3-30-2016

People Counting and occupancy Monitoring using WiFi Probe Requests and Unmanned Aerial Vehicles

Edwin George Vattapparamban
Florida International University, evatt001@fiu.edu

FLORIDA INTERNATIONAL UNIVERSITY

Miami, Florida

PEOPLE COUNTING AND OCCUPANCY MONITORING USING WIFI PROBE REQUESTS AND UNMANNED AERIAL VEHICLES

A thesis submitted in partial fulfillment of

the requirements for the degree of

MASTER OF SCIENCE

in

ELECTRICAL ENGINEERING

by

Edwin Vattapparamban

2016

To: Interim Dean Ranu Jung
College of Engineering and Computing

This thesis, written by Edwin Vattapparamban, and entitled People Counting and Occupancy Monitoring using WiFi Probe Requests and Unmanned Aerial Vehicles , having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this thesis and recommend that it be approved.

Kemal Akkaya

Leonardo Bobadilla

A. Selcuk Uluagac

İsmail Güvenç, Major Professor

Date of Defense: March 30, 2016

The thesis of Edwin Vattapparamban is approved.

Interim Dean Ranu Jung
College of Engineering and Computing

Andres G. Gil
Vice President for Research and Economic Development
and Dean of the University Graduate School

Florida International University, 2016

ACKNOWLEDGMENTS

I would like to thank all the Professors in my committee for their kind guidance and feedback which always motivated, inspired and encouraged me in achieving the final goal of this research. Specially my major Professor Dr. İsmail Güvenç for his mentorship, patience and guidance throughout my graduate studies all the time.

Further, I would like thank Mobile, Pervasive, and Autonomous Communications Technologies (MPACT) group and their members for been supportive and helpful during my graduate studies and to the Department of Electrical and Computer Engineering, FIU for providing me with this research opportunity.

Finally, I would like to thank my family and friends for supporting me in many ways to complete my research work during this entire time duration.

ABSTRACT OF THESIS

PEOPLE COUNTING AND OCCUPANCY MONITORING USING WIFI PROBE REQUESTS AND UNMANNED AERIAL VEHICLES

by

Edwin Vattapparamban

Florida International University, 2016

Miami, Florida

Professor İsmail Güvenç, Major Professor

Smart phones have become an important part of our daily lives due to their capabilities of accessing the web using WiFi and mobile data networks. These WiFi equipment are constantly sending out packets referred as probe requests, which can be tracked using wireless sniffers. In this thesis, first we investigate capturing of WiFi probe request packets using the help of WiFi Pineapple devices, and analyze how we can use signal strength information of probe request data for indoor occupancy monitoring. Applications of such occupancy monitoring into building surveillance and building energy management are also discussed.

After completing the initial test indoors, research was moved to outdoor monitoring with the help of unmanned aerial vehicles (UAVs) flying in various trajectories and capturing probe request messages. The information captured from the probe requests is used to identify and localize WiFi users with a single UAV, which can be instrumental in search and rescue applications. Finally, we study in detail various security, privacy, and public safety issues related to drones equipped with wireless communications capabilities.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER I

Introduction

Occupancy tracking has been active research topic because of the challenges and its importance. Buildings consume around 72% of electricity consumption and are considered to be the largest consumers of electricity in the United States [8]. And occupancy monitoring of people can be very helpful in saving significant energy in buildings. Another ways in which monitoring can be via video processing, camera systems or deploying occupancy sensors in buildings [9]. As installing new equipments can be costly an ideal solution is required. Also in search and rescue operations outdoors with Unmanned Aerial vehicles (UAV) [10] tracking people in disastrous conditions is very vital. And in my thesis Wireless signals can be used as alternative to identify occupants.

Probe requests are signals that are continuously broadcast from devices with WiFi technology, such as smartphones, laptops, and tablets [11, 12, 13, 14]. When a WiFi client wants to get connected to a WiFi network, the first method is scanning for beacon frames, which are frames broadcast by WiFi routers to tell about their presence to WiFi clients [15]. The second method is sending probe requests, which also contains the unique MAC address of the device, as well its type, brand, manufacturer, and model. Since a WiFi client itself can initiate a connection to a WiFi router instead of waiting for a beacon frame from the router, use of probe requests is preferable. The probe requests are not encrypted, and can be captured and decoded with the help of wireless sniffers passively, without connecting to a particular network or transmitting any signal.

It is possible to capture the received signal strength (RSS) information of probe requests using sniffers such as WiFi Pineapple (WiFi-PA) [16]. This information can then be used for occupancy monitoring inside the building. Probe requests are bursty in nature as they are broadcasted in the air in search of WiFi networks to get connected, to get a list of available networks, or to handover between WiFi APs. Frequent transmission of probe requests introduces an opportunity to track occupancy of building by simply monitoring probe requests. To our best knowledge, there are no detailed studies in the literature that report efficiency of occupancy tracking using WiFi probe requests. Our work is unique about using WiFi probe requests for occupancy tracking.

An unmanned aerial vehicle (UAV), also known as drone, is an aircraft with no pilot on board. Enabled by recent technological advances, miniaturization, and open-source hardware/software initiatives [17, 18, 19], UAVs have found several key applications recently [20, 21, 22, 23, 24, 25, 26]. Their use in several different contexts are quickly transforming from a futuristic idea to reality. Amazon, for example, claims that seeing its *Prime Air* order delivery UAVs in the sky is expected to be as conventional as seeing mail trucks on the road within the next few years[1]. Google and Facebook have been investigating the use of a network of high-altitude balloons[2]. and drones[3]. over specific population centers for providing broadband connectivity. Such solar-powered drones are capable of flying several years without refueling. UAVs can also be used to deliver broadband data rates in emergency and public safety situations through low-altitude platforms [27].

In this thesis, occupancy tracking has been done using linear least square estimation method along with security issues in WiFi enabled Smartphones have been discussed [28].

[1]http://www.amazon.com/b?node=8037720011

[2]http://www.google.com/loon/

[3]https://info.internet.org/en/approach/

After that, a detailed survey about Drones have been conducted with issues related to Cybersecurity and Privacy [29].

CHAPTER II

Literature Review

This section summarizes issues regarding location analytics and tracking and also privacy and security related to probe request messages and are classified in Table 1. Also, a study on challenges and concerns to be addressed in the areas of cybersecurity,privacy and public safety.

2.1 Location Analytics, Search and Rescue.

WiFi probe requests messages sent from WiFi enabled Smartphone devices are used in location analytics of shoppers. To improve marketing, business firms are using probe request informations to understand the frequency of shoppers visiting stores. And this information can be used for advertising purposes, to estimate number of people at peak hours. In [30], Cisco Meraki Aps are used to capture probe requests packets from users. This data is sent to the cloud and processed, where a database is generated that uses algorithms to study the patterns of shoppers. As, these are vital informations gathered from Smartphones to maintain privacy, only hashed version of the MAC address is stored in the database so that the users identity is not revealed. Euclid Analytics [31] and Libelium [32] are other products that can be used for similar purposes. In [14, 15], data mining techniques are used to extract meaningful information from database of probe requests.

In [33], it is shown that use of probe request information can be utilized in search and rescue operations. In particular, a WiFi-equipped drone actively broadcasts request-to-send

(RTS) frames (100 per second) to trigger transmission of probe requests from a victim WiFi device. This information is then used to coarsely estimate the location of the WiFi device. Our work is different from the above studies in the sense that we capture the probe request for occupancy monitoring purposes. Privacy is maintained in the sense that the addresses are anonymized as in [30].

Category	References
Location Analytics and Statistics	[30, 14, 15, 32, 31, 33]
Trajectory Tracking	[13, 34, 35, 36]
Privacy and Security	[11, 37, 12, 38, 39, 40, 41]

Table 1: Literature related to WiFi probe requests.

2.2 Trajectory Tracking

Probe request information have been used for second-by-second detection of a moving device (walking, driving) in [13] for outdoor environments. Subsequently, trajectory estimation of the mobile device is obtained using the Viterbi algorithm. Triangulation of mobile user locations by jointly using probe requests at multiple reference positions is not considered. Other recent work that study trajectory tracking using probe request information include [34, 35, 36]. However, typically outdoor locations are considered, and triangulation and zone-level localization of a mobile device's position, as targeted by our work, is not explicitly studied.

2.3 Privacy and Security

In [42], tracking of WiFi users by passively capturing probe requests using WiFi sensors deployed at various locations is presented. The mobile user is assumed to be within the zone of its strongest WiFi sensor, and as opposed to our work, no explicit triangulation techniques are considered to fuse information from multiple WiFi sensors. Various tests were carried out using Android and Apple phones/tablets. Results show that more probe

requests are sent when the device is in active mode (screen on) and not connected to any network, which can be used for tracking individual users based on RSS information. Even though companies such as Libelium [32] and Euclid Analytics [31] claim that privacy preserving techniques are possible, [42] shows that de-anonymization of a protected dataset is possible using different attacks. In another work [12], various privacy-related threats due to use of WiFi probe requests and factors affecting transmit frequency of probe requests have been discussed. Tests were conducted by keeping the phones in many different configurations such as connected mode and airplane mode. The maximum probing frequency is observed to happen when a device attempts to connect to a known network in its area. It is also shown that for a commercially deployed MAC randomization mechanism, it is possible to re-identify anonymized probes.

In [38], an attack referred as KARMA is described, in which the attacker automatically sends beacon and probe response frames for every received probe request, to direct the clients to his own network. This allows full control of the data sent by connected clients over the attacker's network. Authors in [38] introduce a detection mechanism for KARMA attacks, in which directed probe requests are sent with random SSIDs, and based on received probe responses the attacker can be identified. In [39], it is shown that early stage of probe-request WiFi attacks can be identified with help of neural networks. In [37], hash-based anonymization of MAC addresses captured from probe requests are shown to be defeatable, while [41, 40] report various privacy vulnerabilities of mobile devices due to use of probe requests.

CHAPTER III

Occupancy monitoring in Buildings using WiFi Probe Requests

Occupancy monitoring is important as they can be used for applications such as energy management, surveillance, and security. In this section, we passively capture probe request messages using WiFi Pineapple equipment and the information is used to understand occupancy zones and obtain occupancy count with respect to each zone and at different time intervals.

3.1 Capturing Probe Request Data with WiFi Pineapple

In the experiments, eight different WiFi Pineapple equipment are deployed at various locations at FIU EC 3rd Floor as shown in Fig. 1. We use the tcpdump sniffer to capture WiFi packets. As we are only interested in receiving the WiFi probe requests, all the other packets are filtered out and only WiFi probes are received. The data captured at WiFi Pineapple includes time stamps providing the time at which the data was captured, MAC address of the WiFi enabled device, and the signal strength of the WiFi device.

All the WiFi Pineapple equipment are powered using a power adapter that is connected to an electric outlet. The scripts that we use to control each Pineapple can be executed via their terminal. They support boot modes which comes with five user configurable switches that help in automated execution of multiple commands using the terminal. Using the web interface, we specify which script to run on which equipment, and as soon as the device is booted and switches points out to our script, it starts executing commands. One of the

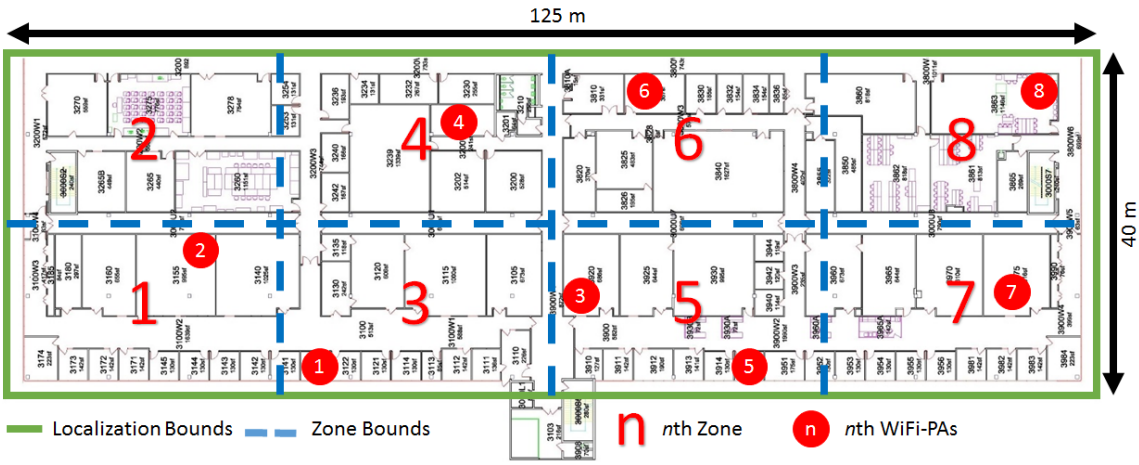


Figure 1: Map of FIU Engineering Center 3rd Floor.

problems faced during our experimentation was that the internal clock time of the device keeps on changing every time it reboots which gives us wrong data capture timings. In order to address this problem, we forced each WiFi Pineapple to automatically connect to a WiFi network. Subsequently, using the Network Time Protocol (NTP) Server the current date and time was synchronized with the time at the local network.

3.2 Database Server for Storing Probe Request Data

In order to obtain the data capture information from the deployed WiFi Pineapple equipment, a Linux server has been set up. This makes it convenient to receive all the information at a centralized location. The data is stored in the SD card of the WiFi pineapple device for a day, and at the end of the day the captured data is sent to the server. Subsequently, each WiFi Pineapple starts capturing new data automatically, while also storing the back-up of the data in the SD card. The data is obtained in the packet capture format and can be viewed in Wireshark software [43], which is a packet analyzer that is used to convert the packets into comma separated variable (CSV) format that can be imported into MATLAB for further processing. The data is transferred from the the device to the server using Linux command `SCP`. In this command we enter the source address and the destination address

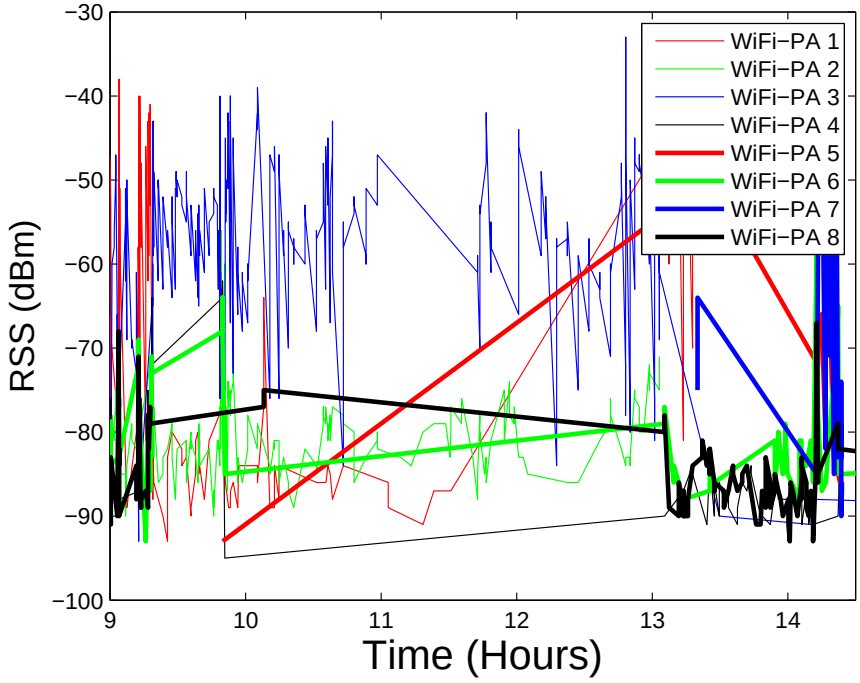


Figure 2: Raw measurements of RSS.

we want to send the file. This also provides the time stamp of the file transfer, making it easy to search for particular packet capture file according to the date and time.

3.3 Pre-Processing Data for Location Estimation

The data captured and saved into server has to be pre-processed before using them in localization and tracking algorithm. First, the RSS data is resampled in time due to its bursty nature. For example, there may be several probe requests from the same user within few hundred milliseconds, followed by a silent period that may last several seconds. In order to have uniformly sampled RSS captures, we average the received RSS values within one second intervals. After resampling the RSS measurements, the data needed to be interpolated for measurement intervals that do not have any RSS readings, and are also within close vicinity of other measurement intervals which have RSS readings. This is critical for

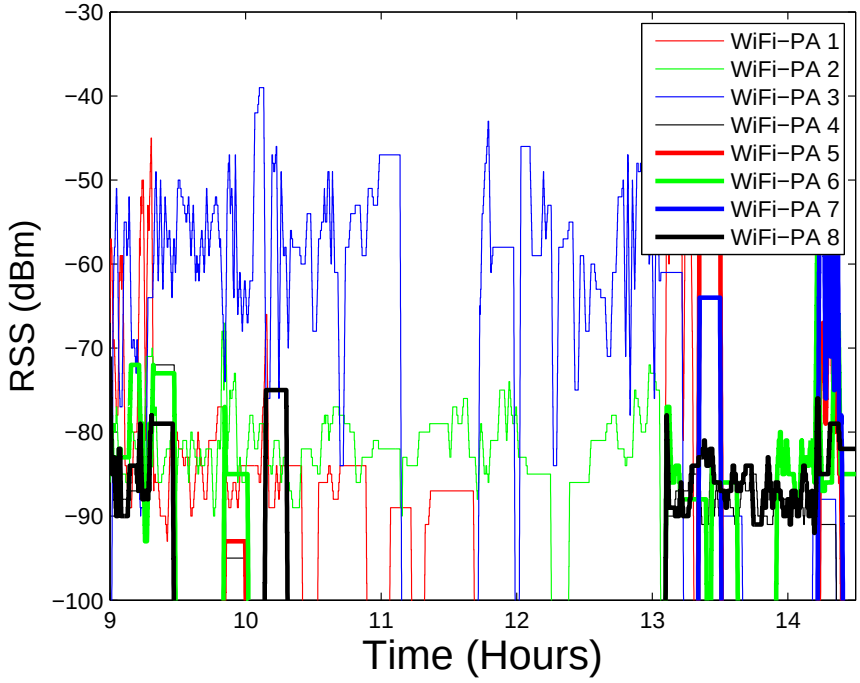


Figure 3: Processed data of RSS.

estimation since it needs at least four different measurements as subsequently stated in 3.4. In Fig. 2 raw measurements and Fig. 3 processed data of RSS from 8 WiFi-PAs are shown, where the locations of WiFi-PAs are as reported in Fig. 1. The post-processed data can be used conveniently for localization of users based on probe request information RSS.

The cumulative distributions of RSS for each WiFi-PAs and overall is shown in Fig. 4. As it can be seen more than 70% of RSS values lie between -90 dBm and -70 dBm. This information is helpful about outlier detection. In this work, an RSS value smaller than -100 dBm or larger than -30 dBm is regarded as outlier and not considered for results.

3.4 Device Localization Using Probe Requests

After post-processing of the data, localization techniques are used to get occupancy counts with respect to time for FIU EC 3rd floor. In [44], localization of WiFi APs are studied

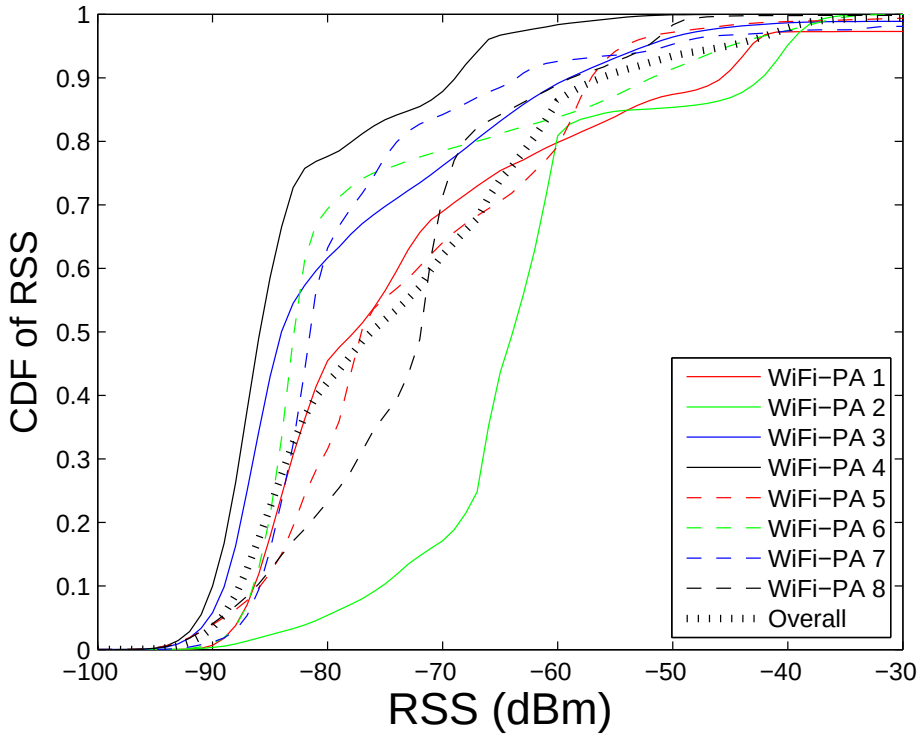


Figure 4: CDF of RSS for different WiFi-PAs.

with unknown transmit power and path loss exponent (PLE). In our case we are localizing the devices with the help of WiFi-PAs, and transmission configurations of different WiFi equipment may have large variations [45]. Therefore, we use a similar technique to that of [44] for localizing users with unknown transmit powers.

The unknown position of a random user is denoted as (x_0, y_0) and the location of the ith WiFi Pineapple is denoted as (x_i, y_i), where $1 \leq i \leq k$. The RSS measurement in the ith position is denoted as r_i. The true distance between the user and the ith reference node is then given by:

$$d_i = \sqrt{(x_0 - x_i)^2 + (y_0 - y_i)^2}, \tag{1}$$

and the path-loss model is for the RSS is:

$$P_r = P_0 - 10n\log_{10}(d/l_0) + X_\sigma, \tag{2}$$

where P_r is the RSS, P_0 is signal strength at the reference distance l_0, n is the PLE which depends on the physical environment, and X_σ is the noise modeled by a Gaussian random variable with a standard deviation of σ and mean of zero. Using (2) the distance between transmitter and ith WiFi Pineapple is estimated as:

$$\hat{d_i} = 10^{(P_0 - r_i)/10n}. \tag{3}$$

Multilateration is the most known and used localization technique when distance estimates of at least three non-collinear reference positions are available. As it can be seen in (1), distance estimation requires the knowledge of transmit power and PLE. A linear approximation is used to overcome this situation in [44] as:

$$10^{(P_0 - r_i)/10n} \approx a_0 + a_1((P_0 - r_i)/10n), \tag{4}$$

where a_0 and a_1 are the linearization coefficients. It is shown that the linearization coefficients do not affect the localization accuracy. Subtracting all approximations of RSS from

different WiFi-PAs:

$$\underbrace{\begin{bmatrix} -x_1^2 - y_1^2 + x_k^2 + y_k^2 \\ -x_2^2 - y_2^2 + x_k^2 + y_k^2 \\ \cdots \\ -x_{k-1}^2 - y_{k-1}^2 + x_k^2 + y_k^2 \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} -2x_1 + 2x_k & -2y_1 + 2y_k & \frac{a_1(r_1 - r_k)}{5} \\ -2x_2 + 2x_k & -2y_2 + 2y_k & \frac{a_1(r_2 - r_k)}{5} \\ & \cdots & \\ -2x_{k-1} + 2x_k & -2y_{k-1} + 2y_k & \frac{a_1(r_{k-1} - r_k)}{5} \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} x_0 \\ y_0 \\ \frac{1}{n} \end{bmatrix}}_{\mathbf{x}} \tag{5}$$

where $k \geq 4$, and the k-th WiFi-PA is selected as reference for linearization. The final solution for $(x_0, y_0, \frac{1}{n})$ is given by:

$$\hat{\mathbf{x}} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{y}. \tag{6}$$

3.5 Occupancy Tracking

Our final goal is real-time occupancy monitoring, by counting the number of users within coarsely defined occupancy zones in the area of interest. To this end, past location estimates of users should be considered collectively for accurate occupancy tracking. The very simple idea of using new location estimates solely to draw a trajectory leads to wasting the history of RSS and the previous estimates. There are also some models which take advantage of previous RSS measurements and location estimates in time.

In this paper, a weighted k-means algorithm is used for tracking users, in order to take advantage of their past location estimates. As a tracking algorithm, k-nearest neighbour

1: **procedure** WKNN TRACKING
2: $\hat{\mathbf{x}}_k = (\mathbf{H}_k^T \mathbf{H}_k)^{-1} \mathbf{H}_k^T \mathbf{y}_k$ ▷ k-th location estimate
3: **for** $\forall t_i < T,\ i \in K$ **do**
4: $w_i = (T - t_i) / \sum_{i \in K} (T - t_i)$ ▷ Weight calculation
5: **end**
6: $\hat{\mathbf{u}}_k = \sum_{i \in K} w_i \hat{\mathbf{x}}_i / \sum_{i \in K} w_i$ ▷ Tracking result
7: $\arg_j \min(D_j),\ \forall D_j < L,\ j \in M$
8: **GridCell**$\leftarrow j$
9: **end procedure**

Figure 5: Pseudocode for the proposed WKNN technique for tracking mobile devices.

is well-known and easy to implement as shown in [46]. Different than [46], we consider here that the algorithm weights the k-nearest neighbour measurements with respect to measurements' time difference to the last one, thus it can be called as time-weighted k-nearest neighbour measurements (WKNN). While using WKNN, a time threshold is also considered to increase the accuracy and reliability of tracking via discarding obsolete estimates. After using WKNN algorithm, the final estimate is mapped to the nearest zone point this time with distance threshold. If the resultant point is further away than a threshold to a zone's center, it can not be mapped to that cell.

The pseudocode for the WKNN tracking algorithm is shown in Fig. 5 where $\hat{\mathbf{x}}_k$ is the kth location estimate, t_i is the time difference between ith measurement and kth measurement, T is the threshold for obsoleteness of measurement, K is the set of last k measurements, w_i is the weight of ith estimate, $\hat{\mathbf{u}}_k$ is updated sequence of location estimates after the tracking algorithm, D_j is the distance between tracking result and jth grid cell center, L is the distance threshold for mapping to a grid location, and M is the set of grid cells.

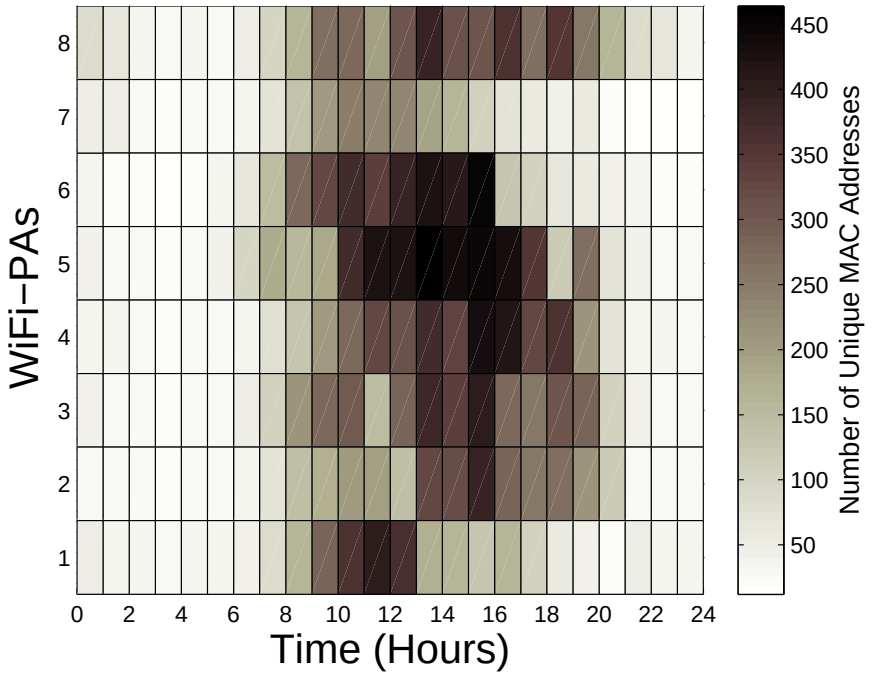


Figure 6: Number of probe requests captured from each WiFi-PA.

3.6 Numerical and Experimental Results

3.6.1 Occupancy Tracking

In this section, we will present our numerical results on occupancy tracking based on the collected and post-processed data. Counting the number of probe requests received by particular WiFi-PA would be helpful about understanding WiFi activity around that WiFi-PA. On the other hand, it might also be misleading since the number of probe requests can be increased solely by a particular device. For example, in our experiments we have seen that wireless printers may increase the number of probe requests heavily. For this reason, number of unique MAC addresses detected by a WiFi-PA gives a better hint about the number of WiFi devices and hence the occupancy of that region.

To this end, in Fig. 6, number of unique WiFi probe requests captured at each WiFi-PA is presented as a function of time. It can be observed that there may be over 400 unique

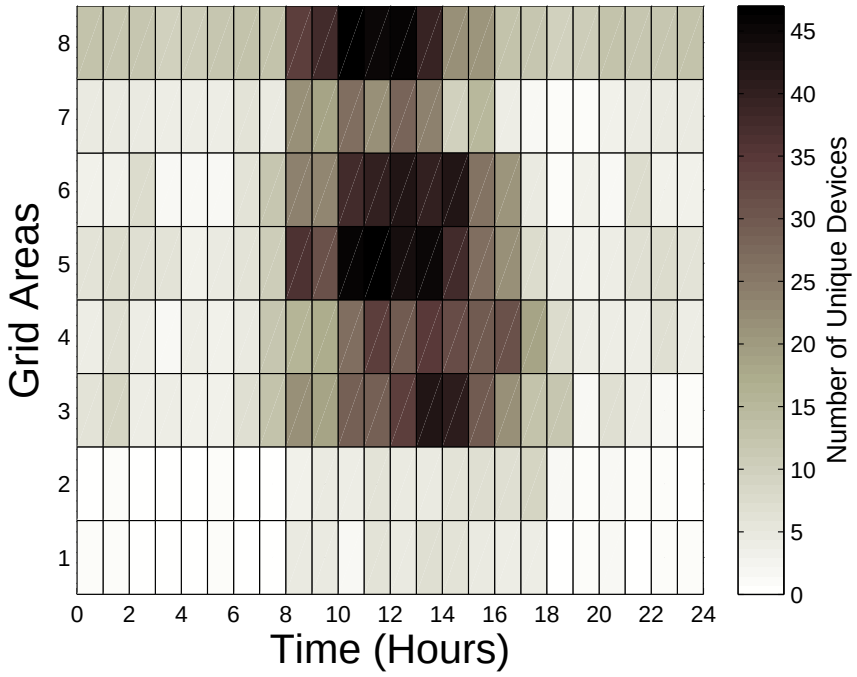


Figure 7: Detected number of people in gridded areas.

MAC addresses, captured within one hour at certain WiFi-PAs. Note that this number may also include devices that are in the vicinity of FIU EC building. In Fig. 7, we explicitly triangulate users into one of the eight zones in Fig. 1, which requires RSS information to be captured at least by 4 WiFi-PAs. While the number of unique detected devices is much lower compared to Fig. 6, Fig. 7 gives more reliable information about the occupancy pattern at different zones within the building. Fig. 8 provides occupancy pattern with one minute sampling interval between 12 PM to 12:30 PM, which can be used to extract fine-grained information for occupancy patterns.

In Fig. 9, total number of unique devices detected in gridded areas is given with respect to time, which shows that the peak hours of occupancy occurs between 10 AM and 2 PM. The number of detected devices drops to less than one fifth of peak hours after 4 PM. The difference between the number of devices detected and located can be explained with the help of Fig. 10. In the figure, the number of detected unique MACs throughout the day is

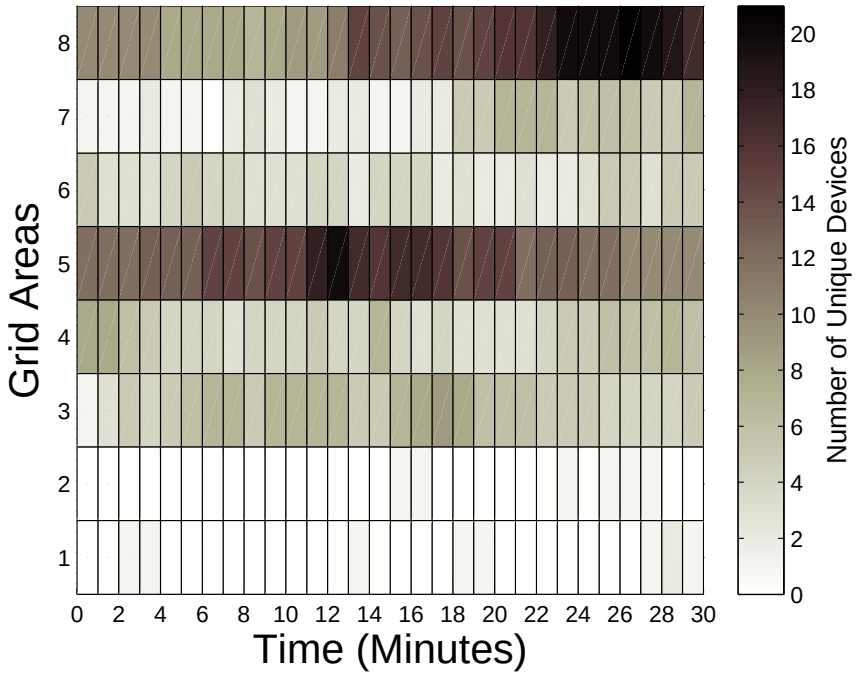


Figure 8: Number of total people in gridded areas between 12 PM to 12:30 PM.

given with respect to number of WiFi-PAs that have detected them. It is clear that more than half of devices are detected by only a single WiFi-PA. As stated earlier, the device needs to be detected by at least four different WiFi-PA to be included in occupancy monitoring.

3.6.2 MAC Randomization

WiFi-based occupancy tracking algorithms generally utilize RSS or connection information matching MAC addresses of devices [9]. Thus the MAC address randomization feature which is available as a privacy preserving method in certain commercial smart phones is investigated in this section. The first 24 bits of the MAC address is considered as the Organizationally Unique Identifier (OUI), which differentiates the devices according to their manufactures. The last 24 bits, on the other hand, are unique serial numbers assigned by manufacturers. The unique MAC address of devices can be easily captured and monitored from probe request messages since they are not encrypted. For example, owner of a net-

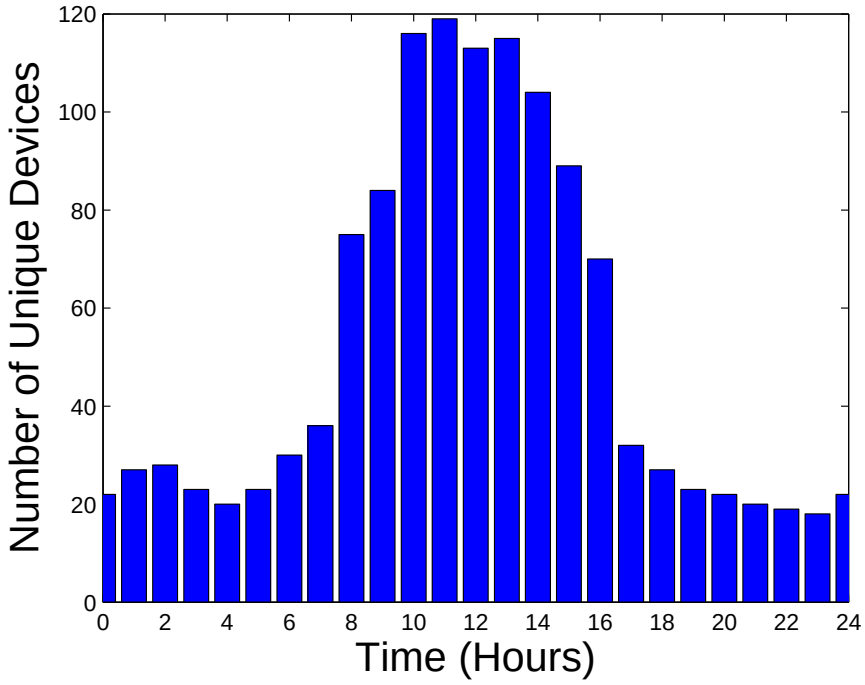


Figure 9: Number of total people in gridded areas per hour.

work can restrict certain MAC addresses not to connect to a network. Having a fixed MAC address also introduces several security and privacy issues as discussed in Section II. In order to address such privacy concerns, recently, Apple introduced the iOS 8 operating system at Worldwide Developers Conference (WWDC) on September 17th 2014. The most interesting feature in the new operating system was that MAC addresses will be randomized automatically making it difficult for an outsider to keep track of a user continuously.

We used Wireshark to capture probe requests of an iOS 8.4 Apple phone and to evaluate the MAC address randomization feature of iOS 8. In order to have a randomized MAC address, the location services should be switched off, and the phone must first go into sleep mode while WiFi is switched on and not associated to a particular WiFi network. When the devices goes into sleep mode, it takes around 150 seconds to send out randomized MACs, and keeps on sending the same randomized MACs at equal intervals. If the device screen is used and then goes into sleep mode again, iO8 will be sending probes with randomized

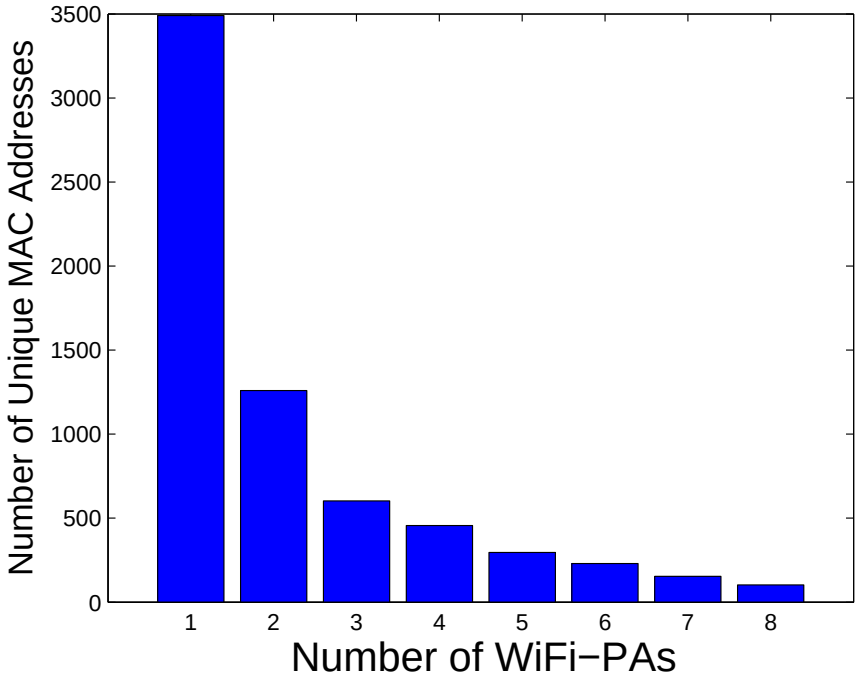


Figure 10: Histogram of number of WiFi-PAs that see unique MAC addresses.

MAC, which is different than the randomized MAC in previous cycle [47]. This feature helps in changing the MAC address at every sleep cycle.

Experimental results for the considered iOS 8 device are shown Fig. 11. The real MAC address of the phone is `dc:86:d8:b7:70:f3` and first 24 bits is the OUI of Apple as identified by Wireshark. After the phone satisfies the conditions described earlier, MAC address gets first randomized into `e6:59:5f:6c:f4:f5`. When we bring the phone back from sleep mode it again starts emitting the real MAC, and when it goes into sleep mode a new MAC `da:94:ba:e1:43:7a` appears, which is different from the earlier randomized MAC. After repeating this cycle, the randomized MAC is observed to change again into another different MAC `82:52:34:80:c3:3b`. The RSS information for static and randomized MACs are also seen to be similar, which can be used as an additional information for detection of randomized MACs. When the device was connected to a network and then goes into the sleep mode, no randomization of MACs are found; instead, iOS will be probing for the

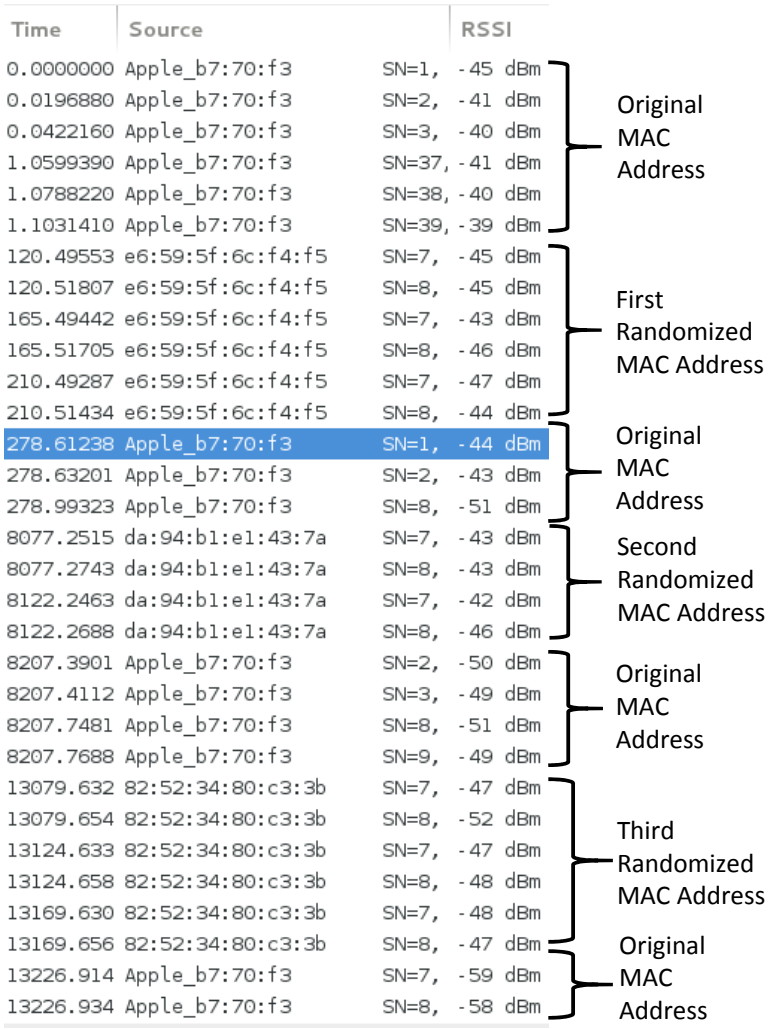


Figure 11: Randomization of MAC address at a smart phone with iOS 8 operating system.

previously connected network, as it gets disconnected with the network in sleep mode.

While the randomization of MAC addresses provides security to WiFi users, there are ways to identify randomized MACs. For example, MAC addresses must be registered with IEEE standards association [48], and Wireshark automatically filters out MAC addresses which do not comply with this criteria. Even though MAC randomization feature is not provided by the Android operating system, it can be achieved in *rooted* Android devices by manually changing the MAC addresses, or using specific apps [49]. In our study we determined 1464 randomized MAC addresses generated by 1639 identified Apple phones.

CHAPTER IV

Unmanned Aerial Vehicles in Cybersecurity, Privacy and Public Safety

In Section 4.1, recent FAA regulations related to use of drones are summarized. Section 4.2 reviews few of the major small commercial drones which are more pervasively available in the market. Section 4.4 discusses how drones can be used as cyber attack tools, while Section 4.3 is on vulnerabilities of drones to cyber attacks. Detection and interdiction of unauthorized and potentially malicious drones are discussed in Section 4.5. Section 4.6 lists some other applications of drones in smart cities, such as for search and rescue, forensics, and public safety communications

4.1 Recent Regulations on Drones

For maintaining the safety of manned aircrafts and the public, the Federal Aviation Administration (FAA) in the United States have developed rules to regulate the use of small UAVs. For example, FAA recently requires each small Unmanned Aircraft Systems (UAS) that weighs more than 0.55 lbs and below 55 lbs to be registered in their system[1]. If not complied with the FAA regulations, the owner of a drone can face civil and criminal penalties. The FAA has also released a smartphone application B4UFLY[2]. which provides drone users awareness about any restrictions in the location they are planning to operate the drone.

[1] http://www.faa.gov/uas/registration/

[2] http://www.faa.gov/uas/b4ufly

The *Know Before You Fly*[3]. campaign by FAA aims to educate public about UAV safety and responsibilities.

Overall, FAA regulations for small UAVs include flying them under 400 feet with no obstacles around, maintaining a line of sight with the UAV at all times, not flying UAVs within 5 miles from an airport unless permission is received from the airport and control tower, avoiding endangering of people or aircraft, and not flying near people and stadiums. FAA exempts public aircraft operations by issuing a Certificate of Waiver or Authorization (COA), which allows operators to use airspace for safety provisions. Most of the public uses of UAVs include firefighting, search and rescue, and disaster relief. For civil operations, FAA authorization can be received either by Section 333 Exemption (i.e., by issuing a COA), or by Special Airworthiness Certificate (SAC) in which applicants describe about their design, software development, and control, along with how and where they intend to fly. FAA also enforces Federal Aviation Regulations along with Law Enforcement Agencies (LEAs) to deter, detect, investigate, and stop unauthorized and unsafe UAV operations[4].

4.2 Review of Major Small UAVs

In this section, we will review and compare four commercially available and popular small UAVs, in terms of their basic features, wireless communications capabilities, and security vulnerabilities: Parrot AR Drone 2.0, Bebob Drone, Phantom 2 Vision Drone, and 3D Robotics Solo Drone.

[3]http://knowbeforeyoufly.org/

[4]https://www.faa.gov/uas/

4.2.1 Parrot AR Drone 2.0

Parrot AR Drone is built by a French company called Parrot and was first revealed in 2010. Subsequently, its version 2.0 was revealed at Consumer Electronics Show (CES) Las Vegas in 2012. It includes a 1 GHz 32 bit ARM Cortex A8 processor with 1 Gbit RAM and supports Wi-Fi b,g,n. It comes along with a Linux machine called BusyBox[5]., which has Unix tools in an executable file and can run over various operating system interfaces like Linux and Android. This drone can be controlled using various interfaces, like using an Android or iOS app to take images and videos to be stored in the phones and also using tools like LabVIEW which has a dedicated tool kit to control it called AR Drone Toolkit[6]. In [50], AR Drones were used in unknown areas for surveillance, for spying and to detect suspicious devices and objects using an extended Kalman filter. Drones can be controlled using various other programming languages such as Python, javascript, and node.js[7]. Due to their ease of programming, AR Drones have been very popular, and have been used in events such as Nodecopter[8] , where many developers come in a group of three, and are provided with AR Drones to showcase their work to other participants.

4.2.2 Bebop Drone

Bebop drone is also built by Parrot, and is much powerful compared to Parrot AR Drone 2.0. Bebop weighs around 400 g and it uses a P7 dual core CPU, quad-core GPU, and 8 GB of memory. It comes with a device called Skycontroller, which is used to improve its range.

[5]https://busybox.net/about.html

[6]https://ardronelabviewtoolkit.wordpress.com/

[7]https://github.com/felixge/node-ar-drone/

[8]http://www.nodecopter.com/

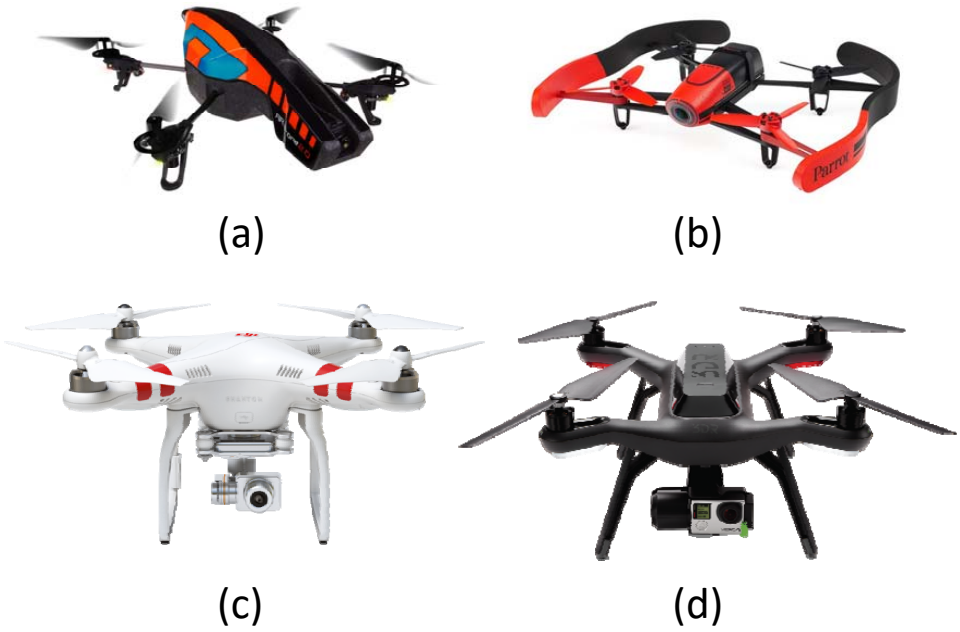


Figure 12: (a) Parrot AR Drone 2.0 [1], (b)Bebop Drone [2], (c) Phantom 2 Vision [3], and (d) 3D Robotics Solo [4].

One of the important features in the drone with respect to the drone regulations is that we can select our country in the network settings of the application, which helps us to fly the drone keeping the drone regulations easy to obey in the selected country. Bebop drone uses an IEEE WiFi radio type of 802.11ac, i.e. they can be operated both in 2.4 GHz and 5 GHz range. To maximize the range it is preferable to use 2.4 GHz WiFi and for better quality of videos it is preferable to use 5 GHz WiFi. Bebop's WiFi network uses an IP address of 192.168.42.1. This drone has an open FTP server which includes images, videos and black box readings. Therefore, flight recordings, GPS locations of the drone, among other information, can be easily accessed by an outsider due to open FTP server[9].

[9]https://github.com/cucx/bebop

4.2.3 Phantom 2 Vision Drone

Phantom 2 Vision is built by DJI, founded in 2006. It manufactures UAVs for aerial photography. This drone uses a mobile phone first person view (FPV) system with wireless connection range of up to 300 m, and supports 14 Megapixels image resolution. Receiver range of this drone is attained with the help of a range extender and is larger when compared to other drones. It uses a 2.4 GHz Direct Sequence Spread Spectrum (DSSS), within 2.4 GHz to 2.488 GHz band. Live video is available using the DJI Vision App, connecting the drone with the help of range extender. However, WiFi network has no security features, and WiFi network name cannot be changed because it should remain fixed to allow the drone to get associated with it. Range extender is a Linux machine that uses OpenWRT[10]., which always has an IP address of 192.168.1.2. Additionally, Phantom 2 Vision has two other Linux systems: One to access the SD card to obtain the images and recordings using IP 192.168.1.1, and other system is used for recording and encoding using IP address 192.168.1.10. Therefore, it is vulnerable to attacks which allows others to access data packets and GPS coordinates of the drone[11].

4.2.4 3D Robotics Solo Drone

Solo Drone is manufactured by 3D Robotics and is considered as the world's first smart drone with dual 1 GHz Linux computers: one on the drone and the other on the controller along with a GoPro attached to the drone. The HD video is obtained straight from the GoPro to 3DR solo application in an Android or an iOS phone. It uses a Pixhawk 2 au-

[10]https://openwrt.org/

[11]https://github.com/noahwilliamsson/dji-phantom-vision

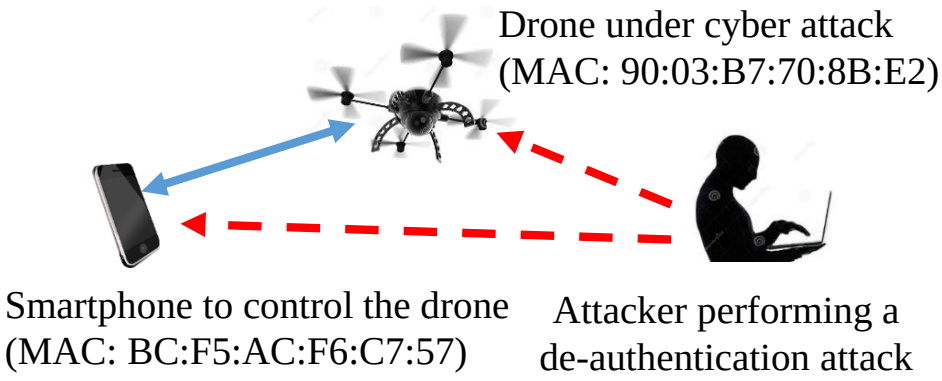


Figure 13: De-authentication attack on a drone.

topilot with a flight time of approximately 20 minutes[12]. This drone is considered to be more secure than the drones discussed in earlier sections, because of its secured password protection. Initially, when we try to access the drone we connect to the network beginning with SoloLink with an initial password *sololink*, which can be changed later. However, since we have to keep the network name as `SoloLink_` followed by alphanumeric characters, it makes it easy to understand that its the Solo drone's network. Therefore, while cyber hacking into the drone might be difficult, a de-authentication attack can be performed to disconnect the communication link between drone and the controller.

4.3 Cyber Attacks on Drones

Drones controlled by WiFi use IEEE 802.11 standards. All the communication between the drone and ground station controller typically use the WiFi network, which is vulnerable to security breaches. In particular, an unencrypted Wi-Fi used at a drone allows any individual to connect and hack the drone. For example, software such as Skyjack[13]. can be used to seek out and wirelessly control other drones within the range, and create an army of drones.

[12]https://3dr.com/solo-drone/

[13]http://samy.pl/skyjack/

In particular, SkyJack can detect all wireless networks and can deactivate clients connected to a drone, and then use node.js with node-ar-drone to control the drone. This drawback can be solved by using Wi-Fi protected access[14]., which provides password authentication to the drone and will not be easy for a hacker to gain access to the drone.

We have performed experiments with the help of a WiFi Pineapple[15]., AR Drone, Bebop and Phantom 2 Drone. Since these drones have WiFi, they create a wireless network which allows users to connect to them. These access point (AP) names are basically the drone name followed by the last two octets of the MAC address which makes it easy to find them; otherwise, they can be found by scanning, using the Organizational Unique Identifiers (OUI) which remains the same for all the drones from a given manufacturing company. Later, we try to connect to their network using the Linux command `iwconfig` and we can check our connection using `ping` command to their IP addresses. If we receive back an IP message we can confirm that our connection is successful and we have complete control over the drone. We perform these tasks using the WiFi Pineapple device which comes with boot switches, and can run the program as soon as the device is turned on. This helps in performing automated tasks, e.g., every time a drone is found, commands will be executed automatically.

4.3.1 De-authentication Attack

As shown in Fig. 13, we can perform a de-authentication attack on drones by using *aircrack-ng*[16]. Initially, a passive scan is made to search for the wireless network. After a network is found, using airodump-ng (from aircrack-ng), packets from only that particular wire-

[14]https://github.com/daraosn/ardrone-wpa2/

[15]http://hakshop.myshopify.com/products/wifi-pineapple

[16]http://www.aircrack-ng.org/

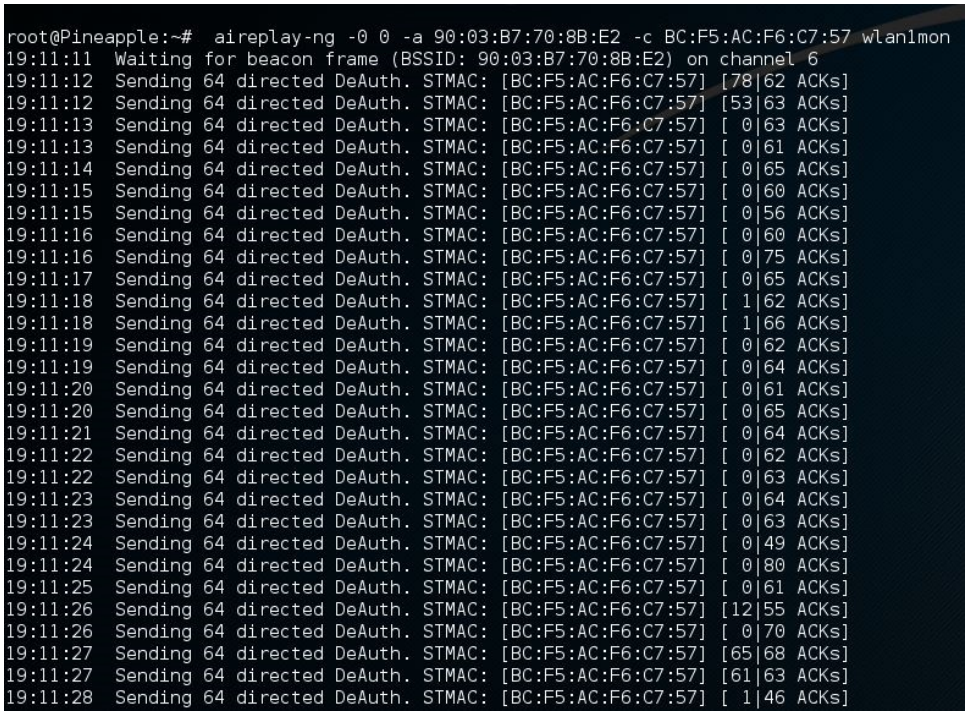


Figure 14: De-authentication packets have been set from an attacker to a Parrot AR Drone 2.0. The MAC address of the drone is `90:03:B7:70:8B:E2`, while the MAC address of the smartphone is `BC:F5:AC:F6:C7:57`.

less network can be filtered and stored. Then the list of clients associated with network is available and de-authentication attack can be performed. The clients are de-authenticated using aireplay-ng (again within aircrack-ng), which sends disassociate packets to connected clients, for disconnecting them from the AP.

In Fig. 14, a screen shot of the sent de-authentication packets are shown, where `90:03:B7:70:8B:E2` is the BSSID of the drone. In this experiment, aireplay-ng software is waiting to receive a beacon frame from the Parrot AR Drone which creates an AP for the clients to connect. After a beacon frame is received it starts sending disassociate packets to its connected clients. This results in the connection between the client (in this case `BC:F5:AC:F6:C7:57`) and the drone being lost. In this case, the disassociate packets are only sent to one particular

client. Even if we try to make the drone secure by using a WPA 2 authentication [17]., it is possible to disassociate them. We can also jam the complete AP network by continuously sending disassociation packets, disallowing anyone to connect to the network. Connection is regained later once the de-authentication packets are stopped being transmitted and when the client sends an association packet to the AP. Once the client gets disconnected from the drone due to the de-authentication attack, a link lost message is displayed.

The aireplay-ng command[18]. for the de-authentication attack firsts waits to receive beacon from the BSSID and then sends directed de-authentications as shown in Fig. 14. In particular, the command sends 128 deauth packets, out of which 64 packets are sent to the BSSID and 64 packets are sent to the selected client. In the last column of Fig. 14, first number represents the number of ACKs received from the client, while the second number represents the number of ACKs received from the UAV. The ACKs for the client can go above 64 packets when the client is actively participating with the BSSID or when ACKs from the previous packets is received. A number very smaller than 64 indicates the client is far away with weak signal strength, and zero value indicates that packets have not reached to the client.

4.3.2 GPS Spoofing Attack

GPS spoofing attack is another type of cyber attack commonly performed on drones. The communication links in drones include incoming signals from GPS satellites, signals notifying the drone's presence, and a two-way link between the ground station and the drone. GPS enables a drone's navigation, and due to no encryption of the signals they can be easily spoofed. In December 2011, Iranian forces claimed to have captured a Lockheed

[17]https://github.com/daraosn/ardrone-wpa2

[18]http://www.aircrack-ng.org/doku.php?id=deauthentication

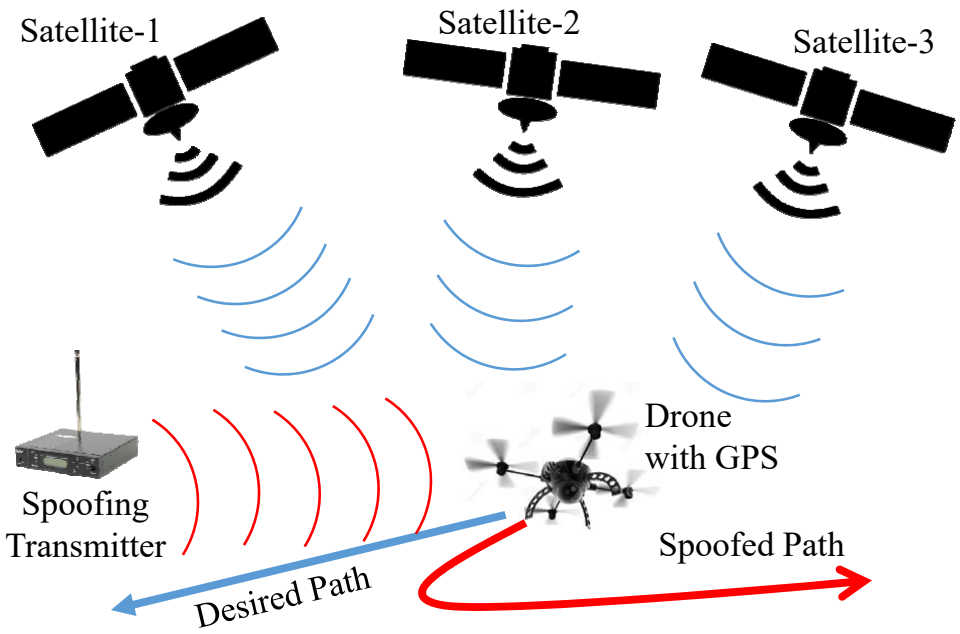


Figure 15: GPS spoofing attack Scenario, which changes the actual path of the Drone with a spoofed trajectory.

Martin RQ-170 Sentinel drone, operated by United States Air Force (USAF), and President Barack Obama asked for the return of the drone which was initially rejected by Iranian officials [51]. A major possibility that may have caused the loss of the drone is a cyber attack on the GPS system, which could be a GPS-spoofing attack.

The basic idea in GPS spoofing is transmitting fake GPS coordinates to the control system of the drone. This will hijack the drone and subsequently it will be in complete control of the attacker. For a GPS spoofing attack, a transmitter is used to transmit false GPS signals forcing the victim to synchronize with the attacker's signals. For example, in Fig. 15, three satellites are sending true GPS signals to the drone, constantly allowing it to fly in a desired path. An attacker can use a transmitter to send false GPS signals, which deters its path and sends the drone in a direction specified by the attacker. A successful attack is conducted when attacker is very close to the drone, or by using a directional antenna with narrow beamwidth aiming the drone. Due to no authentication mechanism, civilian drones can be attacked easily by delaying signals, while attacks on military drones

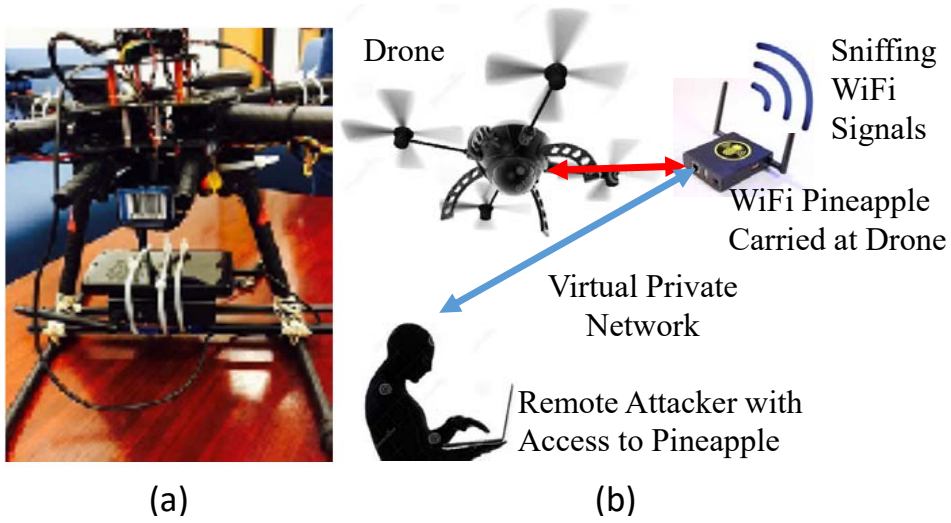


Figure 16: (a) Experimental setup with WiFi Pineapple mounted on a drone, and (b) Attack model using a WiFi Pineapple.

are complicated due to use of authentication mechanisms [52].

Researchers from UT-Austin were successfully able to demonstrate GPS spoofing attack, which can force a UAV to follow a trajectory set by the spoofer, proving that such an attack was technically and operationally feasible. For demonstration, initially the drone was made to hover in a particular location with the help of authentic signals. Later, spoofed signals were transmitted and they were aligned within the legitimate GPS signals received by the UAV. After overpowering the authentic GPS signals, spoofer changed the velocity and position of the UAV and another safety pilot was used to control it from drifting away [53, 54]. Recently in Defcon 23, 2013, a security researcher explained how to carry out GPS spoofing attacks [55] on cars and UAVs. Initially, GPS information is captured using USRP B210 software defined radio (SDR), and it is replayed back using a bladeRF SDR. Using these devices it is also possible to change the time and date of the GPS signal.

4.4 Drones as Cyber Attack Tools

Drones are vulnerable to many of the attacks but can also be used in malicious and harmful ways. As discussed in[19], a young teenager was charged to modify a drone to make it a fire handgun. In Defcon 21[20], a security researcher used a DJI Phantom mounted with a WiFi Pineapple to sniff wireless signals. Built-in tools like airdoump-ng, sslstrip were used to dissect the wireless data. The basic idea was to fly the drone and land it over a building or balcony for a particular time collecting data and bring it back to the starting position using return to home position. The connection with the WiFi Pineapple is maintained over a 3G connection by Reverse SSH tunnel, i.e. by creating an SSH relay server[21]. In this section, we will provide one particular example on the use of drones as cyber attack tools: sniffing signals using a virtual private network (VPN).

4.4.1 Sniffing Signals using a Virtual Private Network

In order to sniff WiFi signals, a WiFi Pineapple carried at a drone can be utilized. In our setup, we are mounting the WiFi Pineapple on the drone along with a smartphone as shown in Fig. 16(a), where the smartphone will provide Internet connection to WiFi Pineapple. There are multiple ways in which we can access the device. One of them is using an Ethernet cable connected to the device and the laptop. And another way is connecting it wirelessly by connecting to its AP within its WiFi range.

Since the device mounted on the drone will be flying at heights and distances that may be far away, it is not possible to connect using any of the above methods. Therefore, as

[19]http://www.smh.com.au/technology/web-culture

[20]https://www.defcon.org/images/defcon-21/dc-21-presentations/Hill/DEFCON-21.pdf

[21]https://hak5.org/episodes/hak5-1520

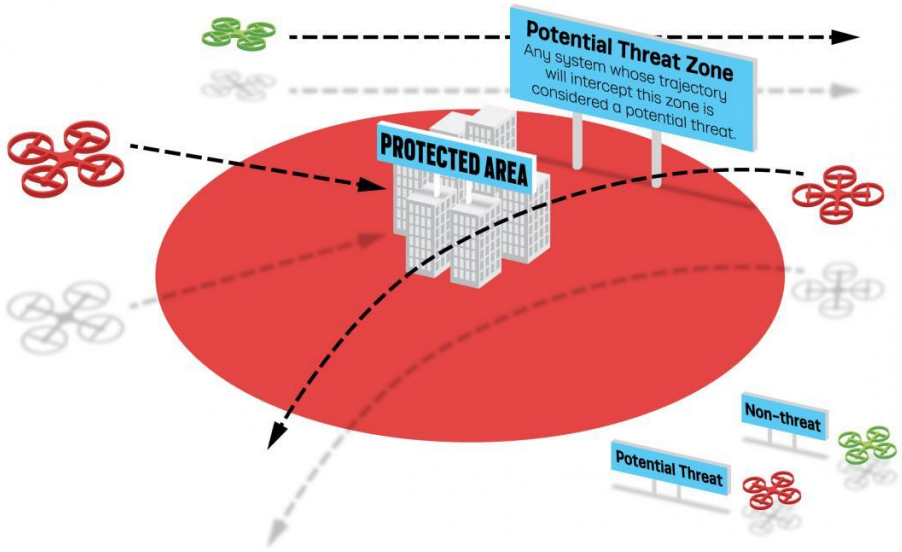


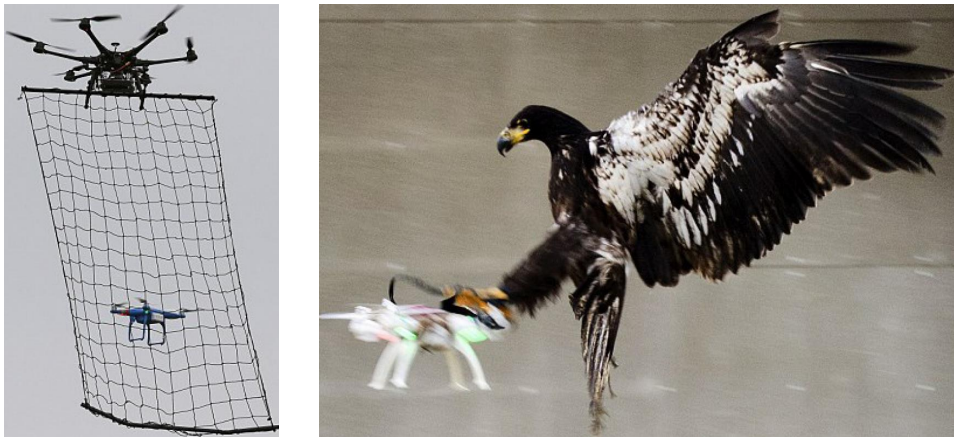
Figure 17: Threat model for UAVs in a protected area based on *MITRE Challenge* framework [5].

shown in Fig. 16(b), we created a VPN between the WiFi Pineapple and our own devices, thus making it possible to access the device from anywhere by just providing Internet connection to the WiFi Pineapple. VPN is basically used for client-server applications. The question might arise about the security of these devices and what if an intruder tries to intercept the server's communication. The OpenVPN[22]., which we are using to create a virtual private network, creates a symmetric key of size 2014 bits and is exchanged between the server and the client using Diffie-Hellman key exchange.

4.5 Interdiction of Unauthorized Drones

UAVs can be launched in any territory for constant surveillance and monitoring, and they can also be used to perform certain types of cyber/physical attacks that may harm property and civilians. As the cost of the drones is going down, the number of hobbyists and industries using them is increasing. Thus, it gets difficult to identify unauthorized drones

[22]https://wiki.archlinux.org/index.php/ OpenVPN



(a) (b)

Figure 18: Taking down malicious drones using (a) Net traps used by other larger drones [6], and (b) Eagle trained to catch a drone and place in a safe zone [7].

and bring down such drones when necessary. They can cause danger for civilian aircrafts, and there have been reports of near misses with UAVs and aircrafts, such as the near collision incidents with Boeing 737 in the British Airport[23]. It is critical that small UAVs are restricted to enter protected areas, and an unauthorized UAV that may be a potential threat should be detected, interdicted, and brought down in a safe zone. These have also been the major goal in *MITRE Challenge*, summarized in Fig. 17 [5], where the threat model shows how the trajectory of the drone can cause a threat. If the trajectory of the UAV is outside the protected area, it is considered to be safe. If the trajectory goes within the protected area it needs to be detected and brought down outside of the protected zone.

One way in which a drone could be brought down is by using another drone. In [6], Tokyo police created a drone squad for privacy breaches. Police department were handed over net-carrying drones that will trap suspicious drones flying in the vicinity as shown in Fig. 18. Before any attempt to trap the drone, suspicious drone operator will be warned and if refused will be trapped and questioned further.

[23]http://i-hls.com/2016/02/iata-chiefdrones-pose-realthreat-to-civilian-aviations

Cyber hacking one drone with the help of another drone is another possible alternative, because of the excess use of commercial drones using a WiFi network to control the flight. This makes it open to security vulnerabilities like de-authentication attacks or brute forcing into the drone's WiFi network. In [56], a Parrot AR Drone is interdicted using a Phantom Vision 2 equipped with a WiFi Pineapple. A shell script is used with command utility called as `empty`, which uses the concept of pseudo-terminals used on SSH sessions. Here, the Parrot drone acts as the slave and WiFi Pineapple at the Phantom drone acts as the master. Thus, as commands are entered in the master device it will be executed in the slave device, making it completely under the control of the WiFi Pineapple device.

As shown in Fig. 18, Scotland metropolitan police came up with an innovative idea of training eagles to take down suspicious drones [7]. This idea was very much successful as it was not required to use any other devices and was very cost effective with no danger to civilians. Eagles were made to consider drones as preys so that they catch them and place them in a safe area. In

4.6 Privacy, Forensics, Search and Rescue

One of the major concerns due to use of drones is the privacy. It is easy to mount a camera or a device to capture information, which may occasionally invade the privacy of people. To overcome such concerns, Center of Democracy and Technology (CDT) have asked FAA to issue rules on privacy and recommends on using data collection statement to know whether the information collected will be retained, used or disclosed. The most suitable means to maintain privacy was considered as Privacy by Design (PbD), which helps in maintaining standards and provides remedies for security breaches. By adopting PbD principles, privacy intrusion becomes limited and privacy can be ensured at an early stage [10].

Privacy was considered a major concern when a UAV goes beyond line of sight, but non line of sight operation is also typically required for UAVs, such as for search and rescue

or taking a survey of an area. In early 2012, when a person flew a UAV over a slaughter house and found a pipe going to the nearby rivers. While this was reported as a case for endangering public health, it also created complex issues since anyone can inspect such violations by overflights. Continuous use of such overflights can reveal trade secrets and if these photos go on the Internet, they can affect the business of companies [57].

Drones have various other applications in smart cities related to search and rescue, forensics, and public safety. During a war or in any emergency situation surveillance of a city becomes a challenging task. In such scenarios drones can be used conveniently to avoid human calamities. Most drones come with HD cameras which provide real time videos back to the ground station when they are flying. With advances in aerial technology, drones can be used in public safety communications. During natural disasters, there can be communication difficulties for public safety and in such times drones prove to be a viable solution in creating unmanned aerial base stations (UABs). Deploying drones in such scenarios can improve throughput coverage and help in saving lives [58, 59].

In [60], use of drones for spectrum monitoring and forensics have been studied. Spectrum measurements collected by drones can be compared to a database, which may help in detecting and solving the interference problems. A prototype was built in [60] using a DJI Phantom 2 to collect raw I/Q samples and obtain the spectrogram data which was used for signal identification. When the drone is launched it starts looking for interference, and when found it will try to localize the interferer. It will also change the flight plan to get next meaningful measurement point reducing flight time

Microdrones[24]. can be used for surveillance, search, and rescue operations. They are easy to deploy and have been tested under various climatic conditions. They provide live camera feed in disastrous situations such as wildfire or heavy snow, and can also carry thermal image cameras to identify people. To better the search and rescue operations

[24]https://www.microdrones.com/en/home/

with drones, in [61], algorithms have been taught to UAVs for autonomous flying through forests. The Search with Aerial RC Multirotor (SWARM)[25]. team have dedicated themselves in search and rescue operations with drones. This worldwide network is spread across 33 countries with over 600 drone pilots. A related organization is UAViators[26]., whose mission is "to promote the safe, coordinated and effective use of UAVs for data collection, payload delivery and communication services in a wide range of humanitarian and development settings".

[25]http://sardrones.org/

[26]http://uaviators.org/

CHAPTER V

UAV Localization Using Wifi Probe Requests

Aerial surveillance has been critical topic because of its various security applications. Drones are considered as favorites for air surveillance [62]. In worse conditions like flood, earthquake or fire a person might be completely obstructed or in severe danger. Under such conditions of limited visibility and clutter, it might be impossible to detect and track multiple people using only a camera or video to implement the occupancy tracking. In this report, we propose an approach of using Wi-Fi probe requests for occupancy tracking in an open field with the help of Wi-Fi pineapple equipment and with help of a drone.

Probe requests are frames that are continuously broadcast from devices with Wi-Fi technology, such as Laptop, Smartphones and tablets. When a Wi-Fi client seeks for any access point to get connected to the Internet, the first method is scanning for beacon frames, which are frames broadcasted by Wi-Fi routers to announce that they are available to Wi-Fi clients. And alternatively it sends probe requests which also contain the unique MAC address of the device, as well as its model, manufacture and type. We prefer probe request since the Wi-Fi client can initiate a connection to a Wi-Fi router instead of waiting for a beacon frame from the router. It‘s able to capture and decode the probe requests with a sniffer.

We are interested in using Wi-Fi Pineapple as a sniffer to collect the data in the probe request. This information can be used for occupancy counting in a certain field. In this paper, we use Wi-Fi probe request captured at a certain locations for occupancy tracking in

Figure 19: Mission Planner user interface

an opening field. We analyze the collected data which is a log using Wireshark and mission planner.

5.1 Mission Planner

Mission Planner is a ground control station for Plane, Copter and Rover [63]. Mission Planner user interface shown in Fig. 19 can be used as a configuration utility or as a dynamic control supplement for your autonomous vehicle. In Mission Planner, it is possible to analyze the telemetry logs from the sd card stored in the copter. Using an sd card reader inserted into a laptop, the telemetry logs will be accessible. In 5.2 it is described more on using Mission Planner in our experimentation.

Telemetry log includes all the communication messages between the ground station and copter can be seen data in the log. These messages are known as the MAVlink command messages which provides information about the position of its roll,pitch,yaw and other various informations about the drone. The most important information which was required in our experimentation is the GPS coordinates which is obtained from the telemetry logs.

The GPS coordinates includes the latitude, longitude and relative altitude which is with respect to the ground surface.

5.1.1 Sniffing Signals using a Virtual Private Network

In order to sniff WiFi signals, a WiFi Pineapple carried at a drone can be utilized. In our setup, we are mounting the WiFi Pineapple on the drone along with a smartphone as shown in Fig. 16(a), where the smartphone will provide Internet connection to WiFi Pineapple or else a USB modem can also be used to reduce payload. There are multiple ways in which we can access the device. One of them is using an Ethernet cable connected to the device and the laptop. And another way is connecting it wirelessly by connecting to its AP within its WiFi range.

Since the device mounted on the drone will be flying at heights and longer distances, it is not possible to connect using any of the above methods. Therefore, as shown in Fig. 16(b), we created a VPN between the WiFi Pineapple and our own devices, thus making it possible to access the device from anywhere by just providing Internet connection to the WiFi Pineapple. VPN is basically used for client-server applications. The question might arise about the security of these devices and what if an intruder tries to intercept the server's communication. The OpenVPN[1]., which we are using to create a virtual private network, creates a symmetric key of size 2014 bits and is exchanged between the server and the client using Diffie-Hellman key exchange and is expected to be very secure.

5.2 Capturing Probe Requests Data and GPS information using Mission Planner

WiFi Pineapple equipment are powered using a rechargeable battery that is connected to its DC port. As they are portable and light in weight, they are ideal to use in our experimenta-

[1]https://wiki.archlinux.org/index.php/ OpenVPN

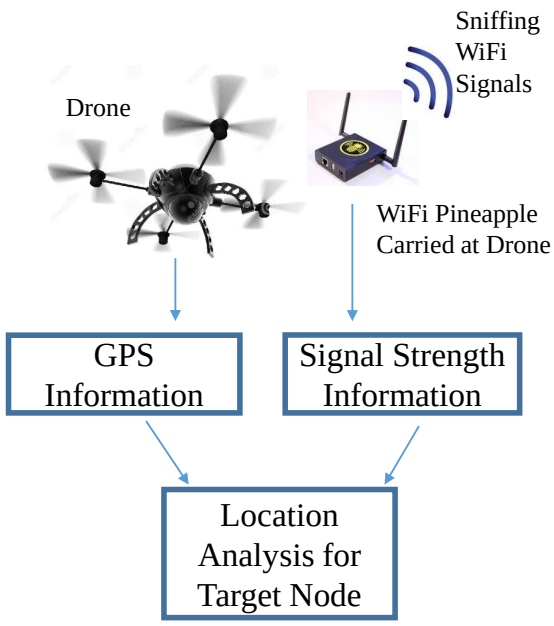


Figure 20: Tracking using UAVs

tion In WiFi Pineapple equipment we use the tcpdump sniffer to capture WiFi packets. We used a Tarot 650 UAV mounted with WiFi Pineapple Mark V and its rechargeable battery. As one end of the telemetry was on the drone other end was attached to a laptop via USB cable. The distance, the telemetry can maintain communication is approximately 1 mile, providing constant status about the UAV. We selected a random path for UAV and started flying by using a RC Controller. And the path was obtained from Mission planner as it helped us in creating a kml and gpx file which was later imported into Google earth.

Initially we also mounted a phone attached to the WiFi Pineapple equipment so that it stores the gps data from the phone to the pineapple and obtained data as shown in Fig. 20. Using an android application BlueNMEA, which sends location data over Bluetooth (RF-COMM) or TCP in the National Marine Electronics Association (NMEA) format GPS data was stored. So using gpsd command in Linux, GPS of the phone was stored directly in the sd card of the WiFi pineapple device. The information included timestamps, latitude and longitude. As, we moved ahead in the research we came across using the telemetry logs from the UAV to access the GPS information, which proved advantageous as the data

provided by it was more accurate which was found after comparing the GPS from the UAV and that of the phone. Also, it provided us altitude information which also made possible way for 3D localization. Another advantage was we no longer need to attach the phone with WiFi pineapple for GPS as it was available from the telemetry logs thus reducing the payload to a certain amount with only the WiFi Pineapple and it battery on the drone.

The scripts that we use to control each WiFi Pineapple can be executed via their terminal or using a web interface. But we used automated execution of scripts using Boot modes which is available in WiFi Pineapple which comes with five user configurable switches that help in execution of multiple commands using the terminal. Using the web interface, we specify which script to run on the equipment, and as soon as the device is booted and switches points out to our script and it starts executing commands. And as soon as the device receives Internet from an access point it sets it correct date and time using an NTP server.

After the whole experiment, it was now time for post processing. So, information from both the WiFi pineapple and drone was collected and converted into an excel sheet. And now the idea was to combine both the data and create a single excel sheet. As, probe request messages are not received every second and only at some intervals. GPS information for those timestamps of that of the probe requests were compared and a new data sheet with all the information's was created. This sheet was imported into Matlab to find reports about our experimentation.

In Fig. 21, various manufacturers found from various probe request messages are shown. Around 13 different manufacturers are found and 4 of the mostly seen ones are shown along with the random MAC address whose manufacturers are unknown in the figure.

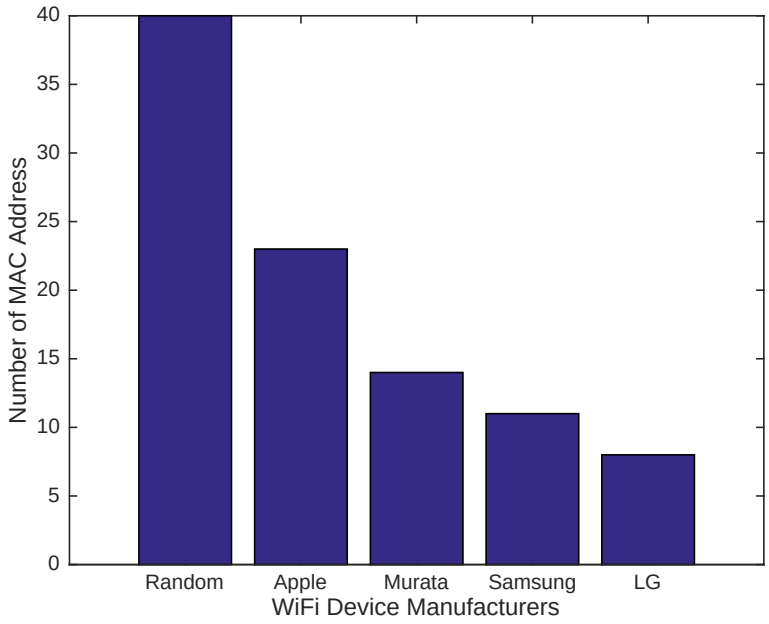


Figure 21: Unique Devices found from different Manufacturers.

5.3 Localization using Probe Requests

After pre-processing of the data, localization techniques are used to track experimental cell-phone. The Drone flies over the area and measures the received signal strength at different locations. The goal is to determine a 3D location of user, and signal power at the reference distance. The k_{th} measurement of received signal power is expressed as

$$P_{r,k} = P_o \left(\frac{d_k}{d_o} \right)^{-n}, \tag{7}$$

where P_o is the received power at the reference distance (d_o), d_k is the distance between Drone and the ground user in k_{th} measurement, n is the pathloss exponent which for outdoor air-to-ground communication is approximately 2. Without loss of generality we can assume $d_o = 1$ and hence $P_{r,k} = P_o d^{-\alpha}$. The true location of user is $L : (x, y, z)$ and the location of Drone at the k_{th} time instant is $U_k : (x_k, y_k, z_k)$. Therefore, $d = \sqrt{(x-x_k)^2 + (y-y_k)^2 + (z-z_k)^2}$.

In our model, the P_o and the position vector $L = [x, y, z]^T$ are unknown. To estimate

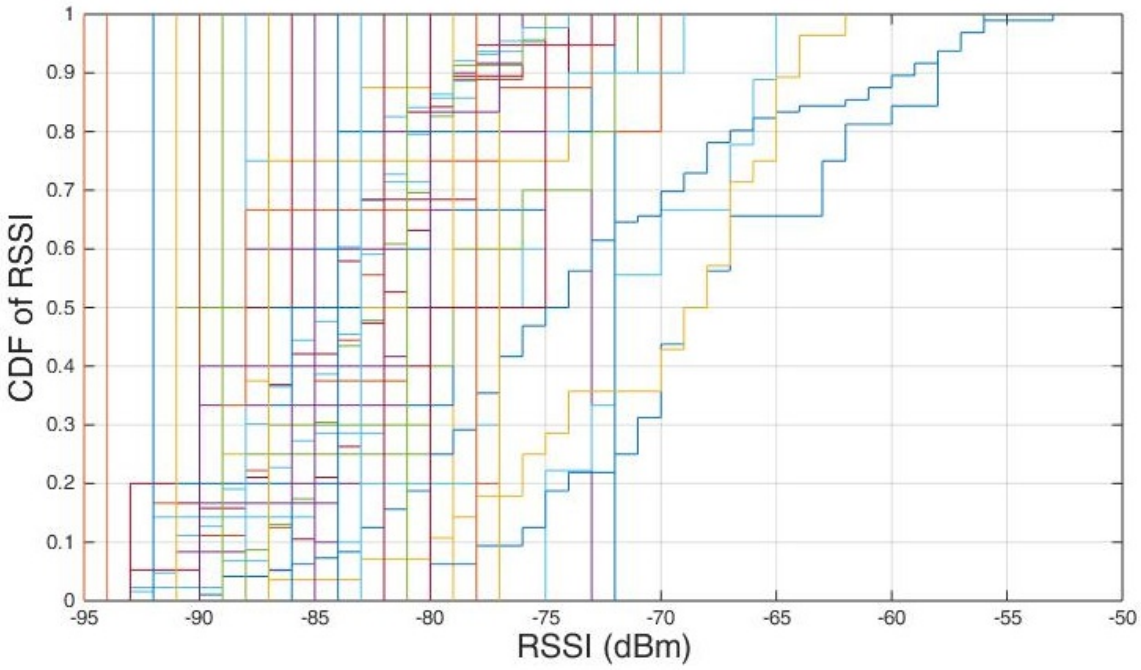


Figure 22: RSS measurements with respect to time.

P_o, we measured received signal power from a known reference location at different time instances. Then, given the distance between the user and the Drone (d_i) during N reference measurements, we estimate P_o as:

$$\hat{P}_o = \frac{1}{N}\sum_{i=1}^{N} P_{r,i} d_i^n. \tag{8}$$

Now, we estimate the position vector $L = [x, y, z]^T$ using least square (LS) estimation.

Consider y as the measurement vector and e the measurement error. Then we have

$$y = HL + e, \tag{9}$$

where L is the position state vector, and H is the design matrix which relates the measurements to the state vector.

From (7) we have:

$$d_k = 10^{\left(\hat{P}_o - P_{r,k}\right)/10n}. \tag{10}$$

As a result, as discussed in [44],

$$\begin{aligned} d_1^2 - d_k^2 &= x_1^2 + y_1^2 + z_1^2 - x_k^2 - y_k^2 - z_k^2 \\ &\quad - 2x(x_1 - x_k) - 2y(y_1 - y_k) - 2z(z_1 - z_k) \\ &= 10^{\left(\hat{P}_o - P_{r,1}\right)/5n} - 10^{\left(\hat{P}_o - P_{r,k}\right)/5n}. \end{aligned} \tag{11}$$

Note that, in (10) and (11), $\hat{P}_o$ and $P_{r,k}$ are in dBw scale.

Now, considering (9) and (11), for m measurements, H can be expressed as:

$$H = \begin{bmatrix} 2x_1 - 2x_2 & 2y_1 - 2y_2 & 2z_1 - 2z_2 \\ 2x_1 - 2x_3 & 2y_1 - 2y_3 & 2z_1 - 2z_3 \\ . & . & . \\ . & . & . \\ 2x_1 - 2x_m & 2y_1 - 2y_m & 2z_1 - 2z_m \end{bmatrix}. \tag{12}$$

Also, we can write the measurement vector as:

$$y = \begin{bmatrix} x_1^2 + y_1^2 + z_1^2 - x_2^2 - y_2^2 - z_2^2 + 10^{\left(\hat{P}_o - P_{r,2}\right)/5n} - 10^{\left(\hat{P}_o - P_{r,1}\right)/5n} \\ x_1^2 + y_1^2 + z_1^2 - x_3^2 - y_3^2 - z_3^2 + 10^{\left(\hat{P}_o - P_{r,3}\right)/5n} - 10^{\left(\hat{P}_o - P_{r,1}\right)/5n} \\ . \\ . \\ x_1^2 + y_1^2 + z_1^2 - x_m^2 - y_m^2 - z_m^2 + 10^{\left(\hat{P}_o - P_{r,m}\right)/5n} - 10^{\left(\hat{P}_o - P_{r,1}\right)/5n} \end{bmatrix}. \tag{13}$$

Finally, the estimation of the position vector is given by [64]:

$$\hat{L} = \left(H^T H\right)^{-1} H^T y. \tag{14}$$

Figure 23: Google Earth Image of the FIU Main Campus Ground.

5.4 Experimental Results

In this section, numerical results obtained from our captured and post-processed data is presented. We were able to obtain only few probe requests as our experimentation was done outdoors in an open field. The data was solely taken from the WiFi pineapple and GPS from the drone.

The flight was conducted in the FIU main campus ground with a flight time of around 10 minutes and the cumulative distribution function of RSS for the WiFi devices used for our experimentation is shown in Fig. 22 As shown most of measurements are between -90 dBm and -70 dBm which proves to be weak WiFi signals and proves us that those devices are in the vicinity of the ground but not inside the ground. This information will be useful while using for further procesing and RSS value smaller than -100 dBm can be neglected to improve the results. But two of the lines in the graph which is between -65 dBm and -55 dBm of those of that of the WiFi enabled devices used for our experiment. The location analytics is done using least square estimation and is shown in Fig. 23 in which the real location and experimental location is very closed to each other and is considered viable solution for search and rescue operations or finding lost hikers. There is an error of

around 14m between the real and experimental locations which we are debugging for more approximation. But in a large area an estimate is enough to find the lost users.

CHAPTER VI

Concluding Remarks

In this thesis, occupancy tracking is studied by passively monitoring WiFi probe requests captured from smart phones and tablets. We used a linear least squares technique to estimate the location of a mobile device based on probe request information captured at multiple reference locations, which is then mapped into a building zone for coarse level occupancy tracking. Our results show that probe requests can be a viable solution for occupancy monitoring in future smart buildings, which can have application such as energy management and surveillance.

Then, a detailed study about security of drones was conducted . As drones will be more pervasively used in future smart cities for communication and surveillance, and the need for security measures for drones is evident. A survey on some commonly used small UAVs, possible ways to provide cyber attacks on them, and potential interdiction mechanisms for malicious drones. We also demonstrated De-authentication attacks to drones, and VPN based sniffing using a WiFi Pineapple carried on a drone.

And at the end we performed experiments of localization in an outdoor environment and detail study about mission planner was made which help us in capturing various information's by passively monitoring WiFi probe requests captured from smart phones and tablets and using telemetry logs for GPS information. Our results show that probe requests can be a viable solution for occupancy monitoring and can be used in search and rescue operation or finding lost hikers.

Bibliography

[1] Parrot AR Drone 2.0. [Online]. Available: http://ardrone2.parrot.com/

[2] Bebop Drone. [Online]. Available: http://store.parrot.com/uk/accueil/336-bebop-drone.html

[3] Phantom 2 vision. [Online]. Available: https://www.dji.com/product/phantom-2-vision

[4] 3D robotics solo. [Online]. Available: https://3dr.com/solo-drone/

[5] (2016) The MITRE challenge: Countering unauthorized unmanned aircraft systems. [Online]. Available: http://www.mitre.org/research/mitre-challenge

[6] S. Liberatore, "How do you catch a drone? with an even bigger drone and a giant net," Dec. 2015. [Online]. Available: http://www.dailymail.co.uk/sciencetech/article-3356746

[7] T. Witherow, "Police set to use eagles to foil terrorist drone attacks," Feb. 2016. [Online]. Available: http://www.dailymail.co.uk/news/article-3436572

[8] V. Erickson, S. Achleitner, and A. Cerpa, "POEM: power-efficient occupancy-based energy management system," in *Proc. IEEE Information Processing in Sensor Networks (IPSN)*, Apr 2013, pp. 203–216.

[9] K. Akkaya, I. Guvenc, R. Aygun, N. Pala, and A. Kadri, "IoT-based occupancy monitoring techniques for energy-efficient smart buildings," in *Proc. IEEE Wireless Commun. Networking Conference Workshops (WCNCW)*, Mar. 2015, pp. 58–63.

[10] A. Cavoukian, *Privacy and drones: Unmanned aerial vehicles*. Information and Privacy Commissioner of Ontario, Canada, 2012.

[11] L. Demir, "Wi-Fi tracking: what about privacy," Ph.D. dissertation, M2 SCCI Security, Cryptologyand Coding of Information-UFR IMAG, 2013.

[12] J. Freudiger, "How talkative is your mobile device?: An experimental study of Wi-Fi probe requests," in *Proc. ACM Conf. Security and Privacy in Wireless and Mobile Networks*, New York, NY, USA, 2015, pp. 8:1–8:6. [Online]. Available: http://doi.acm.org/10.1145/2766498.2766517

[13] A. B. M. Musa and J. Eriksson, "Tracking unmodified smartphones using Wi-fi monitors," in *Proc. ACM Conf. Embedded Network Sensor Systems*, ser. SenSys '12. New York, NY, USA: ACM, 2012, pp. 281–294. [Online]. Available: http://doi.acm.org/10.1145/2426656.2426685

[14] D. Namiot and M. Sneps-Sneppe, "On the analysis of statistics of mobile visitors," *Automatic Control and Computer Sciences*, vol. 48, no. 3, pp. 150–158, 2014.

[15] D. Namiot, "On mining mobile users by monitoring logs," in *Proc. Information Access in Smart Cities Workshop (i-ASC)*, Apr. 2014. [Online]. Available: http://dcs.gla.ac.uk/workshops/iASC2014/papers/iasc2014_namiot.pdf

[16] Hak5. (2013) Wi-Fi Pineapple Mark V. [Online]. Available: http://hakshop.myshopify.com/products/wifi-pineapple

[17] M. Asadpour, B. Van den Bergh, D. Giustiniano, K. A. Hummel, S. Pollin, and B. Plattner, "Micro aerial vehicle networks: An experimental analysis of challenges and opportunities," *IEEE Commun. Mag.*, pp. 1–11, 2014.

[18] İ. Bekmezci, O. K. Sahingoz, and Ş. Temel, "Flying ad-hoc networks (FANETs): a survey," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270, 2013.

[19] J. M. Sullivan, "Revolution or evolution? The rise of the UAVs," in *Proc. Int. Symp. Technology and Society Weapons and Wires: Prevention and Safety in a Time of Fear (ISTAS)*, 2005, pp. 94–101.

[20] J. Villasenor, "Drones and the future of domestic aviation," *Proceedings of the IEEE*, vol. 102, no. 3, pp. 235–238, Mar. 2014.

[21] Z. Sun, P. Wang, M. C. Vuran, M. A. Al-Rodhaan, A. M. Al-Dhelaan, and I. F. Akyildiz, "BorderSense: Border patrol through advanced wireless sensor networks," *Ad Hoc Networks*, vol. 9, no. 3, pp. 468–477, 2011.

[22] C. Barrado, R. Messeguer, J. López, E. Pastor, E. Santamaria, and P. Royo, "Wildfire monitoring using a mixed air-ground mobile network," *IEEE Pervasive Computing*, vol. 9, no. 4, pp. 24–32, 2010.

[23] I. Maza, F. Caballero, J. Capitán, J. Martínez-de Dios, and A. Ollero, "Experimental results in multi-UAV coordination for disaster management and civil security applications," *J. Intelligent & Robotic Systems*, vol. 61, no. 1-4, pp. 563–585, 2011.

[24] E. Semsch, M. Jakob, D. Pavlicek, and M. Pechoucek, "Autonomous UAV surveillance in complex urban environments," in *IEEE Int. Conf. Web Intelligence and Intelligent Agent Technologies (WI-IAT)*, vol. 2, 2009, pp. 82–85.

[25] H. Xiang and L. Tian, "Development of a low-cost agricultural remote sensing system based on an autonomous unmanned aerial vehicle (UAV)," *Biosystems Engineering*, vol. 108, no. 2, pp. 174–190, 2011.

[26] "Amazon Prime Air." [Online]. Available: http://www.amazon.com/b?node=8037720011

[27] I. Bucaille, S. Hethuin, T. Rasheed, A. Munari, R. Hermenier, and S. Allsopp, "Rapidly deployable network for tactical applications: Aerial base station with opportunistic links for unattended and temporary events ABSOLUTE example," in *Proc. Military Commun. Conf.*, Nov. 2013, pp. 1116–1120.

[28] E. Vattapparamban, B. Ciftler, I. Guvenc, K. Akkaya, and A. Kadri, "Indoor occupancy tracking in smart buildings using wi-fi probe requests," in *Proc. IEEE International Commun. Conference Workshops (ICCW)*, 2015, (Submitted).

[29] E. Vattapparamban, I. Guvenc, A. Yurekli, K. Akkaya, and S. Uluagac, "Drones for smart cities: Issues in cybersecurity, privacy and publci safety," in *Proc. IEEE International Wireless Communications & Mobile Computing Conference (IWCMC)*, 2016, (Submitted).

[30] Cisco, "Location analytics," Mar. 2015. [Online]. Available: https://meraki.cisco.com/lib/pdf/meraki_whitepaper_cmx.pdf

[31] Euclid Analytics. [Online]. Available: http://euclidanalytics.com/

[32] Libelium. [Online]. Available: http://www.libelium.com/company/

[33] W. Wang, R. Joshi, A. Kulkarni, W. K. Leong, and B. Leong, "Feasibility study of mobile phone WiFi detection in aerial search and rescue operations," in *Proc. Asia-Pacific Workshop on Systems*, ser. APSys '13. New York, NY, USA: ACM, 2013, pp. 7:1–7:6. [Online]. Available: http://doi.acm.org/10.1145/2500727.2500729

[34] M. Handte, M. U. Iqbal, S. Wagner, W. Apolinarski, P. J. Marrón, E. M. M. Navarro, S. Martinez, S. I. Barthelemy, and M. G. Fernández, "Crowd density estimation for public transport vehicles," in *Proc. EDBT/ICDT Workshops*, 2014, pp. 315–322.

[35] D. Nix, "Analysis of methods for mobile device tracking," *Technical Report*, Oct. 2013. [Online]. Available: https://www.eyeqinsights.com/wp-content/uploads/2013/10/eyeQ-Mobile-Device-Tracking-Study-Nix.pdf

[36] Z. Xu, K. Sandrasegaran, X. Kong, X. Zhu, J. Zhao, B. Hu, and C.-C. Lin, "Pedestrain monitoring system using Wi-Fi technology and RSSI based localization," *Journal of Wireless & Mobile Networks*, vol. 5, no. 4, Aug. 2013.

[37] L. Demir, M. Cunche, and C. Lauradoux, "Analysing the privacy policies of Wi-Fi trackers," in *Proc. ACM Workshop on Physical Analytics*, 2014, pp. 39–44.

[38] T. Kropeit, "Dont trust open hotspots: Wi-Fi hacker detection and privacy protection via smartphone," *BS Thesis*, Mar. 2015. [Online]. Available: https://www.emsec.rub.de/media/attachments/files/2015/03/BA_Kropeit.pdf

[39] D. N. Ratnayake, H. B. Kazemian, and S. A. Yusuf, "Identification of probe request attacks in WLANs using neural networks," *Neural Computing and Applications*, vol. 25, no. 1, pp. 1–14, 2014.

[40] M. Cunche, M.-A. Kaafar, and R. Boreli, "Linking wireless devices using information contained in Wi-Fi probe requests," *Pervasive and Mobile Computing*, vol. 11, pp. 56–69, 2014.

[41] B. Bonné, P. Quax, and W. Lamotte, "Your mobile phone is a traitor!–raising awareness on ubiquitous privacy issues with SASQUATCH," *International Journal on Information Technologies Security*, no. 3, 2014. [Online]. Available: https://brambonne.com/docs/bonne14sasquatch.pdf

[42] L. Demir, "Wi-fi tracking : what about privacy. mobile computing," 2013. [Online]. Available: https://hal.inria.fr/hal-00859013

[43] Wireshark. [Online]. Available: https://www.wireshark.org/

[44] J. Koo and H. Cha, "Localizing WiFi access points using signal strength," *IEEE Commun. Lett.*, vol. 15, no. 2, pp. 187–189, Feb. 2011.

[45] G. Lui, T. Gallagher, B. Li, A. Dempster, and C. Rizos, "Differences in RSSI readings made by different Wi-Fi chipsets: A limitation of WLAN localization," in *Proc. Int. Conf. Loc. and GNSS (ICL-GNSS)*, June 2011, pp. 53–57.

[46] J. Figueiras and S. Frattasi, *Mobile positioning and tracking: from conventional to cooperative techniques*. John Wiley & Sons, 2011.

[47] B. Misra. (2014, Sep.) iOS8 MAC Randomization. [Online]. Available: http://blog.airtightnetworks.com/ios8-mac-randomization-analyzed/

[48] IEEE, "MA-L PUBLIC LISTING," 2015. [Online]. Available: http://standards.ieee.org/develop/regauth/oui/public.html

[49] O. Abukmail. (2015, Sep.) WiFi mac address changer. [Online]. Available: https://play.google.com/store/apps/details?id=com.wireless.macchanger&hl=en

[50] M. Ma'sum, M. Arrofi, G. Jati, F. Arifin, M. Kurniawan, P. Mursanto, and W. Jatmiko, "Simulation of intelligent Unmanned Aerial Vehicle (UAV) for military surveillance," in *Proc. Int. Conf. Advanced Computer Science and Information Syst. (ICACSIS)*, Sep. 2013, pp. 161–166.

[51] CNN Wire Staff, "Obama says U.S. has asked Iran to return drone aircaft," Dec. 2011. [Online]. Available: http://www.cnn.com/2011/12/12/world/meast/iran-us-drone/

[52] "Hacking drones: Overview of the main threats," Jun. 2013. [Online]. Available: http://resources.infosecinstitute.com/hacking-drones-overview-of-the-main-threats/

[53] A. J. Kerns, D. P. Shepard, J. A. Bhatti, and T. E. Humphreys, "Unmanned aircraft capture and control via GPS spoofing," *Journal of Field Robotics*, vol. 31, no. 4, pp. 617–636, 2014.

[54] D. P. Shepard, J. A. Bhatti, T. E. Humphreys, and A. A. Fansler, "Evaluation of smart grid and civilian UAV vulnerability to GPS spoofing attacks," in *Proceedings of the ION GNSS Meeting*, vol. 3, 2012.

[55] H. Lin and Y. Qing, "GPS spoofing," 2013. [Online]. Available: https://media.defcon.org/DEF%20CON%2023/DEF%20CON%2023%20presentations/DEFCON-23-Lin-Huang-Qing-Yang-GPS-Spoofing.pdf

[56] D. Kitchen, "Drones hacking drones," HAK5, Dec. 2013. [Online]. Available: https://hak5.org/episodes/hak5-1518

[57] J. Villasenor, "Observations from above: unmanned aircraft systems and privacy," *Harv. JL & Pub. Pol'y*, vol. 36, p. 457, 2013.

[58] A. Merwaday and I. Guvenc, "UAV assisted heterogeneous networks for public safety communications," in *Proc. IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, 2015, pp. 329–334.

[59] K. Guevara, M. Rodriguez, N. Gallo, G. Velasco, K. Vasudeva, and I. Guvenc, "UAV-based GSM network for public safety communications," in *Proc. IEEE Conf. Southeast*, 2015, pp. 1–2.

[60] A. Anderson, X. Wang, K. R. Baker, and D. Grunwald, "Systems for spectrum forensics," in *Proceedings of the 2nd International Workshop on Hot Topics in Wireless*. ACM, 2015, pp. 26–30.

[61] A. Brokaw, "Autonomous search-and rescue drones outperform humans at navigating forest trails," Feb 2016. [Online]. Available: http://www.theverge.com/2016/2/11/10965414/autonomous-drones-deep-learning-navigation-mapping

[62] A. Birk, B. Wiggerich, H. Blow, M. Pfingsthorn, and S. Schwertfeger, "Safety, security, and rescue missions with an unmanned aerial vehicle (uav)," *Journal of Intelligent & Robotic Systems*, vol. 64, no. 1, pp. 57–76, 2011. [Online]. Available: http://dx.doi.org/10.1007/s10846-011-9546-8

[63] Mission planner. [Online]. Available: http://planner.ardupilot.com/

[64] A. Björck, *Numerical methods for least squares problems*. Siam, 1996.