CMP 408
System Internals and Cybersecurity

Part IV - Mini Project

Name: Dorota Marczak

Student number: 1604779

Who stole the food from the fridge ?
Catching 'refrigerator thieves' with Raspberry Pi and AWS S3

Introduction

Since the dawn of time, a need has always been a mother of invention. Each of the inventions that changed the world – such as steam engine, an electric vacuum cleaner or the first digital computer in the history – were based on the complex research led by specialists in the field of engineering and life sciences, which experiments were often sponsored in millions donated by the government and the most prosperous entrepreneurs. Those and similar experiments would probably stay the domain of elite social groups if not the Do It Yourself anti-commercial movement, commenced shortly after the advent of the digital revolution in the early '50s. Years after, microcomputers such as Raspberry Pi became a powerful tool in developing more or less complex Internet of Things systems for both commercial and home use by individuals interested in IT and Electronics.

The aim of this project is to present the steps needed in order to create own intrusion detection system, which could be used e.g. for detecting 'refrigerator thieves' - often problem of individuals living in a house in multiple occupation.

The project aim is further outlined by the following objectives:

- Setting up the hardware and executing basic tests to review sensor readings accuracy and quality of camera image
- Programming the device to stabilize after the initialization process, and then take a series of pictures whenever the intruder is detected (in short time intervals)
- Uploading taken pictures to AWS Simple Storage Service and downloading them on the target device.
- Securing the system, as well as allocating security keys required for connection with chosen AWS S3.

Methodology

The first step in the process of development of the intruder detection system was establishing a wire connection between the Raspberry Pi Zero W, ultrasonic sensor HC-SR04 and nightvision camera (through flex cable) as presented in Fig.1. Once the hardware setup has been finished, the raspberry pi could have been turned in order to update its settings by enabling direct access to the camera (using *raspi-config*).

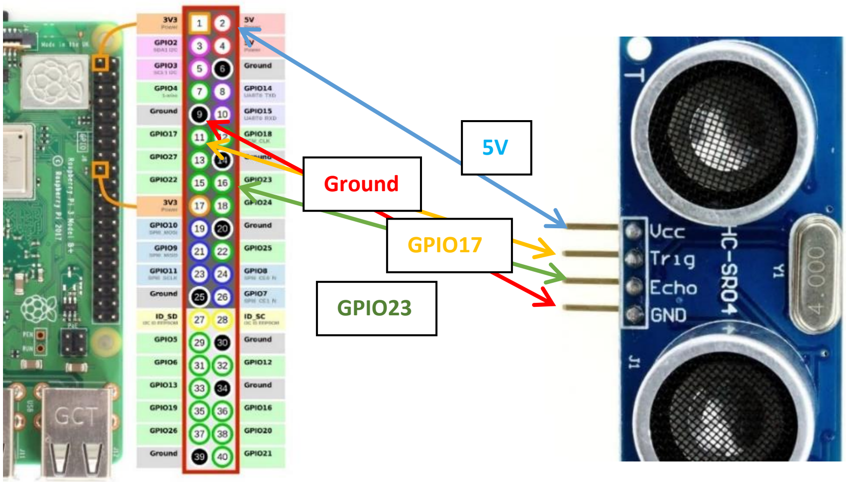

Fig.1

Connecting Raspberry Pi with ultrasonic sensor

The camera module additionally required installation of the picamera library, providing the functionality in the form of python interface. Regarding the ultrasonic sensor, it has been designed to determine the distance to the object localized in front of him in the range between 2 to 400cm. Despite its simplicity HC-SR04 turned out to be significantly more reliable than the passive infrared sensor (PIR) facilitated with Fresnel lenses, which high sensitivity (regardless of changes made to manual settings) to received infrared rays made the readings of dubious quality.

For the purposes of the project, the speed of sound has been calculated from below formula:

$$c_{\mathrm{air}} = (331.3 + 0.606 \cdot \vartheta) \quad \mathrm{m/s},$$

Where v is the temperature in degrees Celsius (°C)

The developer began the implementation of the program aimed at detecting 'the refrigerator thieves' with identifying significant differences in the distance comparing to the original distance between the device and the doors of the fridge. The application would emit the ultrasounds with the transmitter and listen on the receiver port (accordingly port 11 and 16), convert the time that elapsed between sending and receiving the signal into the distance traversed by the sound (divided by half due to the sound wave reflection). Taking into account initial unsteadiness of the sensor it has been decided that the most reliable value of the distance will be obtained by extracting the average of the first ten readings recorded.

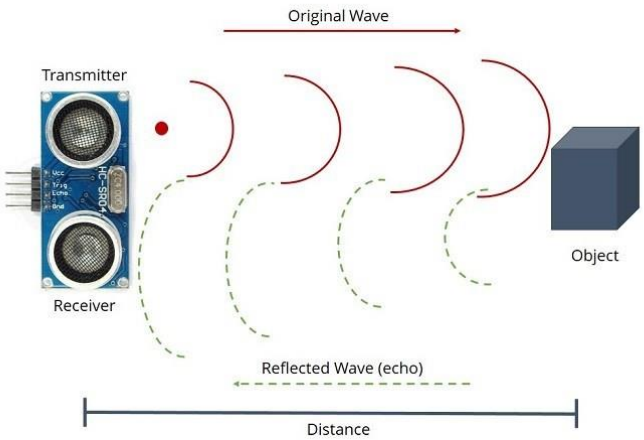

Fig.2

Ultrasonic sensor scheme

Once the device has successfully passed the stabilization stage, it would begin to compare current readings with the average distance and trigger the alarm whenever current distance is bigger than the original, capturing a series of pictures of an intruder whenever current distance ranges between average increased by 5 cm and 400 cm (maximum reliable range). Next, the application has been processed to the raspberry pi and the code has been executed as the device was placed in the refrigerator and powered with the power bank. The success of the first test allowed to implement an improvement to the application functionality, which would enable using the system remotely, without a need of sustaining connection to the IoT device network. The solution chosen by the developer was utilizing AWS storage service offering 5 Gb of storage and up to 20 000 GET requests out of charge within its Free Tier account. Similarly to storage offered by different companies AWS S3 consists of the objects with a globally unique name called storage buckets, created in the region specified by the developer. Taking into account the possibility of attempts of accessing the storage instance by the unauthorized users, the storage bucket settings has been updated to block all public access, which is another level of protection against accidental public exposure.

First step to connect with S3 services was establishing a connection with AWS service through installation and configuration of *aws client*. That, in turn, required providing the secret access key generated by Identity Access Management (IAM) service, which were then stored in hidden *.aws* directory on the host. Next, the connection with the storage has been tested by running the application on the device (Fig.3).

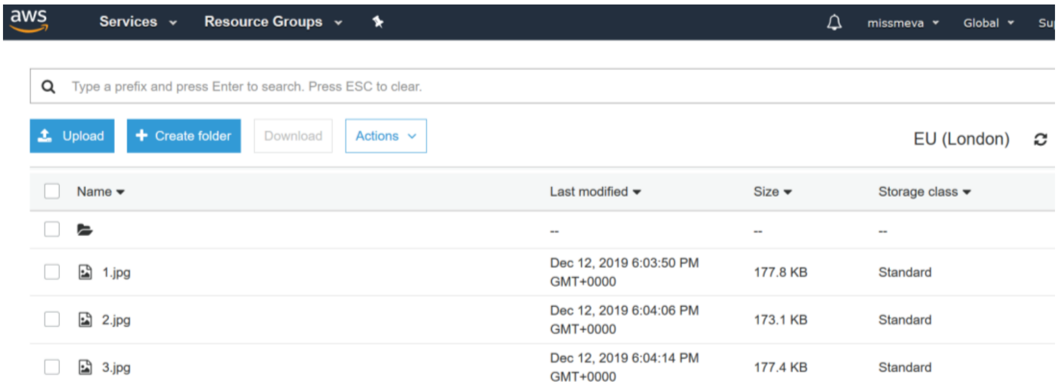

Fig.3

The view on the bucket utilized in the implementation of the application

Once the series of pictures taken by raspberry pi camera has been placed in S3, the developer implemented client-side script, which would allow to get access to the content stored in the cloud. After establishing connection with AWS services and installation of boto3 library the script has been utilized in a system test (Fig.4), carried out in the target conditions. The experiment verified application's efficiency, as well as microcomputer's resistance to lower temperatures (<5°C).

Fig.4

The results of the test executed in target environment by the device placed in an unlit refrigerator

Conclusion

Despite the limited resources, the microcomputers - such as Arduino and Raspberry Pi – has been unceasingly popular for over 20 years due to a wide range of compatible sensors and components ideal for prototyping industrial automatization and IT hobbyists' projects. This document aimed at presenting the implementation of the intruder detection system (Fig.4), which can be achieved with a minimal cost and with the use of professional AWS services.

The application fulfilled the settled goal, achieving desired functionality without compromising security at the expense of ease of use and vice versa. Although the application could be improved with automated execution of the device code, offered by Python SSH module Paramiko, it was decided this action will not be necessary due remote-oriented character of the project.

References

Docs.aws.amazon.com. (2019). *Amazon S3 Server Access Logging - Amazon Simple Storage Service*. [online] Available at: https://docs.aws.amazon.com/AmazonS3/latest/dev/ServerLogs.html [Accessed 10 Dec. 2019].

En.wikipedia.org. (2019). Speed of sound. [online] Available at: https://en.wikipedia.org/wiki/Speed_of_sound [Accessed 12 Dec. 2019].

Python, R. (2019). *Python, Boto3, and AWS S3: Demystified – Real Python*. [online] Realpython.com. Available at: https://realpython.com/python-boto3-aws-s3/ [Accessed 9 Dec. 2019].

Random Nerd Tutorials. (2019). Complete Guide for Ultrasonic Sensor HC-SR04 with Arduino | Random Nerd Tutorials. [online] Available at: https://randomnerdtutorials.com/complete-guide-for-ultrasonic-sensor-hc-sr04/ [Accessed 12 Dec. 2019].