SPACE COAST UNMANNED

CERBERUS

OBJECTIVE

Design a system that:

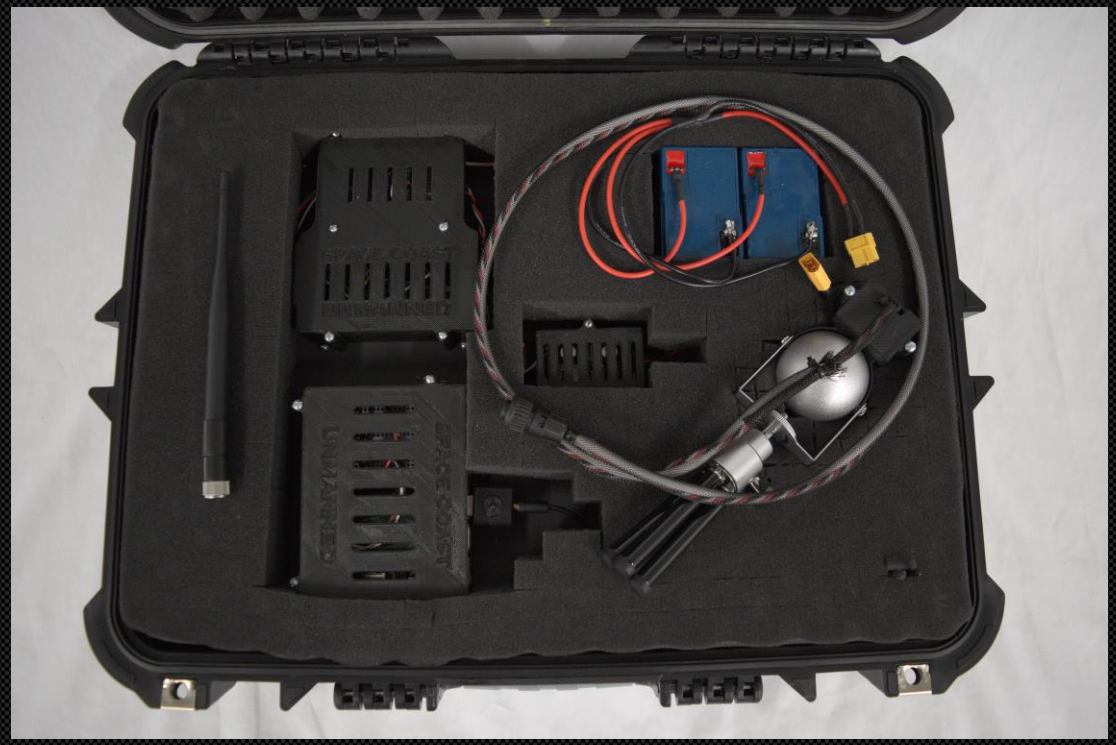
- Utilizes open source hardware and software
- Provides anomaly detection of:
  - Seismic
  - Acoustic
  - Magnetic
  - Infrared
  - Radiofrequency
  - Visible Light
- Transmits information including sensor telemetry utilizing low power RF

SPACE COAST UNMANNED'S SOLUTION:

- Includes sensors for each anomaly type
- Provides high fidelity temperature and battery status
- Provides ample overhead for future functionality
- Protects data using AES128 encryption
- Allows single person carry and deployment in under a minute

COMPONENTS

- Arduino Mega
  - An Arduino Mega serves as the controller for the radio transceivers in the Cerberus and the receiver station. The mega was chosen due over a normal Arduino to its performance gains and larger memory. The current utilization is roughly 23% of the available memory, leaving plenty of room for future expansion.



COMPONENTS (CONT.)

- Raspberry Pi
  - A Raspberry Pi was chosen to monitor the Bosch 9 degree of freedom sensor as well as the RTL-SDR RF sensor. This allows extremely powerful computing to be done in a compact package. As delivered, the scripts utilize around 30% of the Raspberry Pi's CPU. In the future, additional functionality can be implemented with plenty of overhead.

COMPONENTS (CONT.)

- Acoustic Sensor
  - The acoustic sensor is a MAX9814 electret microphone with automatic gain control up to 60dB. This allows for extremely dynamic sound ranges.

COMPONENTS (CONT.)

- Visible Light Sensor
  - The VC0706 weatherproof camera communicates via serial. It has motion detection built in, which allows easy integration. The camera is infrared sensitive, giving it greater performance in low light conditions. There is also a low light level sensor, which triggers infrared LEDs once ambient lighting becomes too dark for the camera alone.

COMPONENTS (CONT.)

- Infrared Sensor
  - There is a Passive Infrared (PIR) sensor installed on the motion camera. It functions by monitoring the infrared spectrum in front of the sensor. When an object that emits infrared (such as a human) moves past the sensor, it will trigger. These sensors are commonly used in household motion detectors and flood lights. This extends the low light capabilities of the Cerberus beyond the range of the infrared LEDs on the VC0076 camera.

COMPONENTS (CONT.)

- Magnetic/Seismic Sensor
  - The magnetic and seismic detection is accomplished via a Bosch BNO055 9 degree of freedom orientation sensor. This sensor combines a 3 dimensional accelerometer, magnetometer, and gyroscope. Initialization of the device includes reading the sensor orientation and magnetic headings. A threshold is set via software for magnetic and seismic activity. Because the magnetometer is very sensitive, rotations of the sensor can produce magnetic disturbance alerts as well as seismic alerts. The sensor has also been test triggered using standard household magnets near the sensor.

COMPONENTS (CONT.)

- Radiofrequency (RF) Sensor
  - The RF sensor is an RTL-SDR unit attached to the Raspberry Pi. This unit is capable of reading frequencies between 24 MHz and 1850MHz. The unit as delivered is monitoring FRS channels 1-7 and 15-22. Channels 8-14 are at a higher frequency and were not included in this demonstration. This unit is compatible with gnuradio, or other software defined radio suites. The unit is currently using the Osmocom RTL-SDR open source library's built in command line functionality to monitor FRS radio channels. By changing the parameters such as center frequency, bandwidth, gain, and integration time, it can easily be modified to monitor any frequency the RTL-SDR is capable of reading.

TELEMETRY

- Temperature Sensor
  - Texas Instruments LM35 3 pin analog temperature sensor
- Battery Monitor
  - Simple voltage divider circuit using a 4k and 10k ohm resistor
- Radio Transceiver
  - HopeRF RFM95 LoRa with up to 2km (1.25 miles) line of sight range
- AES128 Encryption
  - Message is encrypted before transmission, with hardware random number generator used to prevent repetition of cipher

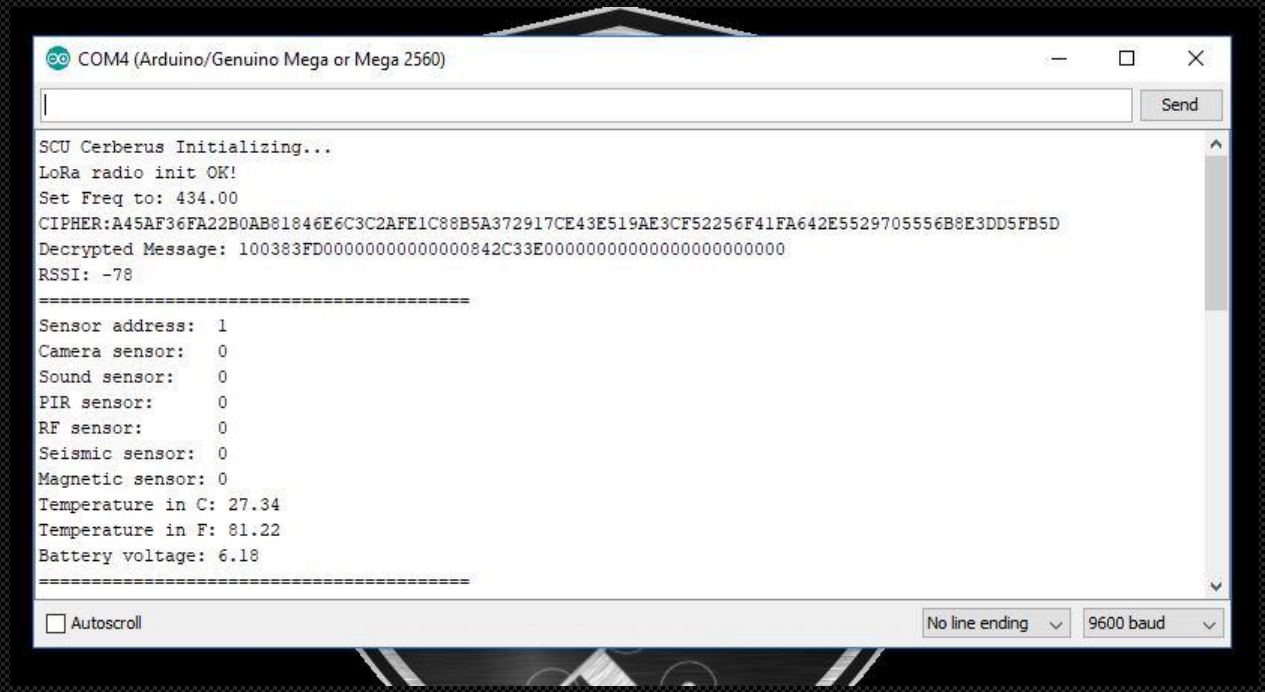
PHYSICAL CHARACTERISTICS

- Size
  - The outer dimensions of the protective case are: 16”x20.5”x7.5” (LxWxH).
- Weight
  - The Cerberus weight as delivered is 15.5 pounds.
- Power
  - The Cerberus draws roughly 4 Watts continuously. This low level of power draw facilitates easy customizability of battery size to meet mission requirements. As delivered the Cerberus has two 6V sealed lead acid batteries with 4.5 Amp-hours of capacity for a combined total of 54 Watt-hours. This should result in over 12 hours of runtime for demonstration purposes. For future units, larger batteries can be incorporated that easily extend the runtime past 24 hours. Another recommended approach would be to install a solar panel to charge the battery during daylight hours. With a properly sized power system the Cerberus should be capable of indefinite fair weather run times.

ASSEMBLY VIDEO

RECEIVER OUTPUT

- The output of the receiver shows status of all sensors, as well as the original Cipher, and Decrypted message in its raw format.
- Lines 1-3 are one time only, and show the initialization status of the Cerberus.
- Line 4-5 show the Cipher message received, and the resulting decrypted data.
- Line 6 shows the Received Signal Strength Indicator, higher numbers are stronger signals
- Line 8 shows the address of the sensor that is transmitting. This can be 0-255
- Line 9-14 shows the status of the individual sensors. A “0” means no activity, a “1” means activity has been detected from that sensor
- Line 15-16 show the temperature reported by the sensor in degrees
- Line 17 shows the voltage of the battery in Volts.



OPERATIONAL DEMONSTRATION

RESOURCES

- This presentation, along with the User Manual can be found at our Github repository:
- https://github.com/Space-Coast-Unmanned/Cerberus
- External resources used:
  - Radiohead library for RFM95: http://www.airspayce.com/mikem/arduino/RadioHead/
  - Ghirlekar library for BNO055: https://github.com/ghirlekar/bno055-python-i2c
  - Python wrapper for RTL-SDR: https://github.com/roger-/pyrtlsdr
  - AES Encryption: https://github.com/spaniakos/AES
  - Tutorials from Adafruit: https://www.adafruit.com/