3-2020

Development of a Drone-Mounted Wireless Attack Platform

Nathan V. Barker

DEVELOPMENT OF A DRONE-MOUNTED WIRELESS ATTACK PLATFORM

THESIS

Nathan V. Barker, 2nd Lieutenant, USAF

AFIT-ENG-MS-20-M-005

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENG-MS-20-M-005

DEVELOPMENT OF A DRONE-MOUNTED WIRELESS ATTACK PLATFORM

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Cyber Operations

Nathan V. Barker, B.S.C.S.

2nd Lieutenant, USAF

March 26, 2020

AFIT-ENG-MS-20-M-005

DEVELOPMENT OF A DRONE-MOUNTED WIRELESS ATTACK PLATFORM

THESIS

Nathan V. Barker, B.S.C.S.
2nd Lieutenant, USAF

Committee Membership:

Barry E. Mullins, Ph.D., P.E.
Chair

Timothy H. Lacey, Ph.D, CISSP
Member

Robert F. Mills, Ph.D
Member

Abstract

The commercial drone market has grown rapidly due to the increasing utility and capabilities of drones. Drones offer an invaluable resource to wireless hackers. Capitalizing on their mobility, a wireless hacker can equip a drone with hacking tools to circumvent physical security (e.g., fences) with relative ease and reach wireless networks.

Wireless networks are inherently more susceptible to passive capture and injection attacks, which is exacerbated by the predominant use of Wi-Fi's vulnerable security algorithms. Despite the impressive leaps drones have made, they are still noisy and hard to conceal. By equipping a drone with a directional antenna, this weakness can be mitigated and significantly improve their effective range.

This research develops skypie version 2 (skypie v2), which is an improved software and hardware prototype designed for directional drone-based attacks. To remain compatible with any drone, it is designed to be lightweight, inexpensive, and easily attachable to most off-the-shelf drones. These design choices also allow the prototype to simulate the capabilities an individual threat actor could produce.

This research experimentally evaluates the ability of a drone-mounted wireless attack platform (DWAP) equipped with a directional antenna to conduct wireless attacks effectively at distances greater than 800 meters. To test this hypothesis, the skypie v2 prototype conducts computer network attacks (CNAs) against a target network then captured data is used to evaluate the effectiveness of the platform.

Results show that conducting CNAs from the prototype is possible well beyond the hypothesized 800 meters when utilizing a directional antenna. Capture of a Wi-Fi Protected Access II (WPA2) handshake is possible at a Received Signal Strength

Indication (RSSI) of -72 decibel-milliwatts (dBm) which equates to 2400 meters from a network located in a open field. Additionally, nmap scans are conducted with a RSSI value of -74 dBm equivalent to nearly 3000 meters from the target network. Packet loss remained below 10% when the RSSI is $\geq$ -72 dBm.

This research demonstrates that platform stealth may be maintained when using a directional antenna. It develops operational drone cyber-attack capabilities, identifies their limitations, and provides potential countermeasures to defend the attack surface that DWAPs are expanding.

Acknowledgements

First and foremost, I would like to thank my Lord, who continually blesses me. Thank you for my lovely wife and our beautiful daughter.

To my advisor, Dr. Mullins, thank you for your encouragement, advice, and sharing your passion for hacking. You have inspired me and God willing many more to come!

Thank you to Dr. Mills and Dr. Lacey for your support and expertise while I pursued our research.

Lastly, I'd like to thank Youngjun for his friendship and aid here at school.

Nathan V. Barker

Table of Contents

Page

Page

Page

List of Figures

Figure Page

Figure Page

List of Tables

List of Acronyms

Abbreviation		Page
HAT	Hardware Attached on Top	45
HMAC	Hash-based Message Authentication Code	22
HTTPS	Hypertext Transfer Protocol Secure	16
ICV	Integrity Check Value	17
IDE	Integrated Development Environment	43
ISIS	Islamic State of Iraq and Syria	9
ISM	Industrial, Scientific, and Medical	14
IV	Initialization Vector	17
KRACK	Key Reinstallation Attack	22
LED	Light-Emitting Diode	45
LTE	Long-Term Evolution	27
MAC	Media Access Control	19
MFP	Management Frame Protection	17
MIC	Message Integrity Code	19
MUAV	Multirotor-UAV	7
mW	milliwatt	59
PAKE	Password Authentication Key Exchange	22
PLA	Polylactic Acid	45
PMK	Pairwise Master Key	19
PRNG	Pseudorandom Number Generator	17
PSK	Pre-Shared Key	19

Abbreviation		Page
PTK	Pairwise Transient Key	19
RDT	Reliable Data Transfer	91
RF	Radio Frequency	102
RGB	Red-Green-Blue	45
RPP	Remote Physical Proximity	1
RSSI	Received Signal Strength Indication	59
SAE	Simultaneous Authentication of Equals	22
SFTP	Secure File Transfer Protocol	16
skypie v2	skypie version 2	34
SOCOM	Special Operations Command	9
SSID	Service Set Identifier	14
SSL	Secure Socket Layer	16
SUT	System Under Test	55
T-Hawk	Tarantula Hawk	9
TKIP	Temporal Key Integrity Protocol	18
TPC	Transmit Power Control	15
U-NII	Unlicensed-National Information Infrastructure	15
UAS	Unmanned Aerial Systems	8
UAV	Unmanned Aerial Vehicles	8
US	United States	6
VoIP	Voice over IP	91

Abbreviation		Page
WASP	Wide Area Surveillance Projectile	9
WEP	Wired Equivalency Privacy	17
WLAN	Wireless Local Area Networks	14
WNIC	Wireless Network Interface Card	25
WPA	Wi-Fi Protected Access	18
WPA2	Wi-Fi Protected Access II	15
WPAN	Wireless Personal Area Network	15

DEVELOPMENT OF A DRONE-MOUNTED
WIRELESS ATTACK PLATFORM

I. Introduction

1.1 Overview and Background

Wireless devices are ubiquitous in home and work environments across the globe today. Unfortunately if an attacker is capable of gaining physical proximity to target devices, they are inherently more susceptible to injection attacks and having their traffic captured. Due to the popularization and significant capability improvements to drone technology, commercial drones can fill the need of physical proximity for wireless hackers. By equipping a drone with sufficient hardware for wireless capture and interaction, a motivated attacker can fly the drone within range of a desired target and gain Remote Physical Proximity (RPP). Not only does this make it easier to reach targets by rendering physical security measures (e.g., walls and fences) ineffective, it allows an attacker to stay hidden and distant.

These 'cyber-attack drones' extend the attack surface that network defenders need to consider. With lightweight hardware and capable commercial drones readily available, they can be developed inexpensively and rapidly. The rise of this threat is likely inevitable and should be evaluated.

1.2 Research Goals

The goal of this research is to further develop skypie, a directional drone-mountable cyber-attack platform previously created [1] and answer the following ques-

tions:

- Can CNAs be accomplished at 800+ meters using lightweight equipment on a cyber-attack drone?
- If so, how long does each attack take?
- At what distance do they become infeasible?
- How effective would these attacks be against a realistic network setup?

Development of this platform also helps identify specific threats that 'cyber-attack drones' pose and aid in the development of countermeasures to minimize those threats.

1.3 Problem Statement

This research aims to investigate the evolving attack vector of Drone-mounted Wireless Attack Platform (DWAP) platforms, specifically those equipped with directional antennas. DWAPs of this kind have been developed, but all suffer from the same limitations because they are equipped with low-gain omni-directional antennas. That is, in order for them to interact with wireless devices, they must be in close proximity. Because drones are relatively loud vehicles, it is nearly impossible for platforms equipped with omni-directional antenna to conduct an attack without being audible.

Stealth is often a necessity when conducting cyber-attacks, and directional antennas are well suited to fill the extended range needs of DWAPs. This work is limited in scope to low-cost consumer hardware with a directional antenna to emulate the limitations of a motivated lone-threat actor. In order to evaluate the new threats and capabilities of this platform, an analysis of its limits and effectiveness is conducted.

1.4 Hypothesis

This research hypothesizes that CNAs can be effective 800 meters or more when leveraging a directional cyber-attack drone. Several attacks are conducted against a target network at increasing distances, and their results are recorded. Additionally, a packet loss evaluation is conducted to help reveal the limitations of the DWAP.

This hypothesized distance is chosen based on the results of previous research that focused on geolocation while using the same hardware prototype [1]. The research found that geolocation using a directional antenna could be accurate, but significantly relied on the accuracy of the antenna bearing. In their work, the authors conducted tests as far as 600 meters; 800 meters is chosen based on their results.

1.5 Approach

Equipment. An existing prototype, which was designed to be mountable to a drone, is modified and further developed in keeping with its design goals. It consists of a directional antenna, wireless interface, computer, flight collection sensors, and a power source. The resulting prototype is used to execute the skypie software package and conduct a set of wireless attacks against a target network.

Data Collection. The attacks chosen in this work are one of several factors in the partial-factorial experiment conducted to determine effectiveness. Attacks are conducted between 200-2200 meters from the targets which places the DWAP between 2-22 times the typical maximum range of Wi-Fi devices (100 m). This typical maximum range is listed in Section 1.6 as an assumption. Additionally, the tests place the DWAP well beyond the hypothesized distance of 800 meters. The collected metrics include time to completion, attack success, signal strength, and number of network devices identified.

Analysis. The data is used to identify the range limits of individual attacks and

if distance has a significant effect on the time to complete each attack. With that information, the attacks' effectiveness against realistic networks are estimated.

1.6 Assumptions and Limitations

This research operates under the following assumptions and limitations:

- All CNAs are conducted in an open field. This prevents any additional attenuations due to obstacles. While this does not simulate a realistic network setup, it eliminates unknown factors and helps control experimental results.
- The location (optimal bearing) of the target network is assumed to be known.
- CNA are conducted from a prototype that is mounted and extended on a telescoping pole to simulate drone flight.
- While variable dependent, the typical maximum range for communication between two consumer grade Wi-Fi devices is assumed to be 50 m indoors and 100 m outdoors [2].
- Although capable of interaction with 5 GHz Wi-Fi devices, the CNAs are limited to a 2.4 GHz network as they make up a larger portion of networks worldwide [3].

1.7 Contributions

This research contributes to the body of wireless attack drone research, specifically airborne CNAs utilizing a directional antenna. It shows empirically that cyber-attack drones can be highly effective tools capable of completing attacks well over a mile (1609 m) from a target.

1.8 Thesis Overview

This thesis arranged in six chapters. Chapter 2 presents a background in drone technology, wireless technology and the associated security, flaws in wireless security, a brief overview of the hacker methodology, and related research in the field of drones and wireless attacks. Chapter 3 discusses the utilized prototype's hardware and software composition. Chapter 4 presents the experiment conducted to evaluate the CNA abilities of the airborne directional attack platform. Chapter 5 reviews the results of the experiment. Lastly, Chapter 6 summarizes the research and discusses opportunities for future work in the field.

II. Background and Related Research

2.1 Preparing For A New Threat

Drones are opening an attack space that once was mitigated by physical security. The growing capabilities and falling cost of commercial drones are a contributing cause. Organizations use walls, fences, and gates to prevent unauthorized access to buildings and to serve as a deterrence for would-be attackers. If attackers are caught trying to circumvent physical security measures, they can be pursued and captured. One of the keys to circumvention of physical preventative measures is knowledge of vulnerabilities. For a technically savvy adversary, drones are an ideal solution to overcoming physical obstacles; the radio frequency spectrum is a particularly susceptible attack vector that can be exploited to great effect. Wi-Fi, Bluetooth, and cellular technology are mediums most people use every day, but they emit signals that can be intercepted. Through this leakage of information, people can be tracked, networks can be mapped, and vulnerable devices can be hacked [4–6]. Drones can provide a low cost of entry for these areas of attack. Through their use, the likelihood of capture is lowered and they can provide intelligence while leaving little to no footprint.

A feature that could make drones used as a wireless attack platform particularly effective is their ability to use cellular capabilities for command and control. With cellular connectivity, there are virtually no limits to where an attacker can be while conducting an operation. This presents more problems for those who would defend against adversaries using these capabilities. Without the range limit of radio frequency or Wi-Fi controls, an attacker need not unnecessarily expose themselves and still can send/receive data to a drone over a cellular connection. Additionally, many of the current mitigation techniques for malicious drones involve identifying Wi-Fi control networks and usurping control of the drone or jamming radio frequency/GPS

signals [7]. When a malicious drone is controlled via a cellular connection, it reduces the likelihood of command and control being usurped. Although jamming is still a possibility, jamming cellular signals is illegal in the United States (US) [8].

2.2 Overview

In order to appreciate the eventuality of drone-based threats, an understanding of the rapid development of relevant technologies over the past several decades is necessary. This research aims to develop drone-based threat capabilities with the intent to evaluate security (i.e, WPA2) currently in place against these threats and raise awareness of drone attacks. This chapter discusses the present state of drones in use by the Department of Defense (DoD) and others in Section 2.3. Section 2.4 and 2.5 explains the different wireless security protocols and their weaknesses. The terms surrounding cybersecurity and information warfare is defined in Section 2.6. Lastly, this chapter introduces the current research involving drones that is relevant to cybersecurity in Section 2.7.

This thesis is an extension of the research preformed by Clint Bramlette [1] and his work developing a man-portable Multirotor-UAV (MUAV) platform capable of CNA and Computer Network Exploitation (CNE) for the purposes of mitigation and understanding the expanding threat space that Consumer-Off-The-Shelf (COTS) drones create. As indicated in Figure 1, the focus of this research is on a narrowed intersection of drones, wireless technology, and cybersecurity/information warfare. With a platform readily available and capable of an array of potential attacks, more safeguards can be developed to protect the wireless technologies on which people are becoming ever more reliant.

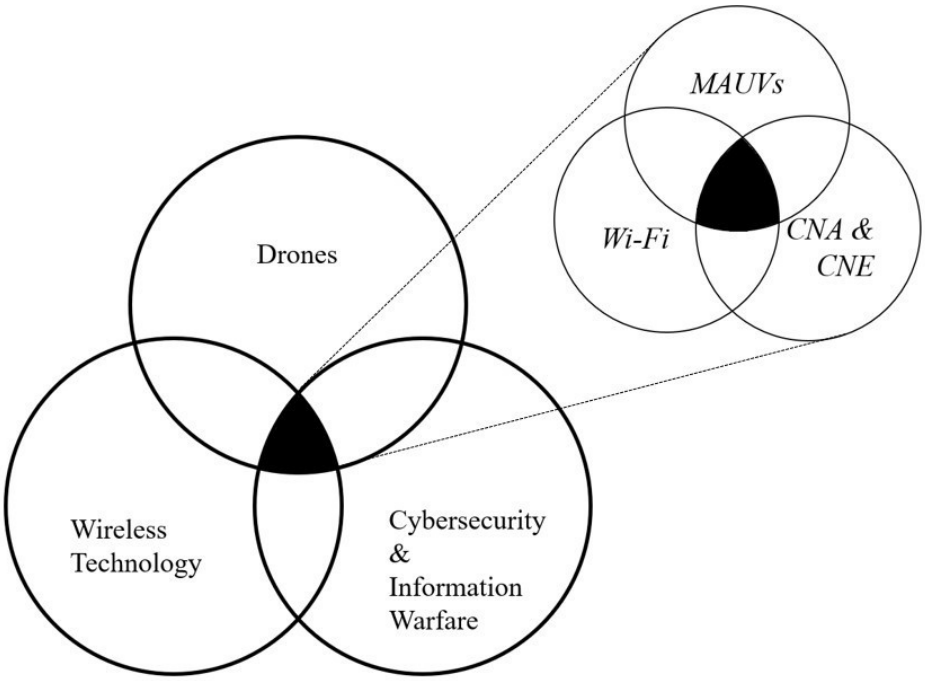

Figure 1: Exploded Venn-diagram highlighting the intersection of MAUVs, Wi-Fi, and CNA/CNE, which is the focus of this research [1]

2.3 Drones

2.3.1 Terminology

The term "drone" has been adopted as a catch-all for Unmanned Aerial Vehicles (UAV) and Unmanned Aerial Systems (UAS). A UAV is strictly an aerial vehicle, and a UAS refers to the vehicle, communication link, and equipment required for operation. While these terms are widely used to describe military capabilities, the term drone also refers to commercial systems. Because of the growth of the commercial drone market in the US, the US Congress defined UAS and public UAS in public law as aircraft that are "operated without the possibility of direct human intervention from within or on the aircraft" [9].

One of the most popular subcategories of drones is multirotor drones. These are

drones that operate multiple fixed-pitch blades attached to motors for flight. In a survey of aerial robotics, Liew et al. point out that 6-rotor hexacopters and 8-rotor octocopters share increased interest in the research community because of their rotor redundancy and ease of operation [10]. Normally, the control of these vehicles is handled by onboard computers and a wireless connection to an operator. The use of the term MUAVs throughout this thesis refers to man-portable UAS.

2.3.2 Military Drones

UAVs have been of great interest to the US military for many decades, but rapid development only began after the U-2 downing incident over the Soviet Union [11] and eventually led to the development of Micro-UAVs (separate from MUAV). These Micro-UAVs began to appear in the mid to late 1990s, prompted by the US Defense Advanced Research Projects Agency (DARPA) [12]. A few of the many developments included Lincoln Laboratorys fixed-wing Micro-UAV and Georgia Institute of Technologies study of vertical take-off and landing (VTOL) with a Micro-UAV [13,14]. The motivation for these research projects was to provide a military capability that was cheap and disposable. These first explorations of Micro-UAV were not exclusively rotor driven, and also included fixed wing vehicles and insect/animal-like flapping wing vehicles [15]. From the multi-phase Micro-UAV program that DARPA initiated, several high-quality UAVs for the time were produced. Two of those UAVs are the Wide Area Surveillance Projectile (WASP) and Tarantula Hawk (T-Hawk) [16]. The WASP was developed by the Massachusetts Institute of Technology for the US Army; it was capable of surviving high-G launches from a 155-millimeter cannon, and could sustain flight for 15 minutes for reconnaissance [17].

With the rapid growth and investment into drone technology has come more capabilities in the matter of only a few decades. Unsurprisingly, the US is not the

only country interested in drones. According to the Goverment Accountability Office (GAO), from the years 2005 to 2012, the number of countries that have acquired UAVs has jumped from 40 to 75; among the countries, there has been an increase in military application [18]. In 2013, the US Special Operations Command (SOCOM) placed orders with the company AeroVironment for MUAVs capable of 15 minutes of flight time and speeds of 100 miles per hour called Switchblades [19]. These MUAVs are equipped with cameras, GPS navigation, and can be operated manually or autonomously to deliver a missile. These drones proved so effective that SOCOM, in its 2016 Joint Urgent Needs Statement, requested 325 additional Switchblades to help combat Islamic State of Iraq and Syria (ISIS) and estimated the procurement cost to total $88.7 million dollars. General Ray Thomas, the SOCOM Commander at the time, recounted the troubling news that some of his operators had discovered a COTS drone that ISIS had modified to carry a 40-millimeter weapon [20]. With the technology being embraced on all sides, it is highly probable that drones will play an increasing role in many different facets of war.

2.3.3 Commercial Drones

Commercial drones have enjoyed explosive growth in capabilities and market growth over the past 15 years. What was once only a hobbyist activity has reached the hands of a much larger user base and is expected to grow to a $6.6 billion commercial worldwide market by 2020 [21]. These advances can be tied to a bevy of technological improvements such as: increased battery energy density, lower power sensor packages, and brushless electric motors [22]. But drones owe much of their growth to the rapid development of smartphones that occurred roughly the same time and brought cheaper accelerometers, cameras, and Wi-Fi chipsets.

Now commercial drones are being used for legitimate purposes ranging from crop

monitoring, power-line/pipeline inspection, wildlife surveillance, and rescue operations [23]. These activities were made legal when the Federal Aviation Administration (FAA), the US's air space governing body, introduced "part 107" which amended its regulations on commercial drone use in 2016. These amendments replaced the lengthy and costly waiver process for commercial drone use with a set of conditions.

With these regulations now in place, the race to integrate drone technology into their businesses has accelerated for tech giants such as Google, Amazon, and Facebook. Google's Project Wing focuses on the delivery of medical equipment, such as defibrillators, for times that require immediate medical response. Likewise, Amazon seeks to integrate drones into its product delivery process [24]. While Google and Amazon focus on physical product delivery, Facebook aims to use solar-powered drones to deliver Internet connectivity to parts of the world that are uncovered.

COTS drones are now meeting and exceeding some of the specifications of the US military's MUAVs and at a comparably low cost. Table 1 lists the specifications for many of the most popular drones available to consumers in 2018. Examining these popular drones shows that for less than $1500, flights times of 30 minutes and speeds of nearly 50 miles per hour can be achieved. Additionally, all of these consumer drones are equipped with cameras and many have a range of several miles. The competitive drone market has helped produce these products and vault them into the mainstream. Notably, over a thousand drones were used in last year's 2018 Winter Olympics opening ceremony to conduct a drone light show [25].

Table 1: Specifications of Popular Consumer Drones in 2018 [26]

Model	**Manufacturer**	**Camera (MP)**	**Maximum Flight Time (minutes)**	**Maximum Flight Distance (miles)**	**Maximum Flight Speed (mph)**	**Weight (grams)**	**Approximate Price**
Phantom 3 Pro	DJI	12	23	3.1	38.5	1280	$800
Phantom 4 Adv.	DJI	20	30	4.3	45	1370	$1200
Phantom 4 Pro	DJI	20	30	4.3	45	1390	$1400
Inspire 1 Pro	DJI	16	15	3.1	40	3500	$3000
Inspire 2	DJI	20	27	4.3	58	4000	$4900
Spark	DJI	12	16	1.2	31	300	$550
Mavric Pro	DJI	12	27	4.3	40	730	$700
Mavric Air	DJI	12	21	2.4	42	430	$800
Bebop 2 Power	Parrot	14	30	1.2	40	530	$600
X-star Premium	Autel	12	25	1.2	35	1600	$1600
Breeze	Yuneec	16	12	0.1	11	350	$180
Typhoon H Pro	Yuneec	12	22	1	30	1695	$1000
H920 Plus	Yuneec	16	24	1	25	4990	$2800
H520 Plus	Yuneec	20	28	1	38	1633	$3000

2.3.4 Academic Interest

Drones have not only grabbed the attention of the commercial market, but also the academic community. Their low cost and many applications have undoubtedly played a role in this. When comparing the three year periods of 2014-2016 and 2017-2019, a three-fold increase of publications can be seen in the IEEE database going from 522 to 1728 using the index term "drone" [27]. Some of the most recent improvements to drone capabilities include wireless mid-air charging [28], a cooperative drone network framework [29], video stabilization [30], autonomous infrared landing system [31], and reliable connectivity via cellular networks [32]. Figure 2 shows the number of UAV papers that were published in the top eight journals/conferences over a fifteen year period. The points indicate the number of papers per year which is growing at an exponential rate [10].

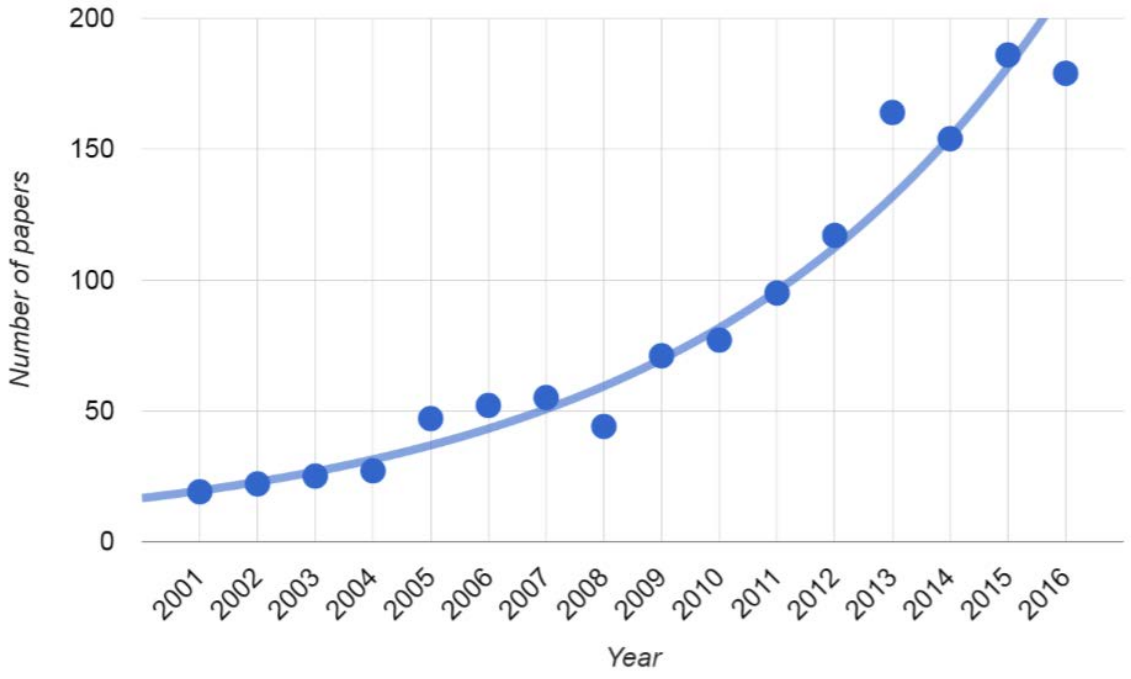

Figure 2: Number of UAV papers identified from the top eight journals/conferences over a 15 year period [10]

With the work from the academic community not appearing to slow, it is likely that more contributions to drone capabilities will be added in the coming years.

2.4 Wireless Technologies

This section describes the technical details of the different wireless technologies utilized by a large swath of devices such as routers, computers, smartphones, drones, smart devices, etc. A focus is placed on Wi-Fi as it is one of the main pillars of this research.

2.4.1 Wi-Fi

Wi-Fi is the family of technologies and protocols defined by the IEEE 802.11 standard (hereafter referred to as simply 802.11) and is widely used throughout the world for Wireless Local Area Networks (WLAN) [33]. Its use can be found in many homes and corporate networks with devices such as routers, computers, smartphones, tablets, smart TVs, printers, etc. The basic structure of the 802.11 standard consists of the following four physical components: (i) Access Points (APs), (ii) interconnection device (switch or router), (iii) a wireless medium, and (iv) stations (devices) [34]. In home networks, it is common for an AP and a router to be integrated into one unit. However this is not always the case or optimal for every network. The 802.11 wireless mediums consist of the unlicensed Industrial, Scientific, and Medical (ISM) radio frequency bands 2.4 GHz and 5 GHz. These two radio bands are also subdivided into fourteen and twenty-five different usable frequency channels respectively. In general, the APs that utilize the 2.4 GHz band can transmit longer distances but at lower data rates; APs that utilize the 5 GHz band can transmit shorter distances at higher data rates.

When connecting, stations must first search the channels on the radio band they

are utilizing to find available APs. APs can generally be identified by a Service Set Identifier (SSID), which is assigned upon setup by a network administrator. Once devices identify an AP available to them, they can attempt to associate with the AP. If security is enabled, stations must first authenticate before they are allowed to associate.

Another feature of 802.11 is Transmit Power Control (TPC) which affects 802.11a devices [33]. This feature is used to automatically reduce the transmit power of network devices when neighboring wireless networks are nearby. The AP can dictate to its clients at what power to transmit in order to reduce interference with other networks. A secondary effect of this feature is increased battery efficiency and decreased power consumption.

2.4.2 Bluetooth Low Energy

Bluetooth Low Energy (BLE) operates as a low-power, short-distance, low data rate technology that can provide a Wireless Personal Area Network (WPAN) [35]. These capabilities are different from Wi-Fi's high power, medium distance, and high capabilities. However, BLE requires little to no infrastructure compared to a WLAN and is an inexpensive solution to connect a wide range of devices wirelessly. It is common to see computer and phone peripherals connected via Bluetooth such as keyboards, mice, and headphones. Like Wi-Fi, BLE utilizes the 2.4 GHz ISM bands, but not the 5 GHz Unlicensed-National Information Infrastructure (U-NII) band which 802.11a uses.

2.5 Wi-Fi Security Protocols and Attacks

Over the past two decades, Wi-Fi security has evolved to address discovered vulnerabilities. However, the rate at which these vulnerabilities are disclosed and

the time between the evolution of the security protocols is concerning. Approximately 65% of today's APs use Wi-Fi Protected Access II (WPA2) even though its vulnerabilities were demonstrated as early as 2006 at the popular conference RECON.CX [3] [36]. With a DWAP, an attacker can tactically exploit the majority of wireless networks in use today and enjoy safety miles away from the target.

2.5.1 Open Configuration

For ease of access, APs offer an unencrypted option. This "open" configuration allows users to connect to an AP without any authentication. The tradeoff for this ease of use is that traffic flows between the AP and station in plain text unless secured by a higher-layer protocol such Secure File Transfer Protocol (SFTP), Hypertext Transfer Protocol Secure (HTTPS), or Secure Socket Layer (SSL). If traffic is left unencrypted, the traffic is vulnerable to eavesdropping. Additionally, injection attacks (sending traffic to an AP or station while masquerading as one of the devices) and spoofing attacks (malicious devices pretending to be a legitimate device to capture traffic) are made possible. Despite the heavy security tradeoffs, open APs are commonly used throughout public spaces and often expected of shops to provide.

Another ease of access mechanism exists in many Wi-Fi devices for connecting to previously used APs, but it also comes with a security tradeoff. This mechanism continuously searches for APs by sending packets called 'Probe Requests.' These packets contain a list of the SSIDs of previous connections. Because of the design of the authentication process, this leakage of SSID information could lead a device to connect to malicious APs. When an AP receives Probe Request packets with its SSID, it can respond to let the station know they are within range. Because authentication only requires that SSIDs match, an 'Evil Twin' (malicious AP set up to look like a legitimate AP) could respond to a probe request to trick the device into

connecting [6].

One final convenience feature of APs that can inadvertently assist a malicious actor conducting an Evil Twin attack is the way stations attempt to connect to APs. When a station reconnects to a previously used AP, if there are multiple AP with the same SSID, it will connect to the AP with the strongest signal. If an AP does not support Management Frame Protection (MFP), which blocks erroneous deauthentication packets, an attacker can disassociate a station from a legitimate AP and have a spoofed AP with a stronger signal for the station to connect to [33]. This scenario becomes more plausible with drones added to the equation as they could carry equipment capable of spoofing APs and have the ability to move closer to a target for greater received signal strength at the target.

2.5.2 WEP

Wired Equivalency Privacy (WEP) was introduced in 1997 as the first Wi-Fi security algorithm [37]. It implements the Rivest Cipher 4 (RC4) stream cipher for encryption, the 32-bit Cyclic Redundancy Check (CRC-32) integrity check algorithm, and protected with a 64-bit key or 128-bit key. This key is composed of either a 40-bit or 104-bit secret key depending on which key size was used and a 24-bit Initialization Vector (IV). Figure 3 depicts WEP's security algorithm. The IV and secret key are concatenated, creating a seed input for the Pseudorandom Number Generator (PRNG). The resulting key sequence is XORed with the plaintext concatenated with a generated Integrity Check Value (ICV). Importantly, the IV which is needed for the decryption is sent in plain text with the cipher-text.

Within a few years, however, it was demonstrated that the key could be cracked with only the cipher-text because of an implementation flaw [38]. IVs are introduced to extend the lifetime of the secret key and should be random without repeats as

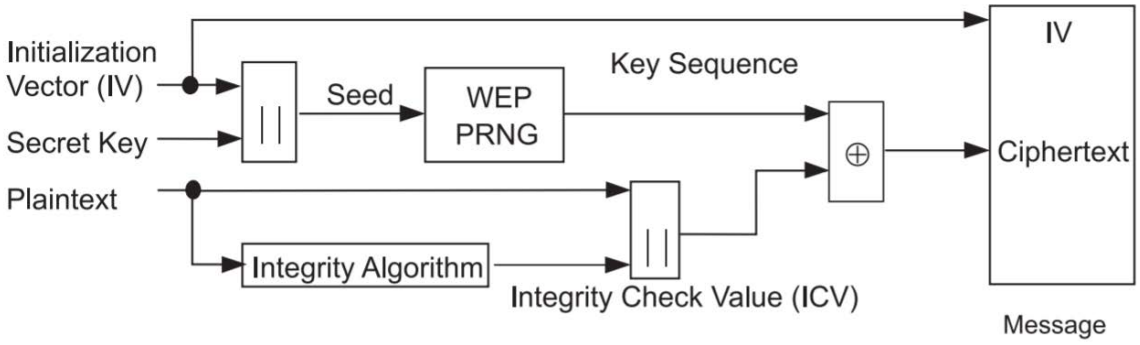

Figure 3: WEP encipherment block diagram [37]

not to reveal portions of the key sequence. Because of the small 24-bit IVs and the pseudorandom implementation of changing IVs, a repeat in the 16.7 million possible IVs can be expected with 99% confidence after 12,400 frames.

Optimizations to the attacks on the security algorithm have led to WEP's security being broken in a negligible amount of time [39]. These attacks take advantage of another flaw in WEP that allows replay attacks to artificially speed up the traffic for frame capture. Since being superseded by Wi-Fi Protected Access (WPA), WEP's peak usage has dropped from 45% to 6% in 2019 [3].

2.5.3 WPA

WPA was introduced in 2003 as a stopgap solution for link-layer insecurity WEP introduced [40]. WPA uses the Temporal Key Integrity Protocol (TKIP) for encryption. This protocol generates a 128-bit key for each packet and is, therefore, more resilient to the IV attacks from WEP. But, WPA2 was ultimately the desired security algorithm. WPA was put in place as a short term replacement because many APs that utilized WEP did not have the requisite hardware required for WPA2.

Rather than offering WEP's Open System and Shared Key authentication types, WPA and WPA2 offer two new modes to accommodate different user architectures. These modes are WPA-Personal and WPA-Enterprise. WPA-Personal is designed

for small-scale use, and users are required to authenticate with a passphrase that is 8 to 63 ASCII-encoded characters. WPA-Enterprise, or WPA-802.1X, requires an authentication server on the network to authenticate stations over the 802.1X control port before traffic is allowed.

2.5.4 WPA2

In 2004, an amendment to the 802.11 standard introduced WPA2 [41]. For authentication between devices, WPA2 implements the Extensible Authentication Protocol over LAN (EAPOL) four-way handshake. When an end-user architecture is set to the Personal authentication mode, a Pairwise Master Key (PMK) is generated using a cryptographic hash function. Through the use of the four-way handshake, WPA2 device pairs can verify independently that the other knows the PMK without sending the PMK over the media. This is accomplished by having a client and AP exchanging nonces, deriving a Pairwise Transient Key (PTK), and verifying the results. Additionally, the RC4 stream cipher was replaced by the Advanced Encryption Standard (AES) block cipher which utilizes a larger 128-bit key.

The EAPOL four-way handshake's unique scheme is what facilitates the secure authentication of devices and is depicted in Figure 4, with a station (Supplicant) on the left and an AP (Authenticator) on the right:

- The AP sends a 256-bit ANonce (Authenticator number used once).
- Utilizing the ANonce, SNonce (Supplicant number used once), the Media Access Control (MAC) address of the AP, and its own MAC address, the station derives the PTK. Then the station sends the SNonce and calculated Message Integrity Code (MIC).
- The AP now derives the PTK as it has finally received the SNonce. Then the

AP verifies that the station knows the Pre-Shared Key (PSK) by comparing the MICs. Upon a successful verification, an install message that includes the Group Transfer Key (GTK) and MIC the AP created are sent to the station. Otherwise, a deauthenticate message is sent to the station.

- Finally, the station verifies the MIC to ensure the AP's PTK is the same and responds with an ACK message that includes the MIC.

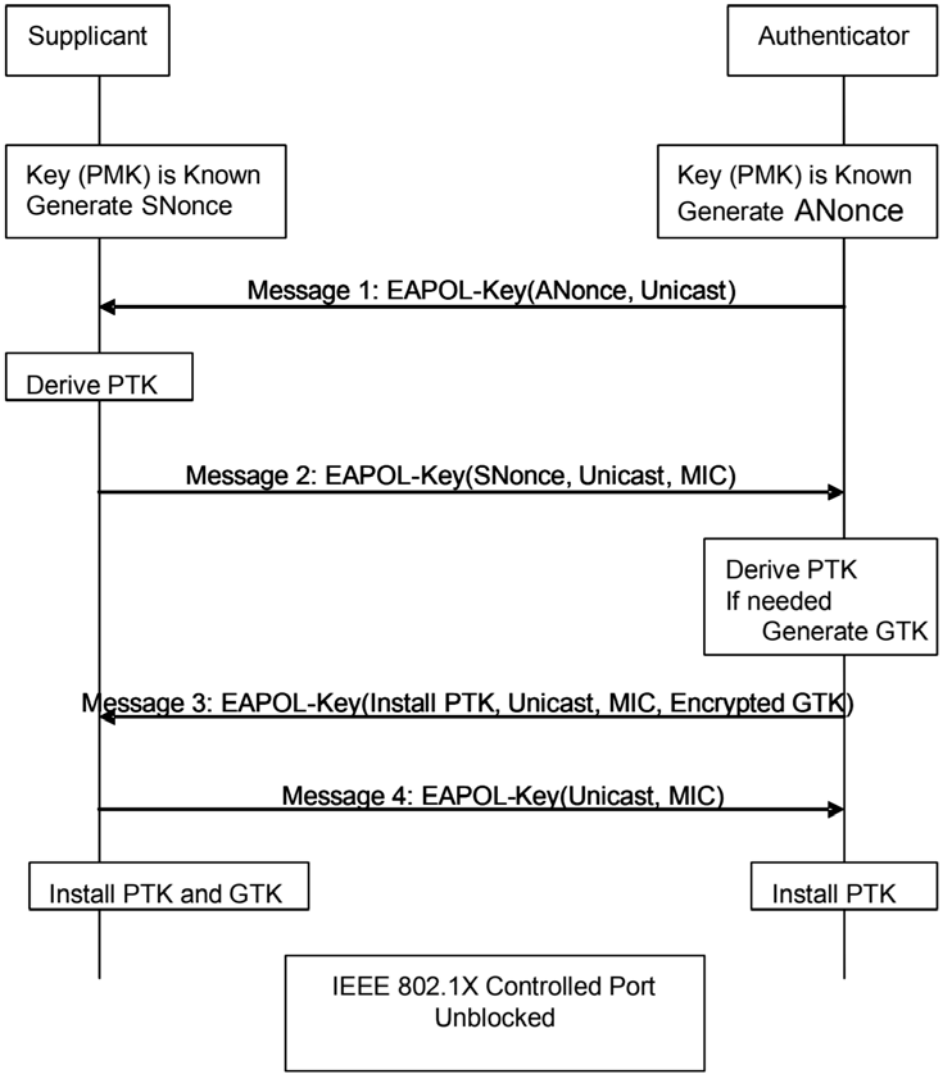

Figure 4: WPA2 four-way handshake [41]

2.5.5 WPA and WPA2 Brute Force Attacks

Unfortunately, due to the implementation of the four-way handshake WPA2 is susceptible to brute-force and dictionary attacks [42]. While this is considerably harder than compromising WEP, it can be accomplished offline after capturing only two of the four-way handshake messages exchanged between a client and AP during a legitimate authentication. Within the first two messages, all of the variables (MAC of client, MAC of AP, and both nonces) required to generate the PTK are present. With brute force guessing or the use of pre-built dictionaries, the PMK can be verified and ultimately the passphrase.

WPA and WPA2 are implemented with features to deliberately deter this class of attacks. When utilizing the WPA-Personal mode, APs generate the PSK using

$$PSK = PBKDF2(PassPhrase, ssid, ssidLength, 4096, 256) \tag{1}$$

where PBKDF2 is a passphrase-to-PSK mapping function and ssidLength is the string length of the SSID. First, PBKDF2 concatenates the passphrase, SSID, and ssidLength. Then it hashes the concatenation 4,096 times with HMAC-SHA1. This value is passed to a RSA key derivation function and results in a 256-bit output mapping. Then, each session established with the AP is encrypted with a unique 128-bit key derived from the 256-bit generated PSK.

The computational intensity of 4,096 hashes is added to slow an attacker trying to brute force a password. In order for an attacker to test a single password they must either preform the hash 4,096 times or utilize a pre-computed PSK and passphrase table (rainbow table). However, these tables are likely to be of limited use to an attacker because passphrases are salted with the SSID of their AP. This makes PSK and passphrase pairs unique to their SSID. Interestingly, the 802.11 standard warns

that "a pass-phrase less than 20 characters is unlikely to deter attacks" [41].

Additionally, other vulnerabilities have been demonstrated over WPA2's lifespan. As recently as 2017, researchers demonstrated a Key Reinstallation Attack (KRACK) against WPA2 [43]. KRACK is used to trick client devices into reinstalling an already in use key to manipulate association parameters and ultimately gain access to the network. These new vulnerabilities make it relatively easy to bypass WPA2's security.

2.5.6 WPA3

With multiple vulnerabilities in WPA2, the need for a new security protocol was answered in 2018 by the Wi-Fi Alliance when they announced WPA3. Some of the notable changes include the implementation of the Simultaneous Authentication of Equals (SAE) handshake, mandatory MFP, and forward security [33].

Figure 5 depicts the process of a client connecting to an AP using WPA3. Three main steps of this process are the SAE handshake, association, and the four-way handshake. Note that the four-way handshake is still in use, but is not vulnerable to offline dictionary attacks because of a new SAE method of generating the PMK. SAE is a Password Authentication Key Exchange (PAKE) and was first introduced in 2008 [44]. Both client and AP can initiate the SAE handshake by sending a commit message containing a scalar and password element. Each peer generates these two variables using two random numbers and the hashed password. Once the confirm messages are received, the peers can use the scalars and password elements to mathematically verify they each know the password. After confirmation they generate a Hash-based Message Authentication Code (HMAC) and send a confirmation to their peer. Upon verification of the HMAC, the client can request association and go through the four-way handshake as described in Section 2.5.4. The WPA3 implementation of the SAE handshake produces a high-entropy PMK, supports mesh networks, and prevents

decryption of captured network traffic if the passphrase is latter discovered (forward security) with the random numbers the client/AP produce.

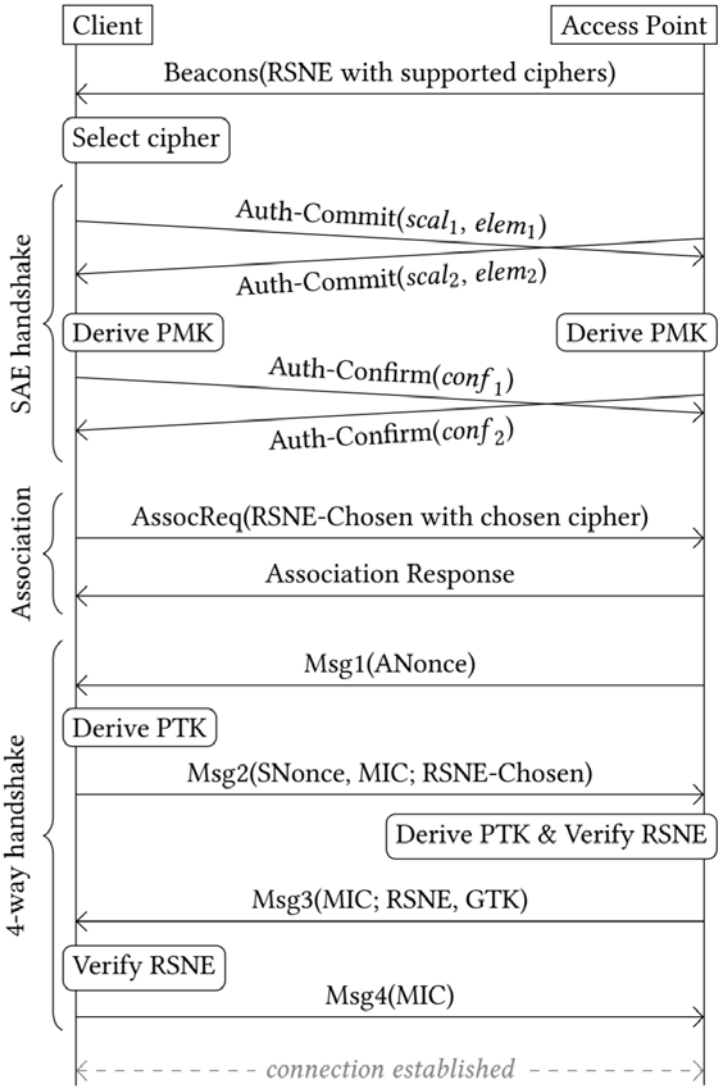

Figure 5: WPA3 authentication [45]

Unfortunately, researchers were quick to find a round of vulnerabilities that can result in password recovery, Denial of Service (DOS), or forcing devices into weaker security groups [45]. These vulnerabilities were disclosed to the Wi-Fi Alliance and mitigations have since been implemented. Due to the limited processing power of

APs, the creation of a secure and efficient security protocol has proven difficult over the past several decades. Therefore, it can be expected that new vulnerabilities will continue to be discovered.

2.6 Hacker Methodology

As the digital age has progressed, so has the hacking community. Hacking as a profession, which was nonexistent several decades ago, is now a viable commodity sought after by the military, government, and even private companies. While the terminology and methods have evolved, a common attack methodology has emerged and is depicted in Table 2 [6]. These six steps are the processes which hackers follow throughout CNA.

Additionally, in 2011 Lockheed Martin Corporation defined a popular methodology known as the Cyber Kill Chain [46]. This methodology is focused on informing Computer Network Defense (CND) resource prioritization decisions, relevant metrics, and identifying patterns to reduce the success of adversaries.

Table 2: Hacker Methodology

1. Reconnaissance
2. Scanning / Enumeration
3. Gaining Access
4. Privilege Escalation / Pivoting
5. Maintaining Access
6. Covering Tracks

Drones have the unique potential to assist an attacker during the first three steps of the hacker methodology. This is made possible with the RPP that they afford.

The first step for attackers is reconnaissance. This is the passive gathering of information relevant to a target. Understanding target specific lingo, management hierarchy, and security practices are some examples. The more attackers understand

a target and its operations, the easier it will be to perform the latter steps. Drones specifically can help attackers in this step with their high quality cameras and mobility. They can usurp physical barriers, capture images, and other intelligence data when properly equipped (e.g., GPS coordinates).

The second step of the hacker methodology is scanning and enumeration. It consists of using the gathered information to conduct an active analysis of a target's computer network. By sending short packets to target devices, revealing information can be returned by the receiving computer. The information can include open ports, operating system, and network topology. A common tool used for this task is nmap, and hackers use it to identify open ports that are often vulnerable such as ports 21, 22, 23, 443, and 445 [47]. If the services using these ports are not properly configured and secured, they can lead to security breaches.

If networks are well isolated, a properly equipped drone can potentially help an attacker gain a foothold in the network with its mobility and Wi-Fi capabilities. Firewalls that block messages from outside devices can be bypassed if a drone with a Wireless Network Interface Card (WNIC) connects to an internal Wi-Fi network and conducts scans.

The last step a drone can help accomplish is gaining access. This can be accomplished by exploiting identified vulnerabilities of the machines on a target network, convincing unsuspecting personnel to give access, also known as social engineering, or even password cracking accounts on open machines and Wi-Fi networks. If the need to connect to a secured internal Wi-Fi network arises, drones that are equipped with properly configured WNICs can facilitate the brute force attacks against Wi-Fi's security algorithms.

2.7 Related Research

This final section discusses the research that demonstrates the offensive capabilities of properly equipped drones. Also discussed are wireless vulnerabilities that could be enhanced if integrated with a DWAP.

2.7.1 Proof of Concept: Drones That Can Hack

Even though drones have received much attention from the academic community in the way of enhancements and their security being scrutinized, there have been few examples of drones being used to augment wireless attacks [48]. The work that does exist strongly suggests that drones used for wireless attacks could provide attackers a powerful tool.

One of the first notable examples is the Wireless Aerial Surveillance Platform, not to be confused with WASP drone discussed previously. Figure 6 displays the topology of the system. This drone was developed in 2011 by a pair of security researchers using an Army surplus target drone (FMQ-117B), avionics components, a small computer running Backtrack 5, and a USB 4G dongle [49]. It was capable of autonomous flight after takeoff and all equipment used were COTS with an approximate cost of $6200 at the time. The Wireless Aerial Surveillance Platform was capable of Wi-Fi password cracking through Aircrack-ng and other software on the Backtrack operating system. Additionally, the drone integrated a GSM cellular attack which involves masquerading as a GSM cell tower to capture cellphone calls and text messages [50].

The next highly relevant proof-of-concept hacking MUAV was developed by the UK security company 4Armed in 2015. This security company developed a drone payload in order to demonstrate the capabilities of a hacker using the leading consumer drone [51]. The result of their development was a DJI Phantom 2 Vision+ equipped with a Raspberry Pi, several Wi-Fi components, a 3G cellular dongle, and

Figure 6: Wireless Aerial Surveillance Platform system design [49]

extra USB batteries. As a demonstration they created a WPA2 Wi-Fi home network, captured the four-way handshake messages of a connecting device, cracked the password, and exploited a vulnerable Windows XP machine. Figure 7 depicts the drone and mounted payload which cost approximately $1150.

More recently, and likely the most advanced open source hacking MUAV, is the "Danger Drone," shown in Figure 8. This drone was developed by two researchers for their security consulting firm, Bishop Fox, as a penetration testing tool and was presented at DEFCON in 2017 [7]. Rather than relying on a COTS drone controlled through a Wi-Fi connection, the researchers created a drone from scratch, controlled by a Raspberry-Pi and interacted with it through a 4G Long-Term Evolution (LTE) cellular connection. This cellular connection allows them to avoid many of the jamming techniques employed against drones. The platform is also equipped with a Wi-Fi Pineapple, which is capable of impersonating APs, and WNIC to perform other wireless attacks. This hacking MUAV had a price tag of just under $500 and is used by

Figure 7: DJI Phantom 2 Vision+ with wireless attack payload [51]

the company as a penetration testing tool.

Over the short few years that these drones were developed, a rapid decline of production cost can be seen. For under $500, many of the capabilities of the Wireless Aerial Surveillance Platform can now be achieved. With a buy-in so low and more capable models to follow, malicious actors using these technologies are an eventuality.

2.7.2 Directional Antenna

Utilizing a DWAP requires a degree of stealth, and the use of a directional antenna provides this functionality. Directional antennas, when compared to omni-directional antennas, provide an extended range in a particular direction. This extended range should be leveraged by MUAVs since they are fairly loud and therefore should maintain a distance from a target in order to avoid detection. The typical consumer drone emits 76 dB which would be audible within 100 meters [52]. Law showed that with a

Figure 8: Bishop Fox Danger Drone [7]

Yagi directional antenna, the ideal relative angle to a target signal could be identified with a median bearing accuracy of 9 degrees [53]. He accomplished this by connecting the antenna to a stepper motor controlled by a Raspberry Pi and collecting signal strength readings at different bearings. With the readings taken at several locations, he could triangulate the target APs.

2.7.3 Cyber-Attack Drone Payload

Continuing the research thread of utilizing a directional antenna in tandem with a DWAP, Bramlette developed a drone payload (Figure 9) and web application for cyber-attack [1]. Unfortunately, in-motion bearing prediction of APs is far less accurate than those of Law's stationary experiments and had a median bearing error of 25 degrees. Correction of the bearing prediction is required before a more accurate geolocation can be achieved.

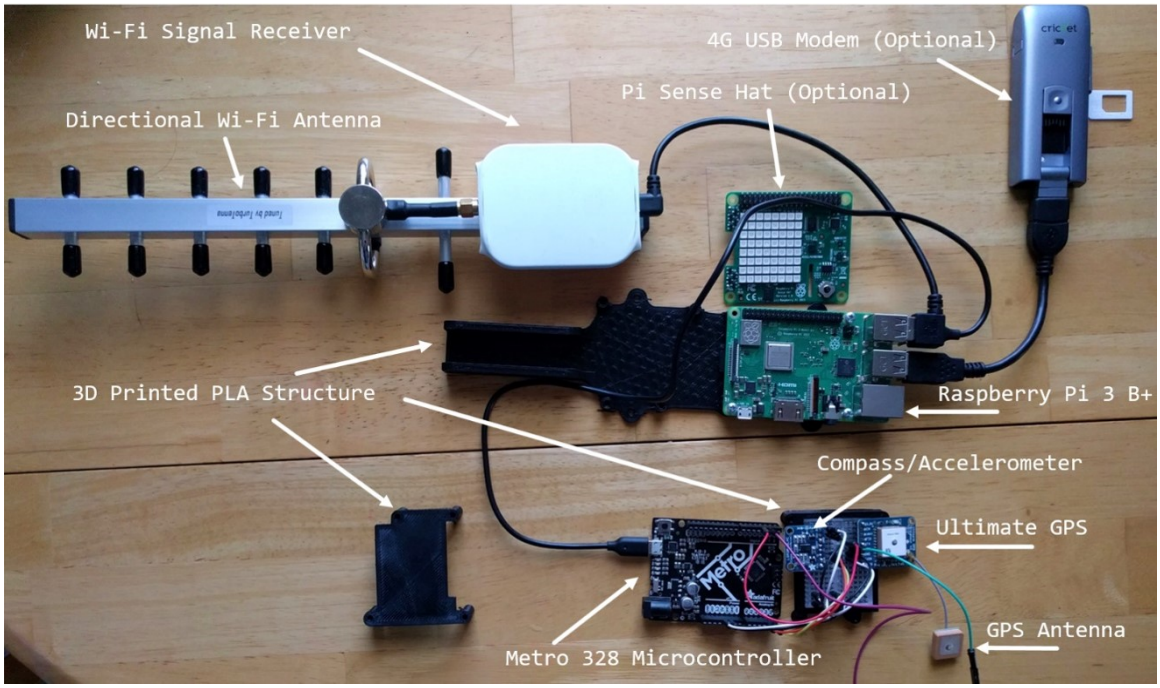

Figure 9: Bramlette's drone payload: "Skypie" [1]

While the geolocation of signal emitters was less accurate than expected, the framework that Bramlette developed is a solid start for further development. Modules can be easily added to perform different attacks, and it can be controlled remotely via a File Transfer Protocol (FTP) server.

2.7.4 Identification from Location

Privacy has been a growing concern for many years, especially with the advent of the digital age, and a team of researchers at the Massachusetts Institute of Technology demonstrated that human mobility data is particularly revealing [4]. With as little as four spatio-temporal points that identify a user hourly, 95% of individuals can be identified. This is because human mobility is highly unique, and this uniqueness can be exploited with even coarse datasets. Although cellular datasets, such as the 1.5 million user set the researchers used, does not include personal identifiers such as address or phone number, "individual's patterns are unique enough" to correlate against outside information. If such a technique was integrated into a DWAPs, it could prove to be a powerful tracking technique. With the simple modification to passively collect beacon frames that smartphones broadcast with static MAC addresses, a covert human mobility dataset can be built.

2.7.5 Data Leakages

In 2018, Beyer constructed a smart home consisting of 18 Wi-Fi and BLE devices and demonstrated that pattern-of-life modeling could be accomplished by passively sniffing wireless traffic [54]. With the collected data, Beyer was able to classify 94% of the Wi-Fi and 75% of the BLE devices with a script. With this device information available to him, he was able to correctly identify 95% of the smart home events (i.e., door locking) that occurred during his test. Even though the captured smart home

traffic was encrypted, Beyer took advantage of the unencrypted lower levels of the Wi-Fi and BLE protocols. Just as drones could augment human mobility tracking, drones could be a key enabler of this type of attack. They can be employed with directional antennas to passively collect wireless traffic well beyond a traditional omni-directional antennas range, providing an adversary with the necessary information to replicate a pattern-of-life attack.

2.7.6 Related Research Summary

Table 3 summarizes the related work conducted in the field of DWAPS. This research does not continue investigation into Wi-Fi localization, but develops wireless network attack capabilities for the skypie prototype [1] and evaluates their effectiveness when utilizing a directional antenna. This novel approach to DWAPs has the potential to significantly increase their operational use and is the first research to cover the four areas: multirotor drone, directional antenna, cellular modem, and wireless network attack.

Table 3: Related Research Summary

	Fix Wing Drone	**Multi-rotor Drone**	**Direc-tional Antenna**	**Cellular Modem**	**Wireless Network Attack**	**Wi-Fi Local-ization**
M. Tassey et al. [2011] [49]	X			X	X	
J. Greenwood [2015] [51]		X		X	X	
F. Brown et al. [2017] [7]		X		X	X	
B. Law [2018] [53]		X	X			X
C. Bramlette [2019] [1]		X	X			X
N. Barker [2020]		X	X	X	X	

2.8 Background Summary

This chapter provides a brief summary of rapid evolution of military and commercial drones. The wireless technologies Wi-Fi and BLE, are discussed as well as the security protocols deployed to protect Wi-Fi networks along with their vulnerabilities. The hacker methodology and how it can be augmented by DWAPs is explored. Lastly, the related drone research is explained and additional wireless attack avenues are presented for DWAPs.

III. Prototype Design

3.1 Overview

This research presents and analyzes data collected from an enhanced hardware and software prototype previously developed by Bramlette [1] as discussed in Section 2.7.3. The sensor payload, skypie, is further developed throughout this research to fully realize the CNA design goals of the prototype and evaluate the performance of the attack capabilities that a directional antenna can afford. It is equipped with GPS and an accelerometer, so that it may operate independently from the flight system of the drone. This also allows for compatibility with drones capable of carrying weights less than or equal to the target of 1 kg. Figure 10 shows the sensor payload's GPS and accelerometer (top), directional antenna and WNIC (middle), and computer and cellular modem (bottom).

Bramlette's sensor payload was only capable of passively collecting wireless traffic autonomously on the 2.4 GHz band, be controlled via configuration files pulled from an FTP server, or directly through a remote shell [1]. The now upgraded skypie version 2 (skypie v2) is additionally capable of passive wireless traffic capture over both the 2.4 GHz and 5 GHz band, interacting with wireless APs, performing WPA handshake attacks, and conducting nmap scans of target networks. The remote configuration and control is made possible by an added 4G LTE USB modem that establishes a cellular connection to the Internet. Cellular connectivity also notably extends command and control theoretically across the globe.

The code responsible for control and collection is written in Python 3.7 and has added many feature to the original skypie codebase. The modifications and added features to Bramlette's repository account for an additional 700 lines of code.

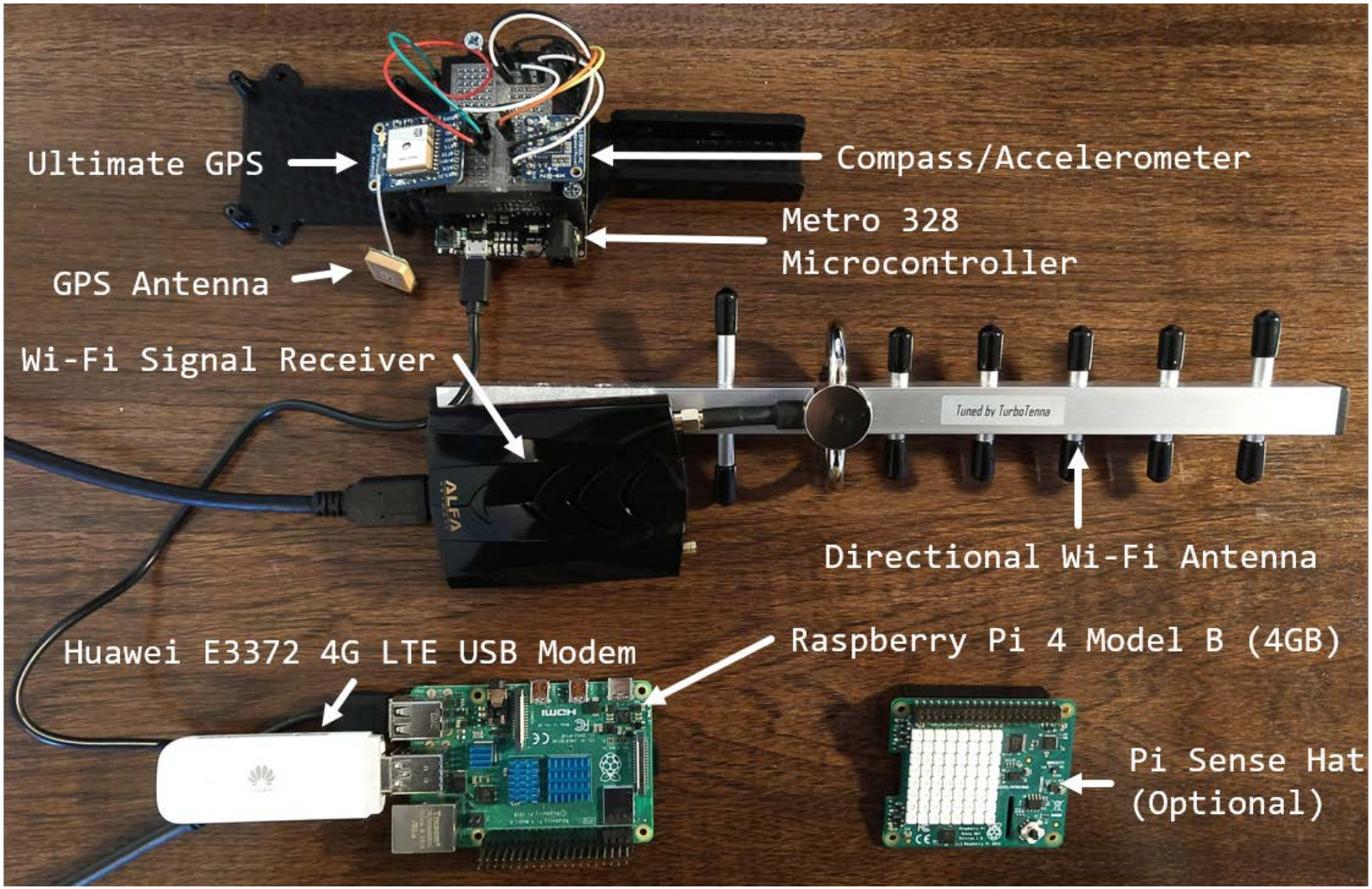

Figure 10: Skypie v2 sensor prototype

3.2 System Summary

The system has two major components: the sensor payload (skypie v2) and the command/analysis web application (skyport). All captured data is uploaded to an Internet-facing server which acts as an intermediary between the two components. When using skyport, an attacker can download all of the captured data files (PCAP, nmap results, and GPS coordinate data), view geolocation history over satellite imagery, and issue commands to a sensor payload. Figure 11 outlines the relationship between the two components and how data flows between the two.

Ideally, skypie v2 should be mounted underneath a MUAV and launched at a distance from a target that is discrete. The drone operator does not have to be skypie v2 operator, but could be someone that works in conjunction with an attacker. An effective team or single attacker could then fly and angle the DWAP's antenna at the target APs to collect wireless traffic, and conduct WPA handshake capture attacks through skyport. Figure 12 depicts a likely attack scenario where skypie v2 could bypass physical security on a drone and conduct CNAs against a distant network.

Captured handshakes are uploaded to the FTP server. This allows an attacker to download and crack the WPA password on a powerful workstation rather than on the limited hardware of skypie v2. After a successful crack, the attacker can simply enter the password into the skypie configuration file (see Appendix A for an example configuration file), select the AP in skyport, and then freely connect/disconnect skypie v2 to the Wi-Fi network.

From this point, many attack options become available to the operator(s). One useful tool is the ability to conduct nmap network scans. Skyport can also be used to direct customized nmap scans and set time limits for each of those scans. If more fine-tuned control is necessary, a remote shell on skypie v2 can be opened and used in the web interface.

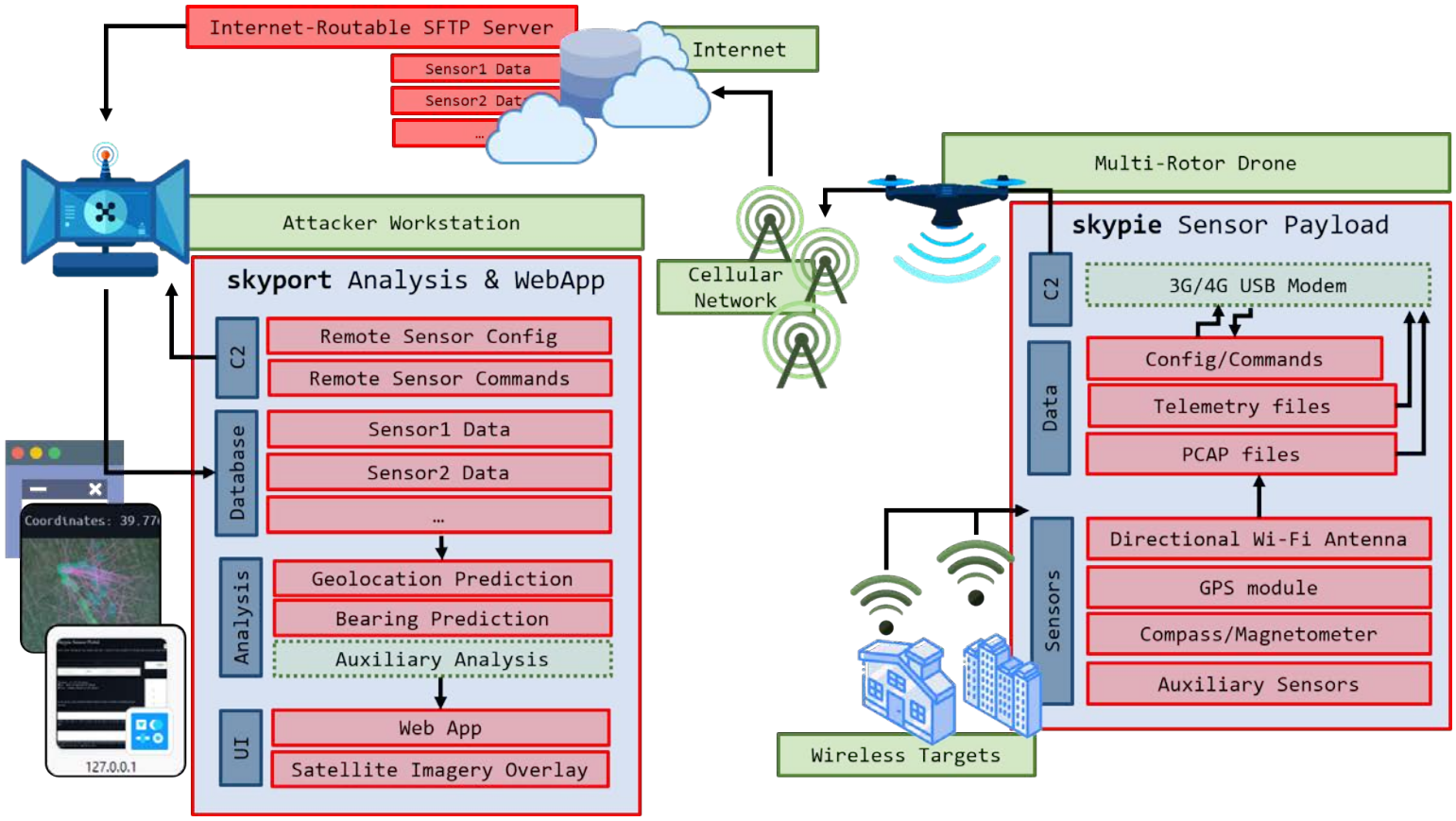

Figure 11: Skypie/skyport system design [1]

On top of being able to download all captured scans and captured data, many collection parameters can be changed through skyport. Parameter changes include adding Wireshark filters to wireless collection, timeout lengths, FTP upload/download intervals, trigger different automated attacks, and more. All of these changes can be sent to the FTP server even when a skypie v2 sensor is not actively pulling configurations. If the sensor experiences a loss in cellular connection, changes are made as soon as connectivity has been restored. If the skypie v2 is set to 'off' mode, periodic checks are executed, and if new configuration changes have been uploaded they take effect.

3.3 Design Goals

As the prototype utilized for this research was originally designed by Bramlette [1], his design goals are adhered to when upgrading the skypie prototype. They are as follows:

- **Low Cost.** With the ever increasing availability of low cost commercial drones as discussed in Section 2.3.3, the need to model poorly resourced, yet motivated threat actors has arisen. Therefore, a target of less than $500 is chosen for the development of the skypie sensor payload. Note that this does not include the cost of a drone.

- **Realistic Utility and Robustness.** In order to ensure that the prototype is a reliable tool for CNA/CNE drone-based operations a set of capabilities are sought in the hardware selection and software development. These capabilities are:

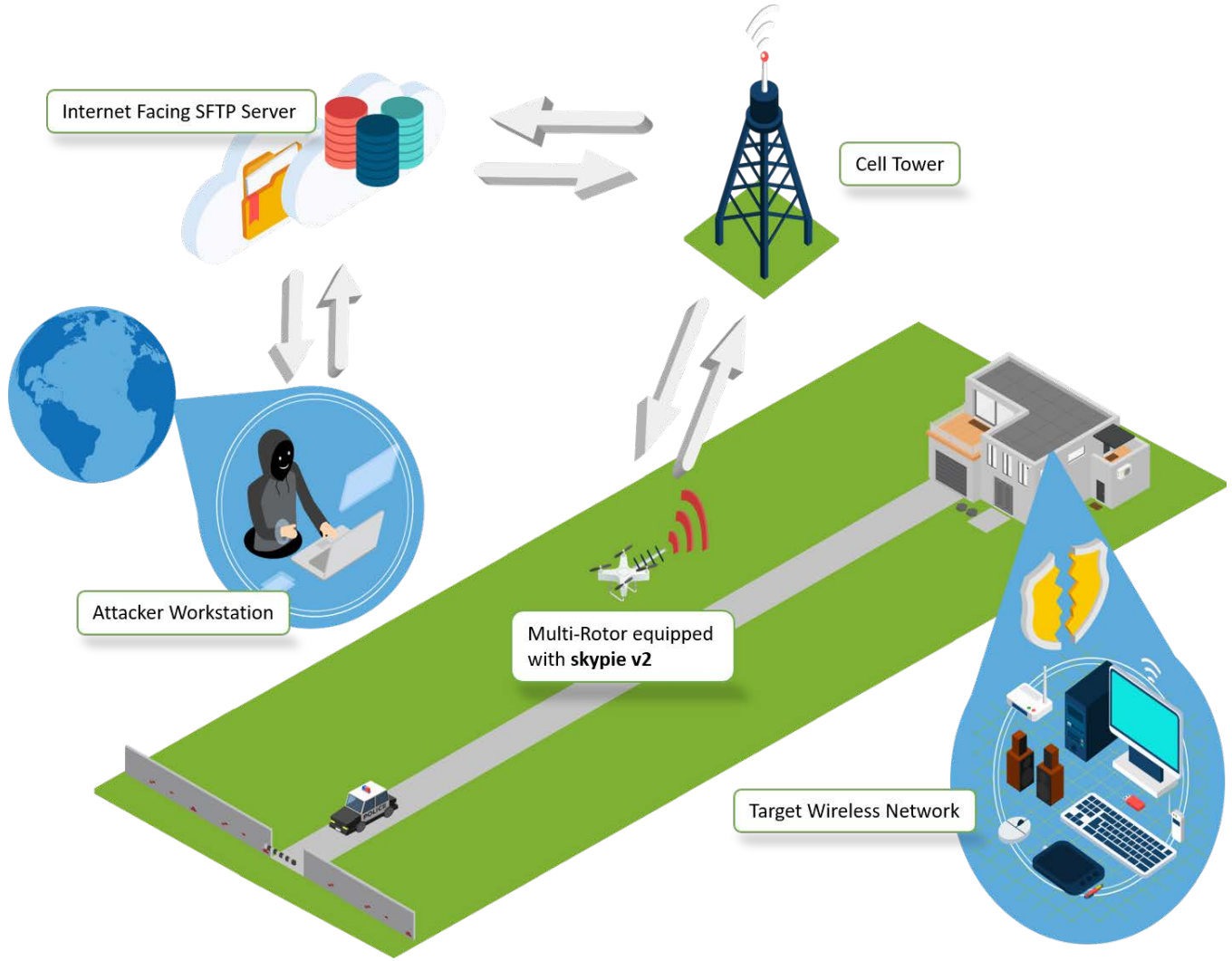

Figure 12: Skypie v2 attack scenario

i. The sensor payload should be able to operate autonomously or be securely controlled via a remote wireless connection. If connection is lost between sensor and attacker, communication resumes when it becomes possible

ii. Near real-time communication between sensor payload and attack should be possible to enable feedback and control.

iii. The sensor payload should be equipped with adequate storage and battery to enable multi-hour operations.

- **Drone Architecture Portability.** The payload should be independent of any drone to which it is attached. This gives skypie the advantage of being attached to any drone capable of the lift requirement and decouples it from the drone's flight system, thereby decreasing its complexity. With a modular payload, rapid development is possible, and mitigates the possibility of becoming outdated if linked to a singular drone. The tradeoff for this design goal is that the payload must include its own sensors for data collection, such as a GPS module and accelerometer. Additionally, a cellular modem is necessary for wireless command and control. Notably, this may cause some overlap between the equipment on the drone and the sensor payload (i.e., both having a GPS module).

- **Low Weight.** As discussed in the previous goal, additional equipment is necessary in order for skypie to operate independently from a drone. According to a 2019 review of a range of medium to large consumer drones, the carry capacity for medium sized drones ranges from 3 kg to 9 kg, and high-end drones reach lift capacities up to 30 kg [55]. In order to most realistically reflect the capabilities of a motivated but ill-funded malicious actor, the weight limit is set to 1 kg. This limit ensures the compatibility with the widest range of COTS drones.

3.4 skypie v2 Hardware Design

Some of the hardware components for the new prototype remain the same from the original design, but select parts are added or upgraded in order to better fulfill the design goals. Table 4 outlines all of the parts required to build the prototype as well as the models used, weight, and price. Highlighted in the table are upgraded hardware components used on skypie v2. Figure 13 is a hardware schematic that details how all of the components are assembled.

Table 4: Prototype Hardware Overview Adapted From Bramlette's Table [1]

Part	**Model / Version**	**Weight (g)**	**Price**
Directional Antenna	Danets USB-Yagi TurboTenna	137	$110
Wi-Fi Interface Card	ALFA AWUS036ACH	52	$60
Computer	Raspberry Pi 4 Model B (4 GB RAM)	46	$55
Digital Storage	16 GB SanDisk Ultra microDSCH UHS-1	1.7	$6
Microcontroller	Adafruit Metro 328	16.5	$18
GPS External Antenna	Passive GPS Antenna uFL - 15mm x 15mm 1 dBi gain	5.5	$4
GPS	Adafruit Ultimate GPS Breakout - 66 channel w/10 Hz updates - V3	8.5	$40
Accelerometer	Adafruit Triple-axis Accelerometer+Magnetometer (Compass) Board - LSM303	2	$15
Power Supply	Charmast 10400 mAh 3 A External Battery Model: W1056	228	$23
4G LTE USB Dongle	Unlocked Modem Huawei E3372-510	18	$30
Optional Sensors	Raspberry Pi Sense HAT	20.4	$38
Structure	3D printed casing	52	$2
Structure	Mini-breadboard	13	$2
Miscellaneous Components	Screws, bolts, wiring, headers, USB cables, solder	36	$16
Total		**636.6**	**$419**

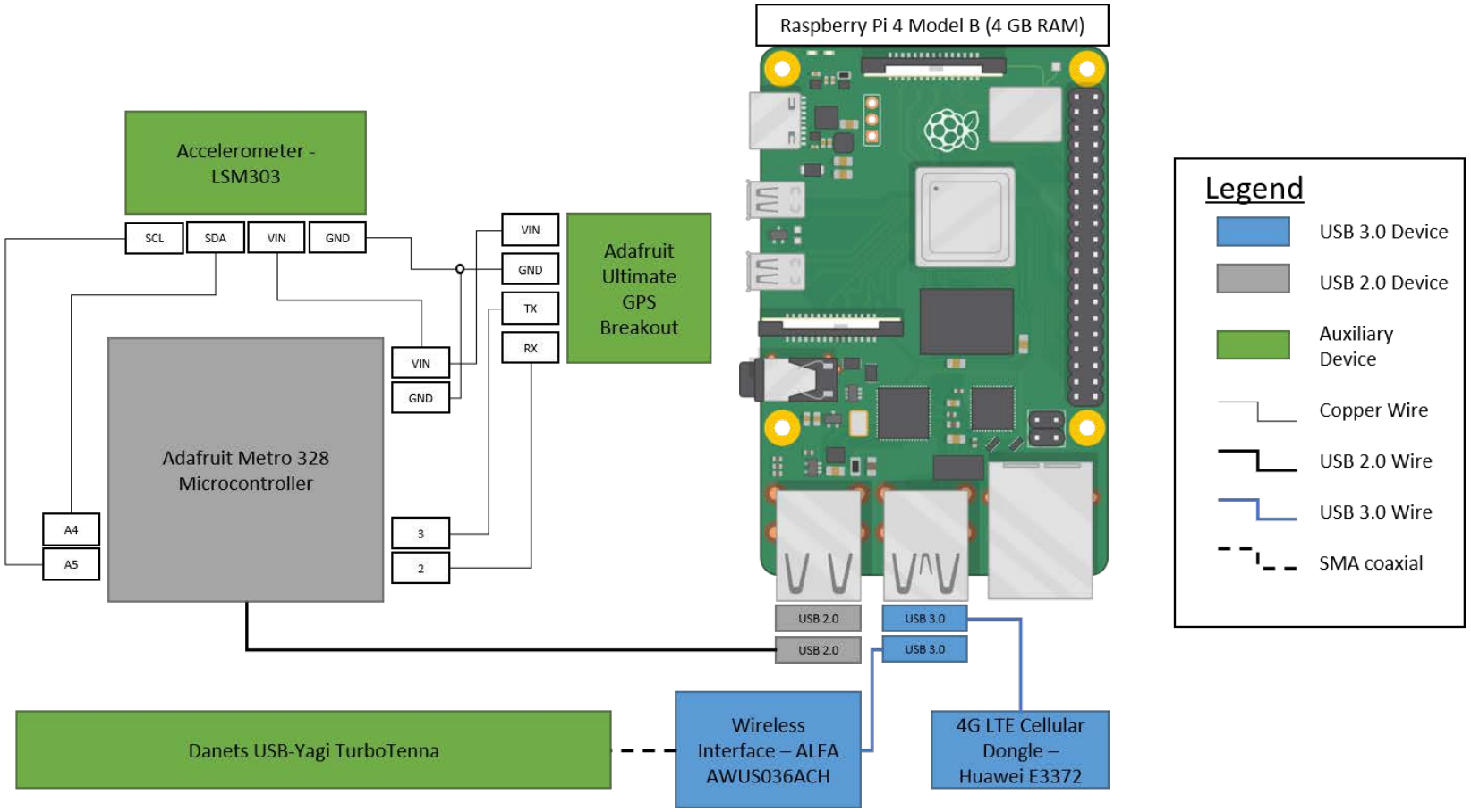

Figure 13: Skypie v2 hardware schematic

3.4.1 Upgraded/New Hardware

In order to better fulfill the design goals described in Section 3.3 and support the new features added to skypie v2, the following hardware modifications are made:

- **Wi-Fi Interface Card.** The ALFA AWUS036ACH Wi-Fi interface card is selected over the previously used DNX10NH-HP for its 5 GHz compatibility. The previous build of skypie was limited to passive capture of wireless traffic on the 2.4 GHz band. In skypie v2, this new Wi-Fi interface card can facilitate the capture of network data and interact with wireless networks operating in both the 2.4 GHz and 5 GHz bands. Another benefit of the ALFA card is the dual antenna coaxial connectors which allows for two antennas to be equipped to skypie v2. As shown in Figure 10, the top antenna is responsible for capture of 2.4 GHz traffic and a bottom antenna (when connected) is responsible for 5 GHz traffic.

- **Computer.** The Raspberry Pi ecosystem remains an ideal platform for this sensor payload as it is has four USB ports and GPIO pins to support multiple sensors. The Raspberry Pi 4 4 GB model's small price increase over the Raspberry Pi Model 3 B+ comes with many advantages. It remains lightweight/low cost and is more powerful, has 4 GB of RAM, and supports dual monitor outputs. These advantages fulfill many of the design goals, especially the drone architecture portability. The added dual screen support and upgraded 4 GB LPDDR4 SDRAM over the previous 1 GB LPDDR2 SDRAM on the Raspberry Pi 3B+ make developing on skypie v2 easier. These quality of life improvements allow for rapid software development all on the Raspberry Pi, enabled by a full-feature Integrated Development Environment (IDE) (in this research: Pycharm) run on two screens.

- **Power Supply.** A new battery is selected, because the Raspberry Pi 4 requires a 5V/3A battery. This battery achieves a better weight to power ratio than the previously used Aibocn battery, and can supply several hours worth of power with a total of 10400 mAh.

- **4G LTE USB Modem.** While previously out of scope of the original skypie design, the addition of a 4G LTE USB modem is necessary to fulfill the realistic utility and robustness design goals. The Unlocked Modem Huawei E3372-510 is a suitable choice for this goal as it has plug and play compatibility with the Raspberry Pi's Linux-based operating system (Raspian).

3.4.2 Retained Hardware

Upon reevaluation, the parts listed below meet or exceed the design goals and are retained in the skypie v2 build.

- **Directional Antenna.** When considering antennas for skypie, the Dantes USB-Yagi TurboTenna fulfilled the design goals of being low cost and low weight. These are especially important considerations as directional antennas are typically longer/heavier than omni-directional antennas. They also have larger cross sections and could affect the aerodynamics of a carrying drone. The Yagi directional antenna is reasonably sized at 31.5 cm in length and weighs 137 grams. The high-power beam in which the Yagi directional antenna is capable of capturing has a beam width of approximately 56 degrees and adds a gain of 18 dBi [56].

- **Microcontroller.** The Adafruit Metro 328 acts as a real-time controller for additional hardware modules added to the system. The Raspberry Pi functions as a full fledged computer with an operating system, but does not provide real-

time support for the required GPS and accelerometer modules. For this reason, and because the Adafruit ecosystem has a vast array of compatible components, the Metro 328 was chosen. Power supply and interaction with the Metro 328 is accomplished through a single micro USB port.

- **Global Positioning Module.** The Adafruit Ultimate GPS Breakout (66 channel with 10 Hz Updates Version 3) was chosen to fulfill the realistic utility and robustness design goals. This module can provide real-time GPS data at a rate of 10 updates per second with 3-meter accuracy. Its signal sensitivity reaches as low as 165 dBm and provides jammer detection and reduction. It is additionally coupled with a GPS 1 dBi gain antenna to enhance its reception capabilities.

- **Accelerometer.** Chosen for its compatibility with the Metro 328 and compact form factor, the Triple-axis Accelerometer + Magnetometer (Compass) Board LSM303 is a necessary module for directing skypie. With the use of a directional antenna, it becomes necessary to know the bearing at which the drone is pointed, so that the antenna's high-powered collection beam is oriented towards target networks.

- **Optional Sensors.** For development purposes, the Raspberry Pi Sense Hardware Attached on Top (HAT) is included in this build. While this sensor includes atmospheric pressure, humidity, and gyroscopic readings, it is exclusively used as a development tool. This module falls under the drone architecture portability goal and its Red-Green-Blue (RGB) Light-Emitting Diode (LED) panel is used to facilitate rapid development as a debug tool. Colors and text are displayed on the panel to indicate the mode and configuration of skypie. This serves as a diagnostic tool when developing and testing the payload. See Table 13 in Appendix B for a description of the RGB indications.

- **Structure.** Shown in Figure 14 are the assembled 3D-printed parts that securely mount all of the skypie's components. The material used in the printing process is Polylactic Acid (PLA), which is a biodegradable filament.

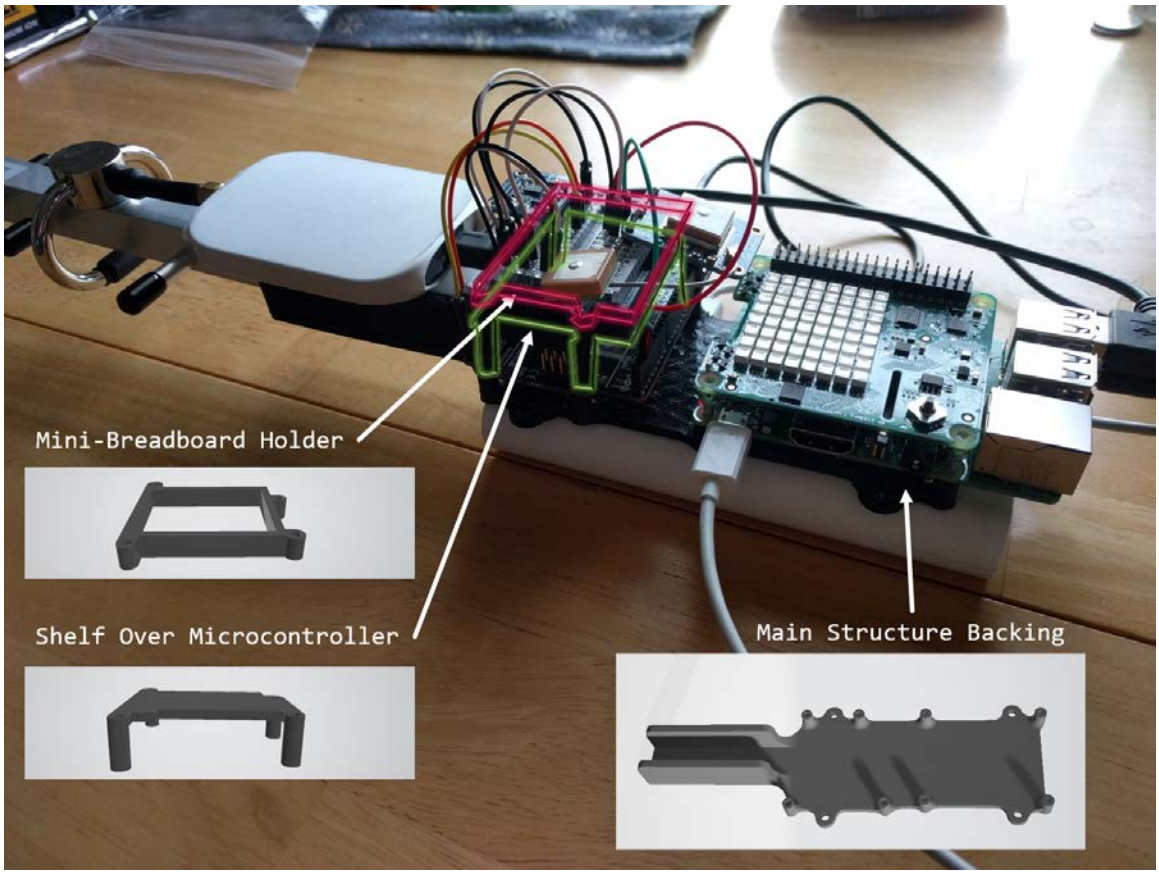

Figure 14: Skypie 3D printed structure components [1]

3.5 skypie Software

The skypie software package is written almost exclusively in Python 3.7. The exceptions are the code to control the microcontroller and the geodesic intersection algorithms which are both written in C++. This C++ code remains in the repository as it was previously developed by Bramlette [1], but is unused in this research. The software repository consists of three packages: skypie, skyport, and shared. Each

package contains modules that control and support the features of the sensor payload and web application. These modules are discussed in the following sections, with an emphasis on the new features added to skypie v2.

3.5.1 Design Constraints

Design constraints followed during the development of the original skypie are adhered to while building skypie v2. These constraints as discussed in [1] are listed below:

- Fine-tuned control of the sensor payload can be achieved through modification of a single configuration file. While manual control is possible through a remote shell, an attacker's workflow is simplified with access to a multi-module configuration file. The skyport web application, provides an attacker a helpful Graphical User Interface (GUI) for modifying this configuration file which facilitates rapid control over several wireless attacks.

- The skypie program runs in a control loop and configuration changes are handled at the beginning of each loop.

- The control loop spawns new threads as directed by the configuration file to complete subtasks (e.g., passive Wi-Fi traffic collection, WPA handshake capture, nmap scans, etc.) and those threads are initialized with the parameters (e.g., Wireshark filters, target MAC addresses, timeouts, nmap search paramaters) also in the configuration file.

  - Threads currently running are not affected by configuration changes. This adheres to Python's best practice guidelines and ensures that unexpected behavior is minimized.

- Threads are completed asynchronously, which allows the main control loop to continue while multiple tasks are completed.
  - Starting threads that can interfere with the Raspberry Pi operation of a running thread will be deferred until the completion of the running thread.
- Threads are initialized with a parameterized run time. This is to ensure that threads lifetimes are finite and prevent the need to interrupt threads. This comes with the benefit of easier code implementation and being less error prone, but has the disadvantage of transition intervals where wireless data could be missed.
- Due to the mobile nature of an attacker and sensor, it is not likely that either will have a static IP address. Therefore, to facilitate secure communication between the two, an Internet-facing FTP server is used as a 'dead drop' location. While in operation, skypie periodically uploads the files it has captured/created and downloads new configurations. This allows an attacker to login and download all captured files and issue new commands at anytime.
- Analysis and cracking is performed on an attacker's workstation through the system's second main component (skyport). Just as skypie is controlled by one configuration file and operates with a control loop, so does skyport.
- To ensure the robustness of skypie's software, the operating system is set up with a chron job. This chron job checks periodically that the skypie program is running and if not, it restarts the program. This protects against unexpected crashes and ensures dormant sensors are routinely "checking in".

3.5.2 Updates to the skypie Package

The skypie package contains the code run solely on the payload's computer. This computer (Raspberry Pi) requires a Linux-based operating system and several software packages. Listed in Table 5 are all of the dependencies with new dependencies highlighted.

Table 5: skypie v2 Dependencies

Package	**Function**
aireplay-ng	Injecting generated packets (i.e., deauthentication packets)
airodump-ng	Targeted WPA handshake capture
dumpcap	Capture packets from Wi-Fi interface
iwconfig	Get Wi-Fi adapter settings, set monitor mode/channel
ifconfig	Prepare WNIC for monitor mode
iwlist	Get Wi-Fi interface current channel
nmap	Conduct network scans
tshark	Makes packet captures available to Python for analysis

The structure of this package builds upon the previous skypie model [1]. As shown in Figure 15, when the software is initialized, the manager module is invoked to manage the main control loop. At each iteration through the control loop, the configuration file is read and requested changes are processed. These changes may be simply variable management or requests to start a new thread. The following sections discuss the modifications and additions to the skypie package. Figure 15 also highlights the upgrades and behavior of the skypie package.

Nearly all Python modules in this package are written as thread classes so that they can run in parallel. In skypie v2, there are six additional threads, and modifications to interface.py. Notably, manager has been modified to prevent clashing threads from starting, a timeout has been added for all attack threads, and target AP information is provided to the threads.

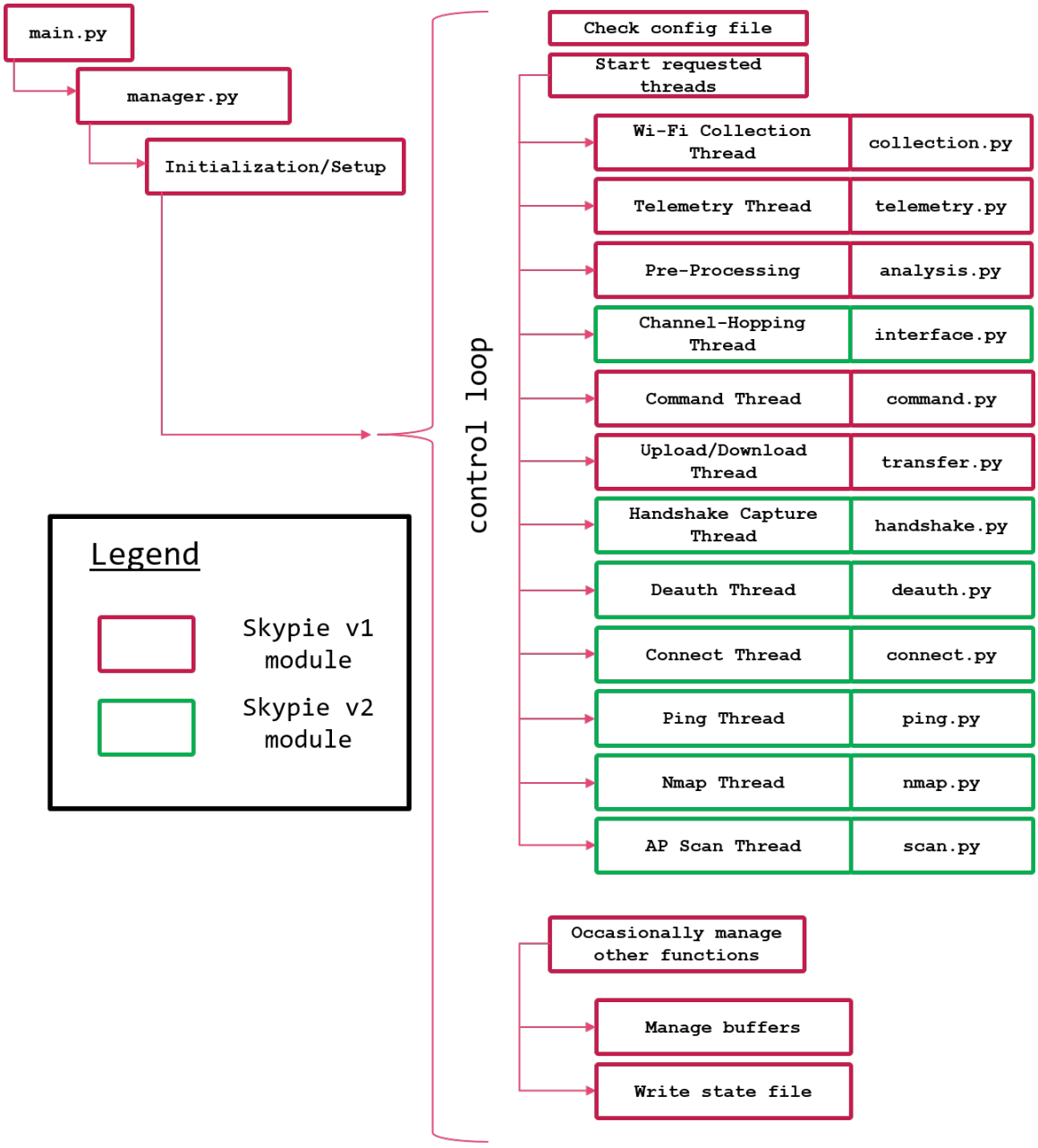

Figure 15: Skypie v2 control flow diagram

3.5.2.1 interface.py

As Bramlette's research focused on the geolocation of all APs in proximity of skypie, this module is built with inherent Wi-Fi channel hopping. In order to allow for focused attacks, modification of this module is necessary. On top of the ability to select and manage the settings of an asynchronous channel-hopping wireless interface, a target AP can be indicated by its MAC address. When a target is specified in the configuration file, the manager module periodically calls scan module which runs the command

iwlist [WNIC] scan

Its results are parsed to identify the current channel of the target AP and if the skypie v2's channel does not match, the module sets the interface to the current channel.

3.5.2.2 handshake.py

This module is responsible for the capture of WPA handshakes of a specific target. This can be accomplished passively or in conjunction with the deauth attack discussed in Section 3.5.2.3. The thread class that makes up this module checks if the wireless interface being used is in monitor mode during initialization, and start an airodump-ng capture with the command

airodump-ng --bssid [SSID] --c [channel] -w [filename] -o pcap [WNIC]

Until the timeout specified in the sensor's configuration file is reached, the airodump-ng output is parsed for a successful WPA handshake indication. Captured handshakes are saved in pcap files in the './data/synch/handshake' directory following the naming convention 'WPA-Handshake-[MAC address].pcap' along with the signal strength of the AP in the file 'signal-strength.txt'.

3.5.2.3 deauth.py

The thread class defined in this module can be used as a helper thread to the handshake module discussed in Section 3.5.2.2, or as a DOS tool. In either case, the use of this module can be detected by the target network and should be used sparingly in order to avoid detection. When initialized, the target AP's MAC address from the sensor's configuration file is fed to aireplay-ng to craft a WPA deauthentication message with the command

aireplay-ng --deauth [number of packets] -a [MAC address] [WNIC]

Until the timeout specified in the configuration file is reached, the crafted erroneous deauthentication message is sent every five seconds. If the targeted AP does not support management frame protection, these deauthentication messages force all connected devices to disconnect. The five-second interval allows enough time for connected devices to reauthenticate, but if the timeout for this thread is set for a long duration, it effectively becomes a DOS attack.

3.5.2.4 connect.py

On the Linux-based Raspbain operating system, Wi-Fi connections are handled by the 'wpa_supplicant' process. This process is controlled through a GUI on the desktop that modifies a configuration file, or by direct modification of the configuration file with the command line interface 'wpa_cli'. The connect module is a thread class that reads the configuration file, adds a specified AP's credentials (if not already present), and reconfigures the 'wpa_supplicant' to connect to the AP. See Appendix C for the code and commands that accomplish these tasks.

3.5.2.5 ping.py

This concise module is a threaded class responsible for managing a ping subprocess. Used in the experiments and discussed in Chapter IV, this module sends 10 ping packets to the IP address specified in the sensor's configuration file with the command

`ping -c 10 [ip address] [filename]`

Then it saves the results file to the './data/synch/ping' directory following the naming convention 'ping-[MAC address].txt'. Appended to the results file is the signal strength of the AP, which is parsed from iwlist's results. Because ping requires a connection to a network, the manager module has logic in place to ensure that the sensor is connected before this thread can be created and started.

3.5.2.6 nmap.py

The nmap module is used to conduct any desired nmap scan per the parameters passed in the sensor's configuration file. The output is saved in the './data/synch/nmap' directory appended with the received signal strength of the AP. The command in this research's experiments is:

`nmap -p 21,22,23,443,445 -T4 -v -oN [filename]`

The results are saved in the following file format 'Nmap-[MAC address]'. Just as all other attack threads, the nmap thread module has a controllable timeout in order to ensure that a scan is completed in a timely manner, and the thread does not get stuck in an infinite loop.

3.5.2.7 scan.py

This thread is periodically called, dependent on the sensor's configuration files, by the manager to fill a dictionary ('./data/synch/available-networks.txt') with all

the available APs in the sensor's collection range. This is accomplished by parsing the data returned from iwlist's scan command. The dictionary created is used by many of the other threads, and is made available to the attacker via the FTP server. The interface thread uses the created dictionary to ensure that the target AP has not hopped channels, and if it has, the thread changes channels accordingly. The connect thread requires the SSID of an AP as a connection parameter and searches the scan dictionary by MAC address to obtain it.

3.5.3 Microcontroller/Geodesic Intersection Algorithms

This research does not modify code for the Adafruit 328 microcontroller, which is responsible for the collection, parsing, and formatting of hardware module data. This data includes GPS coordinates, bearing, altitude, and timestamps. These jobs are controlled by the C++ code loaded on the microcontroller. Additionally, the geolocation algorithms that are used for data analysis are written in C++. While this code is still present, this research focuses on the effectiveness of skypie v2 as a cyber-attack tool.

3.6 Design Summary

This chapter outlines the skypie v2 prototype and its upgraded attack capabilities. Staying true to the original design goals of the skypie prototype outlined in Section 3.3, hardware and software modifications are made to make the payload a more effective cyber-attack tool.

IV. Methodology

4.1 Overview and Objectives

The experiment conducted during this researched is aimed at testing the effectiveness of CNA capabilities using light-weight equipment on a cyber-attack drone. As discussed in Section 2.7.2, to avoid audio detection, a drone should maintain distances greater than 100 meters. Although the typical range of consumer Wi-Fi devices ranges from 50 meters indoors to 100 meters outdoors, utilizing a directional antenna can extend that reach by multiples times.

This research extends Bramlette's work by adding attack capabilities to skypie and attempting to answer the following questions:

- Can CNAs be accomplished at 800+ meters using light-weight equipment on a cyber-attack drone?
- If so, how long does each attack take?
- At what distance do they become infeasible?
- How effective would these attacks be against a realistic network setup?

It became clear during the first experiment where the attenuator factor is set to none, the available open field real estate became a limiting element. To mitigate this, an attenuator is added to the System Under Test (SUT) as a factor to artificially add attenuation and simulate additional distances.

4.2 System Under Test

Figure 16 displays the SUT and Component Under Test (CUT) diagram. The parameters that change through the experiment are factors, and are covered in Section 4.3. These varying factors are measured with the output metrics (discussed in

Section 4.4). The computing parameters and constant variables are those that are held constant throughout the experiment and covered in Section 4.5.

4.3 Factors

Factors are parameters that are varied throughout the experiment. The factors identified in Figure 16 are listed below and summarized in Table 6.

1. **Attack Mode**. Three tasks are chosen to test skypie's attack potential and evaluate the performance of the SUT. The tasks are (1) WPA handshake capture attack, (2) nmap scan of the target network, and (3) 10-ping burst collection. The handshake capture and nmap scan are realistic and useful attacks that are chosen to demonstrate CNAs can be accomplished on the selected hardware. The ping burst collection is used to understand and graph at what distance communication with a target AP becomes infeasible.

2. **Distance**. This is the distance in meters that the sensor payload is from the target AP. The experiments are conducted with 200-2200 m between skypie v2 and the AP for a total of 11 collection points. The varying distances allow for the analysis of attack performance as distance grows.

3. **Attenuator**. To simulate additional distance beyond the physical limitations of the test site, the use of an attenuator is necessary. The experiments are run with the following configurations: no added attenuator and a 15 dB attenuator. Figure 17 shows the attenuator. The attenuator is installed between the wireless interface card and the directional antenna.

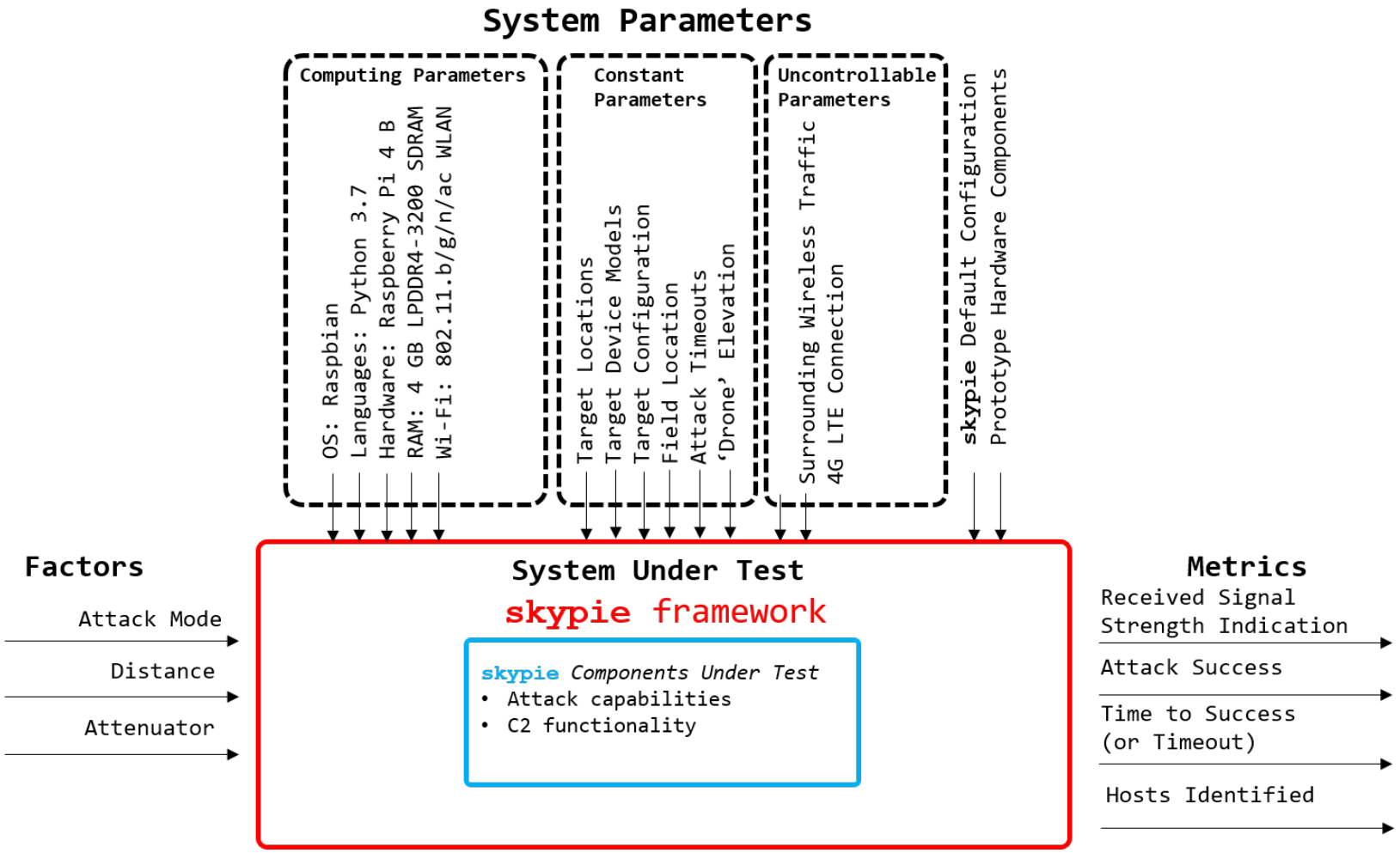

Figure 16: System Under Test and Components Under Test

Table 6: Experiment Factors

Factor	Levels	Description
Attack Mode	[WPA handshake capture, nmap scan, ping burst]	Different automated tasks performed on the target Wi-Fi network
Distance	[200 meter increments] from 200-2200 meters	Open field space where tasks are completed in different intervals
Attenuator	[none, 15 dB]	Added attenuation to simulate additional distance

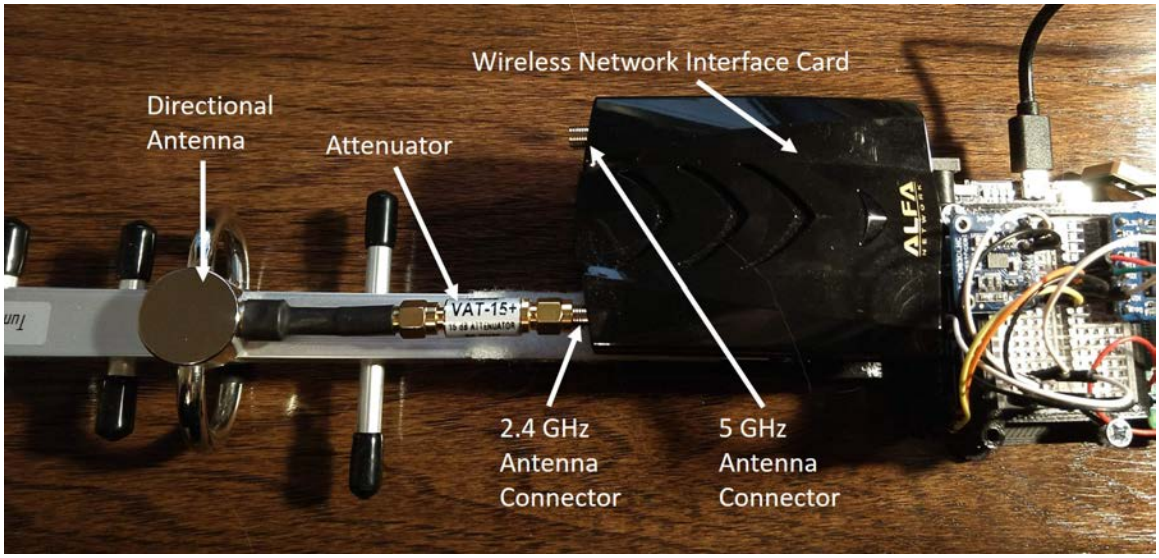

Figure 17: 15 dB attenuator attached between sykpie v2's directional antenna and WNIC

4.4 Metrics

Metric are the output or response variables of the experiment. Their expected ranges are displayed in Table 7 and are discussed in further detail below.

1. **Received Signal Strength Indication (RSSI)**. This is a measurement of the amount of energy that skypie's antenna receives. It is measured in decibel-milliwatt (dBm), which is an electrical power unit in dB relative to one milliwatt (mW). The equation that defines this logarithmic relationship is

$$mW = 10^{\frac{dBm}{10}} \tag{2}$$

   Because dBm is exponentially related to mW, 0 dBm is equal to 1 mW. Decreases in 3 dBm represents the halving of power (i.e., -3 dBm equals 0.5 mW).

2. **Attack Success**. This is a binary measurement of whether the WPA handshake or nmap scan is successful at a given distance.

3. **Time To Success (TTS)**. For the WPA handshake attack, this is the time in seconds it takes for the attack to capture one of the required EAPOL packet pairs (message 1 and 2 or 2 and 3). For the nmap scan, this is the time it takes for the nmap scan to complete. If the attacks fail to complete before the given timeout, they are assigned the time of their respective timeouts (30 seconds for WPA handshake attack and 120 seconds for the nmap scan).

4. **Hosts Identified**. On top of tracking the time nmap scans take, the efficiency of the scans can be measured by the percentage of hosts identified on the target network. This is accomplished by recording the number of host identified through each scan divided by the six known hosts on the target network.

Table 7: Experiment Metrics

Metric	Units	Expected Range
Received Signal Strength Indication (RSSI)	dBm	$-85\ dBm \leq RSSI \leq -30\ dBm$
Attack Success	-	Attack Success = 0 Attack Success = 1
Time To Success (TTS) or Timeout	s	$0\ s \leq TTS \leq 120\ s$
Hosts Identified	-	$0\ ip \leq TTS \leq 5\ ip$

4.5 Constant Parameters

When conducting the experiments, there are several parameters held constant throughout. While varying these parameters may have an effect on the response variables, it is necessary to limit the scope of this research and hold them constant. Table 8 summarizes these parameters, and they are discussed in further detail below.

- **Target Network Orientation**. The target network is set up by placing the four HP Zbook 15 laptops on the corners of a plastic folding table in different orientations, and the Netgear AC1750 AP is placed in the center of the table with the three antennas placed in a standard configuration (see Figure 18). Power to the AP is obtained from a AC inverter plugged into a running car, while the laptops run solely on battery power.

- **Location**. The location is flat (ideal wireless conditions) and has 2200 m of open field available. Figure 19 shows the field used and the target network location which is adjacent to an airfield runway. Figure 20 shows 'drone-mounted' skypie v2 at the 1600 m distance.

- **Number of Targets**. There is a total of five devices that make up the target network. The first is the Wireless AP and the remaining four devices are identical model laptops imaged with Windows 10.

Table 8: Constant Parameters

Parameters	**Proposed Values**	**Controlled By**
Target Network Orientation	Pre-defined layout (see Figure 18)	Experiment Design
Location	Airfield (2200 meters)	Experiment Design
Number of Targets	5	Experiment Design
Target Models	Netgear AC1750 HP Zbook 15 (x4)	Experiment Design
Command and Control Devices	Moto x^4 Cellphone Surface Book 2 Laptop	Experiment Design
Wireless Power Configurations	Full Power 537mw Windows 10 Default Power Settings	Experiment Design
'Drone' Elevation	13 ft	Experiment Design
WPA Handshake Timeout	30 seconds	Device Configuration
Nmap Timeout	120 seconds	Device Configuration
Nmap Scan Parameters	-p 21,22,23,443,445 -T4 -v	Device Configuration
AP Scan Interval	15 seconds	Device Configuration

- **Target Models**. The laptops and AP models are chosen for their sufficient capabilities. The laptops are realistic up-to-date network devices running a Windows 10 version 1903, and the AP has firmware version V1.0.1.52_1.0.36. The AP is configured to use the 2.4 GHz 802.11n standard because it has the greatest data rate.

- **Command and Control Devices**. Two devices are used to control skypie v2 throughout the experiment. The first is a Moto x^4 cellphone that operates in wireless AP mode and facilitates a connection to the Internet with its 4G LTE connectivity. The second device is a Surface Book 2 laptop that is used to control skypie v2.

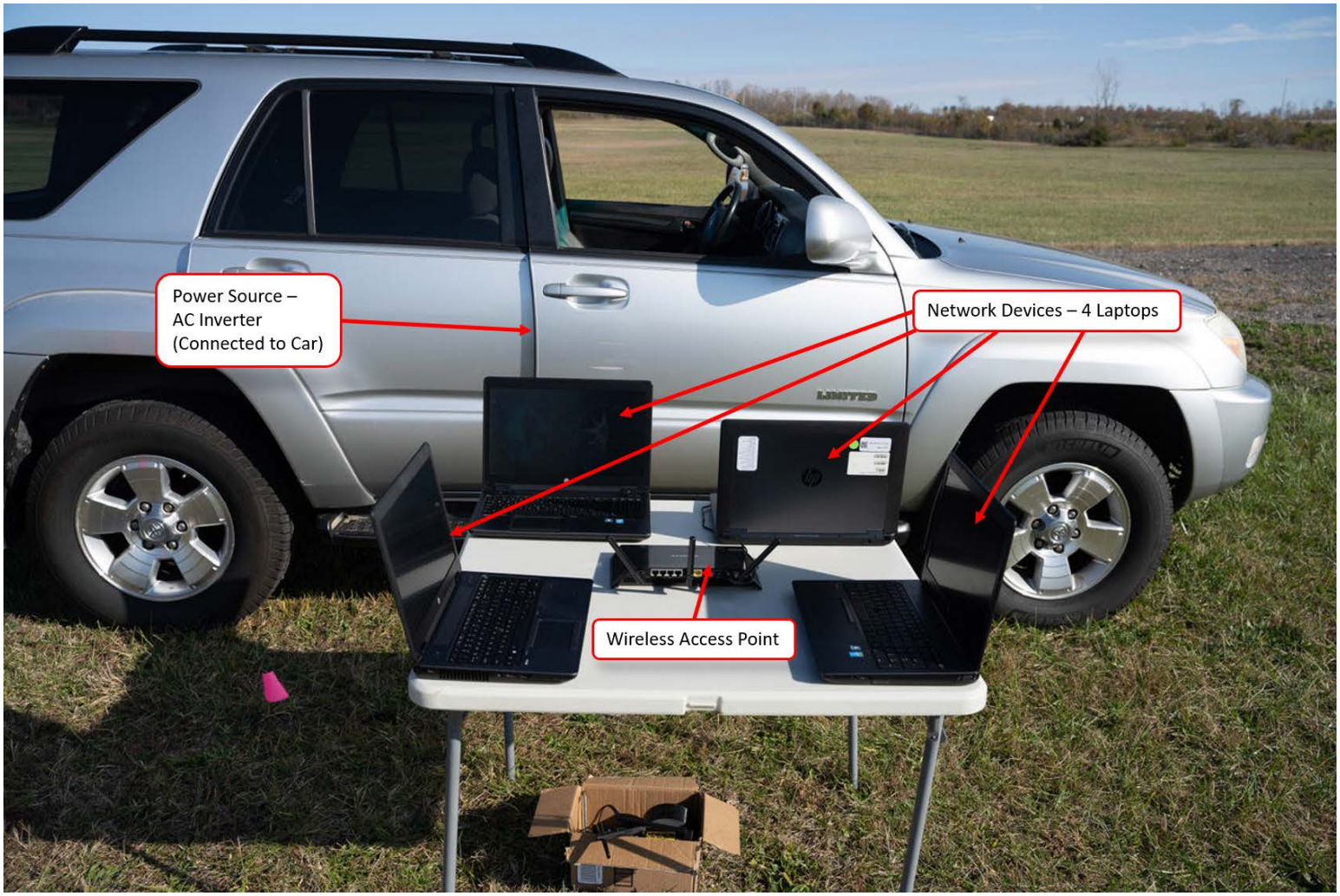

Figure 18: Target network orientation

Figure 19: Open and flat location used in the experiment (map data: Google)

- **Wireless Power Configurations**. The AP has the ability to adjust its power output, but the default configuration of 100 percent transmit power (537 mW) is chosen to reduce the experiment's complexity. Additionally, Windows 10 by default adjusts the power sent to the wireless adapter depending on the power plan selected. For the purpose of this experiment, the power settings are left as default on the four laptops. This affects the distance at which the laptop's wireless traffic can be captured, but simulates a realistic network configuration.

- **'Drone' Elevation**. To simulate flight, skypie v2 and the battery are mounted to a wooden platform shown in Figure 21. The wooden platform is screwed into a telescoping pole via iron pipe fittings (see Figure 22). The telescoping pole is extended to 13 ft, which adequately emulates a drone in flight.

- **WPA Handshake Timeout**. A 30 second timeout is chosen for the WPA handshake attack, because it disconnects and prevents connection to the target network for the duration of the attack. To prevent suspicion, these attacks should be as swift and unobtrusive as possible.

- **Nmap Timeout**. A 120 second timeout is selected for the nmap scan based on the number of devices on the network and scan parameters. Unlike the handshake capture attack, an nmap scan does not cause a DOS. This allows for a lengthy timeout period, and preliminary tests indicated that this is an adequate time for completion.

- **Nmap Scan Parameters**. A standard nmap scan that searches for a select set of often vulnerable ports with the speed 'T4' selected to ensure rapid completion.

- **AP Scan Interval**. In order to ensure that an updated RSSI is documented between each of the different experiment tasks, an interval of 15 seconds is

chosen. This ensures timely updates and that the RSSI reading is as accurate as possible on the given hardware.

Figure 20: Conducting wireless attacks with skypie v2 at 1600 Meters

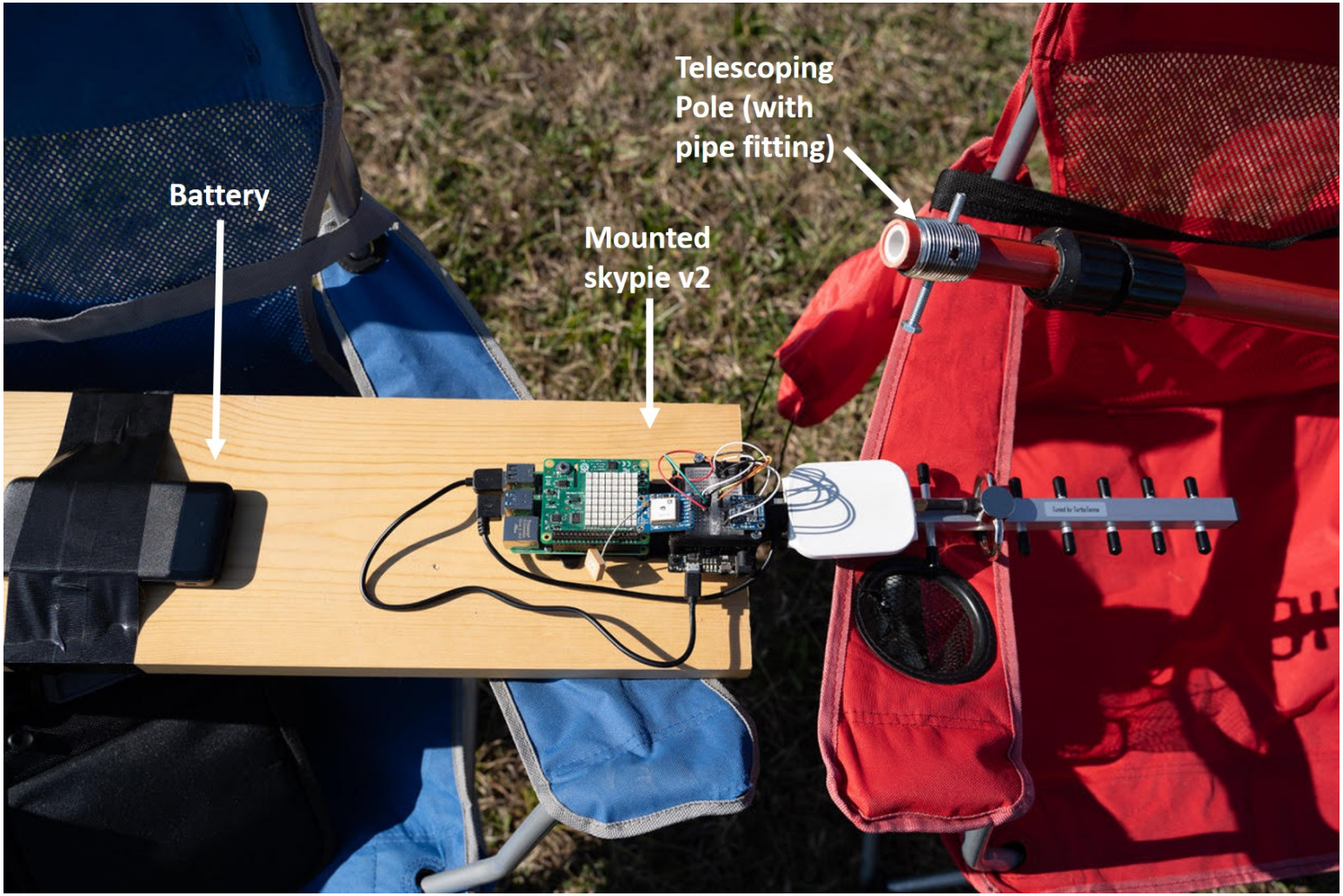

Figure 21: Skypie v2 and battery mounted to platform (left) Telescoping pole outfitted with iron pipe screw fitting (right)

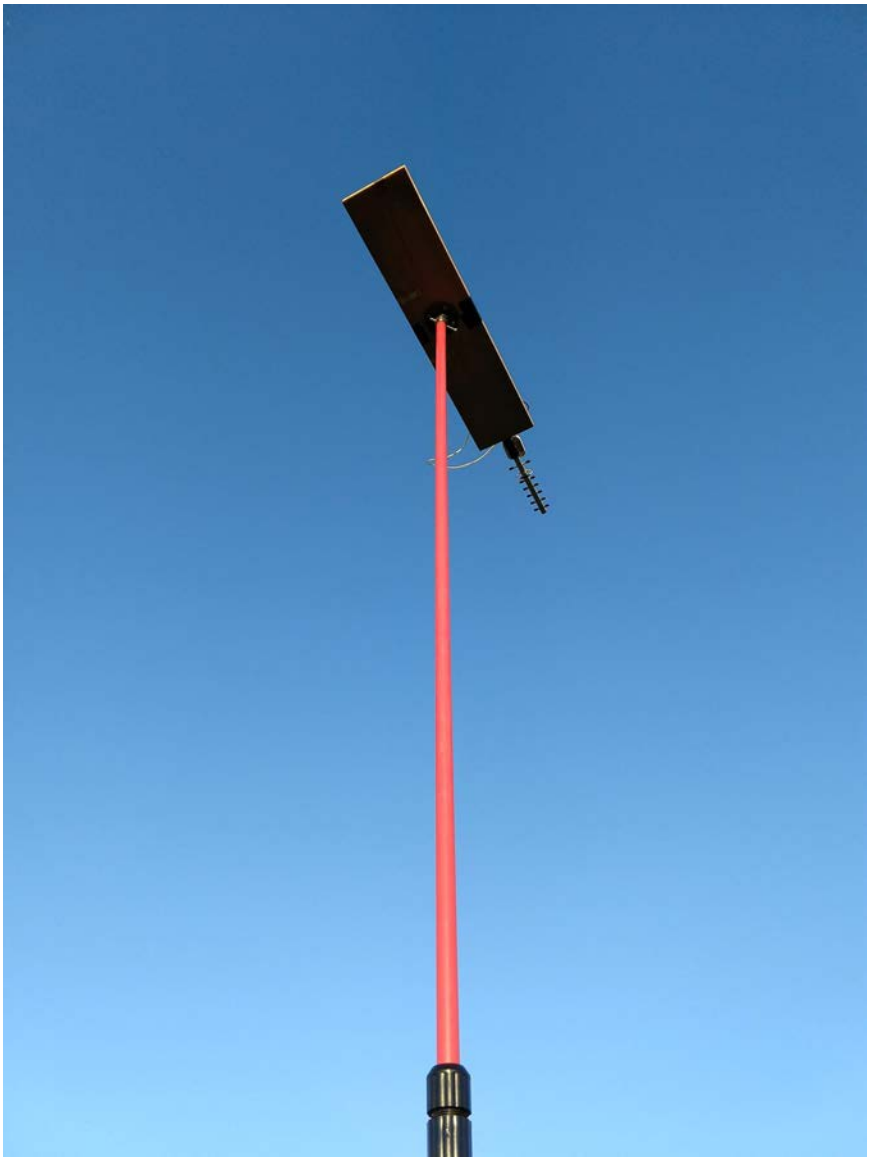

Figure 22: “Simulated drone flight”: skypie v2 platform mounted on telescoping pole via iron pipe fittings

4.6 Uncontrolled Variables

The 2.4 GHz band is highly utilized by public and private entities. Although the experiment location is relatively secluded, the high-gain directional antenna used is likely to pick up signals from non-experiment devices. These signals may cause interference, but the additional traffic simulates a more realistic noisy environment for which the skypie v2 is designed.

An additional uncontrolled variable is the use of a cellular connection to communicate with the FTP server. There are many factors that can affect a cellular connection including distance from cell towers, the weather, and the current congestion of the network. But like the uncontrollable nature of Wi-Fi networks, skypie v2 is designed to operate in these conditions.

4.7 Experiment Design

This section provides detailed steps for each experiment. The attenuator factor changes between each of the experiments. Experiment 1 is conducted with no attenuator and experiment 2 is configured with the 15 dB attenuator (Figure 17) in between the antenna and wireless interface. Figure 23 depicts the relative locations of the devices in the experiment. As no statistical comparisons are made between two devices, treatments are limited to 5 samples each.

4.7.1 Experiment 1: No Added Attenuation

1. Skypie v2 is turned on and set to automatically connect to the command and control cellphone's wireless AP via the Raspberry Pi 4 on board wireless interface card. A command and control laptop is also connected to this network and uses SSH to start a shell script by issuing the command

   `sudo ./skypie_cron.sh`

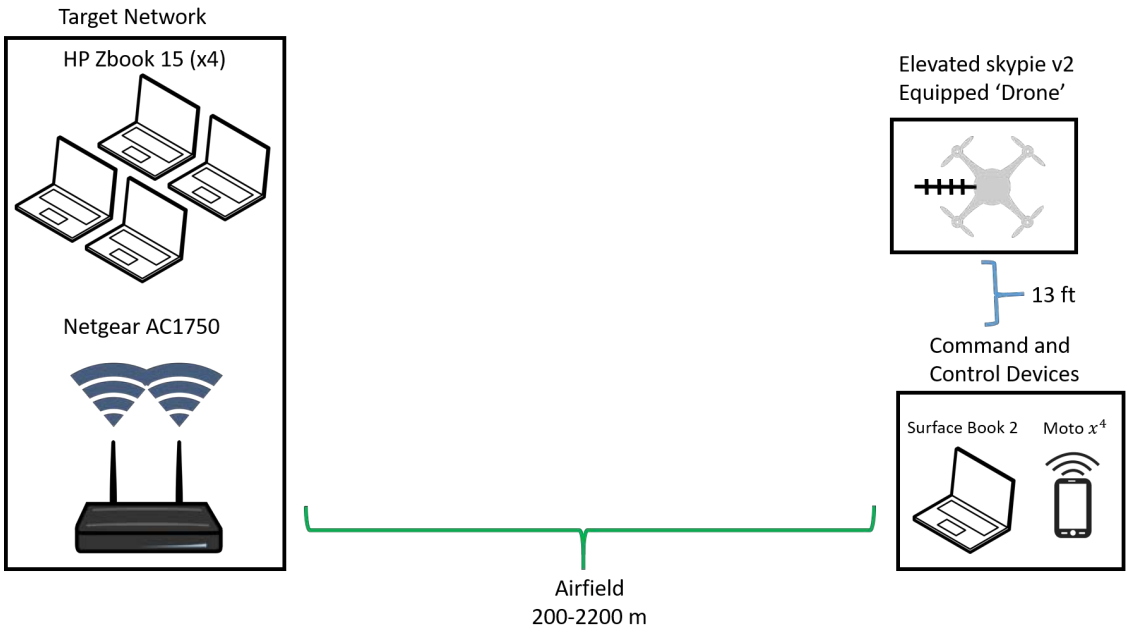

Figure 23: Experimental Design Block Diagram

This script (see Appendix D) checks if a skypie process is already running and if not starts main.py which subsequently initializes manager.py.

2. The mounted skypie v2 is moved west in 200 meter increments away from the target network, which are measured and marked with the pink flags shown in Figure 24. The telescoping pole is placed in the PVC sheath (see Figure 25) at each location to stabilize the platform in order to maximize consistency of data collection.

3. With the directional antenna angled at the target network, the laptop operator modifies the configuration file to change the attack mode of the skypie v2 and connect to the target network. Note that each of the attack thread records the RSSI of the AP 3.5.2 upon completion of it's operation. The attack modes and connection are conducted in the following order with a 15 second interval between each:

- **Capture mode**. This mode simultaneously runs the deauth thread (Section 3.5.2.3) and the handshake thread (Section 3.5.2.2) to capture a WPA handshake.
- Connect to the target network using the connect thread (Section 3.5.2.4).
- **Ping mode**. This mode runs the ping thread (Section 3.5.2.5) to capture network statistics.
- **Nmap mode**. This mode runs the nmap thread (Section 3.5.2.6) to identify the target network devices.
- Cycle through each attack mode and connection to the AP five times.

4. The equipment is moved to the next 200 meter distance.

5. Steps 3 and 4 are repeated until reaching the 2200 meter distance.

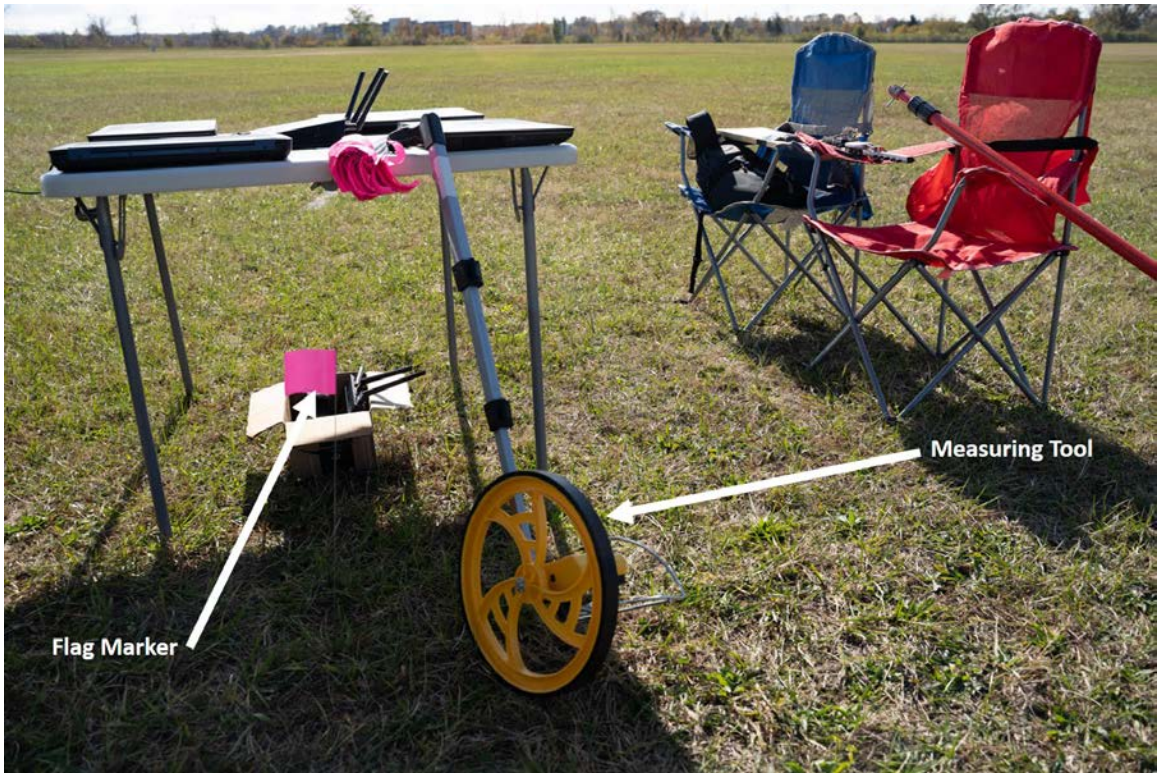

Figure 24: Flag markers and measuring tool

Figure 25: PVC sheath used to stabilize the mounted skypie simulating drone flight

4.7.2 Run 2: 15 dB Added Attenuation

This experiment is conducted to identify the limit of each task and map the packet loss as distance increased.

1. Using the feedback from experiment 1 that all the tasks are running at 2200 meters with no timeouts or failures, the mounted skypie v2 starts at the pre-

marked 2200 meter location.

2. Start up skypie v2 following Step 1 of the first experiment in Section 4.7.1.

3. After using the PVC sheath to stabilize and angle the directional antenna toward the target network, the command and control laptop operator modifies the configuration file to cycle through the attack modes as outlined in Step 3 of Section 4.7.1. For this experiment during the five loops at each location, a task is not repeated if it times out during the first iteration.

4. Move the equipment east (toward the target network) to the next 200 meter marked location.

5. Repeat Step 3 and 4 until either of the conditions are met: ping packet loss is less than 75 percent or the WPA handshake attack does not time out. In order to capture more precise ping packet loss and WPA handshake capture data, collections are conducted to the east and west of where these conditions occur.

4.7.2.1 Ping

Figure 26 depicts the movements of skypie v2 after ping packet loss falls below 75% to conduct more precise cycles of the Ping attack mode. When skypie v2 achieves $< 75\%$ ping packet loss, skypie v2 moves east 25 meters and follows Step 2 a total of two times. Then skypie v2 moves 75 meters (past the starting condition distance) to the west and follows Step 2. Next skypie v2 moves 25 meters and follows Step 2, repeating this until packet loss of $< 25\%$ is achieved.

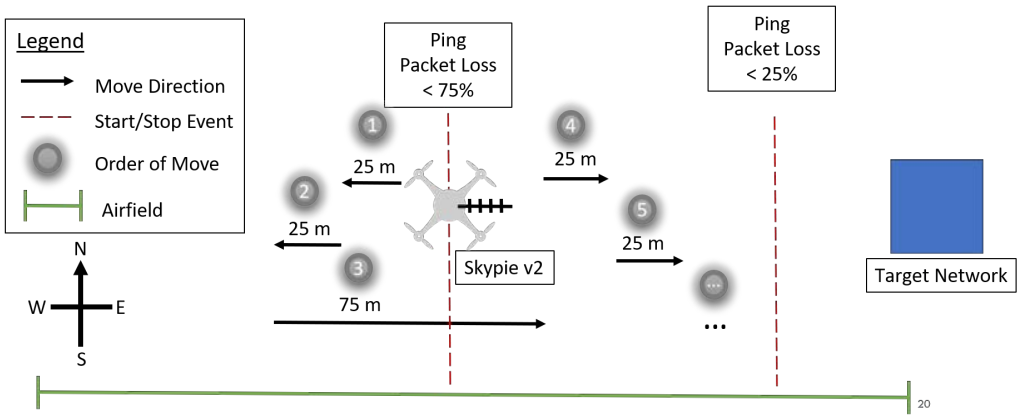

Figure 26: Ping mode's capture process (experiment 2)

4.7.2.2 WPA Handshake Capture

Figure 27 depicts the movements of skypie v2 after the first successful WPA handshake capture to conduct more precise cycles of the capture attack mode. When skypie v2 successful captures a WPA handshake of the target network, skypie v2 moves east 50 meters and follows Step 2 a total of two times. Then skypie v2 moves 150 meters (past the starting condition distance) to the west and follows Step 2. Next skypie v2 moves 50 meters and follows Step 2, repeating this until a WPA handshake is captured in $\leq$ the median capture time of experiment 1 (0.548 s).

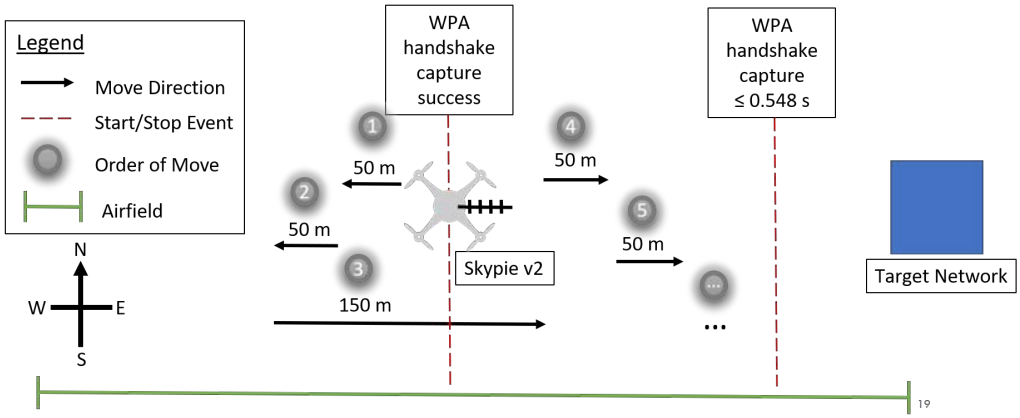

Figure 27: Capture mode's capture process (experiment 2)

4.8 Summary

This chapter outlines the designed system's parameters, outputs collected, and experiment design processes of this research at length. Additionally, the uncontrolled variables are covered. The metrics collected are used to evaluate and assess the effectiveness of the design improvements.

V. Results and Analysis

5.1 Overview

This chapter describes the results obtained from the two experiments outlined in Chapter 4. The results of each experiment is responsible for answering one or more of the research questions from Section 4.1.

Section 5.2 covers the experiment with no added attenuation. The results in this section are used to accept the hypothesis that CNAs are possible on the lightweight equipment of skypie v2 at distances greater than 800 meters.

Experiment 2 helps evaluate the effectiveness of each of the attacks/tasks and their limits. Section 5.3 discusses the limit of each of the attacks/tasks. Lastly, Section 5.4 addresses the final research question of how effective these attacks would fair against a realistic network setup. The added attenuation of buildings and the distance at which a skypie v2 would need to be to remain effective are discussed.

5.2 Experiment 1: No Added Attenuation

Skypie v2 is not equipped with an attenuator in this experiment. But as it shares the same location as the second experiment, its results are useful for comparison and analysis. The experiment location has a 2200 meter space limit, and attacks/tasks are only completed in 200 meter increments unlike experiment 2.

5.2.1 WPA Handshake Capture Results

At each of the 11 collections points, skypie v2 cycled through all of the attacks/-tasks 5 separate times. The first of which is the capture of the EAPOL packets which make up the WPA four-way handshake (as discussed in Section 2.5.4). This step is required in order to infiltrate a network as captured EAPOL packets can be used to

crack the password of a network. While other methods to gain access to a network exist, like KRACK (see Section 2.5.5), those are out of the scope of this research and a handshake capture attack satisfies this step.

With a timeout set for 30 seconds, the WPA handshake capture attack performed well at all locations. The time to capture over distance are depicted in a boxplot graph in Figure 28. At each collection point, the attack's time to completion had a median of under 5 seconds and a average median across all collection points of 0.548 seconds. Out of the 5 runs, only one timed out at 2000 meters, and is considered an outlier. The test locations 1200, 1400, and 1600 had time to capture results slightly longer than the other locations. This is likely caused by the skypie v2's directional antenna being slightly blown out of alignment with the target AP which simulates a realistic flight environment.

To determine if there is a difference in the time to capture a handshake across the 11 capture points (treatments), Analysis of Variance (ANOVA) is conducted over the results. Assuming the null hypothesis (H_0: there is no difference in mean time to capture between treatments), a F statistic of 0.613 is calculated. At significance level of 0.05, the critical value is 2.286751. Because the F statistic is less than the critical value, the null hypothesis, that the mean time to capture is the same for each location, is failed to be rejected. This indicates that across the 11 capture points there is no difference between their capture means, capture of WPA handshakes is possible farther than 2200 meters, and additional data is needed to determine the limit of this attack.

5.2.2 Ping Results

The next task after connecting to the target network, is sending a burst of ping packets and recording the results. To analyze the performance of sending and receiv-

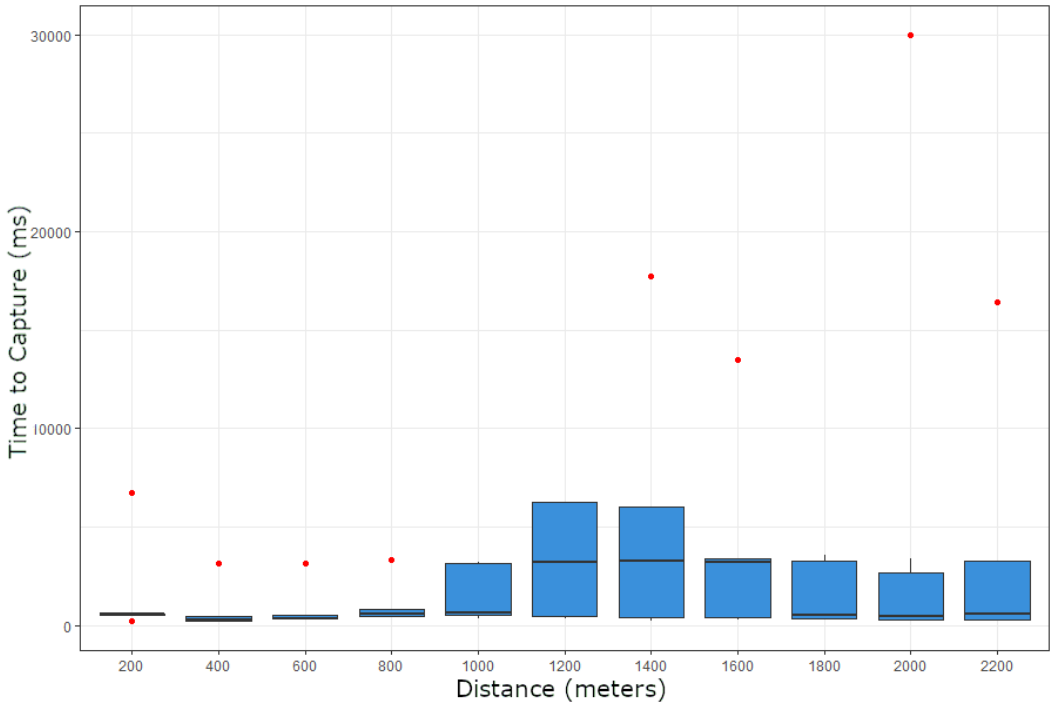

Figure 28: Boxplot of time to capture handshakes at each test location (experiment 1)

ing Wi-Fi traffic via skypie v2's hardware, the packet loss at each collection point is graphed with a boxplot as seen in Figure 29. Similar to the success of the WPA handshake capture, this task's packet loss is low. Each location had a median of 0 percent packet loss. Only the 500 and 1500 locations had trials with packet loss that were not outliers. This indicates that a connection at every location is very reliable and could support a variety of activities.

5.2.3 nmap Results

The final task is an nmap scan of the target network. This task's performance is heavily dependent on the reliability of the network. As nmap scans are conducted by sending packets to potential target devices, it is important to have a reliable connection so nmap can accurately assess the network. As the nmap scans are all conducted with the same parameters (same number of packets sent/received), it is

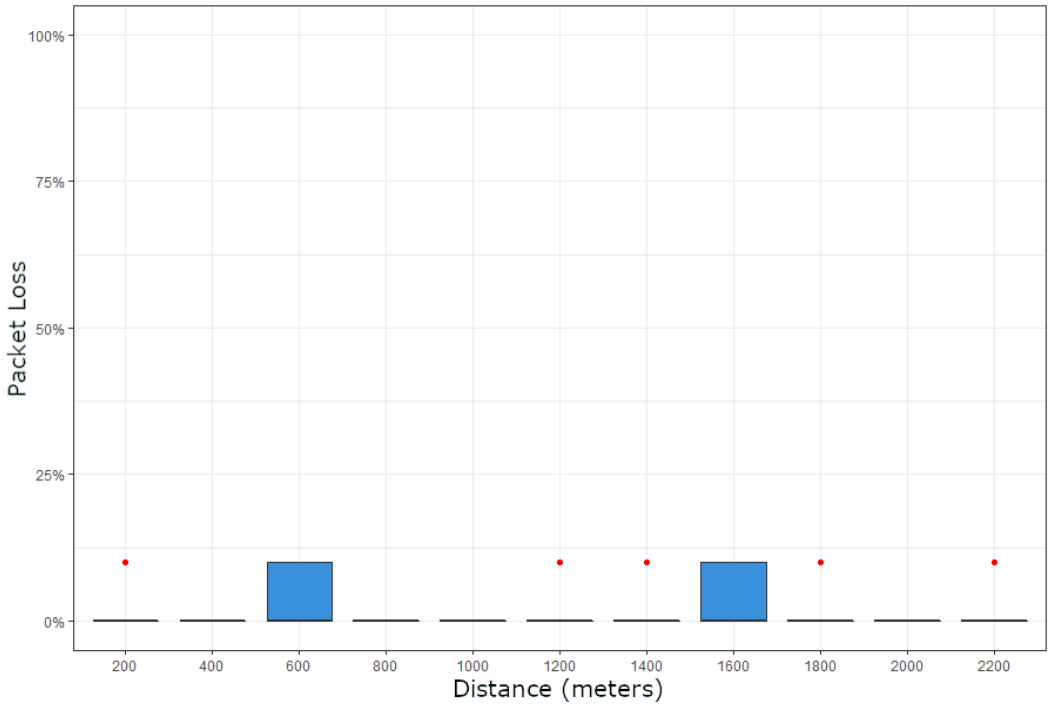

Figure 29: Boxplot of ping packet loss at each test location (experiment 1)

expected that they should all take roughly the same time to complete with negligible increases over longer distance, while the network connection is reliable. From the results of the ping burst test, the nmap attack is expected to perform well and identify all hosts on the network.

Figure 30 shows the results for the nmap attack of experiment 1. For all collection points, the hosts identified had a median of 100%. Only at the 600 location did one trial fail to identify any hosts and this is also likely due to the directional antenna being blown out of alignment with the AP. Additionally, an ANOVA test is conducted with a significant threshold of 0.05 over the completion times of the scans. The F statistic is calculated to be 0.709 and a critical value of 2.291282 is found. Therefore, the null hypothesis is not rejected and the mean scan times are the same at each location. The scans took an average of 20.6564 seconds to complete. These results are expected; a better evaluation of performance is conducted in the second experiment where the

reliability of the network is worse due to the addition of the attenuator.

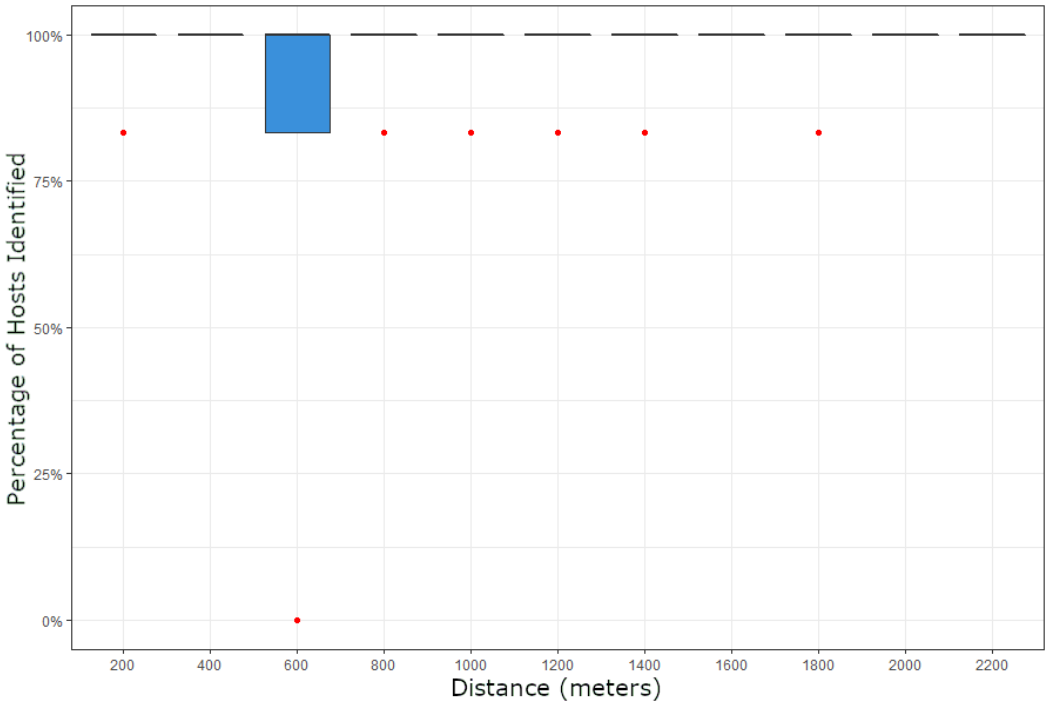

Figure 30: Boxplot of hosts identified by nmap at each test location (experiment 1)

5.2.4 Experiment Summary

Figure 31 shows a boxplot of the RSSI readings at each of the attack locations with weaker signals displayed with reds on a yellow to red gradient. A total of 165 individual readings are taken and boxplot outliers are marked as black dots. At the farthest point (2200 meters), the signals have a median strength of -74 dBm. This strength for Wi-Fi signals is considered weak, but a minimum strength for reliable packet transfer [57].

Using the transmit power of the wireless AP (P_{tx}), the dBi gain of the sending (G_{tx}) and receiving antennas (G_{rx}), and the Free Space Path Loss (FSPL) formula

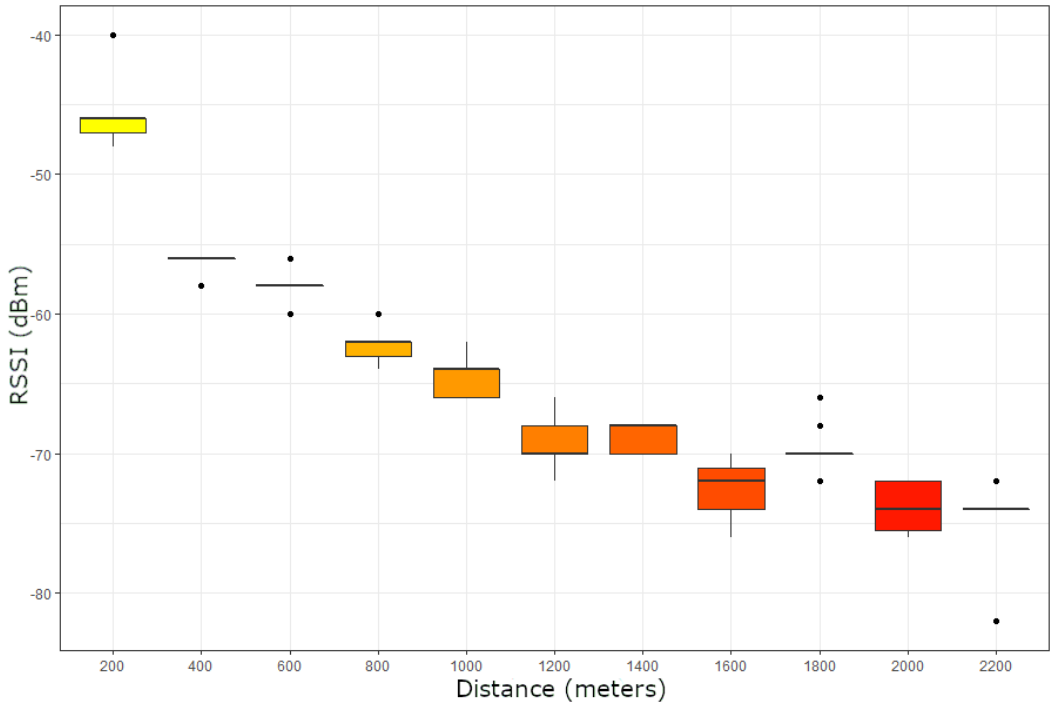

Figure 31: Boxplot of the RSSI at each test location (experiment 1)

the expected RSSI (P_{rx}) can be calculated with

$$FSPL = 10 * \log_{10}((\frac{4\pi d}{\lambda})^2) \tag{3}$$

$$P_{rx} = P_{tx} + G_{tx} + G_{rx} - FSPL \tag{4}$$

where λ is the speed of light divided by the 2.4 GHz frequency and d is the distance in meters between transmitter and receiver respectively.

Once the expected results are calculated, an analysis between the expected and measured signal strength readings are conducted. Table 9 depicts the summary of results at each test location, which displays the expected signal strength value, and the errors that show the difference between expected and measured strength (i.e, expected - measured). Several distances (200, 2000, and 2200) do not have sample sizes of 15 due to human error. The average median error between the expected signal

strength and measured is -13.81 dBm. As discussed in Section 4.6, there are many uncontrollable factors that can attenuate wireless signals. However, the size of this additional signal loss indicates that there is another unknown environmental factor affecting the experiment. Further insight into this error would require testing the accuracy of the skypie v2's WNIC's RSSI readings and the transmit power of skypie v2's AP which is out of the scope of this research.

While 2.4 GHz 802.11 devices' range is typically limited to 100 meters outdoors, point-to-point connections are possible over many kilometers utilizing dual directional antennas. These setups however require a line of sight between the station/client; because of this limitation they are rarely used. Most wireless networks are set up as point-to-multipoint networks using omni-directional antennas.

This experiment demonstrates the potential and utility that a single directional antenna provides when used on drone-mounted wireless attack platform to conduct CNAs. Even when 802.11 traffic has degraded over great distances, the lightweight equipment of skypie v2 can be used to great effect by cyber-attackers. In less than 30 seconds, a WPA handshake can be captured and an nmap scan searching for often vulnerable ports can be conducted on a target network from nearly 2200 meters away. In order to evaluate the prototype's capabilities further, attenuation is added and more measurements taken.

Table 9: RSSI Evaluation Against Expected (experiment 1)

Distance (meters)	**Samples**	**Measured Mean (dBm)**	**Measured Median (dBm)**	**Expected (dBm)**	**Mean Error**	**Median Error**	**Standard Deviation**	**Variance**
200	14	-45.1429	-46	-37.2729	-7.87	-8.7271	2.9051	8.4396
400	15	-56.1333	-56	-43.2935	-12.8398	-12.7065	0.5164	0.2667
600	15	-57.7333	-58	-46.8153	-10.9180	-11.1847	1.0328	1.0667
800	15	-62.4	-62	-49.3141	-13.0859	-12.6859	1.0556	1.1143
1000	15	-64.6667	-64	-51.2523	-13.4144	-12.7477	1.2344	1.5238
1200	15	-69.2	-70	-52.8359	-16.3641	-17.1641	1.8205	3.3143
1400	15	-68.9333	-68	-54.1748	-14.7585	-13.8252	1.0328	1.0667
1600	15	-72.5333	-72	-55.3347	-17.1986	-16.6653	2.0656	4.2667
1800	15	-69.7333	-70	-56.3577	-13.3756	-13.6423	1.4864	2.2095
2000	18	-73.8889	-74	-57.2729	-16.616	-16.7271	1.6047	2.5752
2200	13	-74.308	-74	-58.1007	-16.207	-15.8993	2.4285	5.8974
Averages:					**-13.8771**	**-13.8159**	**1.5621**	**2.8855**

5.3 Experiment 2: 15 dB Added Attenuation

As discussed in Section 4.3 in order to simulate further distances, an attenuator is added to the skypie v2. Then, the distance at which experiment 2's attacks would have occurred (without attenuation) can be calculated with the FSPL. Note that this experiment is conducted at the same location as the first. Figure 32 displays all the RSSI readings of the second experiment in a boxplot graph.

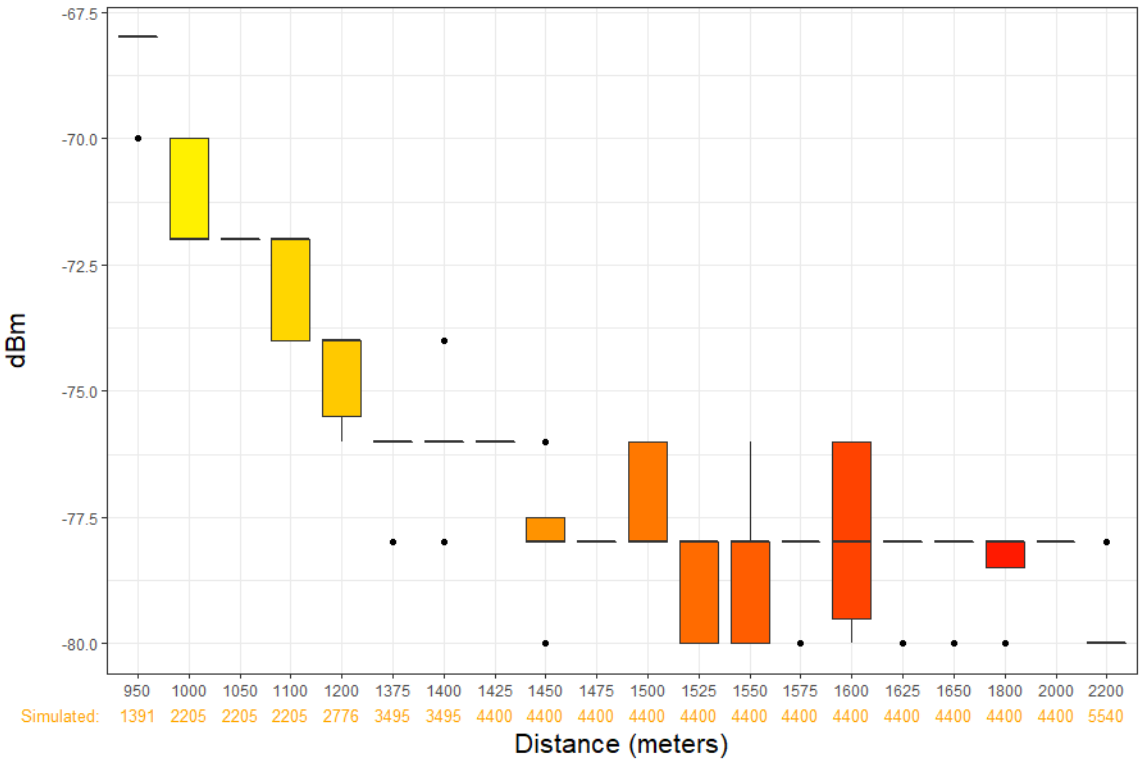

Figure 32: Boxplot of the dBm at each test location (experiment 2)

5.3.1 Attenuator Analysis

In order to verify that the attenuator functioned as expected before field use, it is tested with a signal analyzer. The 15 dB attenuator weakens an incoming 2.4 GHz signal by its advertised dB rating. Interestingly, comparisons of the RSSI measurements of the first and second experiments at seven matching locations (see Table 10)

shows the skypie v2 in the 15 dB attenuator configuration performed much closer to the expected (i.e., Column Measured Mean is closer to Column Expected). In Figure 33 the average measured RSSI values and expected values are graphed at each of the collection points from Table 10. Experiment 2's error is on average -6.5482 dBm which is weaker than the expected compared to the -15.4192 dBm that of the first experiment. These differences are illustrated in Figure 33.

The expected values for the 15 dB configuration were calculated using a similar equation to Equation 4 that subtracts the signal weakening ($Atten$) from the attenuator. The new formula is

$$P_{rx} = P_{tx} + G_{tx} + G_{rx} - FSPL - Atten \tag{5}$$

The summary of the RSSI results for experiment 2 are displayed in Table 11 and averages plotted in Figure 34. The average error over the 20 different distances (-6.9875) is only slightly higher than that of the seven matching locations (-6.5482) between experiment 1 and 2. Comparing experiment 1's average mean error with 165 trials to experiment 2's with 147 trails, experiment 2's mean error improves by 50%. The difference in error could have been affected by the experiments being conducted on different days with different weather conditions, but because of size of the difference in error, it is likely there is another unknown factor at work. While the difference in error between the two configurations may lead to better understanding of the factors affecting the error, additional investigation is outside of the scope of this research.

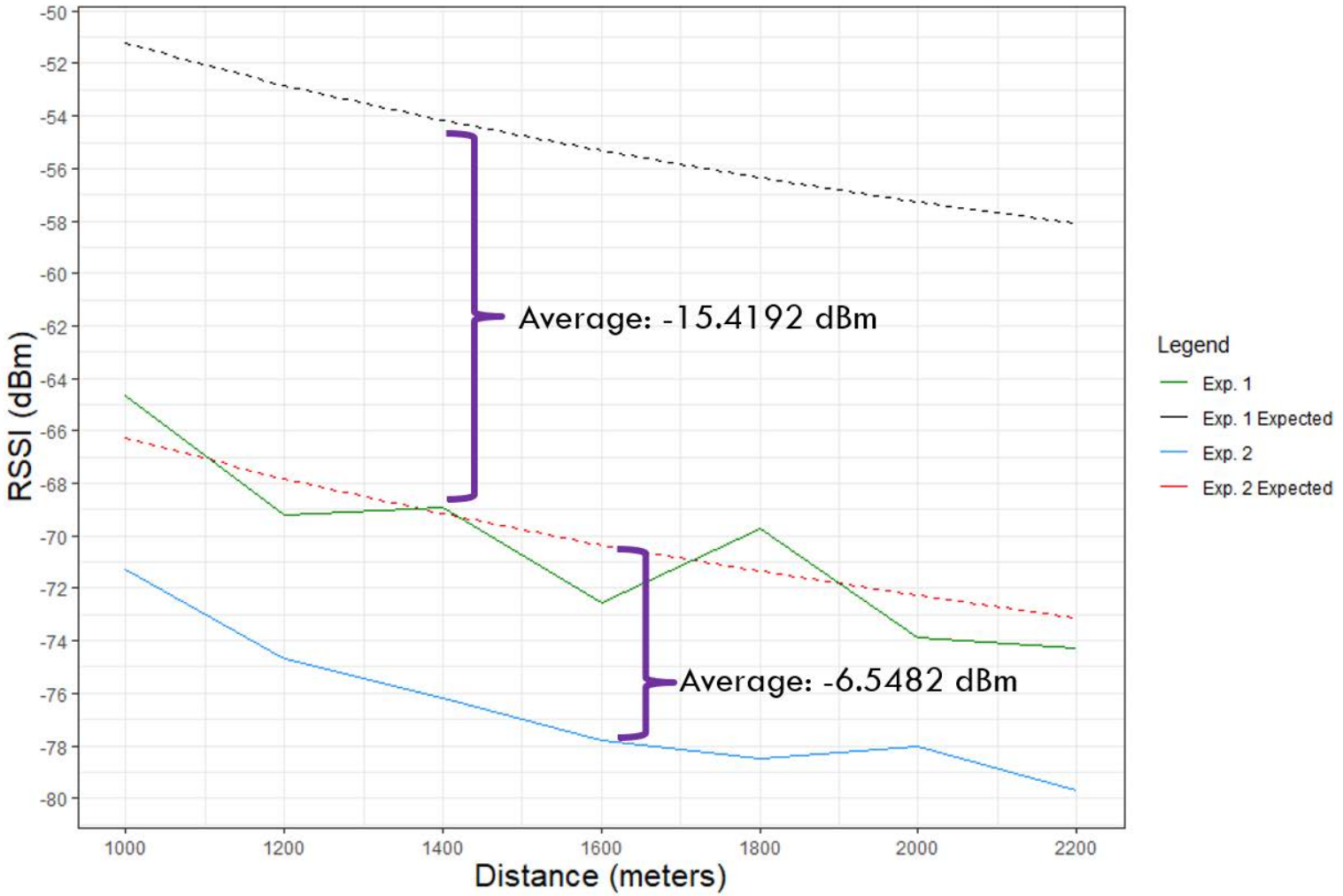

Figure 33: Average measured RSSI values (solid lines) between the two experiments and their expected values (dashed lines)

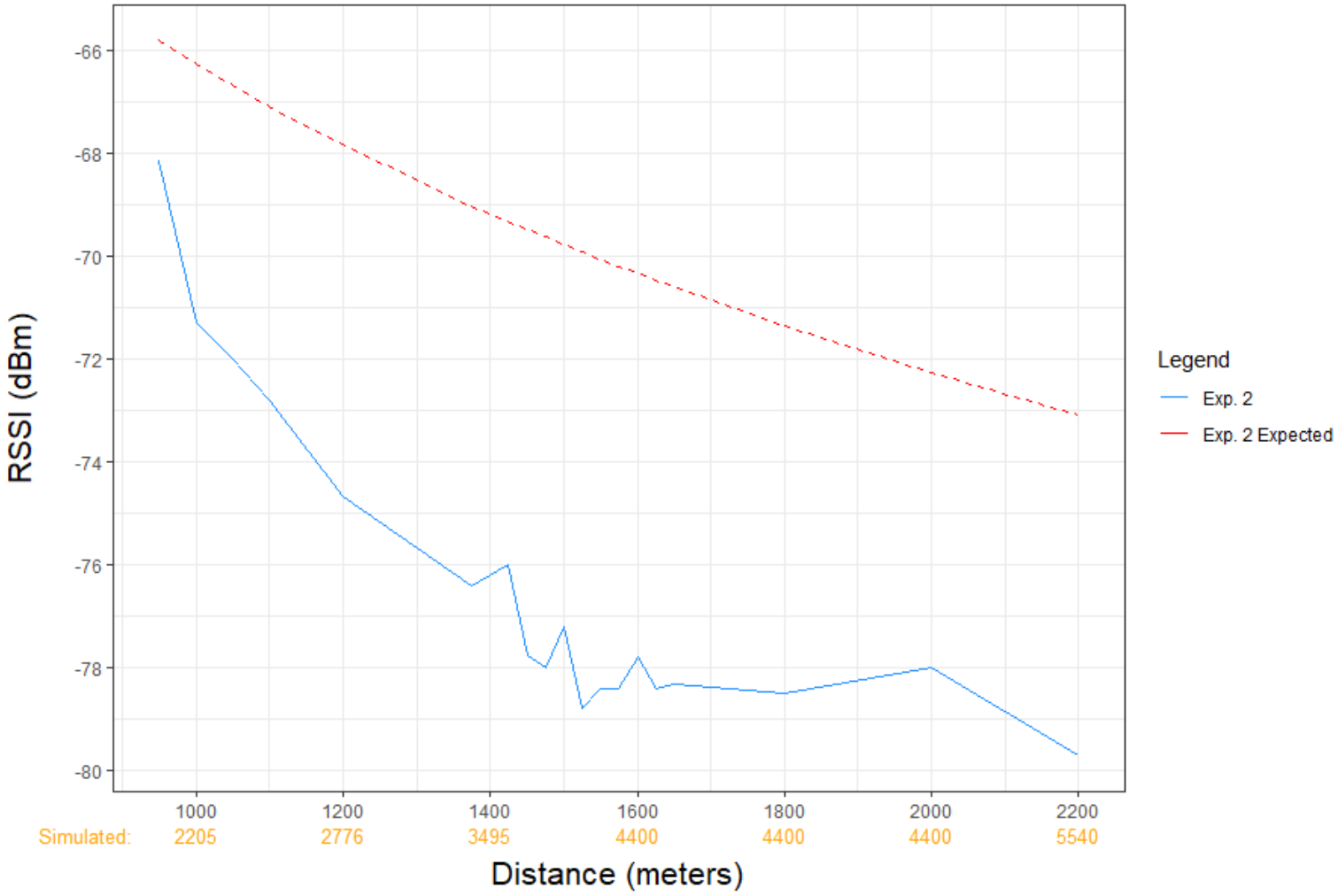

Figure 34: Average measured RSSI values (solid line) of experiment 2 and its expected values (dashed lines)

Table 10: Comparison of RSSI Values Between the Two Experiments.

Exper -iment	**Distance (meters)**	**Samples**	**Measured Mean (dBm)**	**Measured Median (dBm)**	**Expected (dBm)**	**Mean Error**	**Median Error**	**Standard Deviation**	**Variance**
1	1000	15	-64.666	-64	-51.2523	-13.4144	-12.7477	1.2344	1.5238
1	1200	15	-69.2	-70	-52.8359	-16.3641	-17.1641	1.8205	3.3143
1	1400	15	-68.9333	-68	-54.1748	-14.7585	-13.8252	1.0328	1.0667
1	1600	15	-72.5333	-72	-55.3347	-17.1986	-16.6653	2.0656	4.26667
1	1800	15	-69.7333	-70	-56.3577	-13.3756	-13.6423	1.4864	2.2095
1	2000	18	-73.8889	-74	-57.2729	-16.616	-16.7271	1.6047	2.5752
1	2200	13	-74.3077	-74	-58.1007	-16.207	-15.8993	2.4285	5.8974
2	1000	14	-71.2857	-72	-66.2523	-5.0334	-5.7477	0.9945	0.9890
2	1200	6	-74.6667	-74	-67.8359	-6.8308	-6.1641	1.0328	1.0667
2	1400	10	-76.2	-76	-69.1748	-7.0252	-6.8252	1.1353	1.2889
2	1600	10	-77.8	-78	-70.3347	-7.4653	-7.6653	1.7512	3.0667
2	1800	8	-78.5	-78	-71.3577	-7.1423	-6.6423	0.9258	0.8571
2	2000	6	-78	-78	-72.2729	-5.7271	-5.7271	0	0
2	2200	7	-79.7143	-80	-73.1007	-6.6136	-6.8993	0.7559	0.5714
					Experiment 1 Averages:	-15.4192	-15.2387	1.6676	2.9791
					Experiment 2 Averages:	-6.5482	-6.5244	0.9422	1.12

Table 11: RSSI Evaluation Against Expected (experiment 2)

Distance (meters)	**Samples**	**Measured Mean (dBm)**	**Measured Median (dBm)**	**Expected (dBm)**	**Mean Error**	**Median Error**	**Standard Deviation**	**Variance**
950	15	-68.1333	-68	-65.8067	-2.3266	-2.1933	0.5164	0.2667
1000	14	-71.2857	-72	-66.2523	-5.0334	-5.7477	0.9945	0.9890
1050	7	-72	-72	-66.6761	-5.3239	-5.3239	0	0
1100	15	-72.8	-72	-67.0801	-5.7199	-4.9199	1.0328	1.0667
1200	6	-74.6667	-74	-67.8359	-6.8308	-6.1641	1.0328	1.0667
1375	5	-76.4	-76	-69.0183	-7.3817	-6.9817	0.8944	0.8
1400	10	-76.2	-76	-69.1748	-7.0252	-6.8252	1.1353	1.2889
1425	5	-76	-76	-69.3286	-6.6714	-6.6714	-76	0
1450	8	-77.75	-78	-69.4796	-8.2704	-8.5204	1.2817	1.6429
1475	5	-78	-78	-69.6281	-8.3719	-8.3719	0	0
1500	5	-77.2	-78	-69.7741	-7.4259	-8.2259	1.0954	1.2
1525	5	-78.8	-78	-69.9177	-8.8823	-8.0823	1.0954	1.2
1550	5	-78.4	-78	-70.0589	-8.3411	-7.9411	1.6733	2.8
1575	5	-78.4	-78	-70.1979	-8.2021	-7.8021	0.8944	0.8
1600	10	-77.8	-78	-70.3347	-7.4653	-7.6653	1.7512	3.0667
1625	5	-78.4	-78	-70.4693	-7.9307	-7.5307	0.8944	0.8
1650	6	-78.3333	-78	-70.6019	-7.7314	-7.3981	0.8165	0.6667
1800	8	-78.5	-78	-71.3577	-7.1423	-6.6423	0.9258	0.8571
2000	6	-78	-78	-72.2729	-5.7271	-5.7271	0	0
2200	7	-79.7143	-80	-73.1007	-6.6136	-6.8993	0.7559	0.5714
Averages:					**-6.9875**	**-6.8817**	**-3.0121**	**0.9008**

5.3.2 WPA Handshake Capture Results

As skypie v2 is moved towards the target AP (i.e., start at 2200 m and move towards 950 m), the WPA Handshake Capture attack is tested at 8 different locations as shown in shown in Figure 35. Only after reaching 1100 meters did the attack not timeout on its first attempt. From 1100 m to 950 m, the time to success dropped dramatically, which indicates the limit of the WPA handshake capture is near 1200 meters in this configuration.

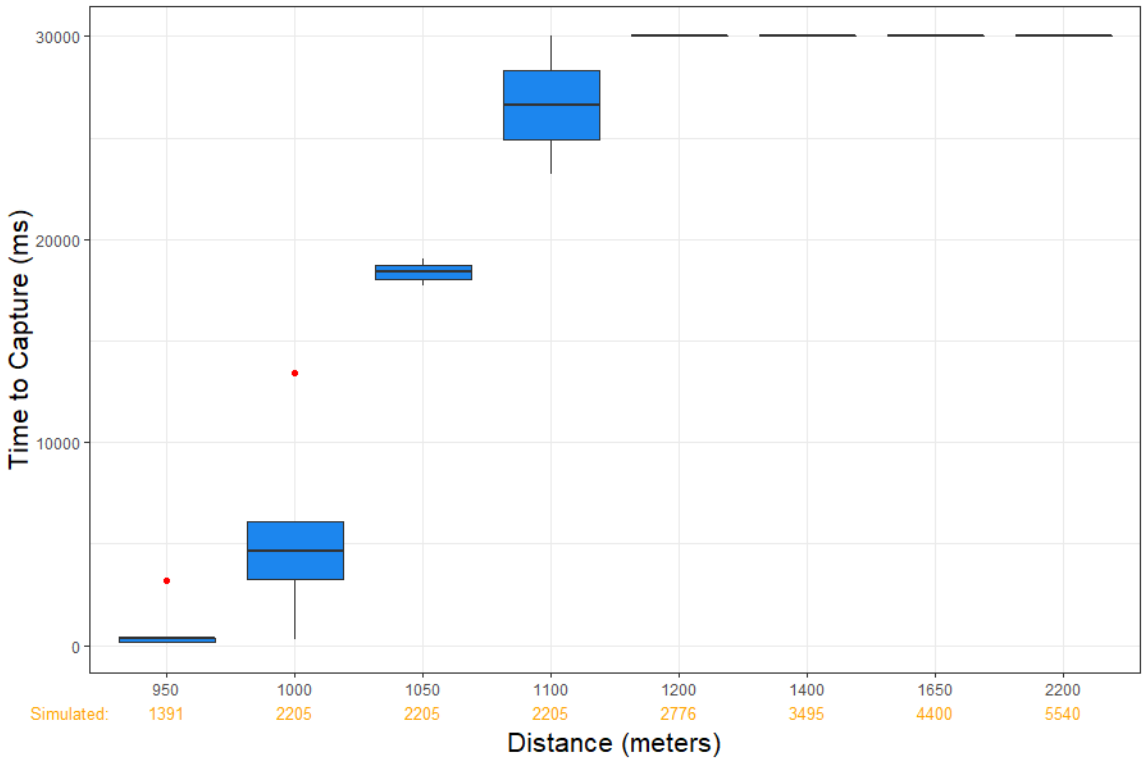

Figure 35: Boxplot of time to capture handshake at each test location (experiment 2)

This limit is not due to skypie v2's inability to capture packets from the AP, as inspecting the Wireshark capture files of the 1200 meter location indicates that messages 1 and 3 (from the AP) of the four-way handshake are still successfully captured. Additionally, the advertised maximum transmit power of the HP Zbook

15 is 22.3872 mW which is lower than the AP. Due to the target network laptops transmitting at a lower power, the traffic attenuates below the sensitivity of skypie v2 over a smaller distance. This transmit power difference is expected and simulates a realistic network setup. One additional variable that can affect transmit power is Windows 10's power configurations that automatically adjusts the transmit power to conserve power as conditions are met. Some of these conditions include whether a laptop is plugged in (experiment 1 and 2 ran on battery power), which power plan is selected, and what battery percentage the laptop is at. These Windows 10 configurations were left as the default.

With the RSSI analysis from Section 5.3.1 and knowing the farthest location where the handshake capture attack did not timeout, it is possible to calculate the approximate max distance an unattenuated attack would have success. By manipulating (3) and (4), and adding an error variable ($Error$) to account for the unexpected signal weakening (discussed in Section 5.2.4), the formulas for calculating distance are

$$FSPL = -P_{rx} + G_{tx} + G_{rx} + P_{tx} + Error \tag{6}$$

$$Distance = \frac{\lambda * \sqrt{10^{\frac{FSPL}{10}}}}{4 * \pi} \tag{7}$$

See Appendix E for the derivation of distance from (3).

Using mean error (-13.8771) and the mean RSSI reading (-72.8 dBm) at the 1100 meter (i.e., the farthest distance where WPA handshake capture was possible) location found in Table 9 and 11 respectively, the max distance for a handshake capture attack is calculated as 2418.42 meters. Using the same mean error and the mean RSSI reading for the 1200 meter location (-74.6667 dBm), it is calculated that the WPA handshake capture becomes untenable between 2418.42 and 2998.24 meters.

5.3.3 Ping Results

The packet loss on this experiment has significantly higher variance compared to the first experiment. As skypie v2 moved farther from the AP, packet loss trended upward in a linear fashion (Figure 36). Similar to the results of the first experiment, the median packet loss stayed within 0 to 10 percent from 950 to 1100 meters. Only after the signal strength fell to a median of -76 dBm at 1375 meters, did the median packet loss exceed 10 percent. This is the point at which the linear trend for packet loss began.

Packet loss is acceptable in some circumstances and is often handled by Reliable Data Transfer (RDT) protocols, but higher packet loss significantly affects the quality of service between devices. This is especially true for real-time services. For example, Voice over IP (VoIP)'s quality is significantly affected by "5% and 10%" packet loss [58]. Therefore, in order to keep skypie v2 operational viability flexible, maintaining distance within this packet loss range is advised.

5.3.4 nmap Results

Results in Section 5.2.3, suggest nmap scans should successfully identify all hosts where the ping results have low packet loss (950-1100 meters). As expected, all hosts are identified in that range with the exception of 1000 meters which identified a median of 90 percent of hosts. The percentage of host identified from the nmap attack are shown in Figure 37. Even at the 1200 meter mark, where packet loss is not recorded, nmap identified all hosts excluding one outlier. This indicates that packet loss is kept reasonably low at this location.

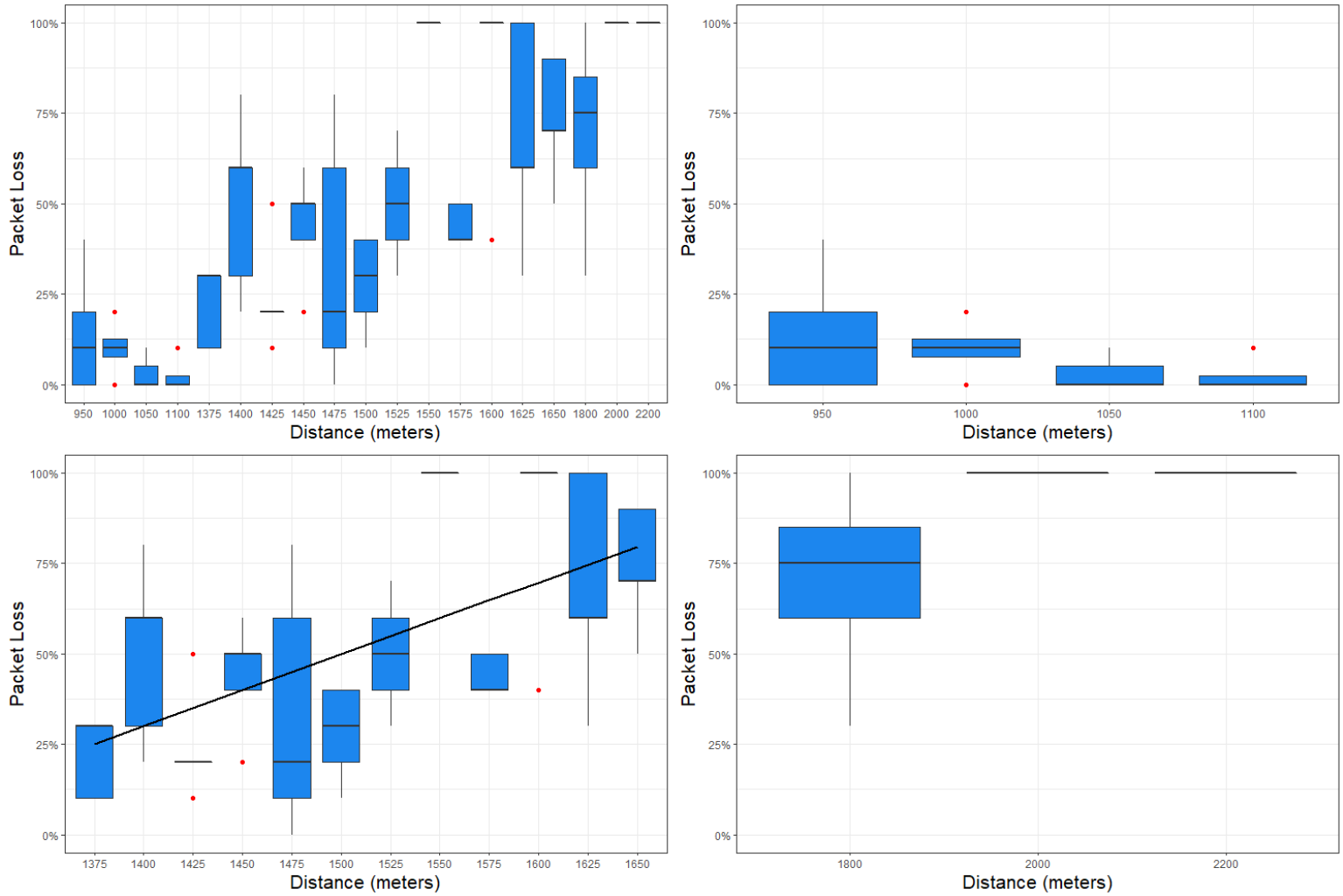

Figure 36: Boxplot of ping packet loss at each test location with a trend line (experiment 2)

The nmap scans took a mean 26.9562 seconds to complete which is slightly higher than the 20.6564 mean completion time of the first experiment. This slight increase in time is expected as nmap automatically slows itself down as it detects dropped packets or higher latency.

Only after the mean signal strength reached -76.2 dBm and the packet loss varied between 20-80% at 1400 meters, did the nmap results also begin to vary widely. As calculated for the 1200 meter location (i.e., the farthest distance where nmap scans identified all hosts) in Section 5.3.2, it is expected for nmap attacks in similar ideal conditions to be viable 2998.24 meters from a target AP. Using the mean RSSI of the 1400 meter location (-76.2 dBm) and in (6) and (7), it is calculated that packet loss would become too high to produce consistent nmap scan results between 2998.24 meters and 3577.10 meters.

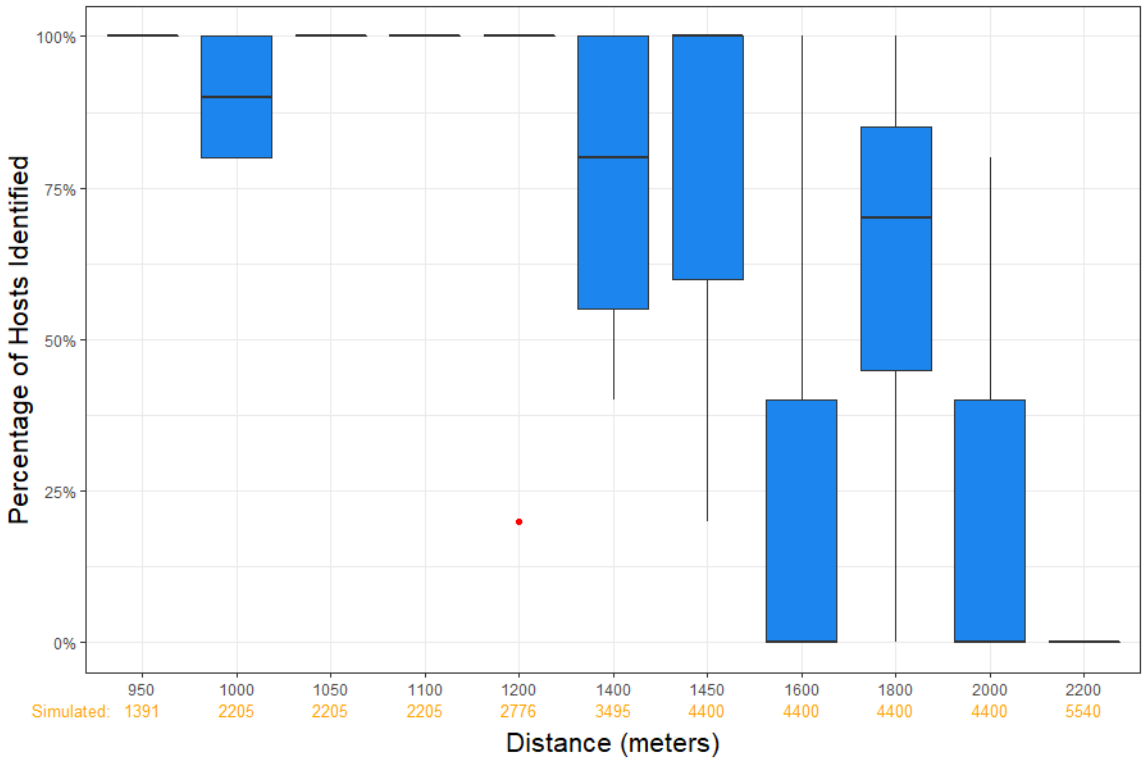

Figure 37: Boxplot of hosts identified by nmap at each test location (experiment 2)

5.3.5 Experiment Summary

Experiment 2 expands on the findings of experiment 1 and demonstrates that all of the CNAs can be conducted at significantly farther distances observed in experiment 1. The WPA handshake capture attack is expected to be effective an additional 200 meters, equating to over 2400 meters away from the target, or at an RSSI reading of -72 dBm. Similarly, the nmap scan is expected to be possible nearly 3000 meters from a target network or at an RSSI reading of -74 dBm. Although packet loss is not measured between the 1100 and 1375, meter locations based on the results of the nmap scan attack at 1200 meters (discussed in Section 5.3.4) indicates low packet loss is maintained at -74 dBm.

Since RSSI is a measurement of power that factors in signal degradation regardless of cause (i.e., distance or passing through materials), RSSI is considered the best measure of effectiveness when discussing skypie v2. Knowing the required RSSI for a particular attack can aid network defenders taking steps to limit their buildings' wireless emissions. It also demonstrates the considerable threat that the DWAP poses.

5.4 Realistic Network

Knowing these attacks are possible at long distances and their limits, their potential against a realistic network setup can be discussed. The experiment in this research does not include any added attenuation from buildings, which is where a large majority of wireless networks are located. Since no one network setup is likely to be the same as another, different building construction materials and variety of different possible configurations are examined.

Table 12 contains the attenuation of common building materials [59] and their calculated effect on the distance at which skypie v2's four-way handshake attack and

nmap scan could be conducted. These values are calculated with (6) and (7) and using the mean RSSI error reading from experiment 1 (-13.8771 dBm). The total added attenuation from building materials are added to the error variable. For example, if a target network is in a building constructed with drywall, fiberglass, plywood, tar paper, and stucco the traffic skypie v2's is -17.3902 dB weaker, which results in the following WPA handshake capture and nmap distance:

$$\begin{aligned} FSPL &= -72dBm + 3.5dB + 18dB + (10 * \log_{10}(537mW)) \\ &\quad + (-13.8771dBm - 17.3902dB) = 89.5324dB \\ handshake\ distance &= \frac{(0.1249m)\sqrt{10^{\frac{89.5324}{10}}}}{4\pi} = 297.86m \end{aligned}$$

$$\begin{aligned} FSPL &= -74dBm + 3.5dB + 18dB + (10 * \log_{10}(537mW)) \\ &\quad + (-13.8771dBm - 17.3902dB) = 91.5324dB \\ nmap\ distance &= \frac{(0.1249m)\sqrt{10^{\frac{91.5324}{10}}}}{4\pi} = 374.99m \end{aligned}$$

In the table each individual material is listed and then some of the worst case scenarios are derived. The worst case scenarios assume a similar interior make up (drywall, fiberglass insulation, fir lumber frame, plywood exterior walls, and tar paper weather proofing) and three different exterior walls (red brick, cinder block, and stucco). Since building frames have gaps and a signal may not have to pass through a fir lumber beam to escape the building, each scenario is calculated with and without its attenuation.

As shown in the table, attenuation can vary widely based on the material the the signal has to pass through. If a wireless AP is placed haphazardly next to a window, the building's signal emissions are bound to be high. The added -0.4998 dB

of attenuation only limits each attack by a few hundred meters each:

$$FSPL = -72dBm + 3.5dB + 18dB + (10 * \log_{10}(537mW)) + (-13.8771dBm - 0.4998dB) = 106.4228dB$$

$$handshake\ distance = \frac{(0.1249m)\sqrt{10^{\frac{106.4228}{10}}}}{4\pi} = 2082.29m$$

$$FSPL = -74dBm + 3.5dB + 18dB5 + (10 * \log_{10}(537mW)) + (-13.8771dBm - 0.4998dB) = 108.4228dB$$

$$nmap\ distance = \frac{(0.1249m)\sqrt{10^{\frac{108.4228}{10}}}}{4\pi} = 2621.45m$$

If it is assumed that a target structure has a similar interior make up of drywall, fiberglass insulation, fir lumber frame, plywood exterior walls, and tar paper weather proofing some of the worst case scenarios can be estimated. If the building is covered with a red brick exterior and has no windows, -6.9621 to -9.751 dB of added attenuation can be expected. For cinder block, the building has between -9.2413 and -12.0302 dB of added attenuation, each of which still puts all of the attacks only hundreds of meters away from the target. The strongest attenuation occurs when a stucco finish is added to an exterior wall. When applied to a plywood wall, diamond mesh is first affixed to the wall and the concrete stucco mixture is applied over top. This alone adds -14.863 dB of attenuation and between -17.3902 and -20.1791 total added dB can be expected. However, this still allows an attack to conduct a handshake capture 216 meters away, and nmap scans 272 meters away, which is still outside the audible range of 100 meters.

Table 12: Common Building Material's Attenuation [59] and the Effects to skypie v2's Attacks

Attenuation:											
-0.4998	-0.4937	-0.0241	-2.7889	-1.9138	-0.0956	-4.4349	-6.7141	-14.863			
Glass	**Drywall**	**Fiber-glass**	**Fir Lumber**	**Ply-wood**	**Tar Paper**	**Red Brick**	**Cinder Block**	**Stucco**	**Total Added Atten-uation**	**hanshake capture Distance (meters)**	**nmap Scan Distance (meters)**
x									-0.4998	2082.29	2621.45
	x								-0.4937	2083.75	2623.29
		x							-0.0241	2199.51	2769.02
			x						-2.7889	1599.87	2014.12
				x					-1.9138	1769.46	2227.62
					x				-0.0956	2181.48	2746.32
						x			-4.4349	1323.69	1666.42
							x		-6.7141	1018.18	1281.81
								x	-14.863	398.45	501.62
	x	x		x	x			x	-17.3902	297.86	374.99
	x	x	x	x	x			x	-20.1791	216.06	272.00
	x	x		x	x		x		-9.2413	761.14	958.22
	x	x	x	x	x		x		-12.0302	552.10	695.05
	x	x		x	x	x			-6.9621	989.52	1245.73
	x	x	x	x	x	x			-9.751	717.76	903.61

Because buildings and wireless network configurations will always be an uncontrollable variable, the max effective range of the skypie v2 is going to vary. Regardless, the results show that even under some heavy attenuation situations the skypie v2 attack range remains large. To defend against this sort of attack it is important for network administrators to consider mitigation and evaluate their building signal emissions.

Physical security such as walls and security personnel can be easily surpassed by a drone, as depicted in Figure 12, to give an attacker RPP. For example, if the target building is in the cinder block configuration discussed above, the attacker could direct the skypie v2 to get within 552 meters of the target (usurping physical security) to conduct a WPA handshake capture attack. Once captured and uploaded to an FTP server via a cellular connection, the DWAP could move 695 meters away from the target and await the attacker to crack the wireless password on a more powerful workstation. Located nearly 700 meters from the target and with a cracked password, the skypie v2 could then connect to the AP and conduct an nmap scan of the target network.

VI. Conclusions and Future Work

6.1 Overview

This chapter summarizes the research and the findings of the experiments conducted throughout. Section 6.2 reiterates the conclusions drawn from the experiments and analysis. Section 6.3 discusses potential countermeasures. Finally, Section 6.4 is dedicated to a roadmap of future work in the field of cyber-attack drones.

This research is successful in demonstrating that an inexpensive and lightweight drone-mounted wireless attack platform (skypie v2) can conduct CNAs against a network at distances greater than 800 meters. When the network's wireless traffic does not experience added building attenuation, attacks via the skypie v2 are highly effective up to 2200 meters. To better understand the attacks' limits and attempt to overcome test distance limitations, an additional experiment is run with an attenuator added to simulate additional distance. The experiment shows that WPA handshake capture is possible as far as 2418 meters away and nmap scans as far as 2998 meters away.

Also discussed is skypie v2's potential effectiveness against a realistic network setup. This is done by using the known attenuation values for common building materials and calculating the range of each attack after factoring in the attenuation. Through this process each of the following research goals were me:

- Can CNAs be accomplished at 800+ meters using lightweight equipment on a cyber-attack drone?
- If so, how long does each attack take?
- At what distance do they become infeasible?
- How effective would these attacks be against a realistic network setup?

6.2 Research Conclusions

It is found that CNAs are possible far beyond the hypothesized distance of 800 meter distance with the lightweight equipment of the skypie v2. Therefore this research proves the hypothesis. WPA handshake attacks can be conducted effectively (under 30 seconds) as far as 2400 meters from the target, and packet loss/nmap scans are reliable until the RSSI reading falls below -76 dBm. Under similar conditions, it is expected that additional network attacks and reconnaissance could be conducted as far as 2998 meters from the target.

After conducting the experiments a median error of -13.8771 dBm is calculated from the expected RSSI over the 2200-meter distance. The experiment is designed to minimize attenuation factors, but often uncontrollable factors affect wireless traffic. Discovery of the factor(s) causing the unexpected loss could enable corrections/upgrades that would significantly improve skypie v2's performance. Regardless of the error, after conducting an analysis of how known construction material would attenuate a 2.4 GHz Wi-Fi signal, it is found that even with heavy building attenuation the skypie v2 can be effective. If a target AP is located in a building that added 20 db of attenuation, it is calculated that a WPA2 handshake capture could be accomplished 216 meters from the AP and an nmap scan 272 meters. This is still outside the audible range of a drone, and if the target AP is located near windows it is expected CNAs can work multiple times those distances, because glass only attenuates a Wi-Fi signal by 0.5 dB.

When conducting WPA handshake capture attacks, the network devices (laptops) limit the range rather than the AP. This is because the network devices transmitted their traffic at a lower power than the AP. When conducting these CNAs, it should be assumed that the network devices transmit at different powers based on their hardware, power configuration settings, and whether the 802.11 TPC feature is being

used. The lower transmit power is only a limiting factor when capturing the WPA handshake and not the nmap scan. Since nmap scan traffic is facilitated by the AP, the range of the nmap scan is only limited by the attenuation between the AP and skypie v2.

The FSPL formula was not considered before the formation of the hypothesis and this impacted the educated guess on the effective distance. It was only after experiment 1 that the FSPL equation was examined. Regardless, through this research it is demonstrated that DWAPs equipped with a directional antenna pose a significant threat. The rise of such a capability requires the need for new countermeasures.

6.3 Research Contributions

This research contributes to the body of wireless attack drone research, specifically airborne CNAs utilizing a directional antenna. It shows empirically that cyber-attack drones can be highly effective tools capable of completing attacks well over 2200 m from a target. Additionally, development of a inexpensive prototype capable of several CNAs which models a motivated lone threat actor's capabilities is accomplished.

6.4 Limitations

This research has the following limitations:

- All CNAs are conducted in an open field. This prevents any additional attenuations due to obstacles. While this does not simulate a realistic network setup, it eliminates unknown factors and helps control experimental results.
- The location (optimal bearing) of the target network is assumed to be known.
- CNA are conducted from a prototype that is mounted and extended on a telescoping pole to simulate drone flight. Testing did not include real drone flight.

- Although capable of interaction with 5 GHz Wi-Fi devices, the CNAs are limited to a 2.4 GHz network.

6.5 Countermeasures

Over the course of this research, development, and experiments in the field of cyber-attack drones several countermeasures have been identified that would significantly increase the complexity or stop CNAs from being conducted by this platform. Below is a list of those countermeasures:

- **Limiting Building Wireless Emissions.** The vulnerability of unintended signal emissions has been around for many years and was given the code name TEMPEST in 1972 [60]. But, with the advent of highly capable COTS drones and inexpensive hacking hardware, the issue of implementing mitigations has become more pressing. The most straightforward approach, while not always the cheapest, is to limit the wireless emissions leaving a building. This can be done several ways. The first is to carefully assess the placement of client and station devices. Unfortunately, client device location is not always in the control of network administrators and stations best coverage locations may not be ideally located. The second way is to construct buildings with Radio Frequency (RF) attenuation materials that are designed to eliminate unwanted wireless emission. These materials come in many forms such as film, foil, paint, and fabrics [61]. While some of these materials need to be applied during construction, some can easily be retroactively applied like paint and window films.

- **Migrate to WPA3.** WPA2's security has been compromised for many years now and even when devices are fully updated (like this research's equipment), they are still vulnerable. Now that WPA3 has been released and devices are

receiving certifications, migration to capable wireless devices should be a top priority. As discussed in Section 2.5.6, WPA3 implements a dragonfly handshake which is not vulnerable to offline cracking like WPA2's four-way handshake. WPA3 also prevents erroneous deauthentication messages with MFP. Some vulnerabilities were already discovered with WPA3, but have since been patched and is still considered more secure than WPA2.

- **Randomized MAC Addresses.** This research's software packages takes advantage of the constant MAC addresses that wireless APs advertise in beacon packets. These beacon MAC addresses make profiling devices trivial and skypie v2's code relied heavily on this information. If a mechanism can be implemented to rotate or randomize the MAC addresses, it would significantly increase the complexity of identifying devices and require another form of identification like radio frequency fingerprinting.

- **Periodic Wi-Fi Passphrase Changes and Unique SSIDs.** If migration to WPA3 is not yet possible, changing the passphrase for a network on a schedule is recommended. Every time this is done, an attacker is required to recapture the four-way handshake and crack the passphrase. Frequently changing the passphrase to strong passphrases and having unique SSIDs can be a strong deterrence against attackers who have previously broken into a network. Breaking strong passphrases requires a workstation with significant power, and unique SSIDs eliminate the option of using precomputed rainbow tables.

6.6 Future Work

The work on skypie v2 was done with the express goal of proving the effectiveness of an inexpensive and lightweight cyber-attack drone equipped with a directional

antenna. There are many avenues for improvements that could enhance this field of research. Some of those improvements include:

- **Drone Mounted/Realistic Network Performance Evaluation.** The experiments conducted in this research simulated flight by mounting skypie v2 to a telescoping pole and elevating it 13 feet in the air. This is a good simulation of drone flight, but may be more stable and less affected by gusts of wind. To fully assess its capabilities in flight, skypie should be mounted to a real drone. This coupled with a set of scenarios targeting networks in buildings would be an excellent showcase of what the cyber-attack drone platform can accomplish.

- **Additional CNA and CNE Features.** The ground work for connecting and interacting with Wi-Fi devices was accomplished during this research's development phase, and several key capabilities were added. But there are many more useful features that can be added. These capabilities could include: probe request client tracking, WPA2 KRACK attack, or MITM attacks. For example, probe requests reveal connection history of client devices to SSIDs. With geolocated-SSID information readily available online [3], probe requests could be used to identify and track individuals.

- **Compatibility With More Wireless Protocols.** The developed software package is extensively developed to capture, transport, and provide analysis of Wi-Fi traffic. Adding software and hardware support for additional wireless protocols would make skypie more of a universal cyber-attack platform. This could include Bluetooth, ZigBee, and even cellular wireless protocols. Expanding the attack surface in this way would allow for targeted attacks against almost any wireless device.

Appendix A. Skypie v2 Default Configuration File

```
1  ## Skypie Config File.  Modifying this file (skypie-config) alters the
       behavior of the program.
2
3  # SFTP Server
4  [fileserver]
5  # Sensor's name, creates unique storage location on skyport.  Useful for
        multiple sensors.
6  name=starchy
7  # Credentials for the SFTP account of the sensor's name
8  verifier=catsWears100Sweaters!
9  # Port to connect over SFTP for uploading/downloading sensor data
10 sftp_port=2222
11 # IP/hostname to connect over SFTP for uploading/downloading sensor data
12 sftp_server=ftp.balllaboratories.org
13 # Weather files will be deleted or kept after uploading to the remote
       server
14 remove_after_upload=False
15
16 # Logging Settings
17 [log]
18 # File logging level.  You may want to set this to 'none' if you are
       worried about the sensor being discovered. [debug, info, warning,
       critical, none]
19 logging_level=debug
20 # Debug file size.  How big (kB) each file will be before split.
       Smaller sizes give feedback faster, but bigger sizes are easier to
       manage.
21 logging_size=50
22
23 #Bluetooth Collection (not implemented)
```

```
24 [bluetooth]
25 # MAC of Bluetooth antenna used for collection.  Bluetooth is not
       supported.  Used as a placeholder.
26 bluetooth_mac=XX:XX:XX:XX:XX:XX
27
28 # WiFi Collection
29 [wifi]
30 # Mode the wifi will be in.  This affects the mirror and collection
       threads [off,collect,mirror]
31 mode=collect
32 # MAC of WiFi antenna used for collection.  Currently supports only 1.
       Can use only first half to denote just manufacturerer (example: aa:
       bb:cc)
33 antenna_mac=00:25:22
34 # Collection interval in seconds
35 interval=30
36 # Size in mB of buffer for preferred packets (see bookmarks file).
       Oldest files will be removed when full.
37 size_bookmarks=500
38 # Size in mB of buffer for envelope data (geo, compass, and packet
       summary data)
39 size_envelopes=500
40 # Size in mB of all packets captured
41 size_raw=500
42 # Turn off collection of all packets, used to save space [on, off]
43 raw_collect=on
44 # Max size in mB of collected files
45 file_size_interval=10
46 # Raw filter (libcap format), the filter the antenna will use as the
       basis for collection.  Only packets in this filter will be collected
47 raw_filter= wlan[0] == 0x80
```

```
48 # Bookmark filters (libcap format). Bookmarks are the only packets that
       are sent directly to skyport.  They are a subsest of the raw packets
        collected.
49 #  Multiple filters are allowed.  Seperate by a new line, be sure to
       indent each line with at least one space.  Each one requires
       processing time, so it's not recommended to do more than 4.
50 bookmarks_filters=wlan.fc.type_subtype == 4
51  wlan_mgt.ssid=="Stowaway Lounge"
52  wlan_fc.type == 2
53  wlan.fc.type_subtype == 8
54
55 # MirrorMode
56 [mirror]
57 # The MAC of the attack platform.  This device must be within range of
       the WiFi interface of the C2 machine
58 attack_mac=AA:AA:BB:BB:CC:CC
59 # The MAC of the victim.
60 target_mac=AA:AA:BB:BB:CC:CC
61 # 'All' will forward any traffic destined for the target's MAC address,
       allowing the attacker to send spoofed MAC frames. [all,attack_only]
62 forward_attackside=all
63 # [all,target_only]
64 forward_targetside=target_only
65
66 # Telemetry
67 [telemetry]
68 # [on,off] Store geo data
69 mode=on
70 # Max size in mB of telemetry data
71 size=80
72 # Length of time before data is written to a file in seconds
73 interval=42
```

```
74 # Calibration for the accelerometer/magnometer.  Adjust so that the
       bearing readings are close to 0 when the sensor is facing north.
       Min= -360, Max= 360
75 bearing_offset = -75
76
77 # Update/Transfer Management
78 [update]
79 # How often config changes are downloaded  (in seconds) from the SFTP
       server. 0 = Constant download attempts
80 download_wait= 30
81 # Time to wait (in seconds) after a data upload completes before
       initiating another.  0 = Constant upload attempts
82 upload_wait= 999
83 #  Changing to 'shutdown' notifies all operating threads they need to
       shutdown.  A gentle way to shut down.  Off is maintained when all
       the threads are done. Selfdestruct will fill the hard drive with 0's
        until the system crashes.  [on,shutdown,off,selfdestruct]
84 skypie_operation=on
85
86 #Attack Parameters
87 [attack]
88 #capture = start deauth and handshake thread to capture 4way handshake
89 #connect = connect to attack_mac AP with given password
90 #nmap = nmap connected network
91 mode = capture
92 #number of deauth packets to send
93 packets = 1
94 #MAC address of the target AP
95 attack_mac = AA:AA:BB:BB:CC:CC
96 #password to be used to connect to attack_mac AP
97 password =
98 #nmap parameters to be used
```

```
99 nmap_params = -sn -T3 192.168.43.1-254
100 #ping IP
101 ping_ip = 192.168.43.32
102 #how often to san for available APs and switch channels if needed
103 scan_interval = 30
104 #attack thread timeout msgs
105 message =
106 #Attack thread timeout variables in seconds
107 capture_timeout = 30
108 connect_timeout = 30
109 nmap_timeout = 30
```

Appendix B. HAT RGB LED Array Indication List

Table 13: HAT RGB LED Array Indication List

Indicator	**Event**
Sky Blue Fill	**Skypie** program initialization
Purple Fill	System is on and waiting to start the control loop. If this color persists, **skypie** is awaiting a GPS fix to synchronize the system clock.
Green Fill	Indicates a passive Wi-Fi collection thread has been started
Blue Fill	Indicates that the telemetry collection thread has been started
Pink Fill	Indicates that an upload or download with the FTP has been started
Yellow Fill	Indicates an attack thread has been started (handshake capture, deauth, nmap)
Pink Flashing	Indicates that upload or download has failed multiple times due to no connection
White Number	Indicates which channel the **skypie** is listening on
Red Number	Indicates a buffer check or change has occurred
Orange Number	Indicates the state file has been written to
Flashing Red	Indicates a self-destruct has been initiated

Appendix C. Python Thread For Managing Wireless Connections

```
1  import logging
2  import threading
3  from skypie import interface
4  import time
5  import shutil
6  from wpasupplicantconf import WpaSupplicantConf
7  from io import StringIO
8  from subprocess import PIPE, Popen
9  import re
10
11 module_logger = logging.getLogger(__name__)
12
13 class ConnectThread(threading.Thread):
14
15     def __init__(self, iface, ssid, address, password,
       connectedNetworkPath):
16         super().__init__()
17         self.daemon = True
18         self._iface = iface
19         self._ssid = ssid
20         self._address = address
21         self._password = '"' + password + '"'
22         self.event = threading.Event()
23         self._connectedNetworkPath = connectedNetworkPath
24         self.connected = False
25
26         module_logger.info("[ ] Starting connect thread instance.
       Attempting to connect to {}".format(self._ssid))
27
```

```
28          # Ensure we are in monitor mode, if not set it
29          while interface.get_interface_mode(self._iface) != "managed":
30              interface.set_interface_mode(self._iface, "managed")
31
32      def run(self):
33
34          #Read wpa_supplicant config file
35          file = '/etc/wpa_supplicant/wpa_supplicant.conf'
36          with open(file, 'r') as myfile:
37              data = myfile.read()
38          data = StringIO(data)
39          conf = WpaSupplicantConf(data)
40
41          #Add attack network and write to disk
42          conf.add_network(self._ssid, psk=self._password, key_mgmt='WPA-
    PSK')
43          output = StringIO()
44          conf.write(output)
45          with open(file, 'w') as myfile:
46              output.seek(0)
47              shutil.copyfileobj(output, myfile)
48
49          data.close()
50          output.close()
51
52          #print(conf.networks())
53          #Force wpa_supplicant to read updated config file
54          reconfigCommand = ['wpa_cli', '-i', self._iface, 'reconfigure']
55          proc = Popen(reconfigCommand, stdout=PIPE, stderr=PIPE)
56          proc.wait()
57          proc.terminate()
58
```

```
59          #Select the target network for connect
60          connectCommand = ['wpa_cli', '-i', self._iface, 'select_network'
    , str(list(conf.networks().keys()).index(self._ssid))]
61          proc = Popen(connectCommand, stdout=PIPE, stderr=PIPE)
62          proc.wait()
63          proc.terminate()
64
65          time.sleep(3)
66
67          #Check if connected
68          stateCommand = ['wpa_cli', '-i', self._iface, 'status']
69          proc = Popen(stateCommand, stdout=PIPE, stderr=PIPE)
70          curr_line = ""
71          while not "wpa_state=" in curr_line:
72              curr_line = proc.stdout.readline().decode()
73          if re.sub(r"[\n\t\s]*", "", curr_line) == 'wpa_state=COMPLETED':
74              self.connected = True
75              module_logger.info("[+] Connected Succesfully to {} ".format
    (self._ssid))
76              with open(self._connectedNetworkPath, 'w') as f:
77                  f.write(self._address + "\n")
78                  f.close()
79          else:
80              module_logger.info("[-] Failed to Connect to {} ".format(
    self._ssid))
81
82          proc.terminate()
83          time.sleep(5)
```

Appendix D. Startup Shell Script

```
1  #! /bin/sh
2  RESULTS=$$(pgrep -af python)
3  SKYPIE_RESULTS=$(echo $RESULTS | grep skypie)
4
5  if [ -z "$SKYPIE_RESULTS" ]
6  then
7          echo "[-] Skypie is not running. Starting Skypie..."
8          cd /home/pi/PycharmProjects/skypie/skypie
9          python3 main.py -a -d -n- w /home/pi/PycharmProjects/skypie/
       skypie
10 else
11         echo "[+] Skypie is running."
12 fi
```

Appendix E. FSPL To Distance Calculation

$$FSPL = 10 * \log_{10}((\frac{4\pi d}{\lambda})^2)$$

$$\frac{FSPL}{10} = \log_{10}((\frac{4\pi d}{\lambda})^2)$$

$$10^{\frac{FSPL}{10}} = (\frac{4\pi d}{\lambda})^2$$

$$\sqrt{10^{\frac{FSPL}{10}}} = \frac{4\pi d}{\lambda}$$

$$\lambda * \sqrt{10^{\frac{FSPL}{10}}} = 4\pi d$$

$$d = \frac{\lambda * \sqrt{10^{\frac{FSPL}{10}}}}{4\pi}$$

Bibliography

1. C. Bramlette, "Cyber-Attack Drone Payload Development and Geolocation via Directional Antenna," Master's thesis, Air Force Institute of Technology, 2018, accessed: Jun 03, 2019. [Online]. Available: https://scholar.afit.edu/etd/2247/

2. B. Mitchell, "What Is the Range of a Typical Wi-Fi Network?" 2019, accessed: Jan 27, 2020. [Online]. Available: https://www.lifewire.com/range-of-typical-wifi-network-816564

3. "Statistics," 2019, accessed: Jun 06, 2019. [Online]. Available: https://wigle.net/stats#

4. Y. A. De Montjoye, C. A. Hidalgo, M. Verleysen, and V. D. Blondel, "Unique in the Crowd: The Privacy Bounds of Human Mobility," *Scientific Reports*, vol. 3, pp. 1–5, 2013.

5. Y. Michalevsky, A. Schulman, G. A. Veerapandian, D. Boneh, and G. Nakibly, "Powerspy: Location Tracking Using Mobile Device Power Analysis," in *24th USENIX Security Symposium*, 2015, pp. 785–800.

6. E. Skoudis and T. Liston, *Counter Hack Reloaded: A Step-By-Step Guide to Computer Attacks and Effective Defenses*. Prentice Hall Press, 2005.

7. F. Brown and D. Latimer, "Game of Drones: Putting the Emerging Drone Defense' Market to the Test," 2017, accessed: Jun 06, 2019. [Online]. Available: http://www.bishopfox.com/files/slides/2017/DEF_CON_25_(2017)-Game_of_Drones-Brown_Latimer-29July2017.pdf

8. Consumer and G. Affairs, "Jamming Cell Phones and GPS Equipment is Against the Law," 2015, accessed: Feb 06, 2020. [Online]. Available: https://www.fcc.gov/general/jamming-cell-phones-and-gps-equipment-against-law

9. Congress, "Public Law 112 95," pp. 1–145, 2012, accessed: Jun 04, 2019. [Online]. Available: https://www.congress.gov/112/plaws/publ95/PLAW-112publ95.pdf

10. C. F. Liew, D. DeLatte, N. Takeishi, and T. Yairi, "Recent Developments in Aerial Robotics: A Survey and Prototypes Overview," pp. 1–14, 2017, accessed: Jun 06, 2019. [Online]. Available: https://arxiv.org/abs/1711.10085

11. D. A. Longino, "Role of Unmanned Aerial Vehicles in Future Armed Conflict Scenarios," Tech. Rep., 1994, accessed: Jun 04, 2019. [Online]. Available: https://apps.dtic.mil/dtic/tr/fulltext/u2/a289777.pdf

12. M. S. Ammoo and M. N. Dahalan, "Micro Air Vehicle: Technology Review and Design Study," in *Proceedings of the 1st Regional Conference on Vehicle Engineering & Technology*, 2006, pp. 1–8.

13. W. R. Davis, B. B. Kosicki, D. M. Boroson, and D. F. Kostishack, "Micro air vehicles for optical surveillance," *Lincoln Laboratory Journal*, vol. 9, no. 2, pp. 197–214, 1996.

14. S. J. Morris, "Design and Flight Test Results for Micro-Sized Fixed-Wing and VTOL Aircraft," in *Proceedings of the First International Conference on Emerging Technologies for Micro Air Vehicles*, 1997.

15. D. J. Pines and F. Bohorquez, "Challenges Facing Future Micro-Air-Vehicle Development," *Journal of Aircraft*, vol. 43, no. 2, pp. 290–305, 2006.

16. A. Prabhakar, "Breakthrough Technologies for National Security," *Defense Advanced Research Projects Agency (DARPA), Tech. Rep.*, 2015, accessed: Jun 04, 2019. [Online]. Available: https://www.darpa.mil/attachments/DARPA2015.pdf

17. R. Martorana, "WASP–A High-G Survivable UAV," *1st UAV Conference*, pp. 1–9, May. 2002.

18. Government Accountability Office, "Nonproliferation: Agencies Could Improve Information Sharing and End-Use Monitoring on Unmanned Aerial Vehicle Exports," 2012, accessed: Jun 04, 2019. [Online]. Available: http://gaonet.gov/assets/600/593132.pdf.

19. P. Tucker, "In Urgent Request, US Special Ops Adds 350 Kamikaze Drones to Fight ISIS," pp. 1–8, 2017, accessed: Jun 04, 2019. [Online]. Available: https://www.defenseone.com/technology/2017/05/Special-Ops-Gets-350-More-Kamikaze-Suicide-Drones-to-Fight-ISIS/137987/

20. DoD, "FY 16-27 PA. Lethal Miniature Aerial Missile System (LMAMS) Reprogramming Action," 2016, accessed: Jun 04, 2019. [Online]. Available: https://admin.govexec.com/media/16-27_pa_lmams_request.pdf

21. The Economist, "Technology Quarterly - Taking Flight - Civilian drones," 2017, accessed: Jun 04, 2019. [Online]. Available: https://www.economist.com/technology-quarterly/2017-06-08/civilian-drones

22. D. Floreano and R. J. Wood, "Science, technology and the future of small autonomous drones," *Nature*, vol. 521, no. 7553, pp. 460–466, 2015.

23. FAA, "Operation and Certification of Small Unmanned Aircraft Systems," pp. 1–624, 2016, accessed: Jun 05, 2019. [Online]. Available: https://www.faa.gov/uas/media/RIN_2120-AJ60_Clean_Signed.pdf

24. D. Bamburry, "Drones: Designed for product delivery," *Design Management Review*, vol. 26, no. 1, pp. 40–48, 2015.

25. B. Barrett, "Technology Quarterly - Taking Flight - Civilian drones," 2018, accessed: Jun 06, 2019. [Online]. Available: https://www.wired.com/story/olympics-opening-ceremony-drone-show/

26. J. Flynt, "The Complete Drone Comparison," 2018, accessed: Jun 03, 2019. [Online]. Available: https://3dinsider.com/drone-comparison/

27. IEEE, "IEEE Xplore," 2019, accessed: Jun 04, 2019. [Online]. Available: https://ieeexplore.ieee.org/search/advanced

28. S. Aldhaher, P. D. Mitcheson, J. M. Arteaga, G. Kkelis, and D. C. Yates, "Light-Weight Wireless Power Transfer for Mid-Air Charging of Drones," in *2017 11th European Conference on Antennas and Propagation*, 2017, pp. 336–340.

29. J. Wang, C. Jiang, Z. Han, Y. Ren, R. G. Maunder, and L. Hanzo, "Taking Drones to the Next Level: Cooperative Distributed Unmanned-Aerial-Vehicular Networks for Small and Mini Drones," *IEEE Vehicular Technology Magazine*, vol. 12, no. 3, pp. 73–82, Sep. 2017.

30. W. G. Aguilar, C. Angulo, and J. A. Pardo, "Motion Intention Optimization for Multirotor Robust Video Stabilization," in *CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies*. IEEE, 2017, pp. 1–4.

31. E. Nowak, K. Gupta, and H. Najjaran, "Development of a Plug-and-Play Infrared Landing System for Multirotor Unmanned Aerial Vehicles," *Proceedings - 2017 14th Conference on Computer and Robot Vision*, pp. 256–260, 2018.

32. H. C. Nguyen, R. Amorim, J. Wigard, I. Z. Kovacs, T. B. Sorensen, and P. E. Mogensen, "How to Ensure Reliable Connectivity for Aerial Vehicles over Cellular Networks," *IEEE Access*, vol. 6, pp. 12 304–12 317, 2018.

33. "IEEE Standard for Information Technology – Telecommunications and Information Exchange Between Systems Local and Metropolitan Area networksSpecific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*, pp. 1–3534, 2016, accessed: Jun 05, 2019. [Online]. Available: https://standards.ieee.org/standard/802_11-2016.html

34. J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach*, 6th ed. Pearson, 2012.

35. "IEEE Standard for Information Technology – Local and Metropolitan Area Networks – Specific requirements – Part 15.1a: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPAN)," *IEEE Std 802.15.1-2005 (Revision of IEEE Std 802.15.1-2002)*, pp. 1–700, 2005, accessed: Jun 05, 2019. [Online]. Available: https://standards.ieee.org/standard/802_15_1-2005.html

36. D. Hulton, "Breaking Wireless... Faster," 2006, accessed: Jan 08, 2020. [Online]. Available: http://2006.recon.cx/en/f/dhulton-breaking-wifi-faster.ppt

37. "IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-1997*, pp. 1–445, 1997, accessed: Jun 05, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/654749

38. S. Fluhrer, I. Mantin, and A. Shamir, "Weaknesses in the Key Scheduling Algorithm of RC4," in *International Workshop on Selected Areas in Cryptography*. Springer, 2001, pp. 1–24.

39. E. Tews, R. Weinmann, and A. Pyshkin, "Breaking 104 bit WEP in less than 60 seconds," in *International Workshop on Information Security Applications*. Springer, 2007, pp. 188–202.

40. I. P. Mavridis, A. E. Androulakis, A. B. Halkias, and P. Mylonas, "Real-Life Paradigms of Wireless Network Security Attacks," in *2011 15th Panhellenic Conference on Informatics*, 2011, pp. 112–116.

41. "IEEE Standard for Information Technology – Telecommunications and Information Exchange Between Systems–Local and Metropolitan Area Networks-Specific Requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Amendment 6: Medium Access Control (MAC) Security Enhancements," *IEEE Std 802.11i-2004*, pp. 1–190, 2004, accessed: Jun 05, 2019. [Online]. Available: https://standards.ieee.org/standard/802_11i-2004.html

42. M. Khasawneh, I. Kajman, R. Alkhudaidy, and A. Althubyani, "A Survey on Wi-Fi Protocols: WPA and WPA2," in *Recent Trends in Computer Networks and Distributed Systems Security*, G. Martínez Pérez, S. M. Thampi, R. Ko, and L. Shu, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 496–511.

43. M. Vanhoef and F. Piessens, "Key Reinstallation Attacks," *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1313–1328, 2017.

44. D. Harkins, "Simultaneous Authentication of Equals: A Secure, Password-Based Key Exchange for Mesh Networks," in *2008 Second International Conference on Sensor Technologies and Applications*. IEEE, 2008, pp. 839–844.

45. M. Vanhoef and E. Ronen, "Dragonblood: Analyzing the Dragonfly Handshake of WPA3 and EAP-pwd," in *IEEE Symposium on Security & Privacy (SP)*. IEEE, 2020.

46. E. M. Hutchins, M. J. Cloppert, and R. M. Amin, "Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains," *Leading Issues in Information Warfare & Security Research*, vol. 1, no. 1, p. 80, 2011.

47. K. Beaver, "Commonly Hacked Ports," 2015, accessed: Jan 3, 2020. [Online]. Available: https://www.dummies.com/programming/networking/commonly-hacked-ports/

48. J. Valente and A. A. Cardenas, "Understanding Security Threats in Consumer Drones Through the Lens of the Discovery Quadcopter Family," in *Proceedings of the 2017 Workshop on Internet of Things Security and Privacy*, vol. 2015, no. 6. New York, New York, USA: ACM Press, 2017, pp. 31–36.

49. M. Tassey and R. Perkins, "Wireless Aerial Surveillance Platform," *DEFCON 19*, 2011, accessed: Jun 06, 2019. [Online]. Available: https://www.defcon.org/images/defcon-19/dc-19-presentations/Tassey-Perkins/DEFCON-19-Tassey-Perkins-Wireless-Aerial-Surveillance-Platform.pdf

50. C. Paget, "Practical Cellphone Spying," *DEFCON 18*, 2010. [Online]. Available: https://www.youtube.com/watch?v=fQSu9cBaojc

51. J. Greenwood, "The Phantom Menace - Weaponising a Consumer Drone," 2015, accessed: Jun 06, 2019. [Online]. Available: https://www.4armed.com/blog/phantom-menace-weaponising-drones/

52. T. Levin, "The Drone Noise Test," 2017, [Accessed: Dec 8, 2019]. [Online]. Available: https://www.wetalkuav.com/dji-drone-noise-test/

53. B. E. Law, "Passive Radiolocation Of IEEE 802.11 Emitters Using Directional Antennae," Master's thesis, Wright-Patterson AFB OH Wright-Patterson United States, 2018, accessed: Jun 03, 2019. [Online]. Available: https://scholar.afit.edu/cgi/viewcontent.cgi?article=2812&context=etd

54. S. M. Beyer, B. E. Mullins, S. R. Graham, and J. M. Bindewald, "Pattern-of-Life Modeling in Smart Homes," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 5317–5325, 2018.

55. L. Brown, "Top 10 Heavy Lift Drones 2019," 2019, accessed: Nov 13, 2019. [Online]. Available: https://filmora.wondershare.com/drones/top-heavy-lift-drones.html

56. D. D. N. Ltd., "NextG USB-Yagi TurboTenna Plug n Play Wi-Fi Antenna," 2018, [Accessed: Nov 21, 2019]. [Online]. Available: http://www.danets.com/turbotenna/UsbYagi.php

57. MetaGeek, "Understanding WiFi Signal Strength," 2019, accessed: Feb 09, 2020. [Online]. Available: https://www.metageek.com/training/resources/wifi-signal-strength-basics.html

58. K. C. Mansfield Jr and J. L. Antonakos, *Computer Networking for LANs to WANs: Hardware, Software and Security*. Cengage Learning, 2009, p. 501.

59. R. Wilson, "Propagation Losses Through Common Building Materials 2.4 GHz vs 5 GHz," *Magis Networks Inc.: San Diego, CA, USA*, 2002. [Online]. Available: http://www.boggstech-consulting.com/Wilson_Propagation_Losses_2_and_5GHz.pdf

60. J. Friedman, "Tempest: A Signal Problem," *NSA Cryptologic Spectrum*, vol. 35, p. 76, 1972.

61. S. Defense, "RF Protection," 2016, accessed: Jan 5, 2020. [Online]. Available: https://www.signalsdefense.com/rf-Protection/

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
26–03–2020	Master's Thesis	Sept 2018 — Mar 2020

4. TITLE AND SUBTITLE

DEVELOPMENT OF A DRONE-MOUNTED WIRELESS ATTACK PLATFORM

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Barker, Nathan V., 2d Lt, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENG-MS-20-M-005

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

None

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The commercial drone market has grown rapidly due to the increasing utility and capabilities of drones. This new found popularity has made it possible for inexpensive drones capable of impressive carry capacities and flight times to reach the consumer market. These new features also offer an invaluable resource to wireless hackers. Capitalizing on their mobility, a wireless hacker can equip a drone with hacking tools to surpass physical security (e.g. fences) with relative ease and reach wireless networks. This research seeks to experimentally evaluate the ability of a drone-mounted wireless attack platform equipped with a directional antenna to conduct wireless attacks effectively at distances greater than 800 meters. To test this hypothesis, the "skypie v2" prototype conducts computer network attacks against a target network and captured data is used to evaluate the effectiveness of the platform. Results showed that capture of a WPA2 handshake was possible at a RSSI of -72 dBm or 2400 meters from a network located in a open field. Additionally, nmap scans were conducted with a RSSI value of -74 dBm or nearly 3000 meters from the target network.

15. SUBJECT TERMS

Offensive Cyber Operations, Drone, Wireless Networking, Directional Antenna

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Barry E. Mullins, AFIT/ENG
U	U	U	UU	140	19b. TELEPHONE NUMBER *(include area code)* (937) 255-3636, ext 7979; barry.mullins@afit.edu