University of Nevada, Reno

Securing a UAV Using Features from an EEG Signal

A thesis submitted in partial fulfillment of the
requirements for the degree of Master of Science in
Computer Science and Engineering

by

Ashutosh R Singandhupe

Dr. Hung M. La - Thesis Advisor
Dr. David Feil-Seifer - Thesis Co-Advisor
December 2017

THE GRADUATE SCHOOL

We recommend that the thesis

prepared under our supervision by

ASHUTOSH SINGANDHUPE

Entitled

Securing a UAV Using Features from an EEG Signal

be accepted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

Dr. Hung La, Advisor

Dr. David Feil-Seifer, Committee Member

Dr. Hao Xu, Graduate School Representative

David W. Zeh, Ph.D., Dean, Graduate School

December, 2017

Abstract

This thesis focuses on an approach which entails the extraction of Beta component of the EEG (Electroencephalogram) signal of a user and uses his/her EEG beta data to generate a random AES (Advanced Encryption Standard) encryption key. This Key is used to encrypt the communication between the UAVs (Unmanned aerial vehicles) and the ground control station. UAVs have attracted both commercial and military organizations in recent years. The progress in this field has reached significant popularity, and the research has incorporated different areas from the scientific domain. UAV communication became a significant concern when an attack on a Predator UAV occurred in 2009, which allowed the hijackers to get the live video stream. Since a UAVs major function depend on its onboard auto pilot, it is important to harden the system against vulnerabilities. In this thesis, we propose a biometric system to encrypt the UAV communication by generating a key which is derived from Beta component of the EEG signal of a user. We have developed a safety mechanism that gets activated in case the communication of the UAV from the ground control station gets attacked. This system was validated on a commercial UAV under malicious attack conditions during which we implement a procedure where the UAV return safely to an initially deployed "home" position.

Dedication

Dedicated to my family.

Acknowledgments

I would like to thank my advisor, Dr. Hung La and my co-advisor Dr. David Feil-Seifer who provided the support, encouragement, and insights to keep me on the right path.

This research is based upon work supported by the National Aeronautics and Space Administration (NASA) under Grant No. NNX10AN23H issued through the Nevada NASA Space Grant, and Grant No. NNX15AI02H issued through the Nevada NASA Research Infrastructure Development Seed Grant and Grant No.115-1528137 issued through the National Science Foundation(NSF).

To my committee members Dr. Hao Xu, thank you for the time and effort which you have put in to review this thesis and for providing advice along the way.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Introduction

The use of unmanned aerial vehicles (UAVs) in civilian airspace has significantly grown from public safety applications for commercial purposes, to personal use by hobbyists and more aggressively in military applications. This increases at various levels has subsequently led to the occurrence of several severe incidents of different type of attacks on both military and civilian UAVs. Security flaws have been demonstrated in recent investigations of both military and inexpensive consumer UAVs, revealing these systems to be vulnerable to attack.

Commercial activities such as Google's "Project Wing" [1] have managed to test their drones for food delivery. Other company projects like Amazon's "Prime Air" service [2] aim to provide same-day package delivery. These successful projects would place several drones in commercial airspace, near population centers, which would increase the number of UAVs in civilian airspace and their proximity to people. It would subsequently increase the potential for, and interest in, potential cyber-attacks on those UAVs. These potential threats need to be addressed to ensure that a UAV completes its mission and is not used for a malicious purpose.

There have been several incidents where aerial robots have been remotely compromised, which includes taking control of the UAV or making it crash-land. The first well known cyber attack on a UAV occurred in 2009, when the Iraqi militants were able to get live video feeds from an unsecured communication link used by a Predator Drone, using readily available cheap software [3]. Other incidents such as in October 2011, a key-logging malware was found on Predator and Reaper ground control stations, which got escalated quickly to both classified and unclassified computers [4].

The claimed theft of a Sentinel RQ-170 UAV by Iranian forces in December 2012 was a troubling incident. Hostile agents were able to compromise the control system of the aircraft and remotely land the UAV, obtaining sensitive data including mission and maintenance data. There are competing theories regarding how the RQ-170 Sentinel may have been lost. The simplest theory is that a technical malfunction caused the UAV to mistakenly land in Iranian territory [5]. A more nefarious possibility is that through a vulnerability in a sensor system, the UAVs global position system (GPS) could have been intentionally fooled into landing to a location where a hostile agent intended. This type of attack is referred to as a “GPS-Spoofing” attack [5, 6]. An example of this type of attack was demonstrated using relatively inexpensive equipment, thereby spoofing the GPS and taking complete control of UAV [7, 8].

Control security for UAVs is an area of active study and development. Flight tests have evaluated a new class of cyber-security solutions on a UAV performing a video surveillance mission. Their goal was to protect military computer-controlled remote systems from cyber-attacks. It included a new cyber-security layer called “System-Aware” which depends on detailed knowledge of the design of the system being protected. This layer of security provides both complement network and perimeter security solutions and protects against supply chain and insider attacks that may be embedded within a system [9].

UAV infrastructure is moving towards more network-centric command and control, where components are interconnected through mesh networks [10]. Some military UAV systems, more specifically the Global Hawk, already has infrastructure of this type. Public safety and disaster management UAVs are also moving to a similar network architecture for planning and communication [11]. This enables fast communication and constant environmental and asset awareness, but introduces security drawbacks. Most elements of the UAV system are interconnected through a network and if one component fails, it would affect the other components which might result in malicious behavior throughout the system.

Certain simulation-based tests with active military UAV pilots have been examined which evaluates whether the autonomous behavior could provide a secure and safe solution to an attack. The results from these tests indicate that the best course of action includes navigating to an earlier way point or switching from GPS-guided navigation to less precise, but more reliable navigation [9]. The UAV pilots presumed that a System-Aware solution could automatically detect cyber-attacks.

An interesting perspective considers scenario of vendor and an attacker as a zero-sum network interdiction game. From vendor's perspective, the aim is to determine an optimal strategy that evades attacks along the way during it's travel from source location to a destination point. It also takes the expected delivery time into consideration, thereby maximizing the security of the UAVs communication. Similarly from attacker's perspective, the aim is to choose the optimal attack locations along the path. This could result in potential physical or cyber damage which would eventually maximize delivery time. Mathematically it was shown that this "network interdiction" game is similar to a "zero-sum matrix game". This results in two linear programming (LP) problems whose solutions attains a Nash Equilibrium (NE). Solving the LP's would give the expected delivery time under different conditions [12].

One potential solution uses bio-metric information to secure communication between a UAV and its command and control station. This would allow the UAV to verify that its stated operator is issuing commands to the UAV. To the best of our knowledge, bio-metric UAV authentication has been limited to facial recognition alone. Facial authentication is problematic since it can be easily deceived by an attacker if they have a picture of the actual operator [13]. In this way, a more secure bio-metric feature is needed. We propose to use the operator's EEG signal characteristics to secure communication between the operator and the UAV.

1.2 Contribution

In this work, we propose a technique which secures the UAV communication with the ground control station using an encryption key generated using features of a user's electroencephalogram (EEG) signal. UAVs communicate using small mobile modules called XBee. XBee's secures communication using the AES (Advanced Encryption Standard) encryption standard [14]. We developed a system to generate an AES encryption key derived from an operator's EEG signal, demonstrated a safety mechanism activated in the case of a third-party attack, securing UAV communication using a bio-metric data. This entire system was validated on a commercially available UAV.

We performed the testing on a UAV, where we encrypt its communication with the ground control station by configuring the XBee's AES encryption key using an EEG bio-metric key. After configuring the XBee, we create a simple attack scenario, where the third party or attacker is aware of the key and tries to attack the communication which connects the UAV to the ground control station. We test our proposed safety solution that enables the UAV to detect that an attack has been attempted and should return to the 'home' station.

1.3 Organization

This thesis is organized as follows. Chapter 2 presents a brief introduction to Unmanned Autonomous Systems (UAS) or Unmanned Aerial Vehicles (UAV), its sensing systems, UAV attacks, and threats and a brief introduction of EEG signal and its parts. Chapter 3 describes the basic UAV communication and the types of communication attacks encountered so far (especially with XBees). Chapter 4 presents the mathematical background for EEG signal feature extraction, Randomness Extraction, and Key Generation and its usage for the UAVs. Chapter 5 presents experimental results and discussion. Finally, Chapter 6 discusses thoughts on future work and concludes the thesis.

Chapter 2

Background

2.1 Unmanned Aerial Vehicles (UAS)

UAV, UAS or drones are all aircrafts which operates without a manned pilot. It doesn't mean that the pilot is absent, rather, piloting is controlled from a ground control station. The rest of the setup is equivalent to any other manned systems or unmanned aerial systems. This type of aircraft is intended to remain inside sight of the administrator and is controlled remotely. Given the extensive research in this field, even very basic UAV systems nowadays are equipped with a flight computer and perform "intelligible" functions. These UAV systems come under the category of Autonomous Systems which has been a leading research field. These UAVs are researched to operate in changing environmental variables [15–17] which allow them to navigate in hostile environments like investigating a forest fire [18], detecting structural faults, nuclear site inspections, etc. There are various types of UAVs being used these days which serve different purposes. These are classified regarding their applications.

- **Micro/Mini UAV:** This category of UAVs are miniature and are light. Micro/Mini UAVs are mostly used in surveillance and scouting operations in small buildings. "PD-100 Black Hornet" as shown in Fig. 2.1, is a good example of UAV under this category.

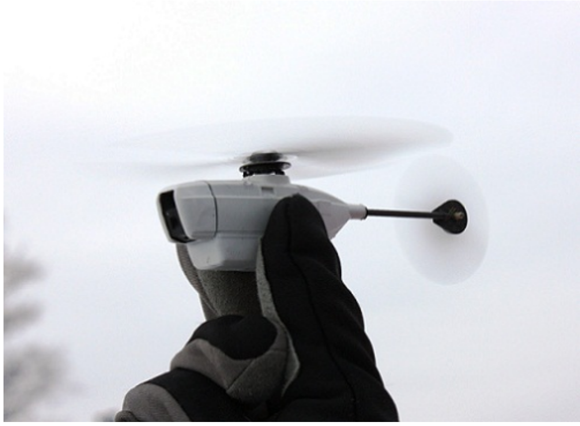


Figure 2.1: PD-100 black hornet: Mini UAV
Source: http://futuristicnews.com/nano-uav-black-hornet-pd-100-prs/

- **Tactical UAV:** These UAVs (see Fig. 2.2) include other sub-types of UAVs based on its functionality like Close Range (CR), Tactical UAVs Short Range (SR), Medium Range (MR), Long Range (LR), Endurance (EN) and Medium Altitude Long Endurance (MALE). These UAVs weigh around 150-1500 kilograms and fly at an height of around 3000-8000 meters. Some examples under this category are the RoboCopter 300.

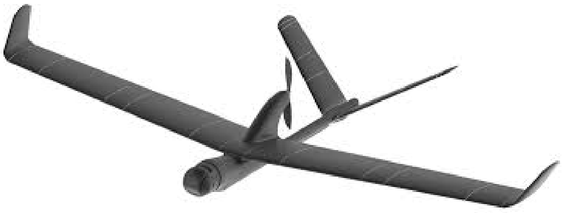


Figure 2.2: A Military Tactical UAV
Source: http://uas.wales/w-101/

- **Strategic UAV:** These UAVs (see Fig. 2.3) have characteristics like heavy weight, altitude, data link, and endurance. Global Hawk is a popular UAV in this category.

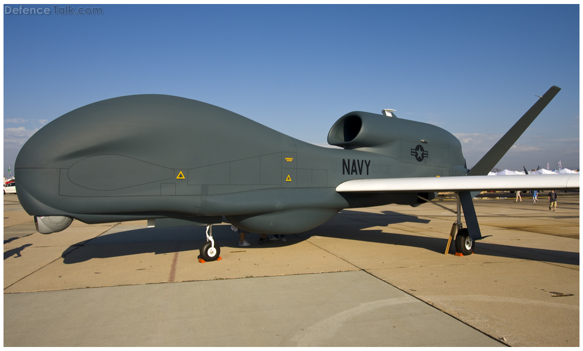


Figure 2.3: Global Hawk Strategic UAV
Source: http://www.defencetalk.com/pictures/mcas-miramar-2010/p40524-mq-4-global-hawk-uavmiramar-2010.html

- **Special Task UAV:** These consists of different subgroups such as Lethal (LET) and Decoys (DECs), Stratospheric(Strato), and Exo-Stratospheric (EXO) UAVs. An example of a DEC Special Task UAVs is the Miniature Air Launched Decoy (MALD) missile (see Fig. 2.4). The missile is incorporated with a Signature Augmentation System (SAS), which makes the missile to behave like any other aircraft and therefore confuses the enemy.

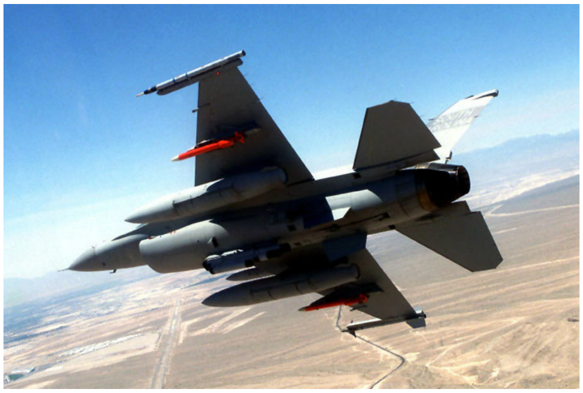


Figure 2.4: An example of MALD UAV
Source: https://en.wikipedia.org/wiki/ADM-160-MALD

Consumer UAVs (see Fig. 2.5) and Professional UAVs (see Fig. 2.6) are the most frequently used types of UAVs in modern days. These fall into the category of mini UAVs. Consumer UAVs are cheap as compared to military UAVs and have limited lifting power and less range, but can be used for variant purposes. Professional UAVs have additional equipment and are more targeted towards superior capabilities for building projects. Professional UAVs are usually costly since it comes with improved lifting power and endurance. Adding more equipment to these UAVs are usually done for a particular purpose. More specifically, consumer UAVs are mostly used by hobbyists for fun like taking videos or only flying, while professional UAVs are used to generate revenue for companies or governmental bodies (e.g. building inspection or power lines inspection). Amid this exploration, an professional UAV is utilized to improve its security.

Figure 2.5: Consumer UAV from DJI

Figure 2.6: A Professional Assembled UAV

To communicate with these UAVs various communication techniques can be implemented. All control signals can be transferred from RC (Radio Controller) to UAV. The best form of communication is the radio waves. Radio waves (see Fig. 2.7) are another sort of Electromagnetic Radiation (ER), that has wavelength more than other communication technologies like Infrared. There are different frequency bands based on various wavelength characteristics. Federal Communications Commission (FCC) in the United States regulates the use of these frequency bands. In Europe, it is governed by the European Communications Committee (ECC). The usage of specific bands is restricted since the frequency bands are controlled. Even private entities acquire license on some frequncies. Fortunately, there are frequency spectra available which are free to use that can be used to control the UAV remotely.

Figure 2.7: Commonly used Radio Modules in a UAV

Different innovations like Bluetooth or Wifi are additionally used to control UAVs. Because of their wavelength, their range is limited. So, it is mostly used to connect to a computer, which makes it convenient to pre-program the flight of the UAV. It gives the advantage of using a smart phone (see Fig. 2.8) to control the UAV and provides a low-cost approach for the whole operation.

Figure 2.8: A UAV controlled using Bluetooth

Other types of UAVs like the fixed wing craft which resemble a mini-plane. An important consideration for a fixed wing drone is the wing type and the airframe configuration. They are often made from lightweight polystyrene with a few reinforcements. Few of them have carbon frame which is tougher. The fixed-wing UAVs usually have a central payload with detachable wings for ease of transport. Some have wingspans measured in meters. However, the flight characteristics of flying wings are not usually quite as good as a more conventional design even though they're a practical proposition for rugged work.

2.2 UAV Sensing

A sensor is an equipment that detects changes in electrical, physical or other quantities in the surrounding environment and produces an electrical output or a signal that specifies the state of the environment which it is sensing. UAVs can be incorporated with sensors to detect changes in their surroundings that give the potential to navigate better and also collect important data about the object that they are inspecting.

2.2.1 Distance Sensors

These sensors are mostly used to measure the distance between a UAV and another object, without physically touching the object. Different sensors which accomplishes these tasks are:

- **Sonar distance sensing:** These sensors (see Fig. 2.9) send Ultrasonic waves and collect back the reflected waves to estimate the distance from UAV to an object. It is in the form of a pulse width of the waves. However, these sensors are affected by noise in the environment.

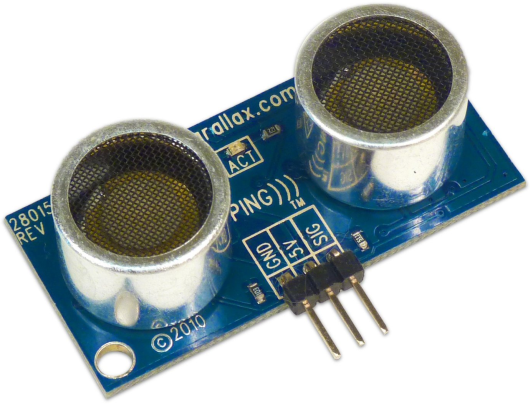


Figure 2.9: UltraSonic Distance Sensor

- **Light pulse-distance sensing:** These sensors are composed of a laser diode which emits extremely short and strong light pulses, which gets reflected by objects of interest and then received by a light sensitive receiver. It comes with an advantage of precision, accuracy and noise immunity.

- **Magnetic-field sensing:** These sensors are used for detecting the presence of Magnetic fields and objects and also helps in determination of the position of the UAV.

2.2.2 Infrared and Thermal Sensors

Infrared sensors (see Fig. 2.10), especially in cameras, are vast and are used in search and rescue, surveillance, crop and forest health, pipeline inspection, leak detection, etc. It gives results depending on the precision of the sensor. Infrared is invisible to human eyes, but one can feel the heat when there is high-intensity infrared radiation. A thermal camera can detect higher temperature areas. It can also indicate the overheating sections of electrical equipment in various devices such as substations and switch-gears. A drone equipped with these sensors can help to detect these parts from a distance, thereby avoiding the inclusion of human beings in those life threatening situations. They are also used for night vision and surveillance.

Figure 2.10: Infrared Sensor

2.2.3 Image Sensors

These are the most commonly used sensors mounted on a UAV. They can be used for various purposes such as object detection, tracking, etc. This provides a platform to integrate other AI (Artificial Intelligence) tools using images like CNN (Convolutional Neural Networks) for complex environment understanding.

As the growth of UAVs continues, demand for accurate sensing and obstacle detection technologies will continue to increase. Lidar is a device that allows an accurate 3D map of the drone's surroundings. It also shows excellent performance in adverse weather conditions and can be easily integrated onto modern UAVs. Lidar has much more flexibility when it comes to its specifications, integration and form factor. Lidar surpasses other technologies in use today for UAV applications such as ultrasound and stereo cameras, both of which have an insufficient range. Although stereo cameras can create very high-resolution images, they are more sensitive to extreme or changing lighting conditions. Certain applications like a high-resolution mapping of terrains and Google self-driving cars use high fidelity sensors such as scanning Lidars. These Scanning LiDAR systems (see Fig. 2.11) have a great potential in generating detail features of the environment,(see Fig. 2.12), but on the downside it is expensive. Other features like a strict requirement to handle the device properly make the device hardly viable for integration into commercial UAVs.

Figure 2.11: Velodyne Lidar Sensor

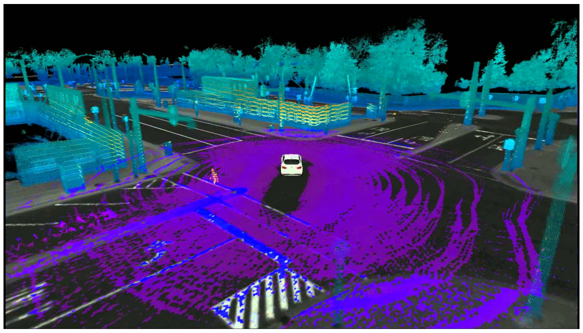


Figure 2.12: Velodyne Lidar Sensor Sample Data

Despite the limitation of lidar, several research institutes and industries are actively using it for various applications like Flood Modelling, Forestry Management, and Planning, Pollution Modelling, Mapping and Cartography, Urban Planning, Coastline Management, etc.

Additional research is being actively pursued in the implementation of path planning using visual cues. A UAV is equipped with the camera. One approach suggests an image based optic flow algorithm including IMU (Inertial Measurement Unit) data is incorporated with Extended Kalman Filter (EKF). It is further integrated with a non-linear controller to accomplish 3D navigation. The IMU measurements were merged with optic flow information to estimate the aircrafts ego-motion and the depth map.

Another technique presents an approach to autonomous exploration where the system takes advantage of a stereo camera and an inertial sensor for mapping and pose estimation. The data shows the occupancy map, raw image data, and the dense point cloud. A visual pose estimation system using multiple cameras mounted on a UAV, also known as Multi-Camera Parallel Tracking and Mapping (PTAM) are also quite popular among various approaches.

2.3 UAV Threats and Attacks

There are various methods where a UAV can be attacked and be used for other malicious purposes. These methods could be classified as Logical and Physical attacks. Physical attacks requires a physical contact with the UAV using some tools, where as Logical assaults focuses on the navigation,position and other information which are utilized by the UAV.

2.3.1 Physical Attacks

In an unmanned aircraft, as the navigation is entirely based on sensor data, it can be affected via logical attacks which could eventually affect its navigation. Another way to deal with bringing down a UAV is to utilize another Interceptor Drone or a UAV. It could be done by flying the Interceptor drone straightforwardly into the target UAV. It provides the benefit of protecting other areas where there is no inclusion of third parties. Another approach directs the second UAV to equip themselves with some tools which brings down safely the target UAV (causing harm only to its functionality as shown in Fig.2.13. Other physical assault vectors, for example, magnetism, wind or powerful microwave beams are additionally utilized. A high-powered microwave beam is expensive as well as it requires the knowledge and expertise to develop, so they might not be applicable for few sectors. These applications are most common for military UAVs. Another approach suggests a UAV equipped with appropriate transport capabilities and power sources that can send a high-powered microwave (see Fig. 2.14) to the target UAV and damages its internal circuitry. This equipment could be installed at a lower cost, thereby avoiding the use of a military UAV. This approach seems very plausible to take down a UAV.

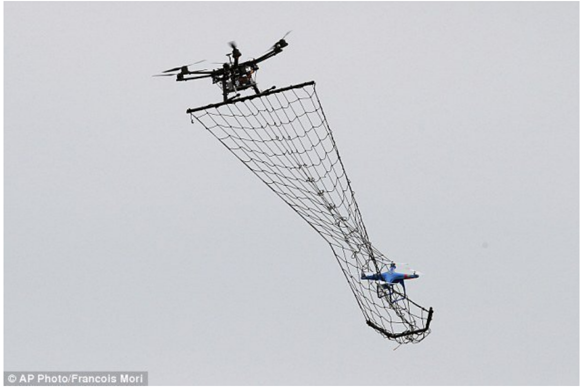


Figure 2.13: Take down a UAV
Source: http://www.tpe-les-drones-1es1-68.webself.net/la-legislation

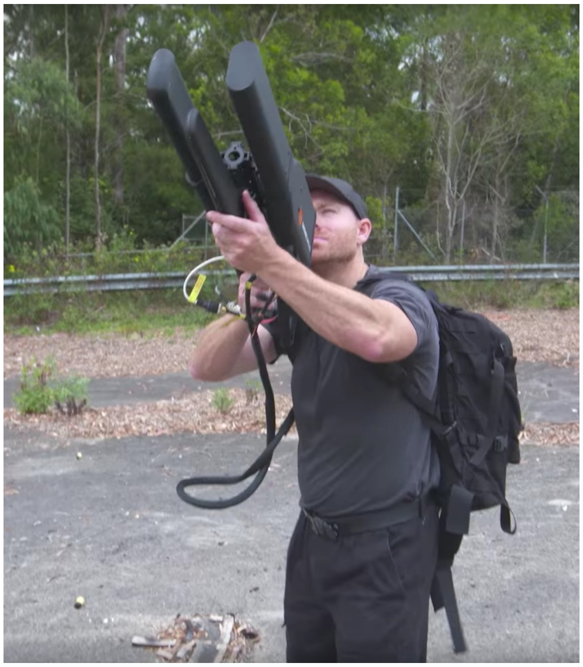


Figure 2.14: Drone Gun that sends waves to take it down.
Source:https://plus.google.com/+AustinPolin

2.3.2 Logical Attacks

UAVs use multiple sensors and technologies that aid the operator and help him to determine its altitude, flight speed, and position, etc. Since UAVs are unmanned, their navigation completely relies on those parameters and commands which they receive from the base station. An attacker can exploit these parameters or could even change the information for his control of the UAV. Eventually, the UAV will obey the attacker's signal. It has gained significant attention lately and needs to be addressed.

Forging Control Signals

In the event that it is an open channel (unencrypted WiFi or radio), it is very simple to send fake information to the UAV without an application layer encryption or any authentication methods set up. Forged signals are eventually assumed as the correct command signals to the UAV. Even the Radio Controller (RC) will accept and process these signals without knowing the difference between the original signals that primarily should come from the ground station. It results in obtaining full control over the UAV to navigate the UAV further. If standard protocols are used for the communication with the UAV, they should also implement proper authentication. Changing protocols without using the available security features can lead to reverse engineering protocol, making it easy to gain control. This type of attack poses a serious threat since the UAV will still obey the commands from the ground control station. This might lead to change in control signals of the UAV.

Denial of Service

For receiver and sender to communicate, they need to be on the same channel. If an attacker is aware of the channels being used, he can surge these channels with junk information which keeps the real control signals originating from a ground control station to be received by the UAV. Another attack uses de-authentication packets to a Wifi network to prevent a connection. Both of these types of attack would result into the end of communication.

Denial of Service attacks also occurs in GPS frequencies which affect GPS data. It is termed as GPS jamming. The channels are overflowed with faulty information to keep away from the real ground station signals going through. The devices needed to perform these types of attacks are cheap and easily available. Evidently, the range of GPS jammers are limited, but professional equipment has a potential to reach far ranges, which allows the attacker to jam the GPS frequencies.

Replay Attack

It is possible to record signals even if the data stream is encrypted, which could aid the attacker to replay these signals towards the target. It will aggravate much faster if the protocol does not utilize cryptographic methods to confirm the legitimacy of the messages. By analyzing the response from the UAV, an attacker could easily estimate the mapping between the message and command. The attacker could gain control of the UAV by analysing all the possible commands. During this situation, the UAV will also still obey the actual user at the ground control station, but it might make controlling the device difficult.

Spoofing

Spoofing means forging signals, and it usually appears in the higher frequency of the communication channel. The victim assumes the attacker's reproduced signals instead of original ones as actual input. The worst case occurs when GPS spoofing takes place. It benefits the attacker by tricking the victim into believing that the victim's UAV is navigating to the correct path, but in reality, the UAV is taking the path that the attacker intends to. GPS Spoofing poses a drastic threat to UAVs. Military GPS system utilizes advanced encrypted and authenticated signals to send data from the satellites to the receivers. Spoofing is more common in civilian GPS, since they do not use military GPS. An attack on military GPS is also possible by adding delays to the system. It is called selective-delay attack.

Wifi and Bluetooth Attack

Wireless communication occurs through an encryption standard named Wired Equivalent Privacy (WEP). In particular cases which can be used to obtain the encryption key, this encryption standard is vulnerable. Recent encryption protocols like WiFi

Protected Access (WPA) and WiFi Protected Access 2 (WPA2) have multiple layers of security. However, if certain conditions are met, it can be hacked.

There are software tools readily available that perform this type of attack in a wifi network. Few commands in Linux systems like “airodump-ng mon0” lists all available network traffic. Since the attacker is now aware of the network traffic, he can select the traffic filters for further investigation. Another software tool called “airrack-ng” can be used to capture all network traffic sent. Address Resolution Protocol (ARP) can be used to generate random packets which could be injected to the network of interest.

To prevent these attacks, one key thing to note is that the password to authenticate the WiFi access point should be volatile and be changed for every device. The use of defaults passwords is dangerous, and the encryption becomes vulnerable to potential attacks. Another way to mitigate this attack is to have another hardware with different encryption. An attacker might not comprehend the technology nor would be able to develop equipment to exploit the encryption layer underneath.

Attack through Video Link

For an attacker, live video feed is vulnerable to attack. However, It depends on the link the ground control uses for live video feed. A good example can be the attack on U.S. Predator UAV for the live video feed in Iraq. Choosing between different products which use different frequencies to transmit the video signal can be helpful to mitigate the problem. Higher frequencies could be used for transmission, since video transmission requires larger bandwidth. A setback we need to look is that higher frequency data comes at the cost of the shorter range of the signal. One trick to mitigate the problem is to split the video data at multiple channels of lower frequencies. The receiver would have to listen to all the channels to reconstruct the original data.

Attack through Flight-Planning Software

The flight plans are conducted on the computer mounted on a UAV. Before the start of a mission, the flight path data is sent from the planning software to the computer mounted on the UAV. The planning software could be on a computer or a mobile device. It is important for the UAV's flight computer to be connected to the planning software throughout the mission. Most configurations of the software provide this functionality. During this time, an attacker can insert malware to the computer, giving him the access to the software through which he can reprogram the flight computer path and aid the UAV to his intended path. An attacker getting access to the UAV through the internet is one of the serious problems that need to be addressed. It happens, if the internet is connected to the ground station during the flight, then the attacker can change the flight coordinates of the system and redirect it to a different location as per the attacker's choice.

2.4 Brain EEG Signal

The brain EEG signal has been of active research, and its fascination grew after its discovery by a German scientist named Hans Berger [19]. After its discovery, new methods for its exploration have been found. They are generally classified into two groups named Invasive and Non-Invasive. Invasive methods need devices composed of electrodes which are tested on humans or animals, which measures single neurons voltage potential or local field potentials. The non-invasive approach utilizes magnetic resonance imaging (MRI), and the EEG technology to gather all the readings. Both approaches give different outlook which enables us to understand the functionalities inside the brain and observe the electrical activity generated by the neurons through these functions. For example, pairs of electrodes made of silver are used to measure this electrical activity. Since the voltage difference between these electrodes is weak (30-100mV), these are amplified for data collection. When neurons communicate current flows. The electrical discharge caused by rapid on and off Na^+ and K^+ ion channels in the neuron membrane, triggers an event called action potential. If it crosses a certain threshold, the neurons fire. Evaluating this “fire” results in a brain activity.

2.4.1 Bands of EEG signal

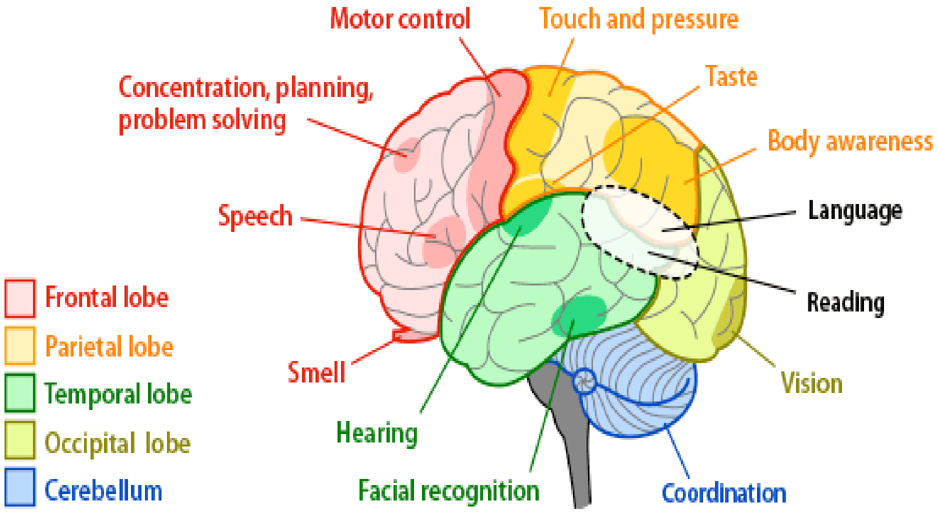


Figure 2.15: Different Parts of the Brain
Source: https://askabiologist.asu.edu/brain-regions

Berger discovered that different electrical frequencies are generated through actions via different stages of consciousness. The different parts of the brain which are involved in generation of different frequencies EEG signals is shown in Fig. 2.15. This was later confirmed by observing and recording different subjects who were performing a different task, like solving analytical problems. This resulted in different parts of EEG wave. The different parts of brain EEG signals are described below.

2.4.2 Gamma Waves (< 25 Hz)

Gamma waves reflects how the human consciousness functions. It is originated from the Thalamus region of the brain. Gamma wave are generated, as the electrical activity traverse in the brain from front to back, about 25 times per second. Sample Gamma wave are shown in Fig. 2.16. If the thalamus is damaged, then the wave generation stops, and the patient goes into a profound coma state. In a way, it links to other parts of the brain. This discovery and research have directed to solve the binding problem, which describes two basic sub-problems. Firstly, how the brain separates complex patterns coming from sensory input to classify them as "distinct" objects. In simpler terms, what neural activity helps to separate two distinct color objects coming from the same visual sensor (human eye). Secondly, how do different objects, background, and emotions combine to form a collective "experience" [20], [21].

Beta and gamma waves combined relate to perception, attention, and cognition. Gamma waves are also seen as a form of neural synchronization which is derived from visual cues in a conscious state. Gamma waves are also found active during rapid eye movement sleep which involves visualizations. Gamma wave is an essential component for learning, memory and processing information. People who are mentally challenged tend to have lower gamma frequency than the average. Even though Gamma waves have the highest frequency of other waves, it has the lowest amplitude. Gamma waves are trainable, which means it's frequency and amplitude can be increased though regular meditation. Research on regular meditation practitioners has suggested an increase in gamma wave activity. It may also explain heightened sense of bliss, consciousness and intellectual acuity [21].

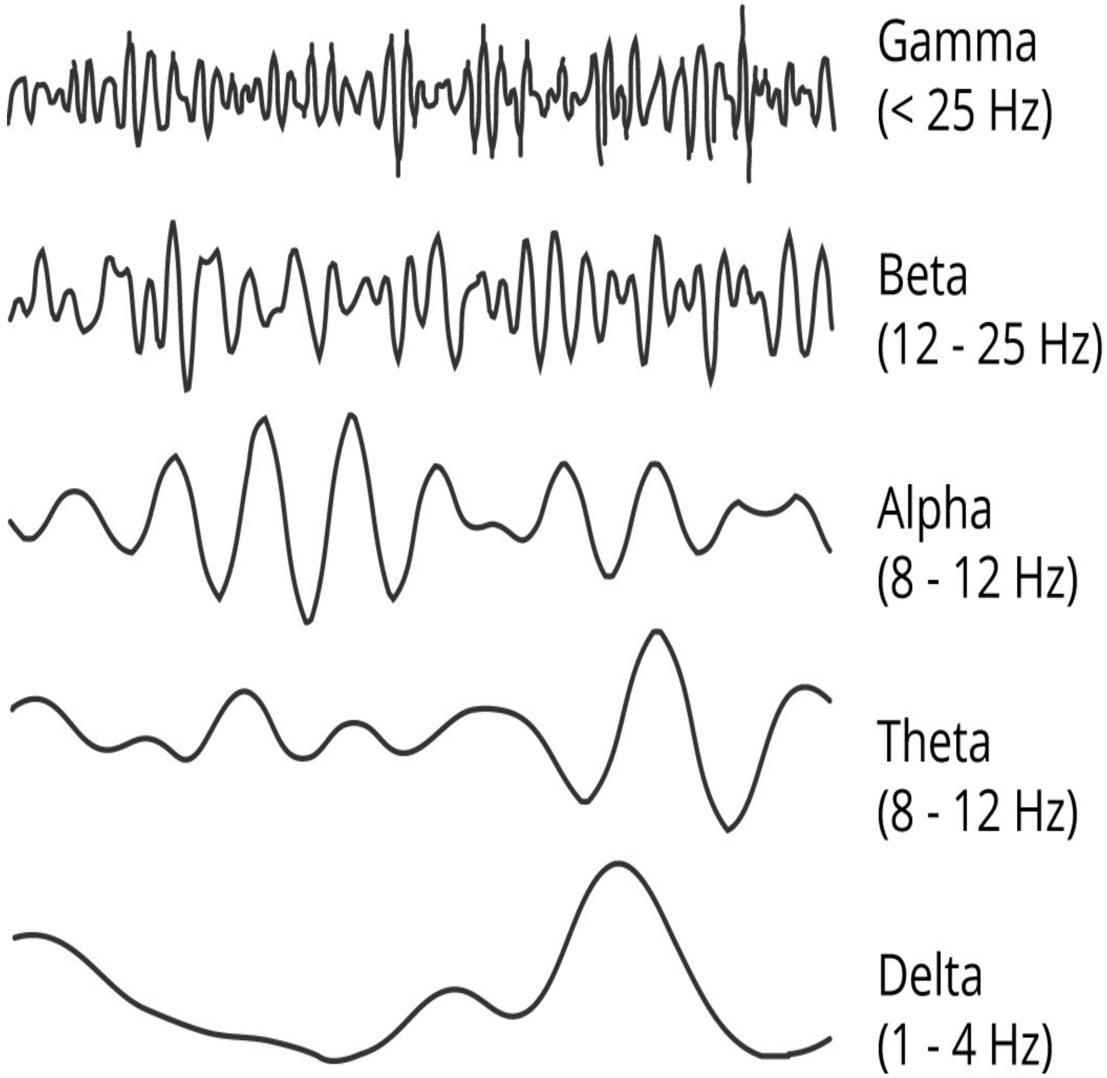


Figure 2.16: Parts of EEG Signal
Source: https://imotions.com/blog/what-is-eeg/

2.4.3 Beta Waves (12-25 Hz)

Beta waves play an important role in this thesis. Beta waves are classified into three types: Low Beta Waves (12.516 Hz), Beta Waves (16.520 Hz) and High Beta Waves (20.528 Hz). These waves are generated in the central and frontal areas of the brain and are usually small and fast. Sample Beta wave are shown in Fig. 2.16. It is mostly associated with focused concentration. A study claims that beta waves are actively seen when there is resisting or suppressing the movement. One research shows a significant increase in Beta power among alcohol dependent subjects. However, the good amount of beta waves can be generated when we practice concentration. If it is hard to concentrate, certain tools like Beta binaural beats help to simulate the brain to go to concentration mode [21]. When two different pure-tone sine waves, having frequencies lower than 1500 Hz, and has a difference of 40 Hz between them is presented dichotically (one on each ear) to a listener, then the listener percieves it as an auditory illusion which is called binaural beats.

However, there is a downside to this. People who often work a lot, for them would result in stress and depression. So concerning brain waves, these people always emit Beta brain waves, even when they are asleep. So increasing the Beta waves through the binaural beats can be fatal for them. Certain experiments have shown that in fact, these people have more Beta activity and less Alpha and Theta waves. So, it is vital to have a proper balance of Alpha and Beta waves. It could be achieved by taking breaks from the Beta binaural wave, stop thinking and listen to Alpha binaural waves [22].

Some researchers claim that Beta waves are directly linked to the conscious mind [23]. During meditation, it is very advisable to make the meditative experience alive or conscious by sharing or writing it down.

2.4.4 Alpha Waves (8-12 Hz)

Alpha waves are slower and are related to relaxation and disengagement. It is detected at good amount when we are thinking of something peaceful. It mostly originates in the frontal lobe and the back of the brain. Sample Alpha wave are shown in Fig. 2.16. A healthy balance of all brain waves is important for the brain. For people who are more stressed, there is significant Beta waves and less of the other waves. Similarly, Alpha waves in excess are not healthy either. A study shows that an individual “become slower” Alpha waves are also called as “relaxing waves”, which apparently plays an important part in human mind. It is shown that they act as a bridge between conscious (Beta waves) part and sub-conscious (Theta waves) part of the human mind. Feelings, information, memories, and creativity present deep inside a person’s mind cannot become conscious if there is no connection between the two states of mind [22].

Some research shows that alpha waves can also occur during the initial stages of meditation. Just like Beta waves, Alpha waves can also be simulated using Alpha binaural beats. However, it varies from person to person. Among the benefits of Alpha waves, the most obvious result is the feeling of relaxation or peacefulness. It also has some effects on the body like heart, where it helps slow down the heart rate, which can be effective for people with heart problems. It also has a positive influence on memorization and learning speed [21].

The measurement of EEG brain waves of a person would show a significant amount of theta brain wave activity and Beta brain activity, but no Alpha waves.

2.4.5 Theta Waves (4-8 Hz)

Theta waves are associated with day dreaming, inefficiency and the lowest of these theta waves signifies the thin line between sleep and awake states. Sample Theta wave are shown in Fig. 2.16. It is sourced from negative thoughts or emotional stress like disappointment and frustration. However, it has also been associated with creative aspiration, unconscious material, and deep meditation. For adults, high amount of these waves are considered abnormal, also related to AD/HD (attention-deficit/hyperactivity disorder) [22].

Theta waves also occur at inactive thinking. It also happens nearly in meditation and dreams. Apparently, these activities require mostly shutting off the brain. However, there is one exception though. People with creative thinking or activity like drawing painting or composing music tend to have more theta brain activity. So, we could even say that creativity comes from the subconscious part and thus the mind inherently has a significant amount of Theta waves [22].

As mentioned, Theta waves play a very important part in human mind. Stressed out people tend to spend much time in Beta brain waves making them deficit in Alpha and Theta waves which results in its inability to be more creative and more relaxed. Also, people feel as not being able to connect to their emotional selves.

So, based on the above discussion, it is healthy to have a good amount of Theta waves to be able to access the “sub-conscious” mind. It also plays an important role in self-healing and learning abilities [21].

2.4.6 Delta Waves (1-4 Hz)

Delta waves are the lowest frequency waves occuring when we are sleeping. These waves are not meant to occur during awake state. So, if they occurs in good amount in the awake state, they indicates physical defects in the brain. Some research suggests that movement can result in artificial Delta waves, but it is yet to be confirmed [22].

Delta waves are also linked with the unconscious mind. Sample Delta wave are shown in Fig. 2.16. They occur in humans when they are asleep not dreaming. They are also found during deep meditations, however, it’s challenging to remain conscious during these meditations, and these meditations are rather harder to reach. Delta waves are also significantly associated with human intuition. People with AD/HD have significant Delta waves activity [21].

2.4.7 Why Beta waves?

Our primary purpose of using Beta waves for our research is that Beta brain waves are associated with normal waking consciousness, logic and critical thinking capability of a user. This ensures that the user is awake during data collection and he/she is performing a mental task. The user is aware that the data is collected for key generation.

2.5 Summary

This chapter provides a brief overview of functioning of the brain. It also provides the idea of the different parts of the EEG signal which is emitted from the brain at different states of mind. Finally, we explained why do we use Beta waves for our work.

Chapter 3

UAV Communication and Types of Hack/Attacks

3.1 Xbee Communication

Figure 3.1: XBEE Radio Module: Essential component for this thesis

For a UAV communication system several things need to be taken into consideration.

- Transmitting device.

- Receiving device.
- Environment for communication.

In wireless communication, transmission is done by a transmitter whose role is to feed the antenna a signal for transmission . With a certain signal strength, the radio transmitter encodes the data and translates it into RF waves. The receiving antenna receives the signal and the receiver decodes the incoming data. The task of accepting and decoding specific RF signals while rejecting unwanted or redundant data is also performed by the receiver.

XBee is one of the mobile communicating device that can be mounted on a UAV for it's communication with the ground control station. XBees only communicate with other XBees. It is shown in Fig. 3.1. XBees operates on Zigbee protocol. Zigbee follows the 802.15.4 international standard. Digi International developed XBee which is a product line and a brand name. In the bottom layer XBee also has the IEEE 802.15.4 standard, but also they have their own suite of protocols on the top. For greater transmission range Zigbee also allows mesh networking functionality in addition to 802.15.4 standard. It helps to forward the messages from one node to other to reach the destination node through the network. The frequency band used for transmission determines the transmission speed. Since it is a low powered device, it mostly acts as an embedded device. XBee devices can also run Zigbee compliant software, but we need to flash it with the Zigbee firmware which results in losing the benefits of XBee's but giving full compatibility with other Zigbee compliant devices. Digi explains regarding the use of proprietary multi-point and Digi mesh protocols as well as the use of proprietary modulation. For example, the manufacturer of XBee 868LP devices which uses proprietary multi point protocols specifies the device's range to be 40 km. To be configured in the network, XBee devices don't require coordinator and end device. In addition to that, for checking whether the channel is independent and the data can be sent, XBee 868LP chip utilizes Listen Before Talk (LBT) and Adaptive Frequency Agility (AFA). This capability allows multiple different networks to coexist which allows dynamic data transfer. Another framework called Killerbee has both Zigbee and 802.15.4 based networks. However, to run the framework specific hardware is required. Killerbee also provides the functionality of integrating itself with GoodFET hardware debug tool. Other research has indicated that GoodFET mentions a common issue in Texas Instruments and Ember chips to obtain RAM (Random access memory) on a locked chip.

To test the experiment 2 XBee modules are needed. The below-mentioned data are needed to transfer the data from one XBee to another, also called as Point-to-Point data transfer.

- PAN ID
- Channel
- Baud Rate
- Device High (DH) address
- Device Low (DL) address

The device comes with the default values for Channel, Baud rate, and PAN ID. It allows the change of DL and DH address for the customer to test the device quickly. The user needs to only change the DH and DL address of the other device so that it can communicate. The UAV manufacturer doesn't modify any parameters except for the DL and DH parameters. It utilizes default values for the rest of the parameters. There are different possibilities to get information about the DH and DL values.

- Physical access: Printed on chip's cover.

- Physical access: Reading the chip's memory.
- Software Defined Radio.
- Brute-Force.

Once the parameters are known by the attacker, fake data can be sent, and different types of attacks can be performed. However, this simple scenario is not fairly possible, since the chips are hidden in the user's hardware, and also the attacker is unaware of the physical address.

3.1.1 Xbee Modes

Xbee has two different working modules. The initial setup for basic communication is named the transparent mode. The chip reads the memory and adds the DL and DH address with the preamble and network ID. To communicate with another chip, the DH and DL values needs to be changed.

Another mode that exists in XBee is the Application Programming Interface (API) mode. In API mode, the receiver expects payload plus API-frame which incudes payload. The destination address can also be specified along with other parameters in the API-frame. This provides a benefit of not editing the DH and DL address every time the destination needs to be changed for the payload. Simply the API-frame can be sent to the device with the destination address specified in the frame itself.

3.1.2 Broadcast Mode

The chip disposes of all packets that are received. However, it contains another address in the destination address in addition to its own. In any case, there is an alternative found through further investigations. The sending and receiving of broadcast packets is also a functionality in XBee devices. In addition to that, regardless of the sender's packet address, each received packet is viewed as legitimate by the chip. It will return an acknowledgment regardless of the possibility that the receiving chip has an alternate destination address put away in its memory.

A tool called "Node Discovery" uses this feature in the software XCTU. Other chips which are using the same preamble and network ID if are in range, then the user can discover it. This could come as great advantage for an attacker since a mere acknowledgement can uncover the destination address.

Changing the preamble and network ID is the main step to complicate detection of the used network. When the parameters are received, the chip's firmware checks the packets. If preamble or network values ID does not match with the ones stored in the memory, then the frame is directly discarded rather than forwading it to the serial connection. The idea behind the implementation is to avoid overloading the serial output of the chip with data coming from other networks and to avoid interference with other networks in range.

The parameters are checked in the firmware of the chip upon supply of a packet. On the off chance that the estimations of the preamble or network ID are not coordinating to the ones present on the chip, the frame is not sent to the serial connection, but rather specifically disposed of. This was initially actualized to keep from interfering with other network in range and over-burdening the serial output of the chip with information from different networks. It is impractical to compel the chip to forward such packets, since the chip is restrictive and only official firmware can be installed.

XBee gives a probability of changing the parameters remotely. The remote chip will temporarily use the newly received address by sending the correct API frame having the DH (or DL) parameter with another value. Another API frame with the Write (WR) command is required to persist the change. The attacker can also remotely change the DH and DL addresses of any given XBee chip and can redirect the data transfer. When this is done, the attacker can control the channel.

3.1.3 Attacks on Xbee communication

Denial of Service (DoS)

For performing a DoS attack, diverse alternatives were proposed. One method is to jam the connection, by over-burdening the utilized channels. Another abuses the conduct and capacities of the protocol and technology being used.

Another alternative will be clarified in more noteworthy detail. At first, the attacker plays out the previously mentioned brute-force attack on possible conceivable network ID and preamble combinations. By sending communication packets utilizing every single conceivable combination, the attacker will locate the dynamic devices. Furthermore, the attacker utilizes an alleged "Remote AT Command" packet, to edit the targeted XBee chip's destination address. The payload of this data comprises of the hexadecimal encoding, trailed by a latest parameter present in the following field. The address of the attacked XBee chip will be changed temporarily by this packet alone. This recent address will not change/overwrite the current address and yet at the same time be utilized as long as the power supply proceeds.

The old address is used again upon restarting the device. Another "Remote AT Command" is needed in order to persist the changes. The hexadecimal encoding for the "WR" instruction is in the second packet. The chip will dependably utilize the new destination address, even after a reboot. The connection will be interrupted if the destination address of the telemetry box and the UAV chip is set to an arbitrary value. The need to address every chip separately can be removed by "Remote AT Command" which can send evry other packet as a broadcast packet. The addresses ought to be set to an arbitrary value.

Man in the Middle Attack

Two things are required to meddle with the flight PC of the UAV: appropriate payload and access to the communication channel.

For the second case, instead of using a random value as used in DoS approach, the destination addresses of the target chips should be configured to the attackers address. At this point, the attacker now can tune into all communication that occurs between the attacked chips. This type of attack is termed as man-in-the-middle.

For the first case, by figuring out the first information which is sent in the channel, the right payload can be known. From this, the attacker can now tune into the communication, subsequently, he may have the capacity to understand the data of the payload and remake the commands. In any case, the user can control the UAV since the producer of the UAV furnishes its customers with a tablet and also a pre-install the application. Since the methods and the used commands implemented in the software can be reverse engineered, the attacker can reveal this information and use it as a payload.

3.1.4 Counter Measures

Hardware Encryption

Hardware encryption provides another feasible option for encryption, since the built-in encryption for the XBee is slow. For allowing decryption and encryption of the incoming serial data, this hardware requires to be present in front of the XBee chip. In transparent mode, which is the original setup, the data gets simply forwarded and doesn't need further interpretation. However, in API mode this approach is impossible, since portion of the data that is given to the XBee chip, requires analysis by the chip. If the data is encrypted, the XBee chip might not be able to estimate the transmission parameters (DL and DH address). It functions similarly to XBee encryption but has dedicated functionality.

Application layer encryption

Another possibility is the application layer. However, no further encryption is required if the exchanged information is now encoded upon it's arrival in the XBee serial interface. Also, since the measure of information remains the same, no execution issues on the XBee chips are expected. Application layer encryption adds a logical layer on the top of the utilized physical layers. Consequently, this would alleviate classification dangers made by one weak connection as well as every single weak connection. Choosing between symmetric and asymmetric encryption is a further step. The upsides of symmetric encryption are performance and simplicity. Two parties that have not met before can have asymmetric encryption. Since the keys can be activated in both the devices in its first use, this case is similar to symmetric encryption, Additionally, asymmetric cryptography is costly due to handshakes and exchange of certificates. Because of this only symmetric encryption is a practical solution.

During symmetric encryption, the data is encrypted prior leaving the computer and decrypted at the planning software and vice versa. This concludes that the processing power for the encryption keys is sufficient and the encryption keys are saved in the device.

Xbee on board encryption

Encrypting the channel would ensure the confidentiality of the information. XBee provides such a feature as mentioned earlier. This would be a convenient method to use since it doesn't take much to implement. It avoids the attacker to modify the internally saved data remotely without having a prior knowledge of the correct encryption key. XCTU software can be utilized to store the encryption key. A user can use the XCTU software to store the encryption key in the XBee chip allowing encrypted communication. Of course, the key should be the similar on both sides, else the packets will be easily removed. The encryption and decryption are performed using AES-128, since both sides use the same key. The encryption occurs symmetrically. It's performance might get affected since the chip requires time to encrypt and decrypt the payload. So, link layer encryption seems to be the most practical approach to the problem, but cannot be applied to the above case.

3.1.5 Additional approach

Man-in-the Middle attack is still possible, if the on-board encryption as mentioned earlier is not used. This will not allow the attacker to read the packet content. In this situation, DoS attack can be performed since the change of address is still possible as the XBee encryption is disabled. There is no way out of it, and the encryption needs to be setup which is not possible in DoS attacks. So, an alternative needs to be found out that provides the same bandwidth and functionality as the XBee although doesn't allow changes of the internal parameters. One option that might work is using duplicate channels by using two Xbee Communication channels with enabled encryption on both of them. It only gives an logical approach to divide the communication and reassemble.

3.2 Summary

This chapter presented an overview of different types of UAVs sensors equipped on the UAV and different types of attacks on a UAV that have been encountered so far.

Chapter 4

Proposed Methodology

This chapter describes the background of our proposed method for securing a UAV. As an outline, we have proposed a technique which secures the UAV communication to the ground control station using an encryption key generated using features of a person's electroencephalogram (EEG) signal. Next, we present the mathematical background for extracting individual characteristics of the EEG signal. Further, we describe briefly on how the XBee communicate with each other and how the data are further translated into control signals. We finally propose a system to generate an AES encryption key derived from an operator's EEG signal and have demonstrated a safety mechanism which gets activated in the case of a third-party attack, thus securing the UAV communication using a bio-metric information. This entire system was validated on a commercially available UAV.

4.1 Mathematical Background

4.1.1 Lengendre's Polynomials

Legendre Differential equation is given by:

$$\frac{\partial}{\partial x}[(1-x^2)\frac{\partial}{\partial x}p_n\left(x\right)] + n(n+1)p_n(x) = 0. \tag{4.1}$$

where n is a real-number, i.e $n = 0, 1, 2, 3, 4, ...$ and x lies in the interval of -1 and 1. $p_n(x)$ are the Legendre's polynomials.

Legendre's polynomials are solutions to the above Legendre's Differential equation. If n is an integer, then at $x = 1$ and $x = -1$ the solution is regular and the series for this solution ends.

The polynomials are computed using Rodrigue's formula:

$$p_n(x) = \frac{1}{2^n n!}\frac{\partial^n}{\partial x^n}[(x^2-1)^n]. \tag{4.2}$$

One significant property of Legendre's polynomial is that it is orthogonal with respect to L2 norm in the interval $-1 < x < 1$. Few more important properties of Legendre's polynomials are:

- Legendre polynomials are symmetric or anti symmetric.

$$p_n\left(-x\right) = \left(-1\right)^n p_n\left(x\right). \tag{4.3}$$

The n-degree equation used for fitting data is given by:

$$y\,(x) = a_0 + \sum_{1}^{n} a_i p_i(x) \tag{4.4}$$

4.1.2 BCH (Bose-Chaudhuri-Hocquenghem) codes

A class of cyclic error-correcting codes can be formed by BCH codes. Polynomials on a finite field can be used to construct BCH codes. During code design, BCH provides an advantage of absolute control over the number of symbol errors correctable by the code. To be more precise, multiple bit errors can be rectified using efficient design of BCH codes. Another algebraic method named syndrome decoding aids easy decoding of BCH codes. This helps in designing a simple decoder for BCH codes using light low powered devices [24].

Assume that there is a prime power (A prime power are those prime numbers that is exactly divisible by one prime number) q. Let there be another 2 positive integers d and m where $d \leq q^m - 1$.

So, on a finite field $GF(q)$ with distance of at least d and code length of $n = q^m - 1$, a primitive narrow-sense BCH code can be constructed based on the below mentioned methodology:

Let the initial element of $GF(q^m)$ be α. Let i be any positive integer. Let the minimal polynomial of α^i over $GF(q)$ be $m_i(x)$.

Then, the least common multiple $g(x) = lcm(m_1(x), ..., m_{d-1}(x))$ is called the BCH code generator polynomial.

It is proven that $g(x)$ is a polynomial with coefficients in $GF(q)$ and divides $x^n - 1$. So, the polynomial code as given by $g(x)$ is a cyclic code. The above procedure is called a primitive narrow sense BCH code.

BCH codes are derived from the above formulas, by adding two more steps:

- α can be non-strict since it is a primitive element of $GF(q^m)$. If so, then the code length varies from $q^m - 1$ to $ord(\alpha)$, as the order of the element α.

- The consecutive roots of the generator polynomial can go from $\alpha^c, ..., \alpha^{c+d-2}$ to $\alpha, ..., \alpha^{d-1}$

Some properties of BCH codes are:

- The generator polynomial of a BCH code has degree at most $(d-1)m$.
- A BCH code is cyclic.
- A BCH code has minimal Hamming distance at least d.

BCH Encoding

BCH encoding can be done in the following way.

- For (n, k) BCH encoder the k message bits are applied to parallel to serial shift register.
- Using those message bits, it computes the parity bits and sent to serial to parallel to serial shift register.
- The parallel to serial shift register's ouput is sent to (n, k) encoder module.
- The the parity bits are concatenated to original bits to get $n - bit$ encoded data.
- The whole encoding process needs n clock cycles.

BCH Decoding

BCH decoding can be done in the following way.

- Assume that we have a (n, k) BCH decoder.

- From the received codeword $r(x)$, Syndrome value S_i $i = 1, 2, ..., 2t$ can be calculated.
- Determine the error location polynomial $s(x)$.
- To correct the errors, the roots of s(x) should be found.

Advantages of BCH codes

- It can be easily decoded with the syndrome decoding method.
- It requires a simple hardware to function. So it obviates the use of complex systems to perform the decoding procedure making it easy to implement on a low powered device.
- It is highly flexible, allowing it to control over block length and acceptable error thresholds. This allows for custom code to be designed on a given specification.
- BCH are useful in theoretical computer science.
- Easy to implement in hardware.
- Widely used encoding and decoding technique.

Disadvantages of BCH codes

- It works on complex and iterative decoding procedure.
- It's hard for decoder to decide whether the decoded package is false or not.

4.1.3 Universal Hashing

From a family of hash functions, we select a random hash function which have a certain mathematical property. It could be understood simply as follows: Assume that we intend to map keys from some universe U into m bins. The algorithm aims to manage some data set $S \subset U$ of $|S| = n$ keys. Normally, the primary aim of hashing is to obtain a low number of collisions (keys from S that maps to the same bin).

The approach to this problem is to select a function randomly from a family of hash functions. A family of functions $H = \{h : U \rightarrow [m]\}$ is called a universal family if $\forall x, y \in U, x \neq y : \underset{h \in H}{Pr}[h(x) = h(y)] \leq \frac{1}{m}$.

In simpler terms, any two keys in the universe clash with a probability of at most $1/m$, if the hash function h is taken randomly from H.

4.2 Getting EEG signal

We used a Mindwave headset from NeuroSky to extract the EEG signal of a user. It is shown in Fig. 4.1.

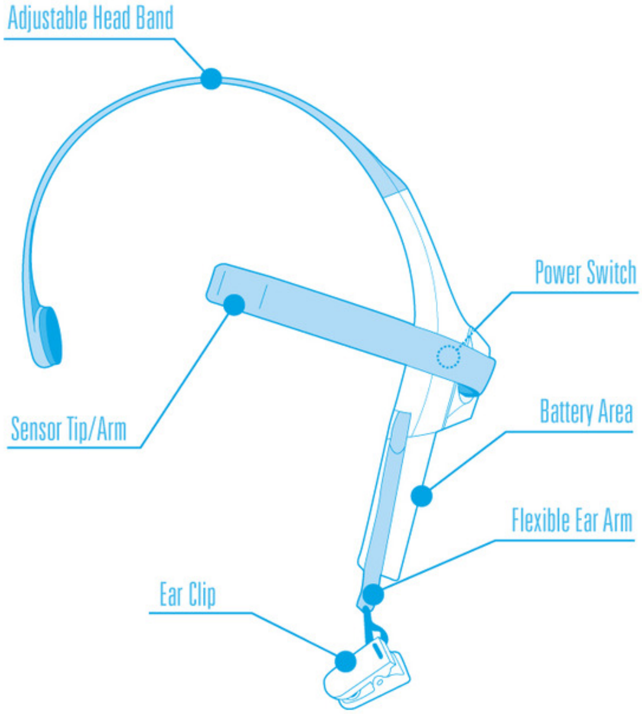


Figure 4.1: Neurosky Mindwave Sensor
Source: http://support.neurosky.com/kb/mindwave/mindwave-diagram

The MindWave headset gives the brain activity in the computer it is connected. It also provides the functionality on mobile devices too. It is safe to use and measures brainwave signals and also monitors the attention levels of user. It samples around a frequency of 1 Hz giving all the components of the EEG signal. Every electrical activity produces waves, and electrical devices create some noise. This noise might change the electrical readings of the brain. This justifies the fact that most EEG devices pick up electrical signals or readings even when the user is not connected to the device. The idea of measuring brain activity through the waves is similar to eavesdroping a conversation at a concert. Earlier, EEG devices avoided this problem by measuring these signals in environments where electrical activity is strictly controlled and increasing the signal strength of the data coming from the brain through the use of a conducting solution.

However, it is inconvenient always to have a room and a conductive solution for reading EEG signals. Thanks to modern science, NeuroSky has developed good algorithms to built into their products which filter out this “noise”. NeuroSkys white paper claims the ThinkGear technology has been tested at 96 percent as accurate as that within research grade EEGs. Sample EEG data from Mindwave Neurosky sensor is shown in Fig. 4.2.

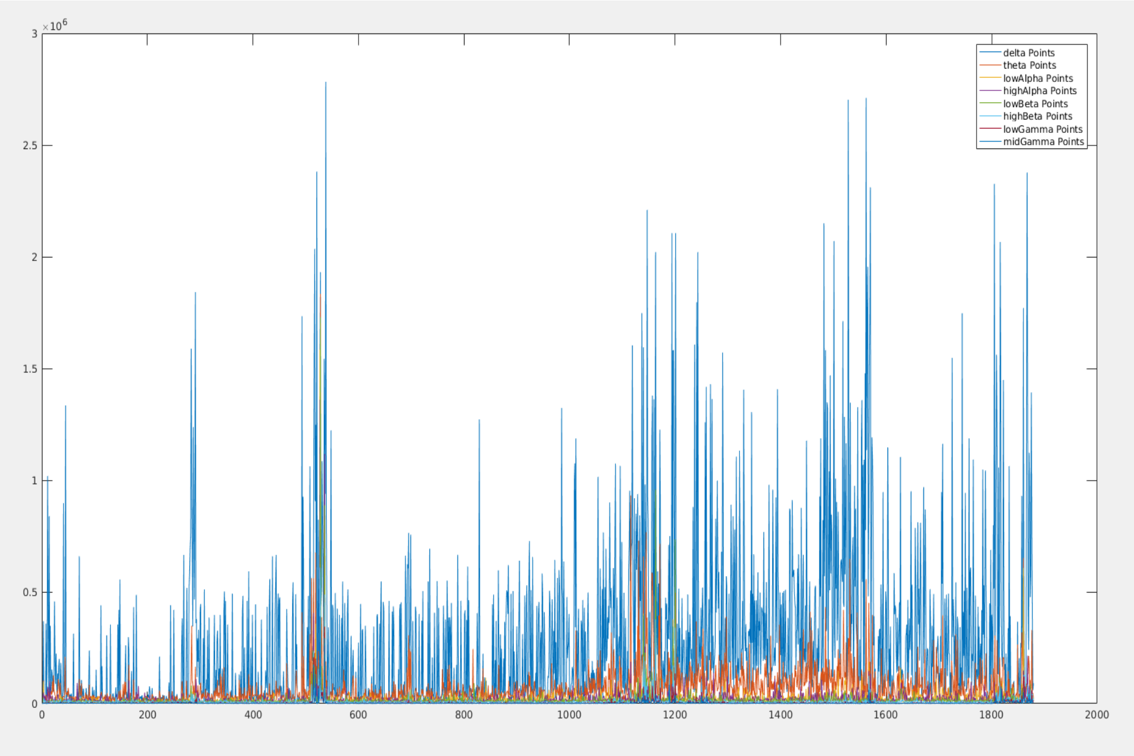


Figure 4.2: Sample EEG data from NeuroSky Mindwave Sensor

Next, we extract only the Beta component of the EEG signals as shown in Fig. 4.3. As we mentioned that this signal is the signal of interest since it is related to concentration in the awake state. So different users are aware of data collection. We collected Beta component of the EEG signal for four different users. The sample wave from is given below.

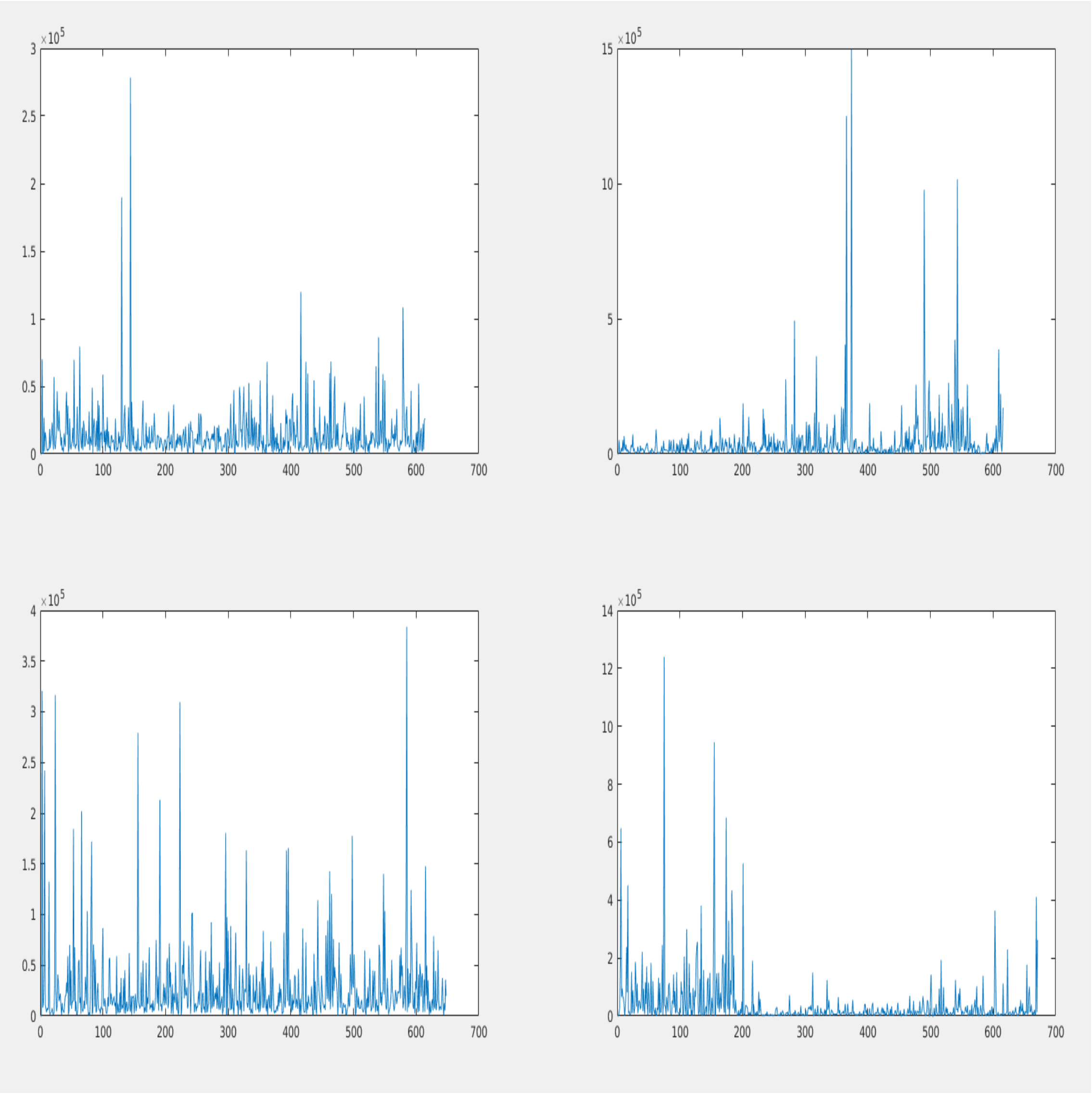


Figure 4.3: Sample Beta component of the EEG signal

4.3 Implemented Algorithm

After collecting the Beta data of the EEG signal, we perform the few steps based on the block diagram as described in Fig. 4.4

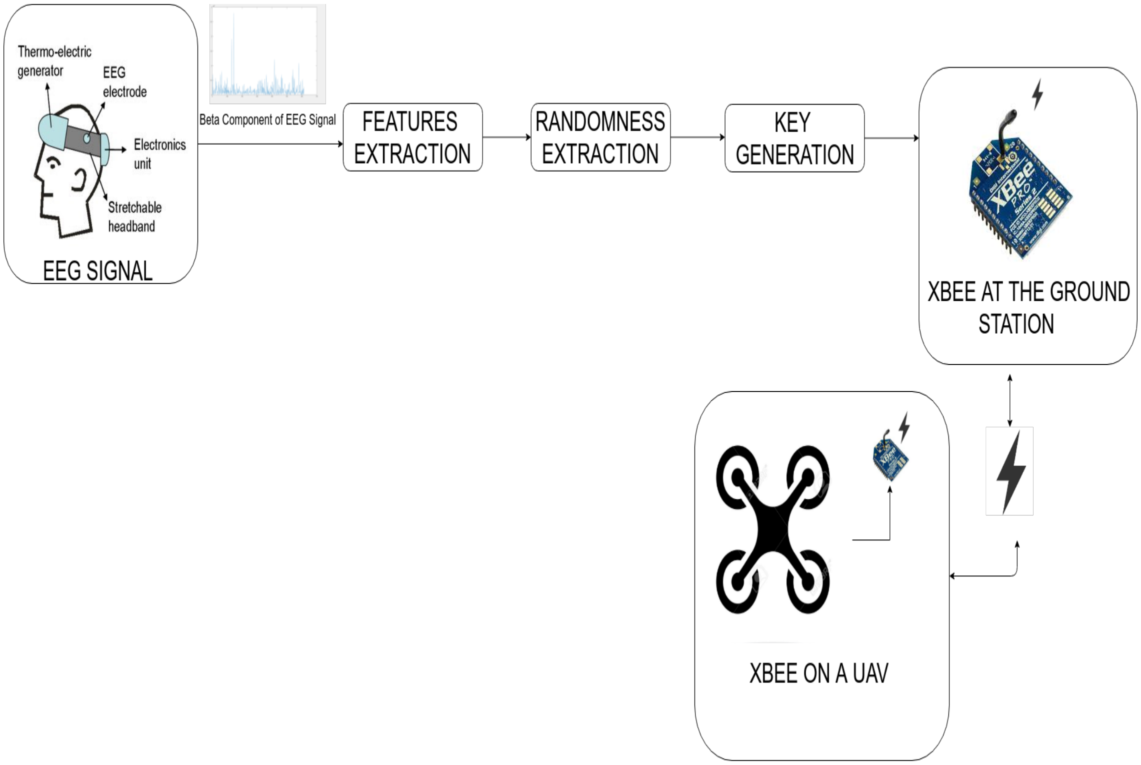


Figure 4.4: Block diagram of our implementation

4.3.1 Feature Extraction

We record an EEG signal (Beta waves) from a specific user for period of time T. The Beta waves are amplified by amplifier value A and matched via high order Legendre Polynomials. Legendre's differential equation is given by Equation (3.1) and its Legendre polynomials are computed by Equation (3.2).

The Legendre's polynomial equations are used to fit the collected EEG data which results in an n-degree equation.

The n-degree equation used for fitting data is given by:

$$y\left(x\right) = a_0 + \sum_{1}^{n} a_i p_{i(x)}. \tag{4.5}$$

The polynomial coefficients $a_0, a_1, ...a_n$ are combined together with the time window of size T and the amplitude multiplier A to form the raw feature vector $z := \{ca_0, ca_1, ca_2, ...ca_n, A, T\}$ where c is a constant to magnify the difference between coefficients. We map z to w such that $w = z \times M + \gamma$ where M is an $n \times n$ invertible matrix which satisfies $\sum_i m_{i,j} = 1$; γ is a random vector whose elements are within the range $\left[2^{-\theta}, 2^{\theta}\right]$.

Since attackers can reconstruct the original EEG waveform given the feature vector, we map the feature vector with some random vector using Linear transformation. This results as a random feature vector w [25].

4.3.2 Randomness Extraction

After obtaining feature vector w, we use a reusable fuzzy extractor constructed from (n, k)-BCH codes (BCH codes form a class of cyclic error-correcting codes to correct errors occurred [25]) with generator function to extract enough randomness from it. Randomness provides the functionality of representing the feature vector in different form so that attacker cannot reconstruct the original signal.

The randomness extracted from each feature r_i is computed as $r_i = H_x(w_i)$, where H_x is a hash function in a universal hash family. The universal hash family H is a class of hash functions. H is defined to be universal if the possibility of a pair of distinct keys being mapped into the same index is less than $1/l$ (l is the length of the randomness string). The hashing operation is performed after making a random choice of hash function from the universal class H. The universal hash function already gives the optimal length of extracted randomness [25].

We also compute the syndrome S_c of feature values for future authentication. If the feature element is viewed as $w_i(x) = w_{i_0} + w_{i_1}x + ... + w_{i_{n-1}}x^{n-1}$, every element

w_i has a corresponding syndrome S_{c_i} for (n, k)-BCH codes:

$$S_{c_i} = w_i(x)\text{mod}g(x) = \left\{w_i(\alpha^1), w_i(\alpha^2), ..., w_i(\alpha^{2t})\right\}. \tag{4.6}$$

4.3.3 Key Generation

Next, we generate the key based on the features and pre-program the specific UAV with that key to secure the communication channel. This is a practical way to ensure that both the ground control station Xbee and the XBee on-board UAV obtain exactly the same key for encryption and decryption. The key K is generated based on chosen extracted randomness from the previous step [25]. The key generation technique is given below.

We randomly choose q constants $1 \leq j_1 \leq ... \leq j_q \leq n$ to pick up several features and produces a permuted feature vector $v := \left\{w_{j_1}, ..., w_{j_q}\right\}$.

The key K generated is based on chosen random extracted randomness $r_{j_i} : K := r_{j_1}||...||r_{j_q}$, where $||$ denotes concatenation.

4.3.4 Configuring XBee with the Key Generated

AES Encryption Key

AES's design methodology is based on substitution-permutation network, which is a combination of both permutations and combination. It works fast in both hardware and software. AES is an adaptation from Rijndael [26], and it has a key size of 128, 192 or 256 bits and a fixed block size of 128 bits. However, Rijndael specification has a specific key and block sizes that could be 32 bits multiples, where $128 <= bitsize <= 256$.

AES has 4×4 column-major matrix composed of bytes, which is termed as the state. However, some variations of Rijndael have larger block length size and have additional columns in the state. Often, AES calculation is done in a specific finite field.

For example , let there are 16 bytes $b_0, b_1, ..., b_{15}$, then these bytes are represented in matrix-form as:

$$\begin{bmatrix} b_0 & b_4 & b_8 & b_{12} \\ b_1 & b_5 & b_9 & b_{13} \\ b_2 & b_6 & b_{10} & b_{14} \\ b_3 & b_7 & b_{11} & b_{15} \end{bmatrix}$$

The total iterations of transformation rounds that convert the input from plain text to cipher text is specified by the AES cipher's key size. The number of iterations for each key size is as follows:

- 128-bit keys - 10 cycles of repetition.
- 192-bit keys - 12 cycles of repetition.
- 256-bit keys - 14 cycles of repetition.

Every round consists of several processing steps. In each processing step there are four similar but different stages. It also includes the steps that depend on encryption key itself. To perform decryption, the same key can be used to cipher text into original plain text by applying reverse rounds.

After generating the key using the above procedure, we configure the XBee's AES encryption key parameter to use the generated key for communication. For this experiment, we used the Mindwave sensor and Alienware 15-inch laptop with i7 6820 HK processor to create the EEG system.

We utilized a commercially available UAV to conduct this experiment. The UAV used the Pixhawk as its controller as the CPU which had the XBee connected to communicate with the ground control station.

The UAV and the base station were wirelessly connected using XBee transmitter and receivers.

After configuring the AES encryption key of the XBee with the generated encryption key, we tested the communication of UAV with the XBee connected to the ground control station. The AES key configuration ensured secured communication of the UAV to the ground control station. However, we have also introduced a scenario where an attacker is trying to intercept the communication between the UAV and ground control station with the primary purpose to gain control of the UAV. For simplicity, we have assumed that the attacker already knows the key generated and has configured its device with that key and to communicate with the UAV maliciously. Our experimental setup is shown in Fig. 4.5.

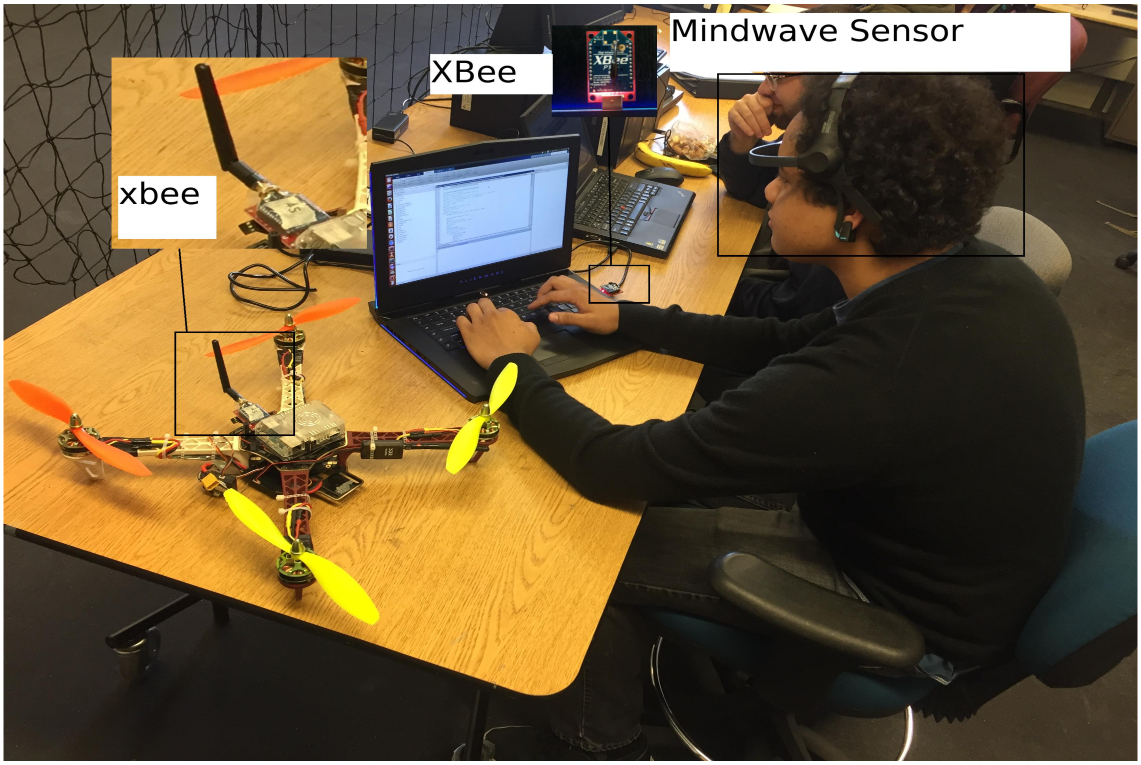


Figure 4.5: Block diagram of our implementation

As a safety measure, we have pre-configured the UAVs XBee to receive the commands from the ground control station Xbee's address. If the attacker tries to send the control signals from its device then from the attacker's packet address, we verify that a third party is intervening, and we activate the Return-To-Launch control signal in the UAV. This would mean that the UAV identified that an attack was attempted and should return to its starting location. The RTL (Return-To-Launch mode) aids the UAV navigation from its current position to hover above the home position. RTL is a GPS-dependent move, so it is essential that GPS lock is enabled before attempting to use this mode.

The algorithm is described below as Algorithm 1.

The LockGPS() function ensures that the sensor is not affected by any other way since it becomes completely independent of the rest of the communication process.

We also propose another methodology where, in case a hack is attempted, the

Algorithm 1: RTL mode activation in UAV

```
$getAddress \leftarrow xbeedata.getAddress()$
**if** $getAddress \neq groundcontrolstation.getAdress()$ **then**
  $LockGPS()$
  $ReturnToLaunch()$
**else**
  $Continue;$
**end if**
```

XBee sends a predefined signal to the ground control station which signals the station to configure the XBees (both at the ground control station and at the UAV) with a new key. We then run at the ground control station the same pipeline of key generation from the EEG signal and generate another key to ensure the communication is secure and configure both the XBees.

We describe the algorithm below (Algorithm 2):

Algorithm 2: Key Change request in UAV

```
$getAddress \leftarrow xbeedata.getAddress()$
**if** $getAddress \neq groundcontrolstation.getAdress()$ **then**
  $LockGPS()$
  $SendKeyChangeToGroundControlStation()$
  $WaitForKey()$
**else**
  $Continue();$
**end if**
```

Another attempt to ensure a secure communication is to regularly change the key generated and configure the Xbees at regular intervals of time. This way we achieve quite robust and secure way of communication in the UAVs.

4.4 Summary

This chapter gives the underlying mathematical frame work and the design of our implementation for Key generation techniques using the individual characteristics of the EEG signal. This chapter also describes the algorithms to detect an cyber attack and evasion techniques in order to secure the communication between the UAV and the ground control station.

Chapter 5

Experimental Results and Discussion

5.1 Results

In the initial setup, we collected the EEG data and activated our key generation pipeline to generate a key. The data were collected from a user performing a specific task which involves activating the Beta component of the EEG signal. The collected data (around 1000 data points) are fed to our key generation pipeline that involves Features extraction, Randomness extraction, and Key generation.

Then we configure the XBee's in AT mode to ensure that XBee's AES encryption mode is enabled and uses the Key generated from our pipeline. Since the EEG data is inconsistent even for the same user at different time intervals, the generated Key from our pipeline would be different, thus ensuring uniqueness of the Key generated. This enables users to configure the XBee's AES key to different values.

A normal EEG waveform of a single person is shown in Figure 5.1: We extract

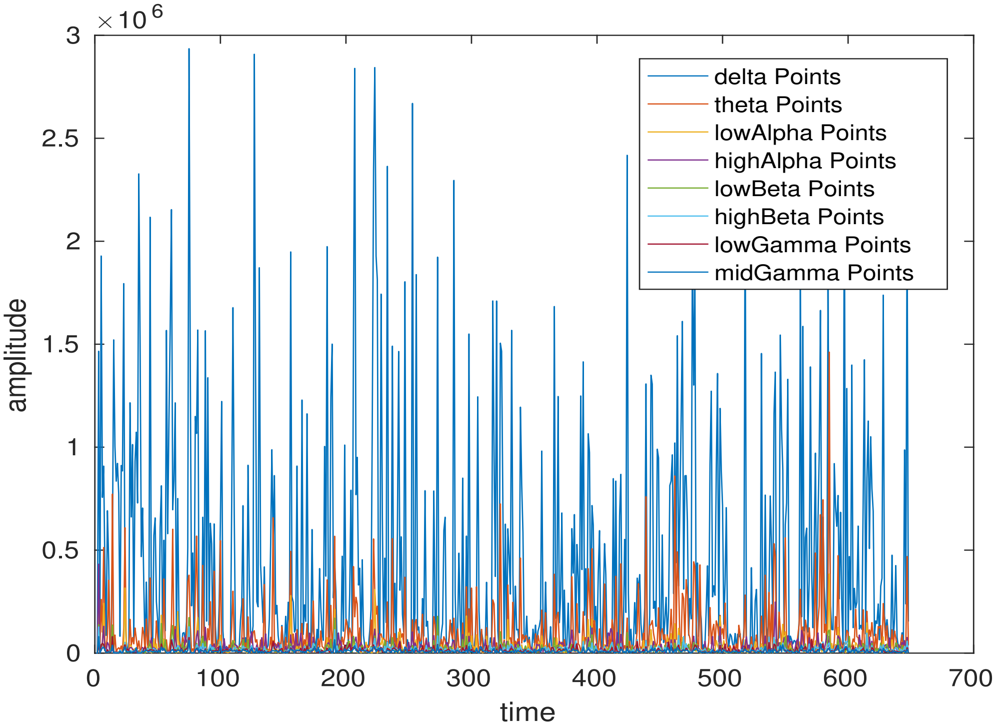
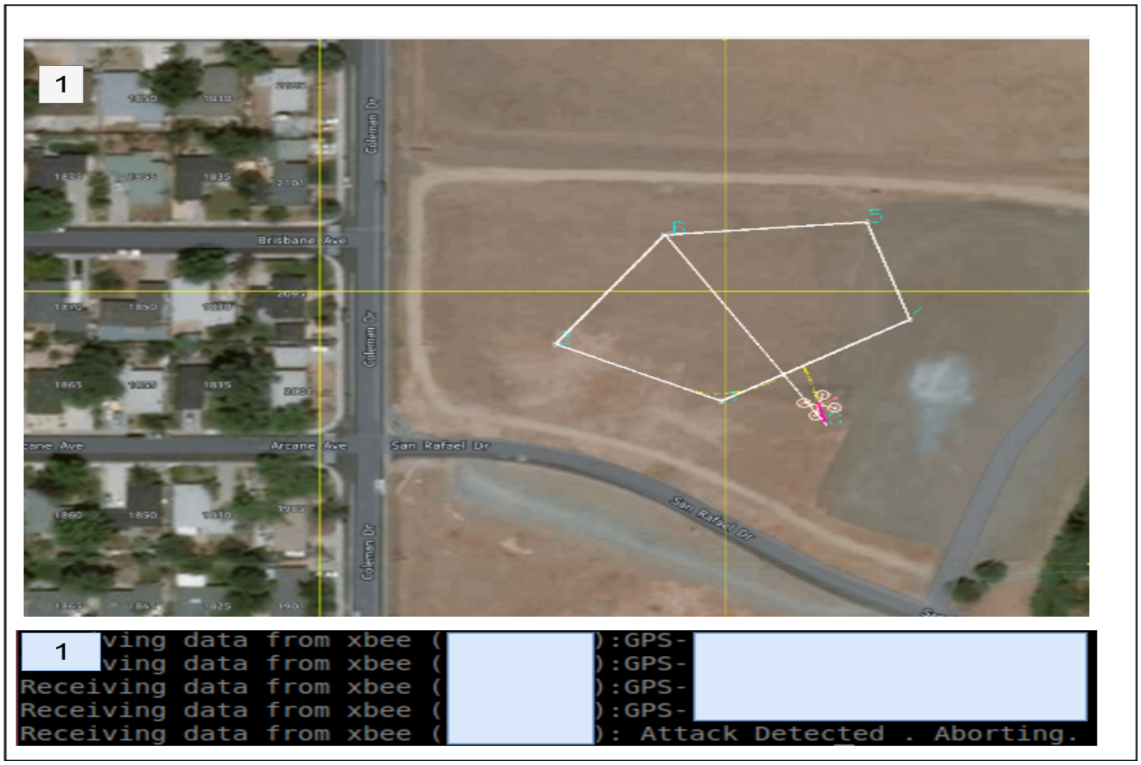


Figure 5.1: Sample EEG waveform (with all the components) of a user performing a specific mental task.

the Beta components of different people which was collected under similar tasks.

XBee has two basic modes of interaction: AT and AP2 mode. AT mode is also referred as Transparent mode. In AT mode, the destination address is specified in the Xbee's memory any data sent to the XBee module is immediately sent to the destination Xbee. A user can configure the Xbee in AT mode by first placing the module in Command mode and then sending predefined AT commands using USB or UART port. It is useful in scenarios where we do not intend to change the destination address often or in cases of simple network or a point to point communication. XBee's in-built encryption also gets disabled in AT mode, so it is important to give proper attention while configuring XBees.

We performed our proposed safety mechanism using an assembled quadcopter with an onboard autopilot and the XBees to communicate with the ground control

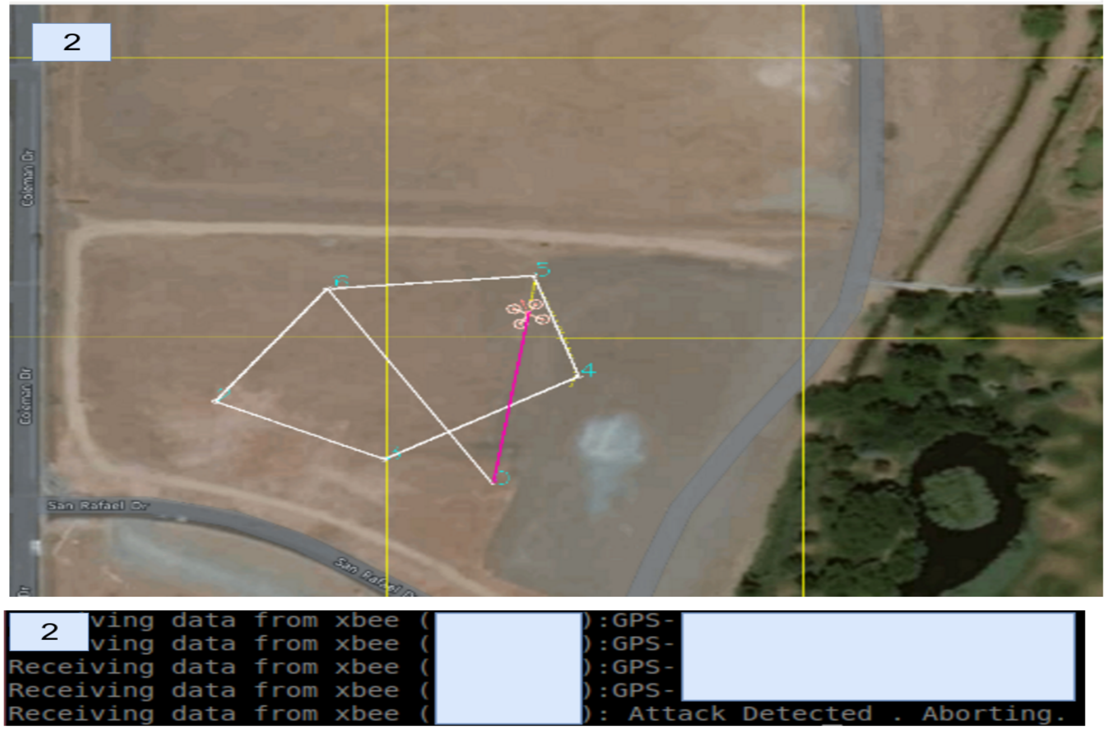


```
1   ving data from xbee ([redacted]):GPS-[redacted]
    ving data from xbee (          ):GPS-
Receiving data from xbee (          ):GPS-
Receiving data from xbee (          ):GPS-
Receiving data from xbee (          ): Attack Detected . Aborting.
```

Figure 5.2: Waypoints set for the experiment in the first configuration. The attack was discovered after the UAV navigated from waypoint 3 and Return-to-Launch (RTL) was enabled.

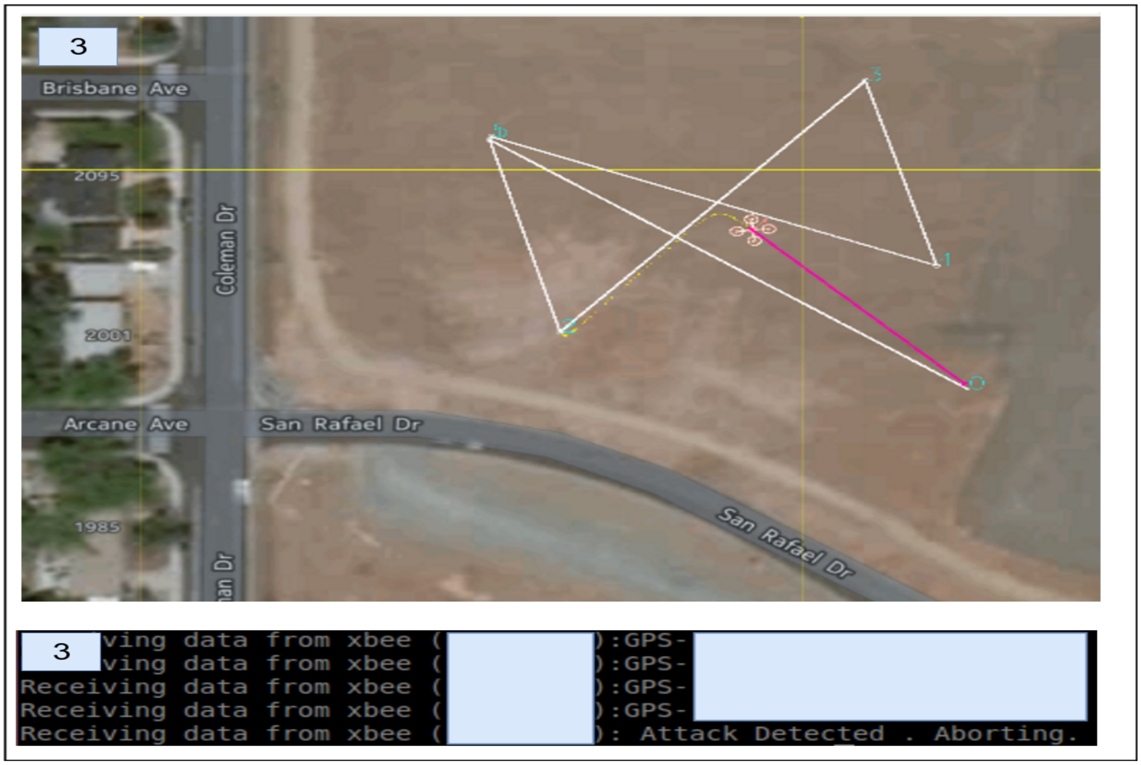
station. We set up the waypoints for the UAV using mission planner software. For our experiment, we set up different waypoints at different configurations and tested our methods at different times. It is shown from Figure 5.2 to Figure 5.5 [27].

The purpose was to travel these way-points and return to the base location in case an attack is detected. We introduced a third party attacking mechanism, and for a simplistic purpose, we made the third party aware of the key generated which the UAVs XBee is using to communicate with the ground control station. As the third party starts attacking and maliciously sending control signals to the UAV, our algorithm successfully detects the intervention (since the received packets at the UAVs XBee has different source address). After detection of the intervention, the UAV initiates its RTL mechanism and return to the base GPS location without completing the directed trajectory [27].



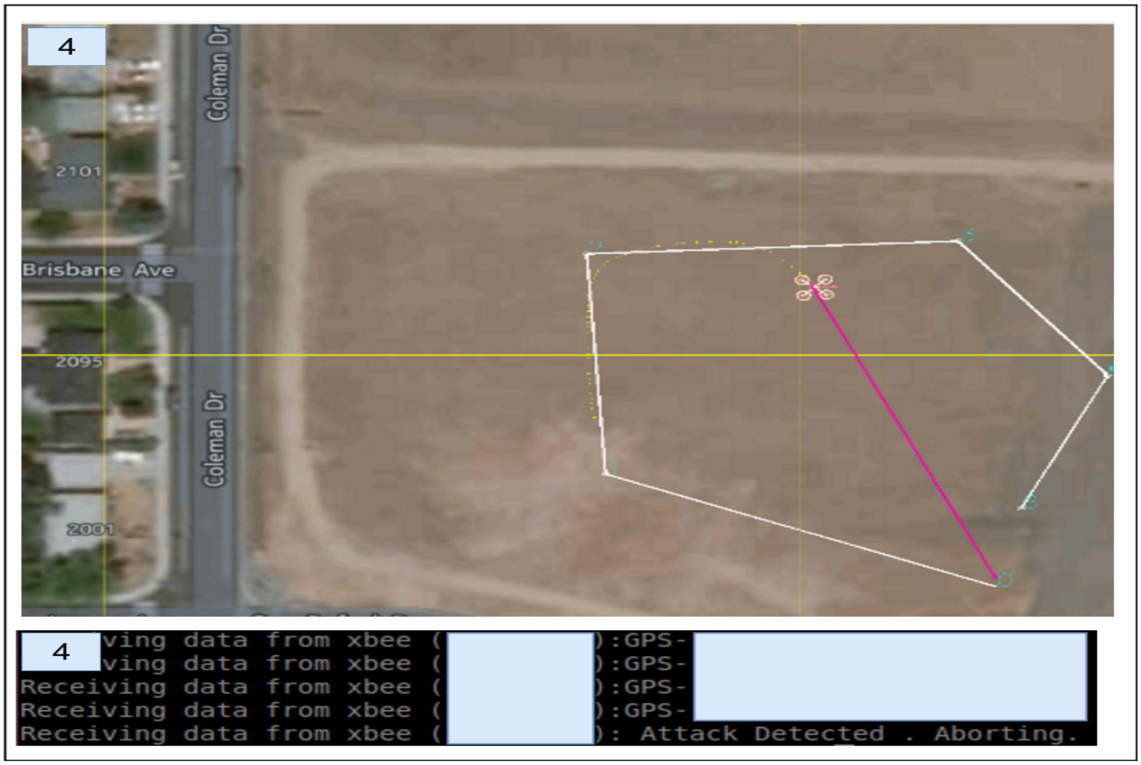
```
2   ving data from xbee ([redacted]):GPS-[redacted]
    ving data from xbee (            ):GPS-
Receiving data from xbee (           ):GPS-
Receiving data from xbee (           ):GPS-
Receiving data from xbee (           ): Attack Detected . Aborting.
```

Figure 5.3: Waypoints set for the experiment in the second configuration. The attack was discovered after the UAV navigated from waypoint 5 and Return-to-Launch (RTL) was enabled.



```
    ving data from xbee ([redacted]):GPS-[redacted]
    ving data from xbee ([redacted]):GPS-[redacted]
Receiving data from xbee ([redacted]):GPS-[redacted]
Receiving data from xbee ([redacted]):GPS-[redacted]
Receiving data from xbee ([redacted]): Attack Detected . Aborting.
```

Figure 5.4: Waypoints set for the experiment in the third configuration. The attack was discovered after the UAV navigated from waypoint 2 and Return-to-Launch (RTL) was enabled.



```
    ving data from xbee ([redacted]):GPS-[redacted]
    ving data from xbee ([redacted]):GPS-[redacted]
Receiving data from xbee ([redacted]):GPS-[redacted]
Receiving data from xbee ([redacted]):GPS-[redacted]
Receiving data from xbee ([redacted]): Attack Detected . Aborting.
```

Figure 5.5: Waypoints set for the experiment in the fourth configuration. The attack was discovered after the UAV navigated after waypoint 2 and Return-to-Launch (RTL) was enabled.

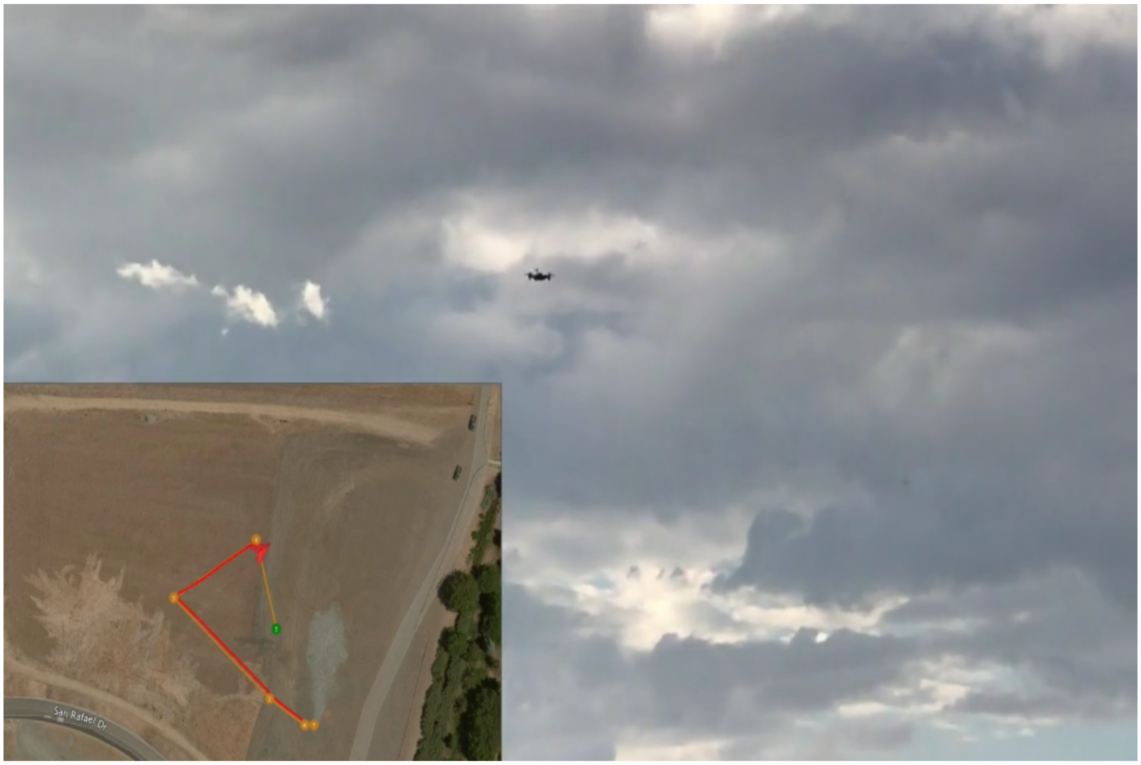


Figure 5.6: Waypoints set for the experiment in real time. The attack was discovered after the UAV navigated after waypoint 4 and Return-to-Launch (RTL) was enabled.

We tested our other approach of changing the key when an attack is detected. During this test, we setup the same waypoints and introduced a similar type of attack along the way. After successful detection of the intervention, the algorithm sent a key change request to the ground control station, during which, the UAVs communication is restricted to the ground control station, and it hovers at a specified location where the attack was attempted. After the XBee is configured to a new AES key, the navigation is resumed to the destined location [27].

Fig. 5.6 shows the results in real-time. The attack was detected after waypoint 4 and Return-to-Launch (RTL) was enabled.

5.2 Summary

This chapter entails the results of proposed algorithm for return to home position of a UAV in case a hack is detected. We performed the detection on a UAV with on-board Xbee. This chapter also gives the results of detected attack at various paths the UAV travels.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This thesis gives a unique approach to biometric encryption of a UAV communicating with the ground control station. We have also provided a safety mechanism for the UAV in case a third-party intervention is detected along the way. This approach can be used for any UAV scenario where cyberattacks are a particular concern. Our approach not only adds a layer of additional security to the UAV but also provides a unique way for securing the UAV with low-cost resources.

6.2 Future Work

In future work, we plan to further extend our authentication scheme to multi-UAV scenarios [28–30], where a cluster of UAVs aims to authenticate their controller. A possible approach is to have each member in all UAVs (a cluster) sequentially verify the controller one by one utilizing the proposed authentication scheme. Formation control and cooperative learning in multi-robot systems can be utilized to enhance the safety security mechanism [31–33].

6.2.1 Statistical Analysis of EEG

We also intend to enhance our EEG key generation technique from extracting polynomial coefficients to perform a statistical analysis of the EEG data. One approach suggests using the averaged event-related potential (ERP) which claims to have the potential to provide more robust bio-metric identification. It describes the Cognitive Event-Related Biometric Recognition (CEREBRE) protocol, an ERP based biometric protocol designed efficiently to express individual's unique responses coming from multiple functional brain systems. The results based on their approach indicate 100 percent identification accuracy in a pool of 50 users.

The idea for statistical analysis for EEG is to extract consistent parameters in an EEG signal of a particular user. This consistency ensures a single key or a single type of key with a known variance. Identify other consistent parameters of brain EEG signals that highlight the task a particular user is performing like sleeping, doing a different mental tasks, etc. It would not only help identifying a state of mind (which is important for generating a key since getting EEG data from sleeping person would defeat our purpose) but also identify the intensity of state of mind of different individuals performing a different task. This must require different volunteers performing different tasks. This requires active data collection for a longer period.

6.2.2 A Bird that never forgets what it sees!

This subsection describes another technique for safe return to home approach for a UAV in case an attack is detected. Given the advancement of miniature storage devices and increased memory capacity of UAVs, it has become affordable to design and implement visual navigation through 3D sensors which justify the title of this section.

The idea is to use only the 3D data acquired over time as the UAV travels in a specified path. Assume a scenario, where the ground control station specifies a path for a UAV to travel from one location to another. Since the UAV is equipped with 3D sensors (Ex: Velodyne) we attempt to store the 3D data in memory and perform the alignment of 3D point clouds to reconstruct the scene as the UAV acquires through time along the way. Assume that at a certain point X along the way, an attack was detected. Our approach is to cut all the communication of the UAV to the ground control station and use only the previously reconstructed scene to navigate and find it's way back to home. It essentially simplifies as follows: understand the environment, detect the key features, find a path to return to the home position based on the reconstructed 3D scene until the attack was detected. We progressed our way though 3D scene construction at a smaller scale (room) using a simpler 3D device (PMD picoflexx).

Overview: 3D sensing

3-D sensing that leads to highly dense mapping of the environment is a key feature of aerial robotics if such system are to be able to execute path planning and navigation. To get such data we require a Time-of-Flight 3D sensor (pico flexx sensor) which gives 3D data points of the perceived environment. These data points are generally called as point clouds which are collection of points that represent the environment.

The per frame point cloud data are generally acquired from different viewpoints of the environment, in this case when the sensor is onboard the UAV. However the point clouds acquired, should be added together to make a perfect sense of the environment which would aid the MAV to localize itself in the environment and navigate accordingly. Aligning the point clouds which are taken at different viewpoints to a global coordinate frame such that the point clouds match as well as possible which helps reconstructing the environment is defined as the Registration. The key aspect in Registration is to find the relative position and orientation between the acquired point clouds. The problem statement narrows down to the registration of two 3D point clouds, i.e., aligning two point clouds by estimating the transformation between the two view poses under which the point clouds have been acquired. This type of registration can be carried out by one of the several variants of ICP(Iterative Closest Point) algorithm. The registration of Point clouds can be split into following steps:

1. Selection: Sampling Input Point Clouds.
2. Matching: Finding the correspondences between the point clouds
3. Rejection: Reducing outlier by filtering the correspondences.
4. Alignment: Using an error metric in order to minimize the error while obtaining the optimal transformation.

The above steps are calculated by the ICP algorithm implemented in PCL (Point Cloud Library). However we process the input cloud data for refining which aids ICP.

We have implemented the following steps in our pipeline:

1. Voxel Grid Filtering:

   Since original point clouds are often redundant and become computationally expensive, we downsample the original point clouds that is, reduce the number of points in the point cloud. It could be done using voxelized grid approach among other methods. This approach creates a 3D voxel grid (Voxel grid can be assumed as a set of tiny 3D boxes in space). Then, in each voxel unit(3D box), all the points present will be approximated (i.e., downsampled) with their centroid.

2. Statistical Outlier Removal:

   It is a technique to perform a statistical analysis of the input point cloud dataset and remove the noisy measurements. The idea is to analyze the number of neighbours for each point, and normally it is a user defined value including standard deviation. So, those points will be identified and removed, which have a distance larger than the user defined standard deviation of the mean distance to the query point. The computed output is stored in a variable.

3. Normals Estimation:

   Surface normals relate to the geometric surface of a point. It is inferred as a normal's direction at a point located on the surface. This surface normal is a vector perpendicular to the surface at that point. It is usually calculated by the analysis of PCA (Principal Component Analysis) which is the eigenvectors and eigenvalues of a covariant matrix which is calculated from the query point's nearest neighbors. So, for every point, the algorithm picks it's neighbors within a sphere of radius r which is user defined.

4. Iterative Closest Point Algorithm(ICP):

   This algorithm is an iterative algorithm designed to find the best transformation (combination of rotation and translation) between two point clouds so that they match each other with minimum error. In this process, one point cloud is taken as target or reference, and the other is taken as the source. ICP iteratively checks the transformation which is required to reduce the distance from the source to the target point cloud. The process can be reduced to the following steps:

- For every point located in the source point cloud, find the closest point in the target or reference point cloud.
- Estimate the translation and rotation combination using an error mean squared cost function that assists best to align every source point in the source point cloud to its match found in the previous step.
- Apply the transformation obtained from the previous step to the source point cloud.
- Iterate the procedure until an accepted error is obtained.

In our approach, the first acquired point cloud is kept as a target and the second point cloud as the source. We iteratively perform the same set of operations for a subsequent pair of point clouds which are acquired for a scene. However, there are certain parameters that need to be taken care of.

- Between two corresponding points in source $< - >$ target we need to set the maximum distance threshold.

- RANSAC Outlier Rejection - This method classifies a point to be inlier or an outlier. Primarily, if the distance between the transformed source index and the target data index is smaller than the user-defined inlier distance threshold, then the point is considered as an inlier.

- For an optimization to be considered as having converged to the final solution, we need to set the Transformation epsilon which is the maximum admissible difference between two consecutive transformations.

- Before the ICP algorithm is considered to have converged, we need to set the maximum admissible Euclidean error between two consecutive steps in the ICP loop.
- Setting the ICP's maximum number of iterations.

Since we are using normals for ICP, the ICP inherently uses point-to-plane error metric for convergence. The result of ICP is the Transformation matrix (PT) between the source and target point clouds. This transformation matrix is used to update the global Transformation (GT) which is set to an Identity matrix in the initial setup.

It is updated as $GT = GT * PT$ where PT is the pair wise transform. The Pairwise transform of the previous step is also used as an initial guess for the ICP in the next pair alignment to perform fast convergence. The data set were collected at a frame rate of 15fps and moved across the room slowly. Data sets collected at smaller distance helped ICP to converge faster since more data would overlap in a pairwise alignment. The results are shown in Fig. 6.1 and Fig. 6.2.

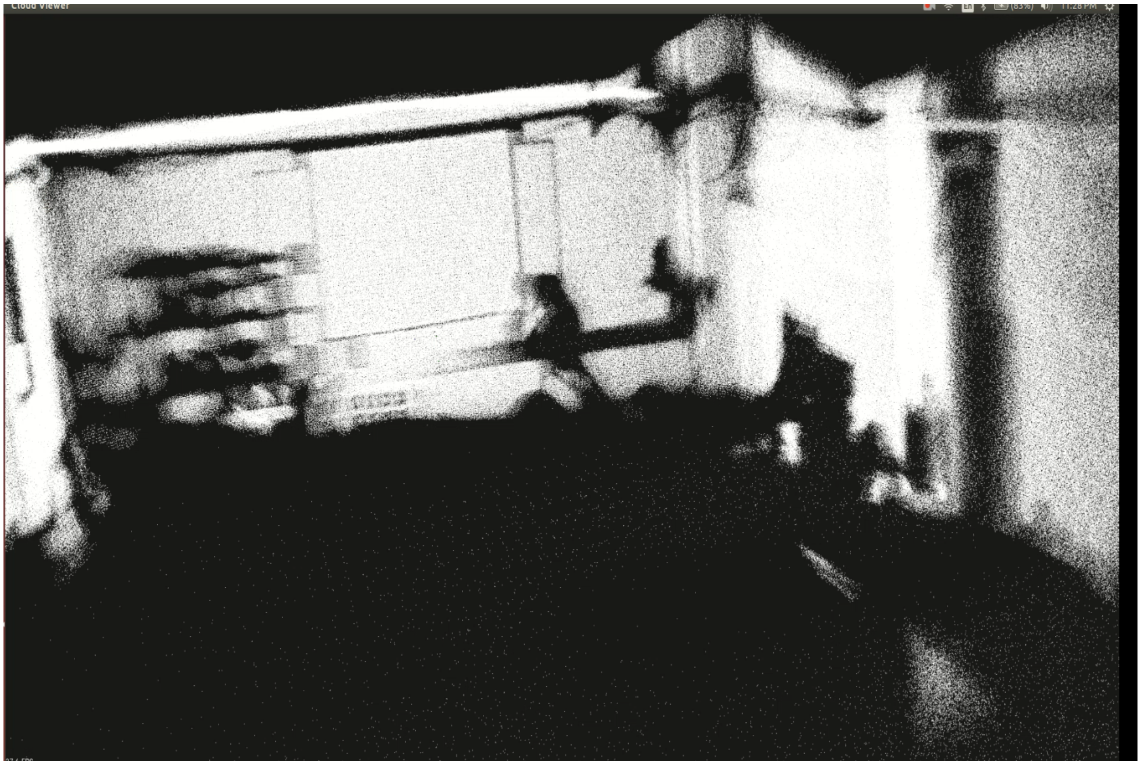


Figure 6.1: ICP results of a room

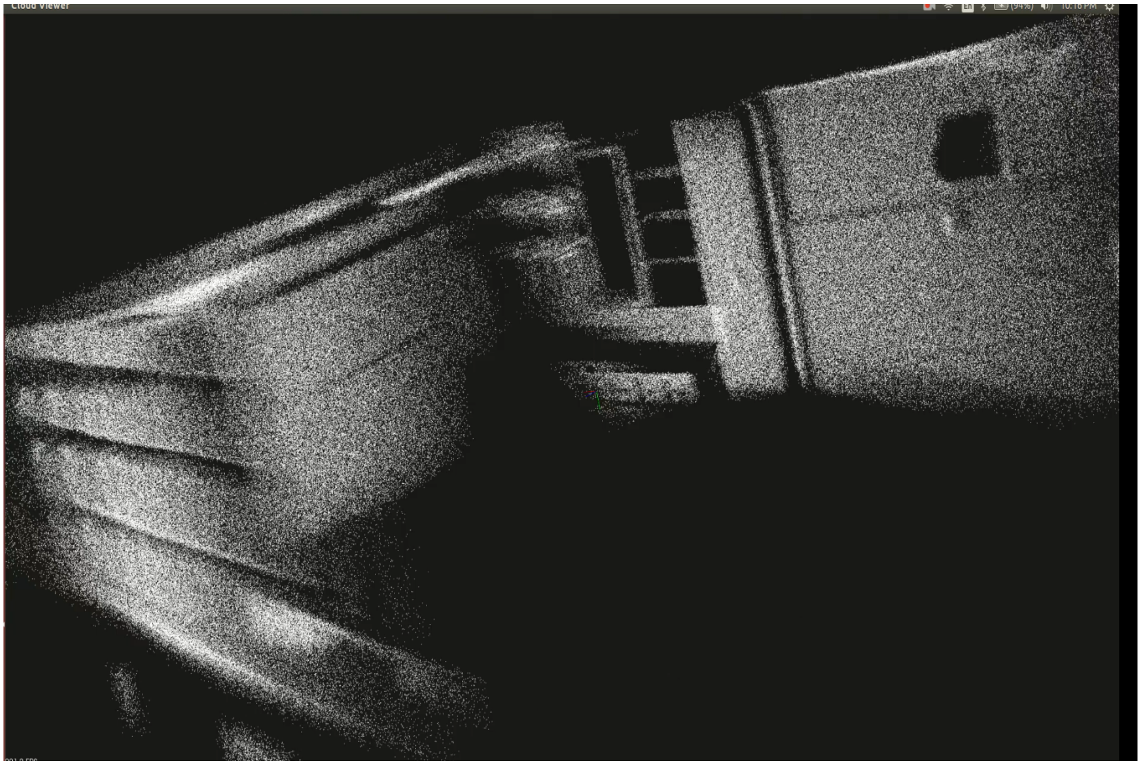


Figure 6.2: Another reconstruction of a room

Path Planning from Visual Data

After creating the 3D construction of a scene, path planning algorithms for returning the UAV to home position needs to be implemented. There are different path planning algorithms that has the potential to determine the paths from visual key points or visual way points. Most popular among these are Rapidly exploring random trees (RRT), Probablistic Road Maps (PRM), Artificial Potential Fields and Mixed Integer Programming.

Bibliography

[1] M. McFarland, “Google drones will deliver Chipotle burritos at Virginia Tech,” September 2016.

[2] Amazon, “Amazon prime air,” 2016. [Online]. Available: https://www.amazon.com/b?node=8037720011

[3] S. Gorman, J. Y. Dreazen, and A. C., “Insurgents hack u.s. drones,” *The Wall Street Journal*, Dec. 2009.

[4] T. C. Nguyen, “Virus attacks military drones, exposes vulnerabilities,” October 2011, retrieved 6/7/13. [Online]. Available: http://www.smartplanet.com/

[5] K. Hartmann and C. Steup, “The vulnerability of uavs to cyber attacks-an approach to the risk assessment,” in *Cyber Conflict (CyCon), 2013 5th International Conference on*. IEEE, 2013, pp. 1–23.

[6] L. Franceschi-Bicchierai, “Drone hijacking? thats just the start of gps troubles,” July 2012. [Online]. Available: https://www.wired.com/2012/07/drone-hijacking/

[7] T. C. Nguyen, “How college students hijacked a government spy drone.” 2012, retrieved 6/7/13. [Online]. Available: http://www.smartplanet.com/

[8] P. Paganini, "Hacking drones ... overview of the main threats. retrieved 6/7/13," 2013. [Online]. Available: http://resources.infosecinstitute.com/

[9] B. M. Horowitz, "Cybersecurity for unmanned aerial vehicle missions." April 2016.

[10] S. M. Diamond and M. G. Ceruti, "Application of wireless sensor network to military information integration," in *Industrial Informatics, 2007 5th IEEE International Conference on*, vol. 1. IEEE, 2007, pp. 317–322.

[11] H.-B. Kuntze, C. W. Frey, I. Tchouchenkov, B. Staehle, E. Rome, K. Pfeiffer, A. Wenzel, and J. Wöllenstein, "Seneka-sensor network with mobile robots for disaster management," in *Homeland Security (HST), 2012 IEEE Conference on Technologies for.* IEEE, 2012, pp. 406–410.

[12] A. Sanjab, W. Saad, and T. Basar, "Prospect theory for enhanced cyber-physical security of drone delivery systems: A network interdiction game," in *Proc. of the IEEE International Conference on Communications (ICC), Communication and Information Systems Security Symposium, Paris, France,*, 2017.

[13] A. Anjos and S. Marcel, "Counter-measures to photo attacks in face recognition: a public database and a baseline," in *Biometrics (IJCB), 2011 international joint conference on.* IEEE, 2011, pp. 1–7.

[14] D. International, "Security and encryption," 2017. [Online]. Available: http://docs.digi.com/display/RFKitsCommon/Security+and+encryption

[15] A. C. Woods and H. M. La, "Dynamic target tracking and obstacle avoidance using a drone," in *The 11th International Symposium on Visual Computing*, Dec 2015.

[16] A. Woods, H. M. La, and Q. P. Ha, “A novel extended potential field controller for use on aerial robots,” in *The 12th Conference on Automation Science and Engineering (CASE)*, Aug 2016, pp. 286–291.

[17] A. Woods and H. M. La., “A novel potential field controller for use on aerial robots.” in *IEEE Transactions on Systems, Man and Cybernetics: Systems*, May 2017.

[18] H. X. P. H. M. La, D. Feil-Seifer, and M. Deans, “Distributed control framework for a team of unmanned aerial vehicles for dynamic wildfire tracking.” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept 2017.

[19] Wikipedia, “Electroencephalography,” 2017. [Online]. Available: https://en.wikipedia.org/wiki/Electroencephalography

[20] ——, “Binding problem,” 2017. [Online]. Available: https://en.wikipedia.org/wiki/Binding_problem

[21] . C. C. Huang, T. L., “A comprehensive review of the psychological effects of brainwave entrainment.” 2008, pp. 38–50.

[22] M. Henry, “The ultimate brainwave entrainment quick start guide,” 2014. [Online]. Available: https://www.linkedin.com/pulse/20141018170122-1767830-the-ultimate-brainwave-entrainment-quick-start-guide

[23] R. B. Liming Chen, Supriya Kapoor, “Intelligent systems for science and information:extended and selected results from the science and information conference 2013,” 2013.

[24] Wikipedia, “Bch code,” 2017. [Online]. Available: https://en.wikipedia.org/wiki/BCH_code

[25] P. Huang, B. Li, L. Guo, Z. Jin, and Y. Chen, "A robust and reusable ecg-based authentication and data encryption scheme for ehealth systems," in *Global Communications Conference (GLOBECOM), 2016 IEEE*. IEEE, 2016, pp. 1–6.

[26] Wikipedia, "Advanced encryption standard," 2017. [Online]. Available: https://en.wikipedia.org/wiki/Advanced_Encryption_Standard

[27] A. Singandhupe, H. La, D. Feil-Seifer, P. Huang, L. Guo, and M. Li, "Securing a uav using individual characteristics from an eeg signal," in *Proceedings of the IEEE Systems, Man, and Cybernetics Conference(SMC)*. Banff, Alberta: preprint, https://arxiv.org/abs/1704.04574, October 2017.

[28] T. Nguyen, T. T. Han, and H. M. La, "Distributed flocking control of mobile robots by bounded feedback," in *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sept 2016, pp. 563–568.

[29] H. M. La and W. Sheng, "Dynamic target tracking and observing in a mobile sensor network," *Robotics and Autonomous Systems*, vol. 60, no. 7, pp. 996 – 1009, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0921889012000565

[30] H. M. La, R. Lim, and W. Sheng, "Multirobot cooperative learning for predator avoidance," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 1, pp. 52–63, Jan 2015.

[31] H. M. La, W. Sheng, and J. Chen, "Cooperative and active sensing in mobile sensor networks for scalar field mapping," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 1, pp. 1–12, Jan 2015.

[32] H. M. La and W. Sheng, "Distributed sensor fusion for scalar field mapping using mobile sensor networks," *IEEE Transactions on Cybernetics*, vol. 43, no. 2, pp. 766–778, April 2013.

[33] T. T. Han, H. M. La, and B. H. Dinh, "Flocking of mobile robots by bounded feedback," in *2016 IEEE International Conf. on Automation Science and Engineering (CASE)*, Aug 2016, pp. 689–694.