\_US.UTF-8 keyboard-layouts=us ocs\_live\_run="clonezilla -l en\_US.UTF-8 -p poweroff -k --skip-lite-menu" ocs\_live\_extra\_param= ocs\_live\_batch=no net.ifnames=0 noeject netboot=nfs nfsroot=192.168.15.50:/tftpboot/node\_root/clonezilla-live/ ocs\_server="192.168.15.50" ocs\_daemonon="ssh" ocs\_prerun="mount -t nfs -o vers=3 192.168.10.33:/volume1/drbl-images /home/partimag"    TEXT HELP  Clonezilla Live 20240715-noble-amd64 runs on RAM  ENDTEXT  **Explication :**   * **nfsroot=192.168.15.50:/tftpboot/node\_root/clonezilla-live/** : Définit l'emplacement du système Clonezilla Live à charger via PXE depuis le serveur DRBL. * **ocs\_prerun** : Monte le partage NFS du Synology avant que Clonezilla ne commence l'opération de clonage ou de restauration. Le partage est monté sur **/home/partimag** pour s'assurer que les images sont enregistrées directement sur le serveur Synology.   **Étapes supplémentaires :**   1. **Appliquez ces modifications** dans votre fichier **/tftpboot/nbi\_img/pxelinux.cfg/default** ou **grub.cfg** si vous utilisez UEFI. 2. **Redémarrez le service TFTP** :   bash  Copier le code  sudo systemctl restart tftpd-hpa   1. **Testez un démarrage PXE** sur un client pour vérifier que Clonezilla utilise correctement **/home/partimag** monté sur le NAS pour sauvegarder ou restaurer les images.   Avec cette configuration, les images seront stockées directement sur le NAS Synology via NFS, et Clonezilla pourra y accéder à partir de **/home/partimag** sur le serveur DRBL. |