

Sécurité et domotique : surveillance habitation

Introduction

Il s'agit de réaliser un système de surveillance d'habitation, c'est-à-dire un détecteur d'intrusion.

Ce détecteur fonctionnera à l'aide des capteurs à ultrasons disponibles dans le Grove starter kit, qui récupéreront le temps moyen entre l'émission et la réception d'un signal dans une pièce (lié à la distance parcourue par l'onde), et analyserons la différence entre une émission/réception et la moyenne des émissions/réceptions depuis la mise en service du système.

Si un écart trop conséquent est constaté, la webcam prendra une photo de la pièce et enverra la photo par mail directement au propriétaire.

Ce système sera activé ou désactivé par le propriétaire grâce à une **interface web**.

Cette interface permettra aussi d'accéder aux photos prises en temps réels, et de définir/changer l'adresse email à laquelle doit être envoyé les photos.

● Mode d'emploi

Montage complet à partir du Raspberry pi 2 :

Le projet s'appuie principalement sur deux modules : un capteur ultrason et un module caméra pour Raspberry.

L'utilisateur doit brancher la caméra sur **le connecteur CSI** (Camera Serial Interface). Pour cela, le câble plat de la caméra doit passer dans la fente prévue à cet effet sur le shield Grovepi.

L'utilisateur doit présenter le câble sur le connecteur, contacts tournés vers la prise HDMI, soulever le verrou du connecteur, brancher le câble et refermer le verrou. Il faut ensuite brancher le shield sur les broches i2c prévues à cet effet sur le Raspberry Pi. Sur le shield, brancher sur le port D4 le capteur ultrason. Enfin, branchez le câble ethernet au port ethernet du Raspberry afin de le mettre sur le réseau.

Si possible, fixer les capteurs caméra et ultrason (à l'aide d'un carton par exemple), afin d'éviter qu'ils bougent et faussent les résultats.

Démarrer le Raspberry Pi, puis tapez la commande suivante (nécessite les privilèges super-utilisateur) :

```
sudo raspi-config
```

Dans le menu qui s'affiche, valider le support caméra, sauvegardez et revenez à la console.

Avant de démarrer le programme, l'utilisateur doit se rendre sur **le site php** hébergé sur le Raspberry (à l'aide d'Apache et de php5) en tapant l'adresse IP du Raspberry dans un navigateur.

L'utilisateur doit alors rentrer l'adresse email qu'il souhaite utiliser pour recevoir les photos dans l'interface "configuration".

Il doit donc installer la dernière version d'Apache et de php5 sur son Raspberry, et mettre les fichiers php et html en respectant l'arborescence dans le dossier /var/www/html.

Afin de pouvoir lancer le programme, l'utilisateur doit avoir préalablement installé la dernière version de Mutt (client de messagerie permettant d'envoyer des pièces-jointes) sur son raspberry, ainsi que le paquet ssmtp (smtp permet à des applications d'envoyer des courriels via la commande /usr/sbin/sendmail.)

```
sudo chmod -r 777 /var/www/html/*
```

Pour lancer le système de surveillance, l'utilisateur peut l'activer via l'interface web ou via le terminal et lancer le programme de la manière suivante (nécessite également les privilèges super-utilisateur) :

```
sudo ./LancerLaSurveillance
```

Lorsqu'une intrusion est détectée, le Raspberry enverra un **mail** sur l'adresse rentrée précédemment par l'utilisateur, qui pourra alors vérifier sur les photos si un intru est effectivement entré dans son habitation ou non.

Il peut également consulter 10 photos qui auront été prises pendant 5 minutes par le Raspberry, hébergées dans l'interface "accès aux photos" du site Web.

Les photos seront triées en fonction de la date et de l'heure de la prise.

● Moyens matériels

Nous avons utilisé un **Raspberry Pi 2**, ainsi que des modules Ultrason et Caméra. Nous avons fait ce choix car il s'agissait pour nous deux de notre premier projet sur des nano-ordinateurs, et que nous avions déjà un peu manipulé le Raspberry Pi en cours de FASO.

De plus, nous avons déjà sous la main le pilote du module Ultrason, qu'il nous a suffi de modifier pour qu'il puisse interagir avec le module caméra.

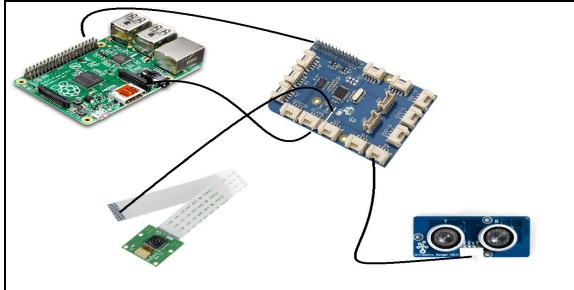
● Moyens humains

Nous avons tous deux réfléchi ensemble aux différentes parties du code, et les avons codées ensemble (en salle de TP principalement). Durant les premières séances, nous avons ensemble recherché comment articuler ensemble les différents modules de notre projet, comment organiser et réaliser le code, à partir de quoi nous pouvions partir, etc...

Nous avons recherché des exemples de projets de ce type déjà existants, mais n'avons trouvé que des résultats de code en python. Cela nous a en revanche permis de découvrir plusieurs méthodes que nous pouvions utiliser et de rechercher la meilleure possible répondant aux fonctionnalités que nous voulions implémenter.

Dans les dernières séances, la principale difficulté a été de réussir à réaliser un site web en php (langage que nous avons découvert durant ce semestre). Il s'agissait de réaliser une interface utilisateur sur le site web hébergé sur le Raspberry (grâce à Apache et php5), où l'utilisateur pouvait consulter des photos, configurer son adresse mail, et lancer ou stopper le programme de surveillance.

- Architecture de votre projet

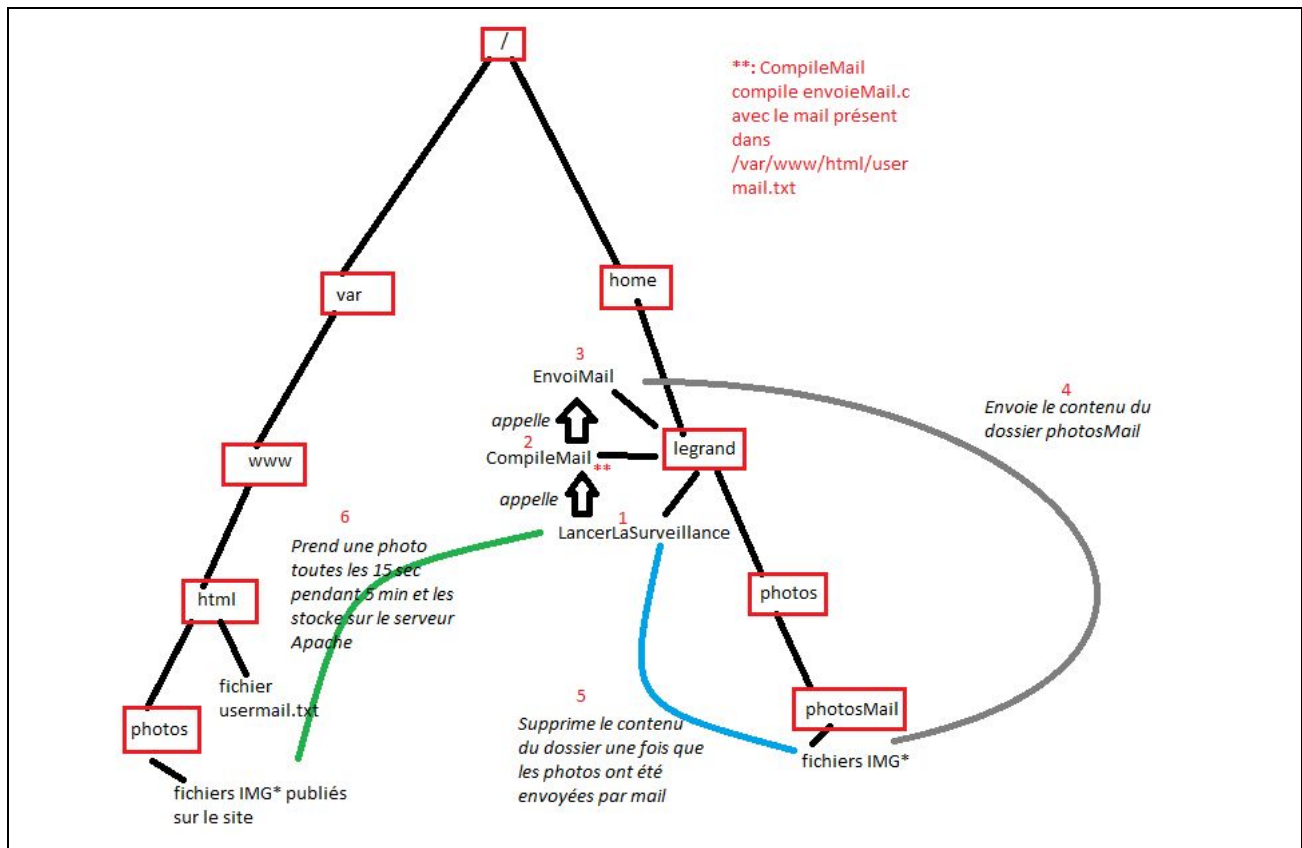


Partie physique

L'**ultrason** est branché sur le shield, lui-même branché aux ports i2c du Raspberry.

La **caméra** est branchée au connecteur CSI du Raspberry, en passant par la fente prévue sur le shield.

Le Raspberry est connecté au réseau par un câble **Ethernet**.



Cf: Code pour plus de détails

Partie logicielle

● Code

Site de gestion excepté, c'est à l'aide du langage C que nous avons codé notre système de surveillance.

Liste des procédures et **exécutables** du système :

- Grovepi_surveillance.c → **LancerLaSurveillance**
 - Grovepi_c
 - Grovepi_h
- Envoiemail.c → **EnvoiMail**
- CompileETenvoi_Mail.c → **CompileMail**
- **usermail.txt**

Grovepi_surveillance.c

Le module repose sur un programme principal **LancerLaSurveillance** dont le code se trouve dans grovepi_surveillance.c.

Ce programme permet de lancer le système de Surveillance, c'est à dire que si le capteur ultrason détecte un mouvement, une série de 5 photos va être prise; les photos sont **envoyées par mail** puis supprimées. Une autre série de 10 photos s'enchaîne ensuite. Ces photos là seront stockés et accessibles depuis le site web.

Le socle de ce code repose sur le grovepi_ultrasonic qui nous a été mis à disposition sur le moodle FAS. Code que nous avons étoffé petit à petit au rythme des contraintes que l'on rencontrait. (Conditions de prises, envoi de mail etc.)

CompileMail

```
int main()
{
    system("gcc envoiemail.c -o EnvoiMail");
    system("./EnvoiMail");
    return 1;
}
```

Cette petite procédure dont on rappelle le code, permet tout simplement de compiler le fichier c qui gère l'**envoi du mail** et de l'exécuter.

Envoiemail.c

Tout d'abord, il faut savoir que l'envoi des emails depuis le raspberry est géré par un serveur sSMTP, configuré au préalable avec l'adresse d'expédition igrasp05@gmail.com.

L'adresse du destinataire est récupérée par le site web, où l'utilisateur renseigne son adresse email. Elle est stockée dans le fichier **usermail.txt**.

Envoiemail.c va récupérer cette adresse email et envoyer l'ensemble des photos via la commande mutt.

```
sprintf(s,"echo \"Ci-joint les photos prises de l'intrusion\" | mutt -s \"Intrusion détecté\" -a IMG_* -- %s",usermail); system(s);
```

- Perspectives

Le projet est à priori opérationnel. Nous avons même été au delà de notre cahier de charge initial, en intégrant des fonctionnalités inédites à notre interface utilisateur, comme l'accès au photo en temps réel et la modification d'adresse mail.

Dans l'immédiat, il n'est pas possible de commercialiser ce système au grand public.

En effet , des branchements et des manipulations, pas toujours à la portée du néophyte, doivent être faits pour jouir de la meilleur utilisation possible.

Le projet pourrait être amené à un autre niveau par la mise en ligne du site web, et l'ajout de nouvelles fonctionnalités tel que le streaming.