Mise en place d'une MakerBox

Une MakerBox est un boitier autonome qui joue le rôle d'un point d'accès sans-fil, un commutateur téléphonique IP et un serveur multimédia. La MakerBox est réalisée à partir d'un routeur sans-fil TP-Link MR3020 alimenté par une batterie à chargeur solaire.

Le choix du routeur TP-Link MR3020 est dû à son succès croissant. Ce routeur portable met en place les standards IEEE 802.11 b/g/n et dispose d'un port LAN/WAN à 100 Mbps, un port USB 2.0 pour un modem 3G/4G, et un port mini-USB pour l'alimentation électrique. La portabilité du routeur MR-3020 en a fait la solution de choix pour le projet PirateBox ou la nouvelle sonde de mesure du projet ATLAS.

Pour fabriquer votre MakerBox, vous avez besoin du matériel suivant:

- Routeur TP-Link MR3020: Les spécifications détaillées du routeur MR3020 sont disponibles sur le site de TP-Link. Vous pouvez acheter un TL MR3020 à 34,90 euros sur LDLC.
- Chargeur solaire: Un chargeur solaire 2600mAh coûte environ 8 euros sur ebay.
- Clé USB: Une clé USB 8Go est largement suffisante et coûte 8.99 euros sur cdiscount.



Figure 1. TP-Link MR3020

Commencez par télécharger la distribution Attitude Adjustment du firmware OpenWRT et procédez comme suit. Connectez votre ordinateur au routeur TL-MR3020 avec un câble Ethernet. Par défaut, le routeur est configuré avec l'adresse IP 192.168.0.254/24. Attribuez donc une adresse de la même plage à votre ordinateur, par exemple 192.168.0.1. Ouvrez votre navigateur préféré et saisissez l'adresse du routeur (par défaut login = admin et mot de passe = admin). Mettez à jour le firmware de votre routeur avec le firmware Attitude Adjustment nouvellement téléchargé. Attendez deux défilements successifs de l'indicateur de progression (le routeur redémarrera entretemps) et procédez à une configuration basique pour connecter votre routeur à Internet.

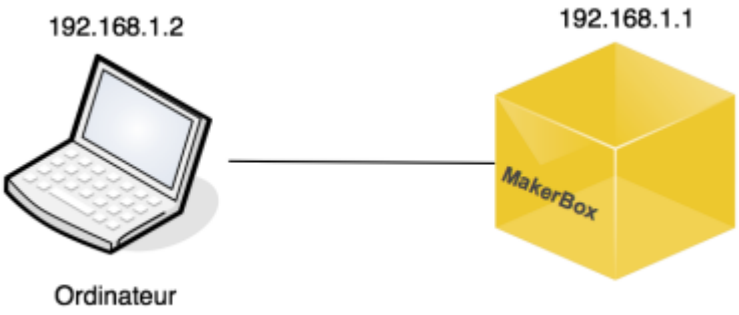


Figure 2. Schéma initial du réseau

Si vous partez d'une ancienne version OpenWrt et que vous souhaitez mettre à jour votre routeur avec la distribution Attitude Adjustment, vous pouvez suivre ce tutoriel.

Avec OpenWrt l'adresse IP configurée initialement sur votre routeur est 192.168.1.1. Modifiez donc l'adresse de votre ordinateur en mettant une adresse dans le même réseau, par exemple 192.168.1.2 comme indiqué sur la figure 2. Connectez vous au routeur avec un câble Ethernet et lancez une session telnet. L'exemple suivant suppose que vous disposez d'une passerelle vers internet. Il s'agit typiquement du boitier ADSL de votre opérateur. L'adresse IP de votre routeur MR3020 est 192.168.1.1, celle de la passerelle par défaut est 192.168.1.100.

La distribution OpenWrt dispose d'une interface de configuration graphique appelée LuCI. Il est donc possible de configurer votre routeur en saisissant son adresse IP dans votre navigateur et en se laissant guider par les menus de l'interface graphique.

Modifiez le contenu du fichier `/etc/config/network` en tapant la commande `vi /etc/config/network` sur votre MR3020 et en suivant l'exemple ci-dessous:

/etc/config/network

```
config interface 'lan'
        option ifname 'eth0'
        option type 'bridge'
        option proto 'static'
        option ipaddr '192.168.1.1'
        option netmask '255.255.255.0'
        option gateway '192.168.1.100'
        list dns '192.168.1.100'
        list dns '8.8.8.8'
```

Modifiez les paramètres sans-fil du MR3020 en activant l'interface radio et en déclarant un nouveau SSID appelé MakerFaire. Le chiffrement WPA2-PSK est choisi dans l'exemple ci-dessous, mais sachez que votre routeur supporte les chiffrements WPA/WPA2 PSK (“WPA Personal”), 802.11i (“WPA Enterprise”), et le chiffrement WEP.

/etc/config/wireless

```
config wifi-device 'radio0'
        option type 'mac80211'
        option channel '11'
        option macaddr 'f8:d1:11:bd:62:ce'
        option hwmode '11ng'
        option htmode 'HT20'
        list ht_capab 'SHORT-GI-20'
        list ht_capab 'SHORT-GI-40'
```

```
        list ht_capab 'RX-STBC1'
        list ht_capab 'DSSS_CCK-40'
        option disabled '0'
        option txpower '27'
        option country 'FR'

config wifi-iface
        option device 'radio0'
        option network 'lan'
        option mode 'ap'
        option ssid 'MakerFaire'
        option encryption 'psk2'
        option key 'makerfaire'
```

Redémarrez votre routeur en tapant `reboot` ou tout simplement en débranchant le câble d'alimentation USB. Branchez maintenant votre routeur à la passerelle par défaut comme indiqué sur la figure 3. Connectez votre ordinateur au réseau sans-fil MakerFaire et lancez une session SSH vers votre routeur. Vérifiez que la connexion Internet est établie.

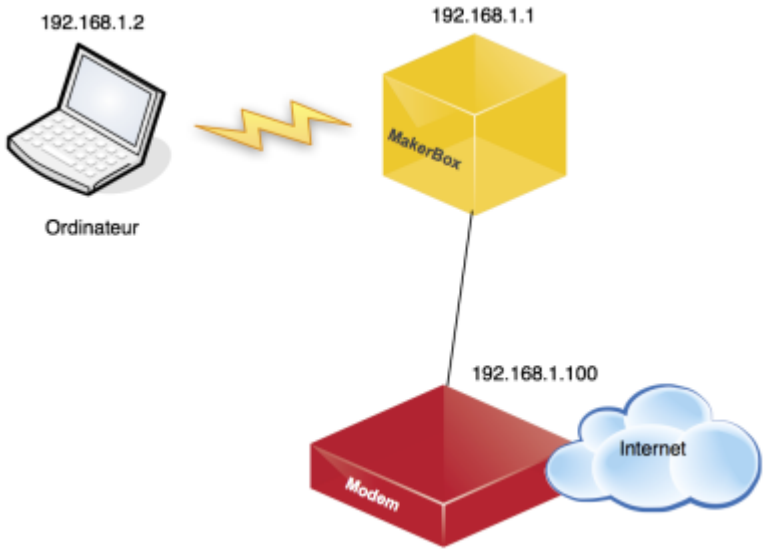


Figure 3. Schéma complet du réseau connecté à Internet



En cas de problème, il est possible de revenir sur une configuration d'usine de votre routeur en utilisant le mode *failsafe*. Pour cela, démarrez le TL MR-3020. Lorsque le bouton WPS commence à clignoter, appuyez jusqu'au clignotement rapide. Le routeur est maintenant passé en mode *failsafe*. Vous pouvez y accéder par telnet à l'adresse 192.168.1.1 et effacer toute installation de paquets en tapant `mount_root` et puis `firstboot`. Votre routeur est maintenant en configuration usine !

Votre routeur MR3020 dispose de 4 MB de mémoire flash. Cet espace est suffisant pour une installation OpenWrt basique. Cependant, comme il s'agit d'intégrer des applications supplémentaires sur votre MakerBox, une extension de l'espace de stockage s'impose. En utilisant le mécanisme `extroot` de OpenWrt, vous avez la possibilité d'étendre l'espace de stockage de votre système de fichier root en utilisant la clé USB.

Commencez par supprimer l'interface graphique LuCI dans le but de libérer l'espace de stockage pour les prochaines mises à jour. Pour cela, il est indispensable de saisir la commande suivante plusieurs fois afin de supprimer toutes les dépendances:

```
root@MakerFaire:~# opkg remove luci* --force-removal-of-dependent-packages
```

Installez les paquets nécessaire pour le montage de la clé USB et pour créer les partitions. Les manipulations suivantes supposent bien évidemment que votre routeur est toujours connecté à Internet pour télécharger les paquets correspondants.

```
root@MakerFaire:~# opkg update && opkg install kmod-usb-storage fdisk kmod-
usb-uhci kmod-usb-ohci
root@MakerFaire:~# insmod usbcore
root@MakerFaire:~# insmod uhci
root@MakerFaire:~# insmod usb-ohci
```

Branchez votre clé USB sur le routeur MR3020. En général, le périphérique correspondant se trouve dans `/dev/sda`, pour en être sûr n'hésitez pas à consulter le journal des messages obtenu en tapant la commande `dmesg`.

Effacez les partitions du périphérique `/dev/sda`:

```
root@MakerFaire:~# dd if=/dev/zero of=/dev/sda bs=64k count=1
```

Créez maintenant de nouvelles partitions: 64 MB swap, 256 MB root et le reste dans une partition séparée:

```
root@MakerFaire:~# cat <<'__EOF__' | fdisk /dev/sda
n
p
1

+64M
t
82
n
p
2

+256M
n
p
3


w
q
__EOF__
```





Il est possible de créer les partitions sur votre clé USB en mode interactif. Commencez par saisir la commande `fdisk /dev/sda` et puis laissez vous guider. Les commandes que vous devez utiliser sont mentionnées successivement dans l'extrait précédent, y compris les entrées.

Vérifiez maintenant les partitions créées:

```
root@MakerFaire:~# fdisk -l /dev/sda

Disk /dev/sda: 8044 MB, 8044675072 bytes
248 heads, 62 sectors/track, 1021 cylinders, total 15712256 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x290acc44

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1            2048      133119       65536   82  Linux swap / Solaris
/dev/sda2          133120      657407      262144   83  Linux
/dev/sda3          657408    15712255     7527424   83  Linux
```

Supprimez `fdisk` pour libérer l'espace de stockage:

```
root@MakerFaire:~# opkg remove fdisk
root@MakerFaire:~# opkg install e2fsprogs
```

Créez les systèmes de fichier `root` et `data`:

```
root@MakerFaire:~# mkfs.ext3 /dev/sda2 -L openwrt-extroot
root@MakerFaire:~# mkfs.ext3 /dev/sda3 -L openwrt-data
```

Supprimez `e2fsprogs` et installez `mkswap`:

```
root@MakerFaire:~# opkg --force-removal-of-essential-packages remove
e2fsprogs libext2fs libcom_err libpthread
root@MakerFaire:~# opkg install swap-utils
```

Activez le swap:

```
root@MakerFaire:~# mkswap /dev/sda1 -L openwrt-swap
```

Installez `blkid`, `block-mount` et les pilotes ext2/ext3/ext4 pour le support `extroot`:

```
root@MakerFaire:~# opkg install blkid block-mount kmod-fs-ext4
```

Vérifiez le résultat avec `blkid`:

```
root@MakerFaire:~# blkid
/dev/mtdblock2: TYPE="squashfs"
```

```
/dev/sda1: LABEL="openwrt-swap" UUID="cdc553ad-b2d8-4d1c-9290-7b8119b5078e"
TYPE="swap"
/dev/sda2: LABEL="openwrt-extroot"
UUID="410340e0-958b-4aee-920f-6fca68aca1ce" TYPE="ext3"
/dev/sda3: LABEL="openwrt-data" UUID="688cdbd1-e288-4128-9582-a134a1710c54"
TYPE="ext3"
```

Effacez les précédentes configurations `fstab`:

```
root@MakerFaire:~# /etc/init.d/fstab stop
root@MakerFaire:~# uci delete fstab.@swap[0]
root@MakerFaire:~# uci delete fstab.@mount[0]
root@MakerFaire:~# uci commit fstab
```

Activez le swap au prochain démarrage:

```
root@MakerFaire:~# uci add fstab swap
root@MakerFaire:~# uci set fstab.@swap[-1].uuid=`blkid -s UUID /dev/sda1 |
cut -d\" -f2`
root@MakerFaire:~# uci set fstab.@swap[-1].enabled=1
root@MakerFaire:~# uci commit fstab
```

Activez `extroot` au prochain démarrage:

```
root@MakerFaire:~# uci add fstab mount
root@MakerFaire:~# uci set fstab.@mount[-1].uuid=`blkid -s UUID /dev/sda2 |
cut -d\" -f2`
root@MakerFaire:~# uci set fstab.@mount[-1].options=rw,sync,noatime
root@MakerFaire:~# uci set fstab.@mount[-1].fstype=ext3
root@MakerFaire:~# uci set fstab.@mount[-1].enabled_fsck=1
root@MakerFaire:~# uci set fstab.@mount[-1].enabled=1
root@MakerFaire:~# uci set fstab.@mount[-1].target=/
```

Activez la partition `data` au prochain démarrage:

```
root@MakerFaire:~# mkdir -p /mnt/data
root@MakerFaire:~# uci add fstab mount
root@MakerFaire:~# uci set fstab.@mount[-1].uuid=`blkid -s UUID /dev/sda3 |
cut -d\" -f2`
root@MakerFaire:~# uci set fstab.@mount[-1].options=rw,sync,noatime
root@MakerFaire:~# uci set fstab.@mount[-1].fstype=ext3
root@MakerFaire:~# uci set fstab.@mount[-1].enabled_fsck=1
root@MakerFaire:~# uci set fstab.@mount[-1].enabled=1
root@MakerFaire:~# uci set fstab.@mount[-1].target=/mnt/data
root@MakerFaire:~# uci commit fstab
```

Désactive l'option `anon_mount`:

```
root@MakerFaire:~# uci set fstab.automount.anon_mount=0
root@MakerFaire:~# uci commit fstab
```

Vérifiez le résultat dans le fichier `/etc/config/fstab` qui doit être similaire à l'extrait suivant:

/etc/config/fstab

```
config global 'automount'
    option from_fstab '1'
    option anon_mount '0'

config global 'autoswap'
    option from_fstab '1'
    option anon_swap '0'

config swap
    option uuid 'cdc553ad-b2d8-4d1c-9290-7b8119b5078e'
    option enabled '1'

config mount
    option uuid '410340e0-958b-4aee-920f-6fca68aca1ce'
    option options 'rw,sync,noatime'
    option fstype 'ext3'
    option enabled_fsck '1'
    option enabled '1'
    option target '/'

config mount
    option uuid '688cdbd1-e288-4128-9582-a134a1710c54'
    option options 'rw,sync,noatime'
    option fstype 'ext3'
    option enabled_fsck '1'
    option enabled '1'
    option target '/mnt/data'
```

Copiez le contenu du répertoire racine:

```
root@MakerFaire:~# mkdir -p /mnt/sda2 /tmp/cproot
root@MakerFaire:~# mount --bind / /tmp/cproot
root@MakerFaire:~# mount /dev/sda2 /mnt/sda2
root@MakerFaire:~# tar -C /tmp/cproot -cvf - . | tar -C /mnt/sda2 -xf -
root@MakerFaire:~# sync
root@MakerFaire:~# umount /mnt/sda2
root@MakerFaire:~# umount /tmp/cproot
```

Activez `extroot` au prochain démarrage:

```
root@MakerFaire:~# /etc/init.d/fstab enable
root@MakerFaire:~# /etc/init.d/fstab start
root@MakerFaire:~# /etc/init.d/fstab whole_root_enable
```

Redémarrez votre routeur, ... Vérifiez que tout est en place (attendez un instant pour le montage de

/dev/sda3 soit effectué):

```
root@MakerFaire:~# df -h
Filesystem                Size      Used Available Use% Mounted on
rootfs                  246.0M     19.8M    213.4M   8% /
/dev/root                 2.0M      2.0M         0 100% /rom
tmpfs                    14.3M      1.8M     12.5M  13% /tmp
tmpfs                   512.0K         0    512.0K   0% /dev
/dev/sda2               246.0M     19.8M    213.4M   8% /
/dev/sda3                 7.1G    130.5M      6.6G   2% /mnt/data
```

```
root@MakerFaire:~# free
             total         used         free       shared      buffers
Mem:         29220        23268         5952            0         3088
-/+ buffers:              20180         9040
Swap:        65532            0        65532
```

Maintenant que l'espace de stockage est étendu, remettez en place quelques outils intéressants:

```
root@MakerFaire:~# opkg update && opkg install luci fdisk e2fsprogs tcpdump
screen mtr wget luci-app-samba
```

Commencez par installer Asterisk sur votre routeur:

```
root@MakerFaire:~# opkg update && opkg install asterisk18
```

Editez le contenu du fichier /etc/hosts pour ajouter une équivalence entre 127.0.0.1 et le nom de votre routeur:

/etc/hosts

```
127.0.0.1 MakerFaire
```

Dans le fichier /etc/asterisk/sip.conf, déclarez les paramètres des utilisateurs SIP. Deux utilisateurs, appelés Alice et Bob sont déclarés dans l'exemple qui suit. Un mot de passe est aussi attribué à chaque utilisateur secret=100.

/etc/asterisk/sip.conf

```
[alice]
type=friend
secret=100
host=dynamic
context=default
allow=all

[bob]
type=friend
secret=100
```

```
host=dynamic
context=default
allow=all
```

Dans le fichier `/etc/asterisk/extensions.conf`, déclarez les extensions 100 et 200 qui permettent de joindre Bob et Alice respectivement. Il est aussi possible de déclarer l'extension 600 qui permet de tester la latence du système avec un mécanisme d'écho.

/etc/asterisk/extensions.conf

```
[default]
<del>include => demo</del>
exten => 100,1,Dial(SIP/bob)
exten => 200,1,Dial(SIP/alice)
exten => 600,1,Echo
```



Pour noter ou changer les modules utilisés par Asterisk, vous pouvez consulter le fichier `/etc/asterisk/modules.conf`. Dans ce fichier, vous avez la possibilité de configurer les modules de la musique d'attente, de chiffrement, de l'encodage de la voix, ...

Dans le fichier `/etc/asterisk/asterisk.conf`, supprimez le commentaire (`!`) qui figure à côté de `[directories]`.

/etc/asterisk/asterisk.conf

```
[directories]
astetcdir => /etc/asterisk
astmoddir => /usr/lib/asterisk/modules
astvarlibdir => /usr/lib/asterisk
astdbdir => /usr/lib/asterisk
astkeydir => /usr/lib/asterisk
astdatadir => /usr/lib/asterisk
astagidir => /usr/lib/asterisk/agi-bin
astspooldir => /var/spool/asterisk
astrundir => /var/run/asterisk
astlogdir => /var/log/asterisk
```

Démarrez Asterisk et activez le au prochain démarrage:

```
root@MakerFaire:~# /etc/init.d/asterisk start
root@MakerFaire:~# /etc/init.d/asterisk enable
```



Le journal de messages d'Asterisk se trouve dans le fichier /var/log/asterisk/messages.



Consultez ce fichier pour identifier tout problème de démarrage ou de fonctionnement de votre serveur téléphonique.

Le serveur Asterisk que vous venez d'installer permet aussi la visioconférence. L'activation de la visioconférence est très simple: dans le fichier `/etc/asterisk/sip.conf` supprimez le commentaire devant `;videosupport=yes`.

/etc/asterisk/sip.conf

```
videosupport=yes                ; Turn on support for SIP video. You
need to turn this
                                ; on in this section to get any video
support at all.
                                ; You can turn it off on a per peer
basis if the general
                                ; video support is enabled, but you
can't enable it for
                                ; one peer only without enabling in the
general section.
                                ; If you set videosupport to "always",
then RTP ports will
                                ; always be set up for video, even on
clients that don't
                                ; support it.  This assists callfile-
derived calls and
                                ; certain transferred calls to use
always use video when
                                ; available. [yes|NO|always]
```

L'utilisation de la visioconférence est un peu plus subtile. Comme Asterisk ne supporte pas le transcodage (cf. https://wiki.asterisk.org/wiki/display/AST/Video+Telephony), il est donc indispensable d'utiliser le même encodage aux deux bouts de la communication. A titre d'exemple, vous pouvez établir une visioconférence entre:

- Le logiciel Linphone avec l'encodage H263-1998 sur Mac OS X.
- Le logiciel Antisip avec le même encodage H263-1998 sur Android.

Cliquez pour visualiser la partie optionnelle

Cette partie est optionnelle et permet de personnaliser le service `echo` joignable en composant le 600. Pour cela, commencez par télécharger le paquet correspondant:

```
root@MakerFaire:~# opkg install asterisk18-sounds
```

Dans le fichier `/etc/asterisk/extensions.conf`, insérer les messages vocaux du service `echo` en début et en fin de l'appel.

/etc/asterisk/extensions.conf

```
exten => 600,1,Playback(demo-echotest)
exten => 600,2,Echo ; Do the echo test
exten => 600,3,Playback(demo-echodone)
```

Et si vous souhaitez que ces messages vocaux soient en français, vous pouvez télécharger les fichiers correspondants et suivre la procédure ci-dessous:

```
root@MakerFaire:~# cd /usr/lib/asterisk/sounds
root@MakerFaire:~# mkdir fr
root@MakerFaire:~# cd fr
root@MakerFaire:~# wget
http://downloads.asterisk.org/pub/telephony/sounds/asterisk-core-sounds-fr-gsm-current.tar.gz
root@MakerFaire:~# tar zxvf asterisk-core-sounds-fr-gsm-current.tar.gz
```

Dans ce cas, le répertoire `fr` doit être ajouté dans le fichier `/etc/asterisk/extensions.conf`:

/etc/asterisk/extensions.conf

```
exten => 600,1,Playback(fr/demo-echotest)
exten => 600,2,Echo
exten => 600,3,Playback(fr/demo-echodone)
```

Redmarrez Asterisk en saisissant `/etc/init.d/asterisk restart`, appelez le 600 et testez le service.



La commande scp peut être utile pour transférer des fichiers depuis votre ordinateur vers le routeur MR3020. Par exemple, `scp asterisk-core-sounds-fr-gsm-current.tar.gz root@192.168.1.1:/usr/lib/asterisk/sounds/fr` permet de transférer les fichiers sons pour asterisk que vous avez téléchargé sur votre PC au préalable.

Pour établir des communications téléphoniques avec le serveur téléphonique Asterisk que vous venez d'installer sur le routeur MR3020, deux types de terminaux peuvent être utilisés: des téléphones IP (par exemple les téléphones AASTRA 6757i), ou des téléphones logiciels disponibles pratiquement pour tout type de système d'exploitation (une liste non exhaustive est disponible sur wikipedia).

La configuration d'un softphone est très simple: il suffit de renseigner le nom de l'utilisateur, son mot de passe, et l'adresse du serveur Asterisk, généralement appelé domaine. La figure 4 présente une capture d'écran de la configuration du logiciel X-Lite sous Mac OS X.

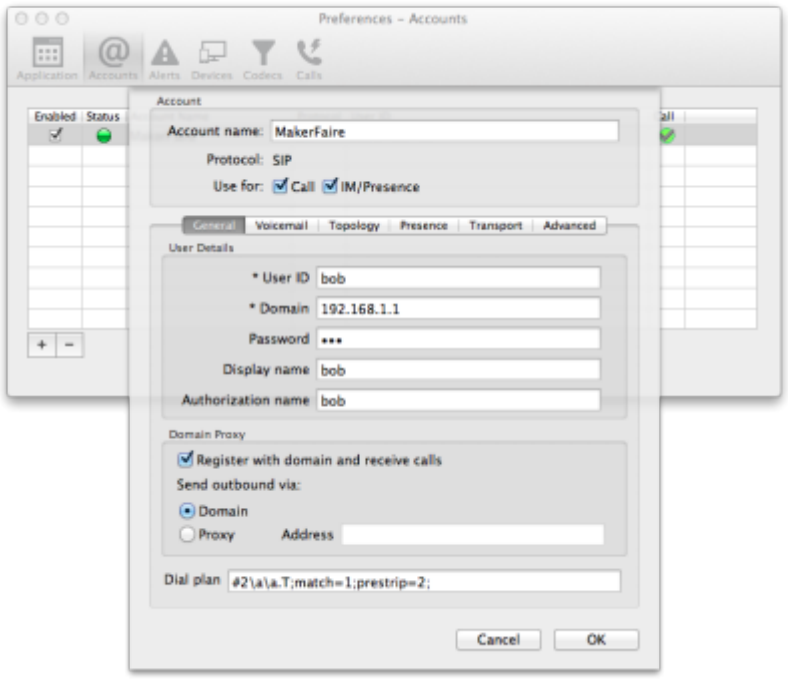


Figure 4. Capture d'écran de X-Lite sous Mac OS X

Un serveur multimédia vous permet de partager sur le réseau de votre MakerBox vos photos, musiques, ou vidéos. Un grand nombre de clients existent pour consulter le contenu multimédia diffusé par votre serveur (cf. liste de clients UPnP par Wikipedia). Commencez donc par installer un serveur multimédia sur le MR3020. La distribution choisie est miniDLNA:

```
root@MakerFaire:~# opkg install minidlna
root@MakerFaire:~# opkg install luci-app-minidlna
```

L'installation de l'interface LuCI est optionnelle mais vous permettra de configurer votre serveur multimédia via une interface graphique simplifiée. Créez maintenant des répertoires pour stocker le contenu multimédia.

```
root@MakerFaire:~# cd /mnt/data
root@MakerFaire:~# mkdir music
root@MakerFaire:~# mkdir video
root@MakerFaire:~# mkdir picture
```

Modifiez le fichier de configuration de miniDLNA pour prendre en compte les répertoires de stockage déjà créés:

/etc/config/minidlna

```
config minidlna config
        option 'enabled' '1'
        option port '8200'
        option interface 'br-lan'
        option friendly_name 'MakerFaire JukeBox'
        option db_dir '/var/run/minidlna'
        option log_dir '/var/log'
        option inotify '1'
        option enable_tivo '0'
```

```
        option strict_dlna '0'
        option presentation_url ''
        option notify_interval '900'
        option serial '12345678'
        option model_number '1'
        option root_container '.'
        list media_dir 'A,/mnt/data/music'
        list media_dir 'P,/mnt/data/picture'
        list media_dir 'V,/mnt/data/video'
        option album_art_names
'Cover.jpg/cover.jpg/AlbumArtSmall.jpg/albumartsmall.jpg/AlbumArt.jpg/a
lbumart.jpg/Album.jpg/album.jpg
/Folder.jpg/folder.jpg/Thumb.jpg/thumb.jpg'
```

Enrichissez maintenant votre bibliothèque multimédia avec de la musique ou des vidéos libres de droits (par exemple des films réalisés avec Blender ou de la musique sous licence CC du Free Music Archive). Pour une démonstration placez vous dans le répertoire `/mnt/data/video` et saisissez la commande suivante pour télécharger la fameuse vidéo du Big Buck Bunny:

```
root@MakerFaire:~# wget
http://download.blender.org/peach/bigbuckbunny_movies/BigBuckBunny_320x180.m
p4
```



Pour relancer l'indexage de votre bibliothèque multimédia, vous pouvez supprimer la base de données en saisissant: `rm /var/run/mindlna/file.db`. Pour une solution pérenne, modifiez le fichier `/etc/init.d/minidlna` et ajoutez l'option `-R` au démarrage de miniDLNA: `service_start /usr/bin/minidlna -R -f “$MINIDLNA_CONFIG_FILE”`. Un simple redémarrage `/etc/init.d/minidlna restart` mettra à jour votre base de données multimédia.

Pour profiter de votre contenu multimédia partagé sur le réseau, il suffit de lancer le logiciel VLC sur l'un des ordinateurs connectés au même réseau. La MakerBox apparaît automatiquement dans l'onglet `réseau local` de votre liste de lecture. Sur Android, vous pouvez opter pour des logiciels comme MediaHouse ou Pixel Media controller.

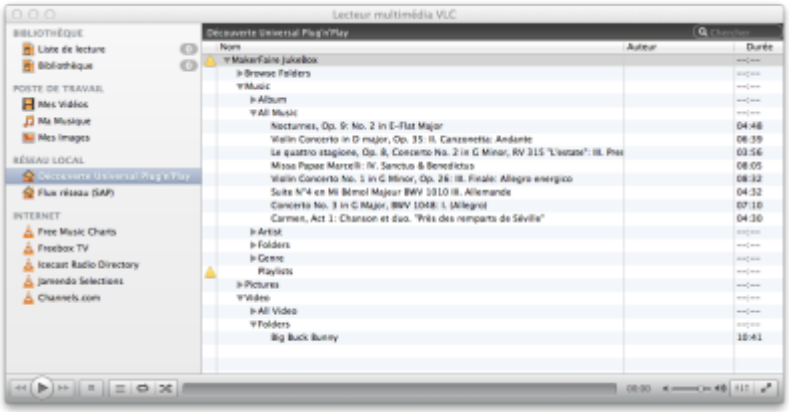


Figure 5. MakerBox dans une liste de lecture VLC

From:
http://wiki.lahoud.fr/ - **wikiroute**

Permanent link:
http://wiki.lahoud.fr/doku.php?id=makerfaire&rev=1428099245

Last update: **2015/04/04 00:14**