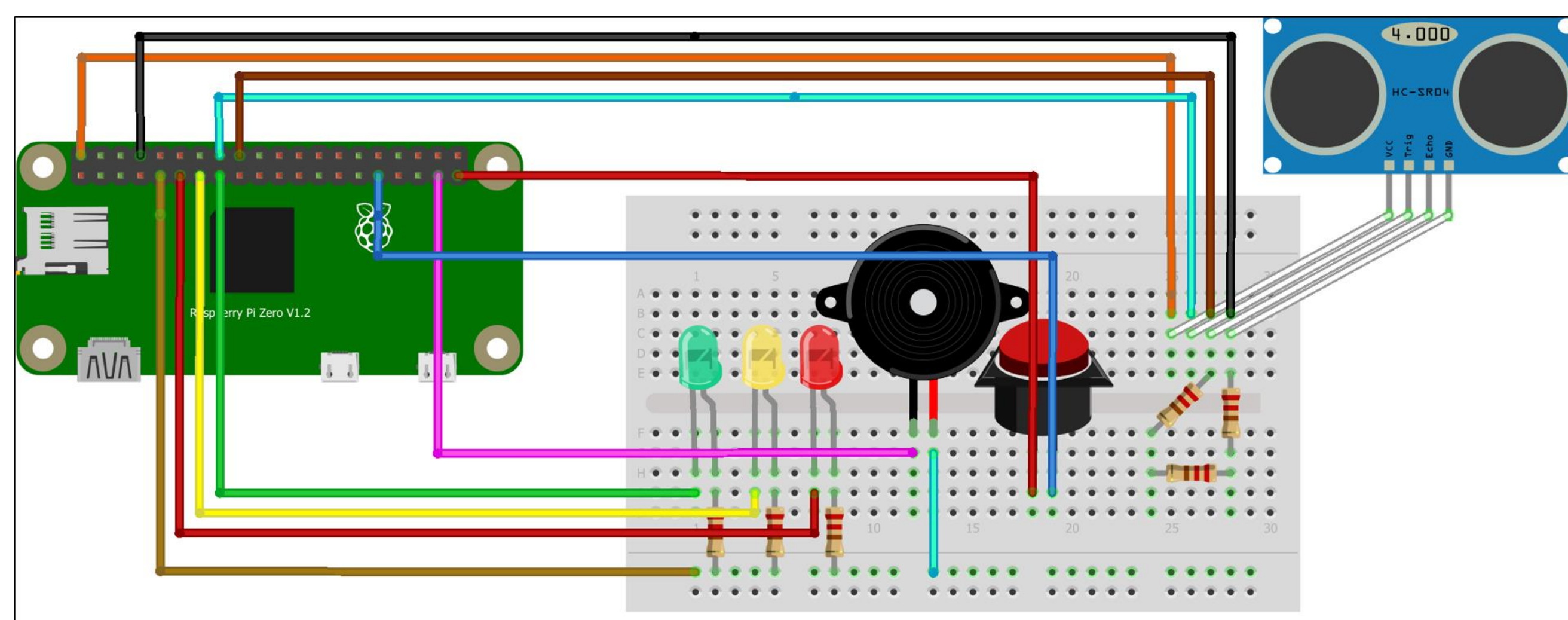


## INTRODUCTION

Project: Perimeter intruder detection system.

Components:

- Ultrasonic sensor
- 3 LEDs
- Buzzer
- Push button
- Raspberry Pi Zero
- Resistors
- Jump wires



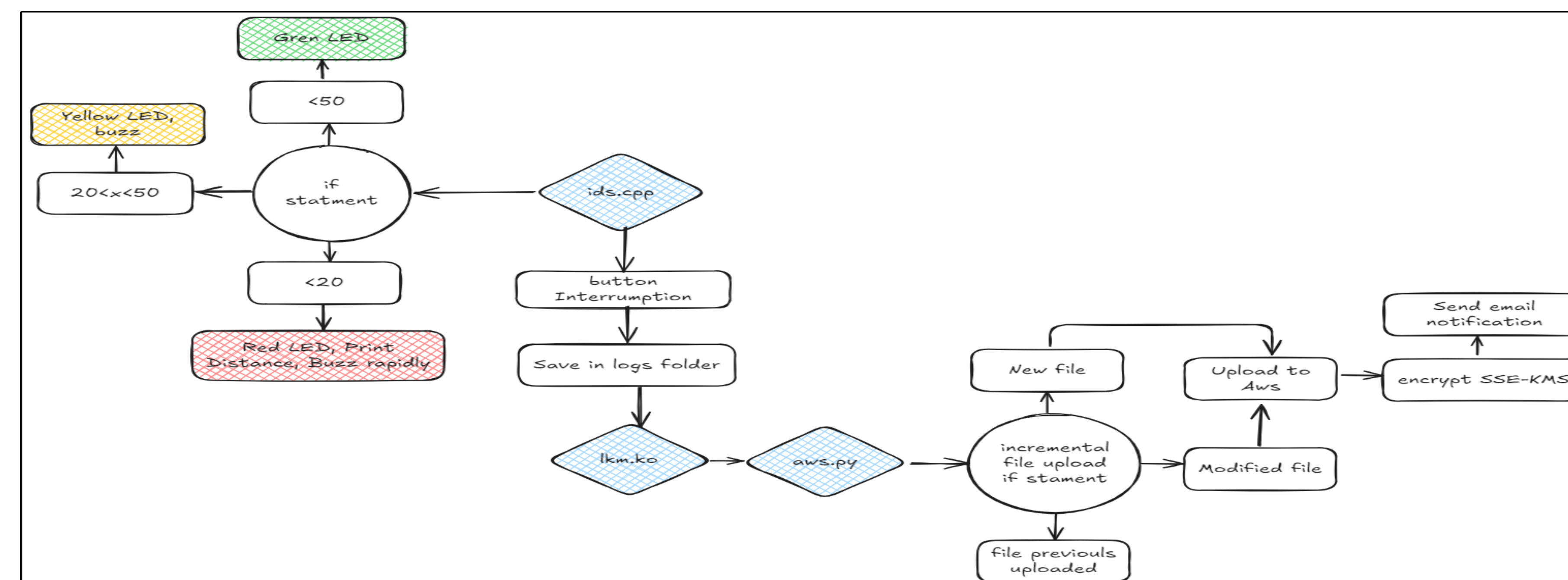
IoT: Took advantage of the GPIO pin of the RPi to connect the component mentioned above. A loadable kernel module was cross-compiled from a fedora Virtual machine and used to call a Python script.

Cloud: Amazon web service was used as cloud storage for sensor's logs. The serves that were used were S3 and notification service. These logs are stored incrementally so only new and modified will be uploaded reducing bandwidth.

Objective:

1. Measure with the sensor to determine the distance from an object.
2. Alert when appropriate (<20cm)
3. Save output in the log folder
4. Utilise loadable kernel module to call python script
5. Incremental upload log files to Amazon Web Services using S3 bucket
6. Receive a notification when the AWS S3 bucket has been modified for any reason

## METHODOLOGY



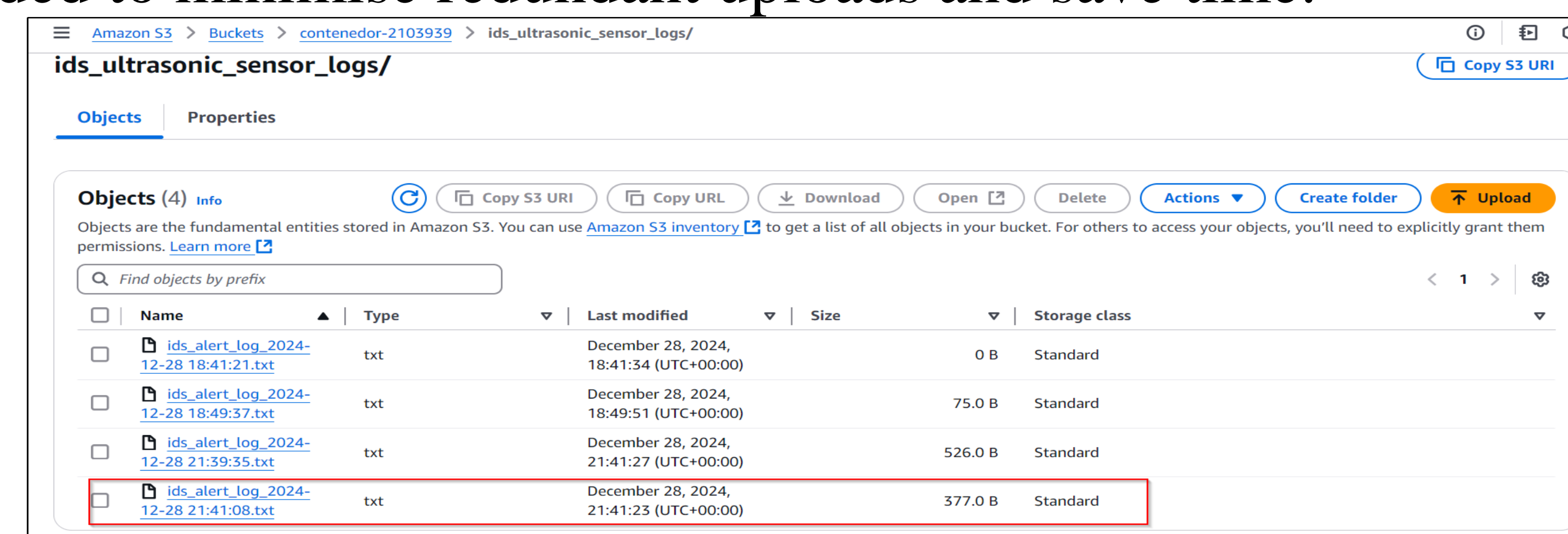
IDS.cpp:

1. Proximity Measurement: Uses an ultrasonic sensor to calculate distance based on sound pulse timing.
2. Alert System: Activates LEDs and buzzer based on distance—green for safe, yellow for caution, and red with rapid beeping for danger.
3. Event Logging: Logs intrusion events with distance and timestamps for later review.
4. Button-Controlled Shutdown: Monitors a button on GPIO 6 to save logs and terminate the program safely when pressed.
5. Kernel Module Integration: Uses an LKM to load aws.py on program shutdown.

```
!Intrusion detected!. Proximity distance of: 15.2 cm at 2024-12-28 21:41:04
Button pressed. Exiting...
Alerts saved to file: /home/pi/Desktop/ultrasonic_alerts/logs/ids_alert_log_2
Kernel module loaded successfully!
```

AWS.py:

1. AWS S3 Integration: The code connects to Amazon S3 using provided AWS credentials to upload files incrementally from a local folder to a specified S3 bucket and folder.
2. File Metadata Check and Incremental Upload: Before uploading, it retrieves metadata (size and last modification date) of each file in S3 to compare with the local file, ensuring only updated or new files are uploaded to minimise redundant uploads and save time.



Name	Type	Last modified	Size	Storage class
ids_alert_log_2024-12-28 18:41:21.txt	txt	December 28, 2024, 18:41:54 (UTC+00:00)	0 B	Standard
ids_alert_log_2024-12-28 18:49:57.txt	txt	December 28, 2024, 18:49:51 (UTC+00:00)	75.0 B	Standard
ids_alert_log_2024-12-28 21:39:55.txt	txt	December 28, 2024, 21:41:27 (UTC+00:00)	526.0 B	Standard
ids_alert_log_2024-12-28 21:41:08.txt	txt	December 28, 2024, 21:41:23 (UTC+00:00)	577.0 B	Standard

## PROJECT HIGHLIGHTS

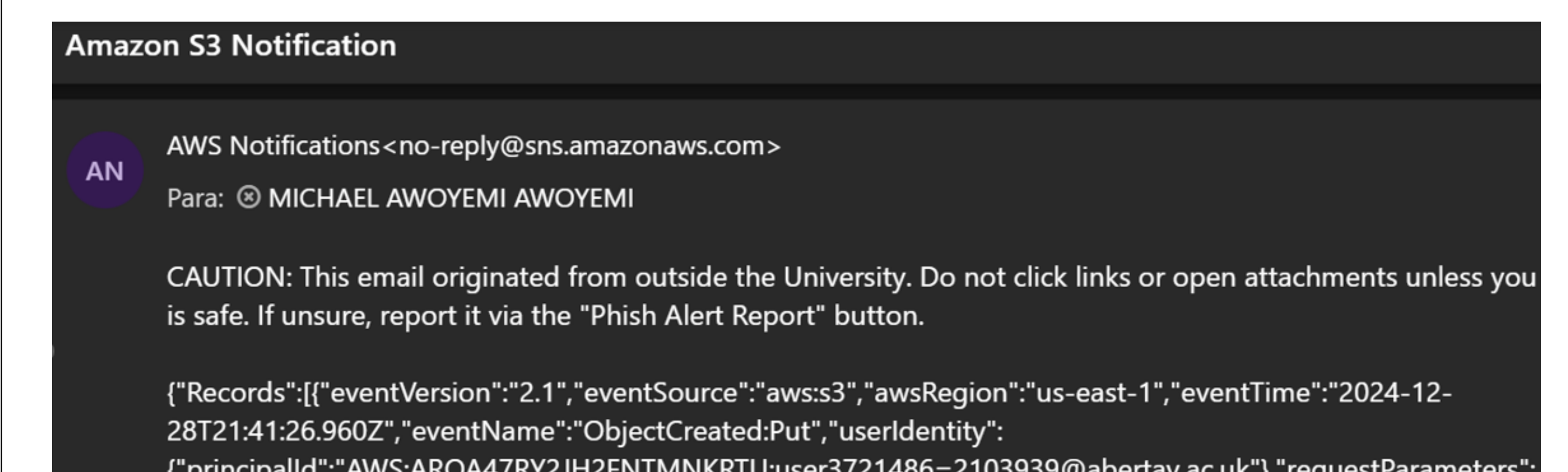
IoT perimeter intrusion detection system using Raspberry Pi.

Program Structure: If statement nested in a while loop, stops with button or ctrl + c.

Sensor Function:

- Sends and receives waves through trig and echo pins.
- Distance evaluation determines actions.
- > 50cm Green light, 20cm - 50cm: Yellow light and buzz every 300ms, < 20cm: Red light and rapid buzz every 50ms. Red-light output (below 20cm distance) logged with distance and timestamp.

Call loadable kernel module to run aws.py script. AWS Function: Logs into S3 bucket, incremental backups (new/modified files), sends email notification of new S3 object.



## REFERENCES

- <https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html>
- <https://docs.aws.amazon.com/AmazonS3/latest/userguide/EventNotifications.html>
- <https://www.geeksforgeeks.org/arduino-ultrasonic-sensor/>
- <https://www.youtube.com/watch?v=0jedQM7qS90>
- <https://projects.raspberrypi.org/en/projects/button-switch-scratch-pi/1>
- <https://www.circuitbasics.com/how-to-use-buzzers-with-raspberry-pi/d>