Abertay University

IoT Door Alarm System

Declan Woodham

CMP408

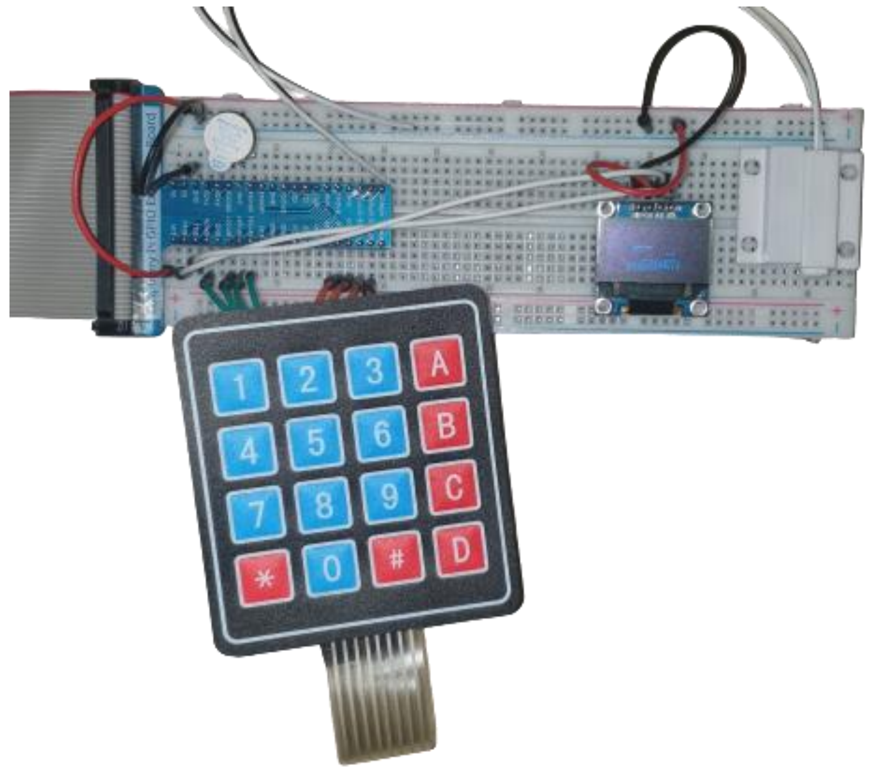
Introduction

This project set out to develop a prototype home security system to detect when doors are opened and raise an alarm if the system is armed.

The security device will be able to be remotely monitored from the cloud using various AWS Services.

Objectives

- Develop Linux Loadable Kernel Module in C to allow communication between connected hardware devices and the user space application.
- Develop a user space application in Python to handle input from Keypad, display various information on a screen, and provide communication of data to and from the cloud
- Develop a develop web panel for users to communicate and monitor the device.
- Ensure security across all attack surfaces including the device and the cloud.



Methodology

Hardware and Kernel

The device required 4 separate external hardware devices connected via GPIO.

- Keypad
- Door Sensor
- OLED Display
- Passive Buzzer

To communicate between the hardware and user space, a Linux Loadable Kernel Module was developed to read the GPIO data from the Door Sensor.

When an interrupt is detected, a signal is sent to the user space application using signals which is registered using IOCTL.

Finally, SysFS is used to read the updated value from the GPIO.

User Space

The user space application is responsible for handling input on the device from hardware, such as the Keypad and the Door Sensor, as well as outputting data to the connected OLED display.

The user space application is also responsible for handling communication to and from the cloud using IoT Core and MQTT. To allow user monitoring, updates are sent from the user space application at regular intervals, and when a user makes a remote change to the device it is handled.

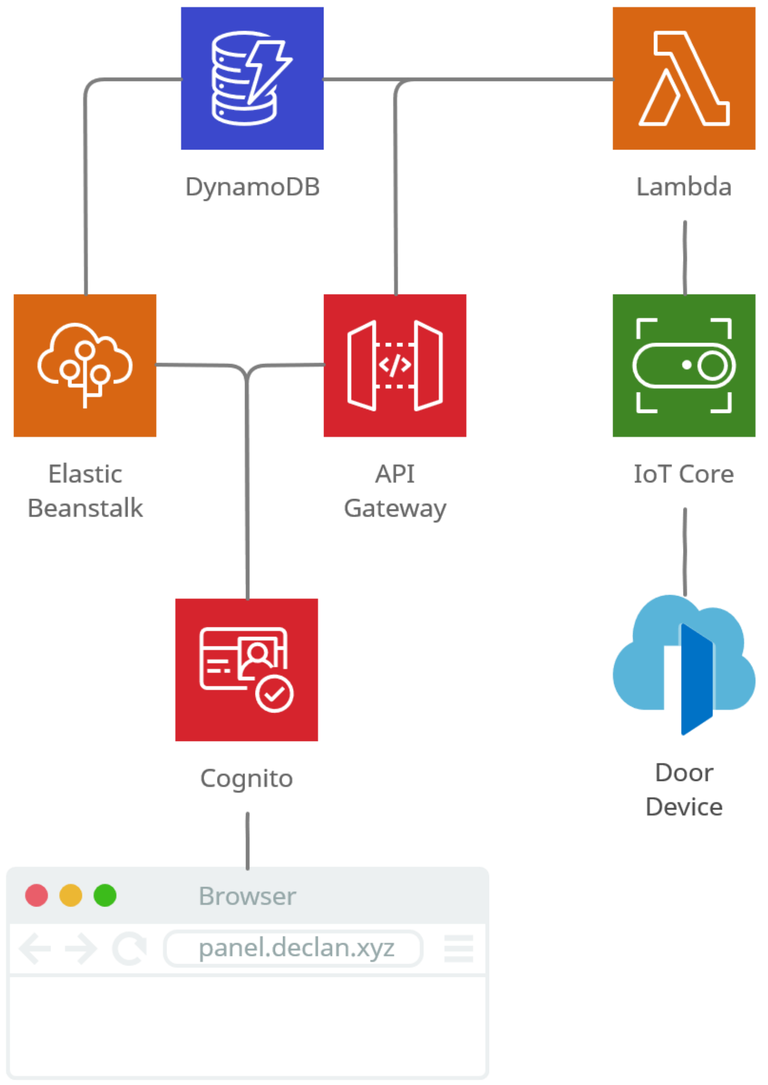
To increase device security, the udev rule was added to allow the application to communicate with the kernel's device file, and SysFS without sudeor permissions.

Cloud

To provide high availability, low latency cloud access to the devices state, 6 different AWS Services were used together.

Users authenticate to the web panel hosted on Elastic Beanstalk using Cognito, all traffic is encrypted using TLS. Once logged into the web panel user actions which influence the device are sent using Web Sockets to API Gateway which is then forwarded to Lambda to send the data to the connected device in IoT Core.

All communication from the device to the Cloud is handled using IoT Core and sent using MQTT.



Project Highlights

Device Features

- Alarm System
- Device Display

Cloud Features

- Remote Monitoring of Door and Alarm State
- Change Passcode Remotely

Security Features

- Cognito Authenticated API's and User Panel
- TLS Encrypted communication
- Separate MQTT Channel Per Device
- Strict AWS Policies between services

Future Work

- Creation of a complete firmware image
- Increased Security
- Multiple Device Support
- More Hardware communication in Kernel Module

References

En.wikipedia.org. 2021. *Loadable Kernel Module*. [online] Available at: <https://en.wikipedia.org/wiki/Loadable_kernel_module>

Python.org. 2021. *Python Programming Language*. [online] Available at: <https://www.python.org/>.