Monter les périphériques (externes ou non) facilement

- Objet : Un script permettant de monter les périphériques externes comme avec pmount, mais en se reposant sur udisks/udisks2.
- Niveau requis :
  avisé
- Commentaires : *Pour les gens qui n'aiment pas utiliser l'interface graphique pour monter leurs périphériques.*

Introduction

- Monter les périphériques externes automatiquement, c'est mal.
- Passer root pour monter les périphériques externes, c'est mal.
- Utiliser le fstab pour monter les périphériques externes, ne tiens la route que s'ils sont utilisés comme une extension des supports de stockage internes, ce qui n'est en général pas le cas.
- Utiliser la ligne de commande pour monter un périphérique externe, c'est bien, mais utiliser pmount offre deux inconvénient.
  - D'abord, cela installe un nouveau fichier setgid sur le système, donc une nouvelle faille de sécurité potentielle
  - Ensuite, le paquet risque d'être délaissé car le travail est fait par udisks/udisks2 dans les système modernes.
- La syntaxe de udisksctl (udisks2) est un peu lourde pour la resaisir à chaque fois (`udisksctl mount -b`), et demande l'emplacement exact du fichier de périphérique (ex. `/dev/sdf1`.)

Installation

Attention, pensez bien à désactiver le montage automatique avant d'utiliser ce script, sans quoi il risque de se révéler relativement inconfortable. Voir Comment désactiver l'automount sous Gnome3.4#modifier-montage-auto

Copier le fichier suivant dans votre répertoire `~/bin/`, à créer s'il n'existe pas.

~/bin/fmount

```
#!/bin/sh

usage()
{
  echo "Utilisation :
  $0 [-u] [[/dev/]device | [/dev/disk/by-uuid/]uuid |
[/dev/mapper/]uuid]
```

```
Monte un système de fichier dans un sous-répertoire de /media via
udisks ou
udisks2.
Si aucun périphérique n'est donné, fmount tente de deviner le dernier
périphérique inséré.

Option de l'application :
  -u Démonte le système de fichier
  -h Affiche ce message d'aide"
}

finddev()
{
  DEV=$1

  if (echo $DEV | grep -q '^/dev/') && [ -e $DEV ]
  then
    echo $DEV
  fi

  if [ -e "/dev/disk/by-uuid/$DEV" ]
  then
    echo "/dev/disk/by-uuid/"$(echo $DEV | sed 's/.*/\L&/')
  fi

  if [ -e "/dev/$DEV" ]
  then
    echo "/dev/$DEV"
  fi

  if [ -e "/dev/mapper/$DEV" ]
  then
    echo "/dev/mapper/$DEV"
  fi
}

wrapper()
{
  ACTION=$1
  DEVICE=$2

  case $ACTION in
    "mount")
      if [ -e /usr/bin/udisksctl ]
      then
        udisksctl mount -b $DEVICE
      else
        udisks --mount $DEVICE
      fi
      return $?
      ;;
```

```
    "umount")
      if [ -e /usr/bin/udisksctl ]
      then
        udisksctl unmount -b $DEVICE
      else
        udisks --unmount $DEVICE
      fi
      return $?
      ;;
    *)
      echo "Action invalide"
      return 1
      ;;
  esac
}

guesslast()
{
  LOG=$(dmesg| tail -n 50 | grep 'Attached scsi generic' -A 11 | tail -
n 12)
  PARTS=$(echo $LOG | grep "sd.: sd.")
  if [ "$PARTS" = "" ]
  then
    DEV=$(echo $LOG | sed '/\[sd/s/.*\[\(sd[^\]]*\)\].*/\1/')
  else
    DEV=$(echo $PARTS | sed '/sd.:/s/.*sd.: \(sd[^ ]*\).*/\1/')
  fi
  echo $DEV
}

ACTION="mount"

if [ $# -ne 0 ] && [ $1 = "-u" ]
then
  ACTION="umount"
  shift
fi

if [ $# -ne 0 ] && [ $1 = "-h" ]
then
  usage
  exit 0
fi

if [ $# -ne 1 ]
then
  DEV=$(guesslast)
else
  DEV=$1
fi
```

```
DEV=$(finddev $DEV)

if [ -z "$DEV" ]
then
  echo "Impossible de trouver le périphérique $DEV."
  exit 1
fi

wrapper $ACTION $DEV
exit $?
```

Rendez ce fichier exécutable :

```
chmod +x ~/bin/fmount
```

Si le dossier bin n'existait pas, indiquez au shell qu'il est dans le PATH :

```
PATH="$HOME/bin:$PATH"
```

Utilisation

```
Utilisation :
  fmount [-u] [[/dev/]device | [/dev/disk/by-uuid/]uuid |
[/dev/mapper/]uuid]

Monte un système de fichier dans un sous-répertoire de /media via udisks ou
udisks2.
Si aucun périphérique n'est donné, fmount tente de deviner le dernier
périphérique inséré.

Option de l'application :
  -u Démonte le système de fichier
  -h Affiche ce message d'aide
```

Par l'exemple

Si des disques occupent sda sdb sdc sdd sde, taper pour le nouveau disque à monter :

```
fmount sdf1
```

Pour démonter le même :

```
fmount -u sdf1
```

Et voilà ! 😀

Problèmes rencontrés

Montage via ssh

Udisks utilise policy-kit et son système de session actives. Par défaut, seul l'utilisateur de la session active (affichée à l'écran) peut utiliser udisks. La conséquence directe est qu'un utilisateur se connectant par ssh ne peut pas monter les disques externes.

Une solution est de changer la politique de montage en copiant le fichier suivant dans le dossier `/etc/polkit-1/localauthority/50-local.d/` :

/etc/polkit-1/localauthority/50-local.d/50-mount.pkla

```
[Mount a system-internal device]
Identity=*
Action=org.freedesktop.udisks.*
ResultAny=yes
ResultActive=yes
ResultInactive=yes
```

Au prochain chargement des règles[1)], tout ira bien \o/

1)
le reboot

From:
http://debian-facile.org/ - **Documentation - Wiki**

Permanent link:
http://debian-facile.org/utilisateurs:captnfab:tutos:pmount-udisks

Last update: **07/07/2014 20:57**