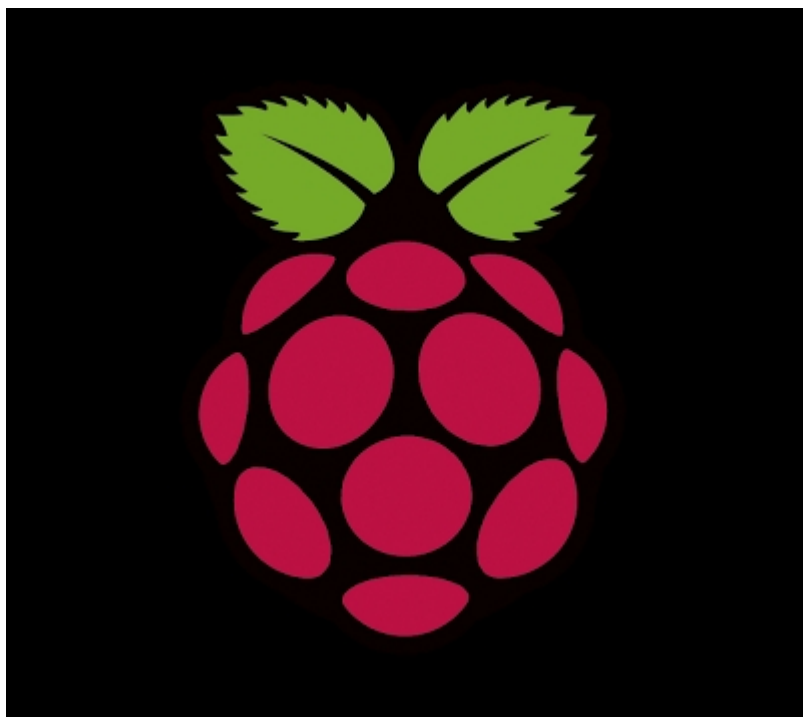


Aujourd'hui nous allons installer un petit Mediacenter sur un Raspberry-Pi le tout piloté par une Nintendo WiiMote. Sympa non ?



Ce Mediacenter va s'intégrer à mon installation existante qui est axée sur un serveur Plex hébergé sous Linux.

Ce Raspberry-Pi va simplement me servir de client et les films et séries seront décodés sur le serveur, je ne vais donc pas réellement tester le Raspberry-Pi en tant que Mediacenter, cependant il y a de très nombreux retours sur internet affirmant que ce petit boîtier à 35€ n'a pas de problème de performance (après rédaction de cet article, j'ai eu l'occasion de l'essayer grâce à un montage NFS et il fonctionne très bien, quelle que soit la qualité de la vidéo).

Raspberry, XBMC et Domotique

Écrit par ElkAin

Vendredi, 29 Mars 2013 07:00 - Mis à jour Jeudi, 28 Mars 2013 23:06

Mon installation sera donc simple et sans stockage, mais il suffit d'y ajouter un disque ou une clé USB et de les déclarer dans la configuration pour le rendre autonome. Vous pouvez également partager un dossier sur un Serveur/NAS, un PC de bureau, votre FreePlayer, ... ou tout autre matériel hébergeant vos données.

Je ne souhaite pas utiliser ce montage pendant très longtemps, uniquement le temps de réinstaller mon HTPC, les configurations seront donc très basiques mais de toutes façons, chaque configuration est différente selon les utilisations et il y a une très grande quantité de sites internet parlant de la configuration de XBMC.

Je vais également, en plus de l'installation du Raspi et de la configuration de base de XBMC, vous montrer comment le contrôler grâce à une WiiMote. Je choisis ce contrôleur car contrairement à l'Infra-Rouge habituellement utilisé dans nos télécommandes, la WiiMote n'a pas besoin de viser le Raspi, il peut donc être cachée derrière la TV ou dans un meuble.



Le Matériel

- Un [Raspberry-Pi](#) : 33€
- Une [Carte SDHC](#) de plus de 8Go pour être tranquille: 26€ (pour 32Go si vous voulez être vraiment-vraiment tranquille ;-)
- Un Dongle Bluetooth USB: ~10€ attention qu'il soit bien compatible avec linux. Pour mes tests j'utilise un vieux D-Link DBT-120
- Une [WiiMote](#) : 20€
- Une [clé Wifi USB](#) (optionnel): 15€, pour ma part j'ai celui-là, mais il existe plus petit si la taille

Raspberry, XBMC et Domotique

Écrit par ElkAin

Vendredi, 29 Mars 2013 07:00 - Mis à jour Jeudi, 28 Mars 2013 23:06

est importante pour vous.

Pour un budget total maximum de 100€ vous pouvez donc vous offrir un chouette petit mediacenter.



Pendant les tests et lors de la rédaction de cet article, j'ai pu constater que le protocole CEC qui permet de passer les informations de la télécommande de la TV (par exemple) par le câble

HDMI et ainsi piloter le Raspberry-Pi, marche en natif sur RaspBMC.

Pour utiliser cette fonction, et donc se passer de la WiiMote ou de toute autre télécommande mis à part celle de la TV, il faut que tous les éléments d'un bout à l'autre de la chaîne soient compatibles CEC (inclus depuis le standard HDMI 1.3a). Le Raspberry-Pi est compatible, il vous reste donc à vérifier la TV, la plupart du temps, les TV utilisent les standards HDMI 1.4 donc cela ne devrait pas poser de problème. L'élément que l'on néglige habituellement est le câble : si cela ne fonctionne pas du premier coup essayez avec un câble plus récent ou moins bon marché.

J'ai également constaté que le Raspberry-pi peut-être alimenté par tous les ports USB, donc si vous utilisez un disque dur avec une alimentation à part (la plupart des 3,5" le sont) vous n'aurez certainement pas besoin de brancher le mini-USB. C'est une bonne nouvelle si vous avez un encombrement ou un nombre de prises limité.

Ces deux petites remarques faites, passons au vif du sujet.

Installation du Raspberry



Raspberry, XBMC et Domotique

Écrit par ElkAin

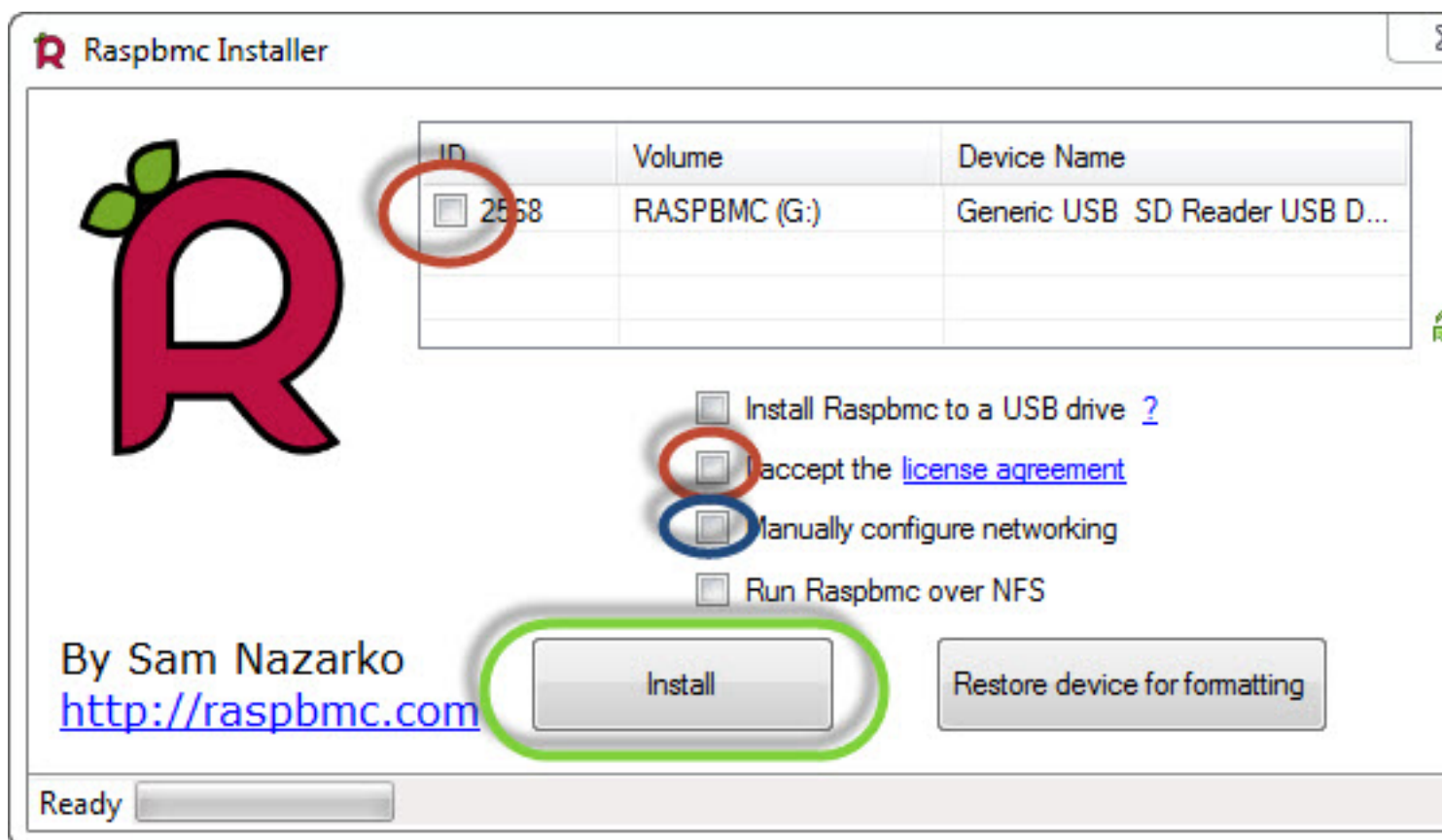
Vendredi, 29 Mars 2013 07:00 - Mis à jour Jeudi, 28 Mars 2013 23:06

Je vais installer le Raspberry-Pi avec la distribution RaspBMC car elle est très simple à déployer, basée sur une Debian et qu'elle fait son office.

Si vous êtes sous Windows, commencez donc par télécharger cette [archive](#) sinon rendez-vous à cette adresse pour les versions Linux, Mac ou tout type d'installation: <http://www.raspbmc.com/download/>

Pendant le téléchargement vous pouvez formater votre carte SD en Fat32. Et dès la réception de l'archive, je vous invite à l'extraire et à exécuter le fichier « installer.exe » qu'elle contient.

Après un message d'avertissement, vous vous retrouvez face à la fenêtre suivante :



Les 2 champs entourés de rouge doivent être cochés. Si vous avez plusieurs périphériques USB connectés, c'est le moment de choisir le bon et de ne pas se tromper sous peine de formater le mauvais volume.

Vous pouvez configurer le réseau dès maintenant en cochant la case bleue avant de valider par la case verte "Install".

Si vous avez coché la case bleue, vous tomberez sur la fenêtre suivante. J'ai renseigné les champs suivants car ils correspondent à mon réseau et car j'aime savoir quel périphérique porte quelle adresse IP.

En fixant cette adresse IP, vous saurez plus facilement à quelle adresse vous connecter par la suite, je vous invite donc à la choisir maintenant.

Network Configuration

Please select network type: **Wireless**

Obtain an IP address automatically

IP Address: 192.168.0.10

Subnet Mask: 255.255.255.0

Gateway IP: 192.168..0.254

DNS Server: 192.168.0.254

SSID: ElkAin2

Encryption: WPA/WPA2

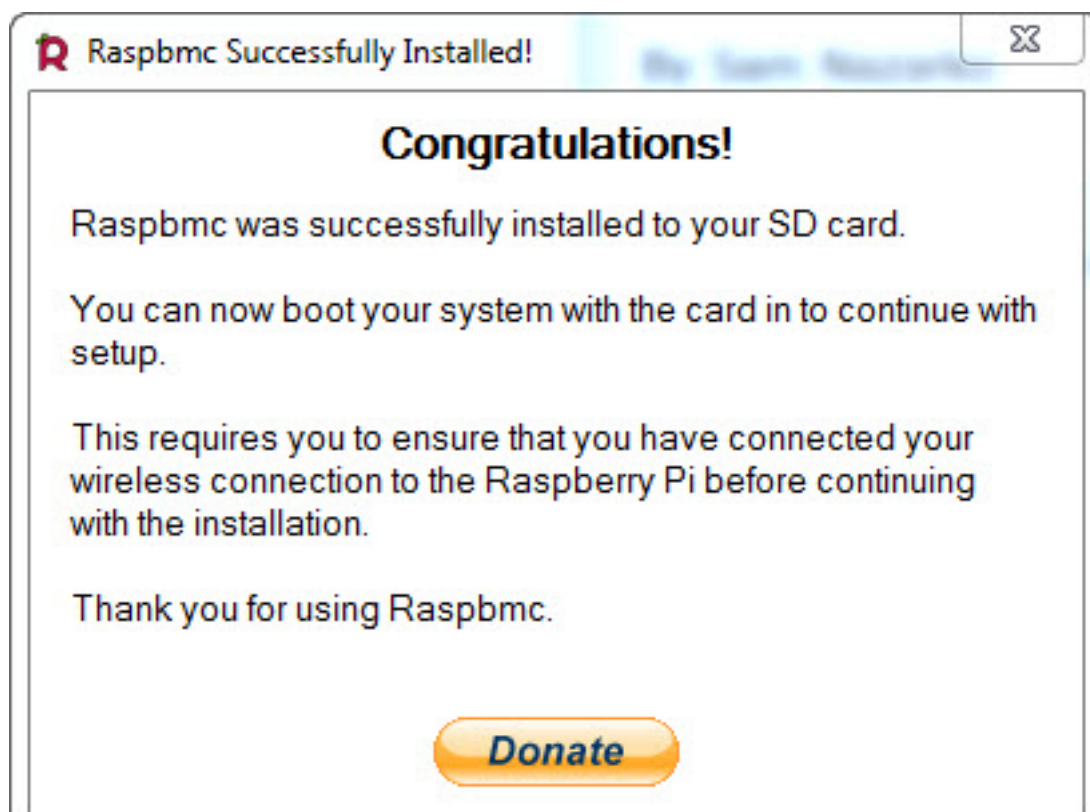
Passkey: [REDACTED]

Ad-hoc network

Use 5Ghz only

Save

Vous pouvez télécharger cette fenêtre et l'installer sur votre ordinateur. Après un petit temps d'attente, vous pouvez télécharger cette fenêtre et l'installer sur votre ordinateur.



Configuration

Je n'ai donc ni écran ni clavier à y relier et vais ainsi utiliser une connexion SSH, si vous travaillez directement sur un clavier/écran relié au Raspi, vous pouvez passer le paragraphe suivant.

Particularité du SSH (Optionnel)

C'est à ce moment que l'IP que nous avons précisé en préparant la carte SD va nous servir.

Si vous ne connaissez pas l'adresse du Raspi (DHCP ?) regardez la configuration de votre routeur si vous en avez un routeur suffisamment manageable. Sinon, vous pouvez aussi utiliser des outils tels que « Nmap » (« `nmap -sP 192.168.0.0/24` » pour un simple scan ping).

La dernière solution est d'utiliser un clavier et un écran et une fois connecté (login : pi, mot de

Raspberry, XBMC et Domotique

Écrit par ElkAin

Vendredi, 29 Mars 2013 07:00 - Mis à jour Jeudi, 28 Mars 2013 23:06

passer : raspberry) d'utiliser la commande « ifconfig » et recherchez l'adresse IP.

Si vous êtes sous Windows, vous pouvez vous connecter au Raspi en utilisant des logiciels comme "putty", "cygwin" ou "MobaXterm". Je recommande ce dernier car il nous permettra d'utiliser l'interface graphique du raspberry-pi dans une autre étape. Si vous utilisez Linux ou Mac vous devez déjà savoir comment atteindre la console.

Une fois sur la console ou sur MobaXterm, vous pouvez rentrer les commandes suivantes.

Dans mon cas 192.168.0.10 est l'adresse IP de mon Raspi.

Par défaut le login est "pi" et le mot de passe est "raspberry"

```
ssh 192.168.0.10 -l pi
```

Une fois connecté, vous verrez les écrans suivants qui permettent de choisir le clavier et la zone géographique (je pense pour le fuseau horaire).

Écrit par ElkAin

Vendredi, 29 Mars 2013 07:00 - Mis à jour Jeudi, 28 Mars 2013 23:06



Package configuration

Configuring locales

Many packages in Debian use locales to display text in the correct language for the user. You can choose a default locale for the system from the generated locales. This will select the default language for the entire system. If this system is a multi-user system where not all users are able to speak the default language, you may experience difficulties.

Default locale for the system environment:

None
 fr_FR@euro

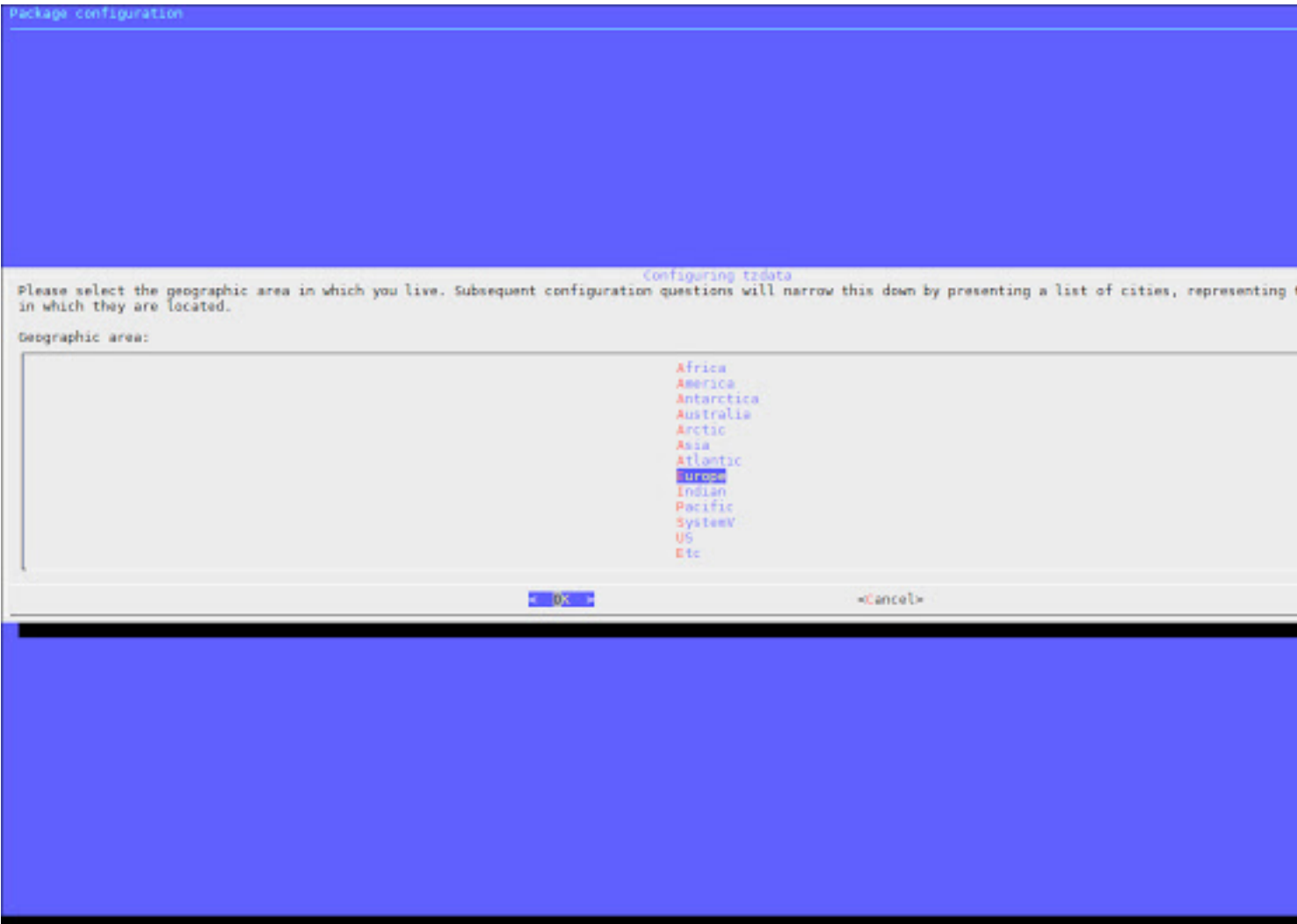
< OK >

<Cancel>

Raspberry, XBMC et Domotique

Écrit par ElkAin

Vendredi, 29 Mars 2013 07:00 - Mis à jour Jeudi, 28 Mars 2013 23:06





Installation de la WiiMote

Pour cette étape, il vous faut impérativement un dongle Bluetooth connecté et une WiiMote à portée de main.

Commençons par installer les prérequis.

```
sudo apt-get update
sudo apt-get install bluez-compat
sudo apt-get install libbluetooth-dev g++ libcwiid1 xbmc-eventclients-common make git-core
bluetooth
```

Raspberry, XBMC et Domotique

Écrit par ElkAin

Vendredi, 29 Mars 2013 07:00 - Mis à jour Jeudi, 28 Mars 2013 23:06

Cette étape peut-être (et devrait être) très longue, pour ma part elle a duré près de 20 minutes.

Les étapes suivantes vont préparer la compilation à venir.

```
sudo mkdir /usr/local/src/xbmc
sudo git clone git://github.com/xbmc/xbmc.git /usr/local/src/xbmc
```

Une nouvelle étape un peu longue.

Maintenant il nous faut travailler en modifiant le fichier
"/usr/local/src/xbmc/tools/EventClients/Clients/WiiRemote/Makefile"

En ajoutant "-l bluetooth" (L minuscule et non i majuscule) à la fin de la ligne contenant
"\$\$(OBJS) -o \$(BIN)".

Ce qui nous donne la ligne suivante :

```
$(CXX) $(CFLAGS) -I./wiiuse_$(VERSION)/src WiiUse_WiiRemote.cpp $(OBJS) -o $(BIN) -l  
bluetooth
```

Maintenant nous pouvons lancer la compilation :

```
cd /usr/local/src/xbmc/tools/EventClients/Clients/WiiRemote/      sudo make
```

Encore une petite attente mais, rassurez-vous, c'est l'une des dernières.

Avant de continuer, il nous faut trouver l'adresse MAC de la WiiMote afin de la connecter plus

Écrit par ElkAin

Vendredi, 29 Mars 2013 07:00 - Mis à jour Jeudi, 28 Mars 2013 23:06

rapidement par la suite. L'adresse MAC correspond à la partie surlignée renvoyé par la commande suivante.

```
sudo /etc/init.d/bluetooth start
sudo hcitool scan
Scanning ...
00:1B:EA:00:81:1A Nintendo RVL-CNT-01
```

Maintenant, nous allons créer un script qui sera lancé en même temps que XBMC et qui permettra que la WiiMote soit reconnue automatiquement.

```
sudo mkdir /home/pi/wii
sudo vi /home/pi/wii/wii.sh
sudo chmod a+x /home/pi/wii/wii.sh
```

Dans la dernière commande rentrée, j'utilise « vi » par habitude mais vous pouvez utiliser tout autre éditeur de texte : nano, gedit, ... L'important et juste de créer ce fichier avec le contenu suivant :

```
#!/sbin/sh
/etc/init.d/bluetooth restart
sleep 2
# WiiRemote should be launch in the same directory
cd /usr/local/src/xbmc/tools/EventClients/Clients/WiiRemote
./WiiUse_WiiRemote --btaddr XX:XX:XX:XX:XX:XX
```

Pensez bien à remplacer "XX:XX:XX:XX:XX:XX" par votre valeur, dans mon cas "00:1B:EA:00:81:1A"

Donnons maintenant l'ordre à XBMC de lancer ce script au démarrage, pour ce faire nous devons éditer le fichier "/scripts/xbmc-watchdog.sh".


```
# MODIFIED TO GET THE WIIMOTE WORKING      sh /home/pi/wii/wii.sh &
```

Pour ajouter les lignes précédentes entre les lignes "date -u +%s > /var/run/splash-timestamp.." et "if [-f /usr/lib/libcofi_rpi.so]; then"

La section finale doit donc être :

```
#!/bin/bash      [...]      test "x$xbmc_libs" != "x" || ldconfig      setcap
cap_net_admin,cap_net_bind_service,cap_net_raw=ep
/home/pi/.xbmc-current/xbmc-bin/lib/xbmc/xbmc.bin      date -u +%s >
/var/run/splash-timestamp      # MODIFIED TO GET THE WIIMOTE WORKING      sh
/home/pi/wii/wii.sh &      if [ -f /usr/lib/libcofi_rpi.so ]; then      su - pi -c "export
XBMC_HOME=$(readlink /home/pi/.xbmc-current)/xbmc-bin/share/xbmc' ;
LD_PRELOAD=/usr/lib/libcofi_rpi.so /home/pi/.xbmc-current/xbmc-bin/lib/xbmc/xbmc.bin
--standalone -fs --lircdev /var/run/lirc/lircd 2>&1 | logger -t xbmc"      else      su - pi -c
"export XBMC_HOME=$(readlink /home/pi/.xbmc-current)/xbmc-bin/share/xbmc' ;
/home/pi/.xbmc-current/xbmc-bin/lib/xbmc/xbmc.bin --standalone -fs --lircdev /var/run/lirc/lircd
2>&1 | logger -t xbmc"      fi      [...]
done
```

```
{jumi [*34]}
```

ATTENTION : Sur mon installation, j'ai constaté d'énormes ralentissements qui, au final étaient dûes à la WiiMote. En effet, un processus consomme toute la mémoire disponible ce qui ralentit grandement le Raspberry-pi et m'a donné une première impression du matériel vraiment négative alors qu'il n'y a au final aucune raison.

Raspberry, XBMC et Domotique

Écrit par ElkAin

Vendredi, 29 Mars 2013 07:00 - Mis à jour Jeudi, 28 Mars 2013 23:06

Je n'ai pas cherché à résoudre ce problème mais je ne pense que cela doit être faisable. Si vous rencontrez le même problème et avez besoin de la WiiMote, n'hésitez pas à me contacter afin que je me penche sur le sujet.

Nous pouvons, après avoir sauvegardé ce fichier, redémarrer le Raspberry-pi.

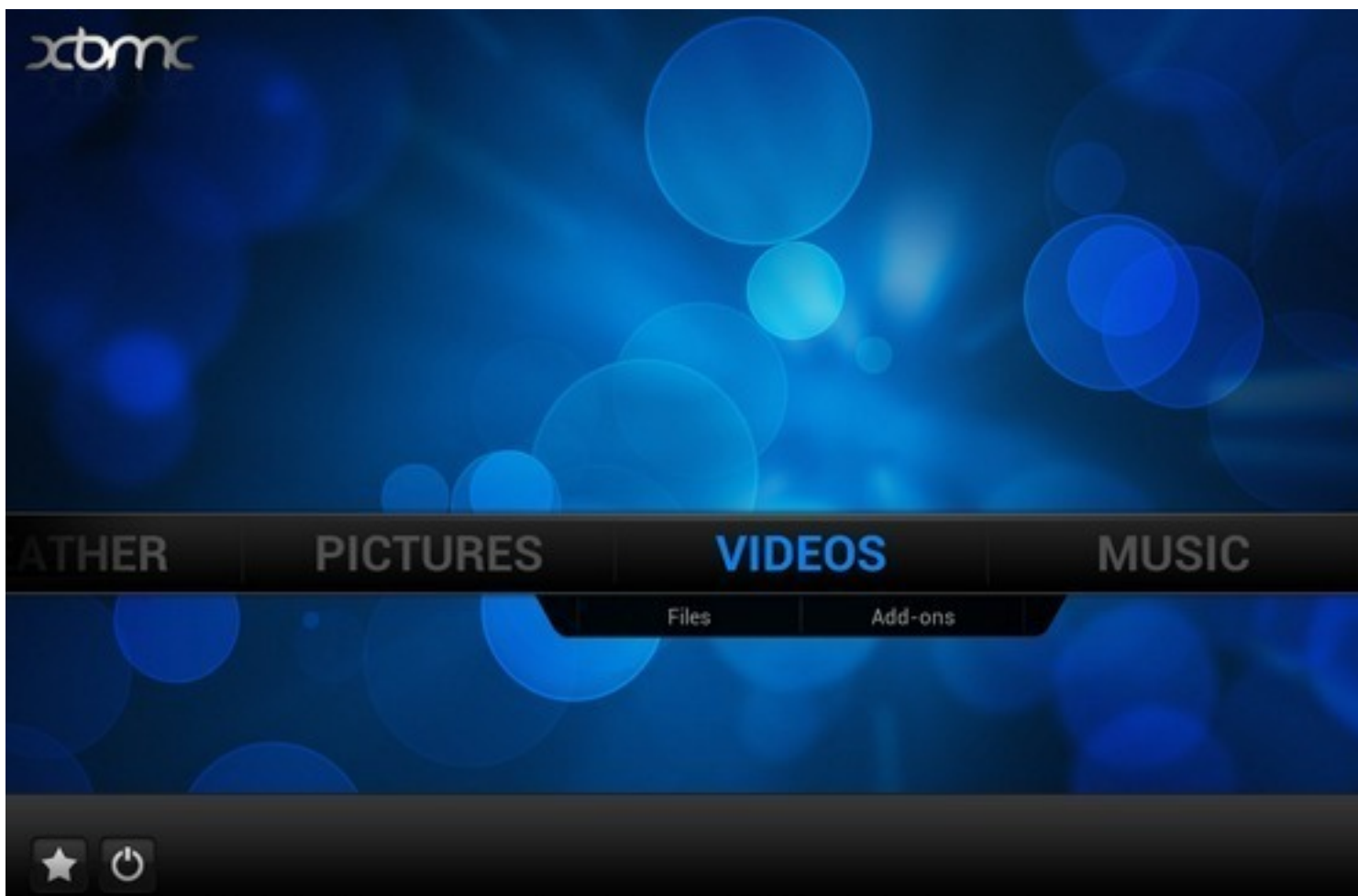
```
sudo reboot
```

Si vous n'avez pas encore relié le Raspberry-pi à votre TV c'est le moment de le faire.

Une fois le RasPi redémarré, vous pouvez allumer la WiiMote en appuyant sur les boutons 1 et 2 et admirer le résultat.

ATTENTION : J'ai eu quelques surprises au niveau de la résolution en branchant la TV après le démarrage du Raspi.

Vous voilà l'heureux propriétaire d'un Serveur XBMC.



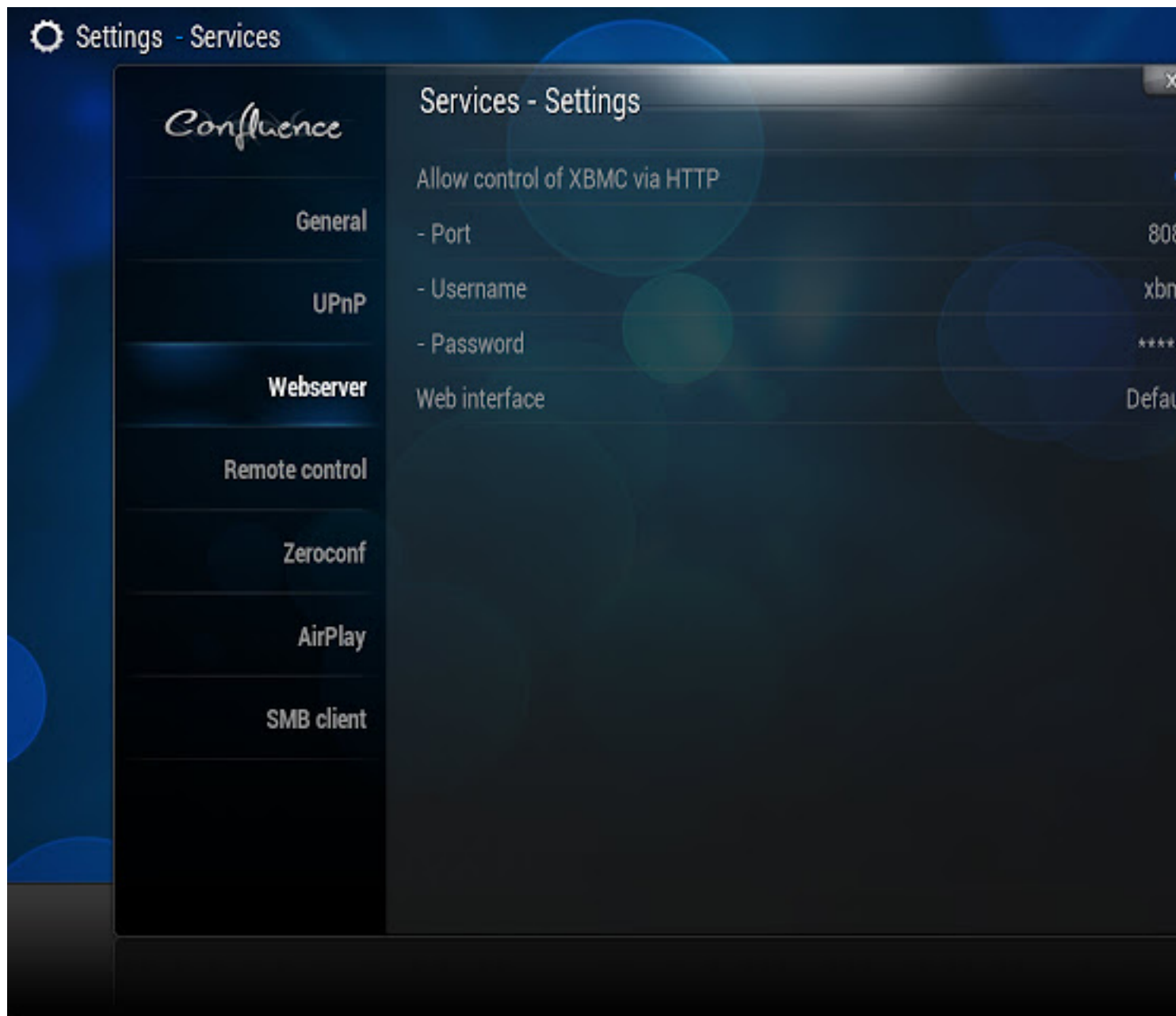
La Domotique et XBMC :

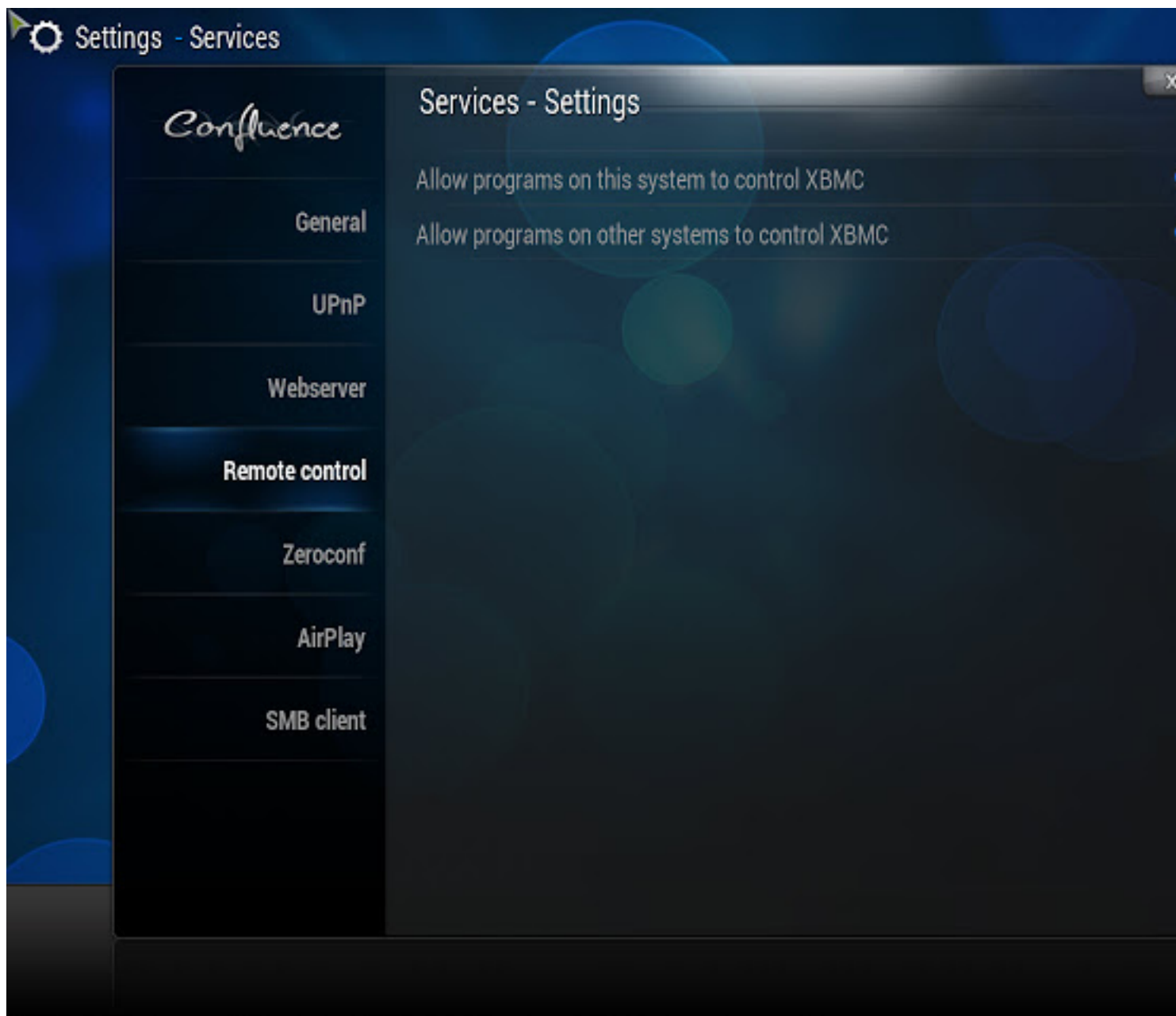
Je vais essayer de mettre suffisamment d'exemples pour répondre à tous vos besoins et envies.

Dans tous les cas, n'hésitez pas à réclamer si j'oublie quelque chose.

Avant tout, vous devez configurer le serveur web et la télécommande HTTP de XBMC. Allez donc dans « System » -> « Services » et configurez les 2 menus comme suit.

Les champs login et password sont facultatifs mais conseillés car sans eux, n'importe qui pourra interagir avec votre système.





my applications in a more elegant way. It will be a nice addition to the standard examples that you can find in the XBMC documentation.

Afficher des notifications

Affiche un message qui s'intégrera à votre thème que vous soyez en train de regarder une vidéo/images, écouter de la musique, ou même si vous êtes simplement sur l'écran d'accueil ou un quelconque menu.

Cette fonction est utile pour afficher toute notification de votre centrale domotique, vos e-mails, un appel téléphonique, ...

Raspberry, XBMC et Domotique

Écrit par ElkAin

Vendredi, 29 Mars 2013 07:00 - Mis à jour Jeudi, 28 Mars 2013 23:06

Depuis une ligne de commande Linux/Unix :

```
curl -v -H "Accept: application/json" -H "Content-type: application/json" -X POST -d
'{"id":1,"jsonrpc":"2.0","method":"GUI.ShowNotification","params":{"title":"Titre_du_message","m
essage":"Corps_Du_Message"}}' http://xbmc:secret@192.168.0.10:8080/jsonrpc
```

En envoyant une requête HTTP GET:

```
http://xbmc:secret@192.168.0.10:8080/jsonrpc?request={"id":1,"jsonrpc":"2.0","method":"GUI.S
howNotification","params":{"title":" Titre_du_message ","message":" Corps_Du_Message "}}
```

Dans cet exemple les parties à changer sont :

- Titre_du_message = Le texte qui apparaîtra dans la barre de titre de la notification. Il peut y avoir des problèmes sur certains thèmes de XBMC.
- Corps_Du_Message = Le texte en corps de message.
- 192.168.0.10 = L'adresse IP de mon RaspBMC
- 8080 = Le port http configuré dans les paramètres système de la télécommande http.
- xbmc = login configuré dans le paragraphe webserver
- secret = mot de passe configuré au même endroit

ATTENTION : Pour les caractères spéciaux de vos messages, il faut utiliser le code ASCII Hexadécimal précédé par « u00 », par exemple :
u0027 pour l'apostrophe

Hex	Caractère	Detail
-----	-----------	--------

--- -----

27	'	Apostrophe
----	---	------------

2F	/	Slash
----	---	-------

5C Anti-Slash

Pour faire un retour à la ligne dans vos messages il faut mettre « n » en lieu et place du retour.

Faire Play/Pause

Pour vos différents scénarios, la fonction suivante va sûrement vous être utile car elle permet de mettre en pause ce que vous regardez/écoutez ou de relancer la lecture.

Les seules choses à changer sont « xbmc » et « secret » en plus de certains changements de l'étape précédente

```
curl -v -H "Accept: application/json" -H "Content-type: application/json" -X POST -d
'{"id":1,"jsonrpc": "2.0", "method": "Player.PlayPause", "params": { "playerid": 0 }}' http://
xbmc:secret@192.168.0.10:8080/jsonrpc
```

```
http://xbmc:secret@192.168.0.10:8080/jsonrpc?request={"id":1,"jsonrpc":"2.0","method":"Player
.PlayPause","params":{"playerid": 1 }}
```

Affichage flux camera IP

Nous allons afficher le flux venant d'une caméra IP comme une vidéo classique.

Les paramètres sont les mêmes que d'habitude avec en plus :

- Adresse_CamIP = lien permettant l'accès à la webcam précédé de elkain:testelkain pour les logins et mot de passe.

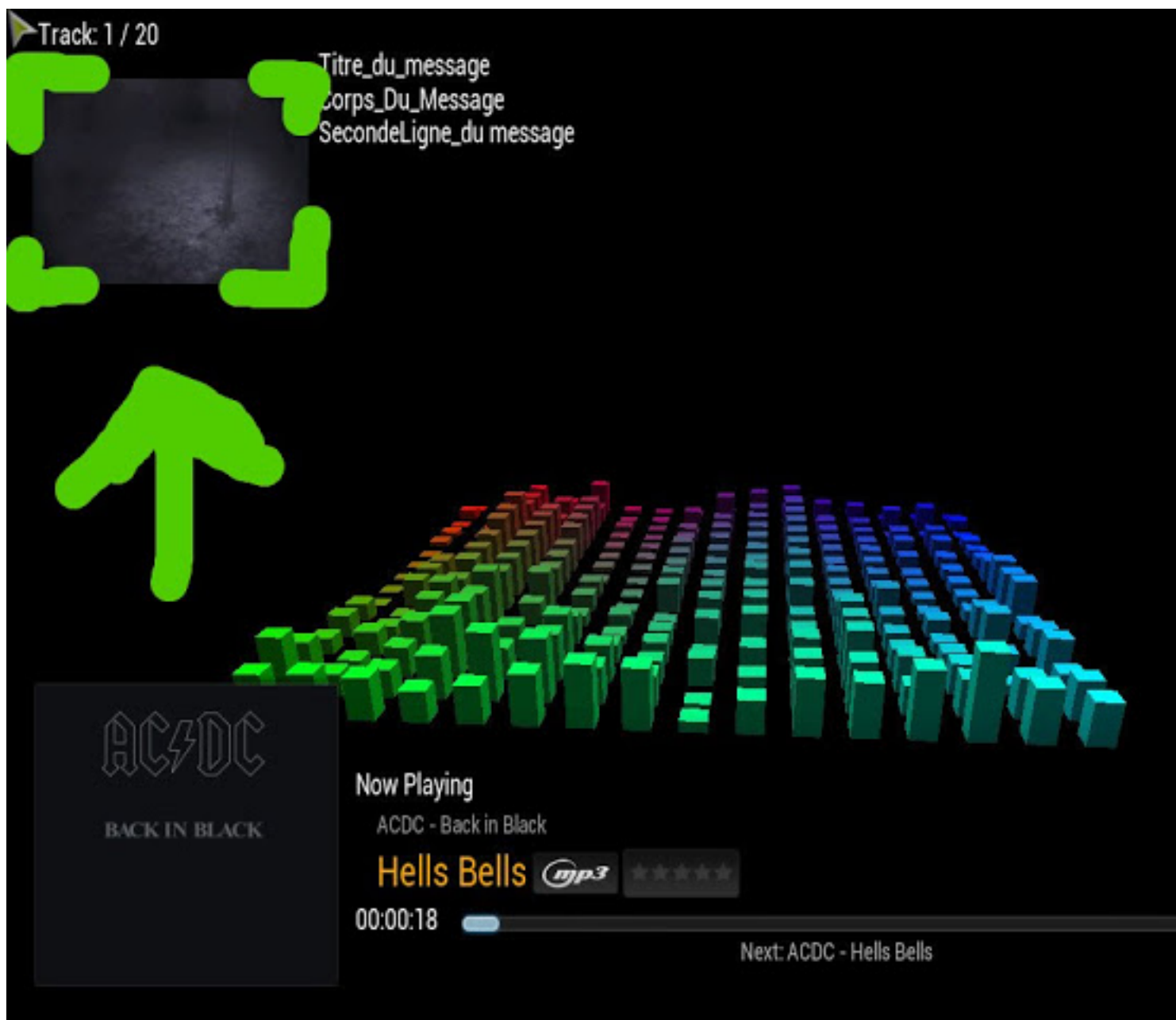
Raspberry, XBMC et Domotique

Écrit par ElkAin

Vendredi, 29 Mars 2013 07:00 - Mis à jour Jeudi, 28 Mars 2013 23:06

```
curl -v -H "Accept: application/json" -H "Content-type: application/json" -X POST -d  
'{"id":1,"jsonrpc":"2.0","method":"Player.Open","params":{"item":{"file": "http://elkain:testelkain@  
Adresse_CamIP/videostream.cgi}}}' http://xbmc:secret@192.168.0.10:8080/jsonrpc
```

```
http://xbmc:secret@192.168.0.10:8080/jsonrpc?request={"id":1,"jsonrpc":"2.0","method":"Player  
.Open","params":{"item":{"file": "http://elkain:testelkain@Adresse_CamIP/videostream.cgi}}}
```



Affiche photo

Nous allons maintenant afficher une photo avec un titre et un court texte (comme les notifications mais avec une grande photo). Le but étant de pouvoir vous permettre d'afficher pendant l'utilisation du mediacenter une photo de la personne qui sonne à votre porte.

Pour cela il nous faut créer un script qui sera appelé par notre commande.

Sur le serveur vous devez créer un dossier contenant 2 fichiers.

```
mkdir ~/.xbmc/addons/script.picpopup
touch ~/.xbmc/addons/script.picpopup/default.py
touch ~/.xbmc/addons/script.picpopup/addon.xml
```

Le fichier default.py doit contenir le texte suivant:

```
import xbmc
import xbmcgui
import sys
import urlparse class PopupWindow(xbmcgui.WindowDialog):
    def __init__(self, image, title, message):
        self.addControl(xbmcgui.ControlImage(x=25, y=25, width=260, height=195,
        filename=image[0]))
        self.addControl(xbmcgui.ControlLabel(x=295, y=25, width=500, height=25, label=title[0]))
        self.addControl(xbmcgui.ControlLabel(x=295, y=50, width=500, height=25, label=message[0]))
    if __name__ == '__main__':
        params = urlparse.parse_qs('&'.join(sys.argv[1:]))
        window = PopupWindow(**params)
        window.show()
        xbmc.sleep(10000)
        window.close()
    del window
```

Quand à addon.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<addon provider-name="ElkAiN" version="1.0" name="Pictures Pop-Up Script"
id="script.picpopup">
<requires>
<import version="2.1.0" addon="xbmc.python"/>
</requires>
<extension library="default.py" point="xbmc.python.script"/>
<extension library="default.py" point="xbmc.python.library"/>
<extension point="xbmc.addon.metadata">
<language/>
<summary>Pictures pop-up script</summary>
<description>Display a popup with an image, a title and a quick message.</description>
<disclaimer>Needs skin support for it to be used properly</disclaimer>
<platform>all</platform>
</extension>
</addon>
```

Je ne vais pas détailler ces 2 fichiers car ce n'est pas le sujet de cet article. Sachez simplement que en modifiant la valeur de « xbmc.sleep(10000) » vous modifiez le temps pendant lequel la fenêtre restera affiché (ici 10 secondes).

Vous devez maintenant redémarrer XBMC et configurer votre sonnette pour envoyer l'une des 2 trames suivante :

```
curl -v -H "Accept: application/json" -H "Content-type: application/json" -X POST -d
'{"id":1,"jsonrpc":"2.0","method":"Addons.ExecuteAddon","params":{"addonid":"script.picpopup",
"params":{"title":"Titre_du_message","message":"Corps_Du_MessagenSecondeLigne_du
message","image":"http://elkain:testelkain@Adresse_CamIP/snapshot.cgi}}}'
http://xbmc:secret@192.168.0.10:8080/jsonrpc
```

```
http://192.168.0.10:8080/jsonrpc?request={"id":1,"jsonrpc":"2.0","method":"Addons.Executo
n","params":{"addonid":"script.picpopup","params":{"title":"Titre_du_message","message":"Corps
_Du_Message hbisnvdohorb
beiyhfkdjghnk","image":"http://elkain:testelkain@Adresse_CamIP/snapshot.cgi}}}'
http://xbmc:secret@192.168.0.10:8080/jsonrpc
```

Raspberry, XBMC et Domotique

Écrit par ElkAin

Vendredi, 29 Mars 2013 07:00 - Mis à jour Jeudi, 28 Mars 2013 23:06

Et voilà une nouvelle utilisation du Raspberry. Je pense que ce n'est qu'un début et que Domotics et moi-même avons encore plein de choses prévues avec ce petit jouet.

Ce qui est bien avec cette petite bête c'est qu'il est très facile de faire des tests avec, il suffit de copier le contenu de la carte SD (ou d'en avoir plusieurs) je vous invite donc à nous parler de vos envies.

L'album Picasa de cet article est [disponible ici](#) .

{jumi [*33]}

{jumi [*24]}