Complete Takeover of Unmanned Aerial Vehicles (UAVs)

Samuel Clay

December 3, 2023

Abstract

This paper aims to provide an updated description of the state of UAV security, especially when it comes to controlling them. In this paper, WiFi and GPS based attacks are considered and a potential solution is proposed with future work to be done in increasing the security of UAVs even further.

1 Introduction

UAVs are becoming more prevalent in society. They are used by hobbyists as a form of recreation and to collect scenic photos. Companies such as Amazon are making strides to deliver packages and goods via UAVs. The United States Military makes use of UAVs for reconnaissance and automated air strikes. These devices are among the most powerful IoT devices that exist because they can directly and physically affect their surroundings. Malicious use of these devices could result in invasion of privacy, injury, endangering the safety of larger Aerial Vehicles, and even loss of life.[3] It is imperative, therefore that great care must be taken into securely operating these devices and in keeping attackers out.

This paper is focused on analyzing potential attack vectors that malicious actors can use to take control of UAVs and suggest alternative measures in order to mitigate potential attacks. Specifically, this paper analyzes WiFi-based communications as well as a GPS side-channel attack. Due to the constraints of limited resources, this paper only analyzes the security of one device, but it applies general mitigations that should work for nearly any UAV. When it comes to hardening the system against GPS based attacks, this paper will not focus on improving the protocol, but instead on improving a device's use of GPS to make it harder to attack.

2 Related Work

Similar work has been done in this field, but most sources are from 2016. Due to the rapidly advancing nature of technology, these sources may be considered out-dated. Still, these sources do outline several weak spots to look out for when performing security analysis on UAVs. Common exploits include: Open ports directly allowing access to a root shell without authentication, controller spoofing, and GPS Spoofing. [4]

The first exploit is relatively straightforward. Most Commercial UAVs also function as as wireless access point with no authentication. This allows essentially anyone to connect to any drone and start poking around. A simple nmap scan can usually reveal something like such:

```
PORT    STATE SERVICE
23/tcp  open  telnet
```

This telnet port has no authentication and the client instantly has root access. This was a major security flaw for a few drone models and allows any attacker complete control over the drone in question.

The next exploit deals with the aspect of pretending to be the controller itself. Some controllers communicate with the drone via WiFi-direct. While there still is little to no authentication to connect to the drone via WiFi-direct, the controller still sets up a handshake with the drone to establish that it is this drone's controller and will issue commands. After this handshake, the drone will accept commands coming from the controller. Once this handshake is set up, an attacker can spoof packets as if they come from the controller to then control the drone from an unauthenticated source. [5]

The third exploit is not specific to UAVs, but rather to any device using the public GPS protocol. GPS is the most popular unencrypted protocol in use and as such any attacks targeting GPS are powerful since they can be used on a wide array of devices. One such attack against a UAV flying with GPS waypoints would be to simply spoof the GPS signal. This is a trivial attack, and can make the UAV believe that it is in a location that it is not. As such, the UAV may perform corrective actions to conform to its preprogrammed path. This attack allows a malicious actor to make the UAV fly toward a new destination while the UAV's systems all show that it is heading toward its programmed path. This very attack was used by Drug Cartels along the US-Mexican border to evade detection by UAVs and cross the border illegally in order to traffic drugs [6]

These exploits were viable several years ago. Since then, drone manufacturers have made improvements to the security and quality of drones and UAVs. Now this paper serves to provide an update on the state of drone security and viable exploits.

3 Methodology and Results

This paper attempts to test all possible exploits possible over the Wifi protocol and the GPS protocol on a specific UAV and generalizes the data as much as possible. The UAV being tested is a ScharkSpark GPS Drone. At the time of testing, it was rated the Top Pick on Amazon under the query "gps drone". This UAV represents the mid-range capable drone for hobbyists. This paper is thus limited in its conclusions and analysis by resources since multiple devices were not tested, and different calibers of UAVs (specifically Commercial and Military) are out of reach. (Though one may speculate that commercial UAVs are quite similar to consumer UAVs in their capabilities, modes of operation, and protocols)

The ScharkSpark GPS Drone connects to 2 controllers simultaneously: the rf controller that comes in the same package, and the user's smartphone via an app over WiFi-direct. The package also claims that only the first connected smartphone can send commands to the drone. Thus the list of analyzed attacks is as follows:

- Deauthentication attack A malicious actor deauthenticates the original controller and connects a second device before the first has a chance to reconnect. This will make the drone only listen to commands sent by the second device.

- Port Command Any exposed ports will be scanned and penetration tested to find any potential backdoors.

- Packet Capture Since anyone can connect to the phone, data can be freely scanned and interpreted using tools.
- GPS Spoof Side-channel attack where the drone is not commanded directly, but is directed to non-commanded areas by interfering with its sensors.

The deauthentication attack was performed with aireplay-ng. Several attempts were made to deauthenticate the user's smartphone controller using different tools, packet-injection capable WiFi adapters, and in different modes of operation. None of them worked. It appears that the drones made today (as are most devices) are resilient to this type of attack. In fact, the WiFi standard IEEE 802.11w-2009 protects against such attacks and it appears to be implemented here as well. This protects against Evil twin attacks such as the deuthentication attack meaning that an attacker cannot simply swap what smartphone controller the drone listens to.

As far as commanding via exposed ports goes, a comprehensive nmap scan was performed against the drone on the entire port range on TCP and UDP. The output of this scan showed that only port 2222 was open. Further analysis and packet capture revealed that this port was used to transmit commands and a live video feed from the drone. The drone was completely closed off to backdoor access via WiFi. However, one thing an attacker is able to do is to also view the video feed. This is a potential privacy risk which could be solved by adding a layer of authentication to connecting to the drone. Therefore, while the drone may be adequately protected from takeover, it is not adequately protected against spying from users within the drone's WiFi range.

GPS Spoofing, the last attack, was the hardest to implement. According to the United States Communications Act of 1934, in Section 333 "No person shall willfully or maliciously interfere with or cause interference to any radio communications of any station licensed or authorized by or under this Act or operated by the United States Government." [2]. Since GPS is a publicly broadcast protocol, there are no barriers to protection against a GPS spoof attack. Additionally most cars, other vehicles, and phones use GPS. Based on the minimum transmit power of my GPS transmitter in order to test this without violating the law, I needed to move to a secluded location at least 1km away from any device that this test could potentially interfere with. This made testing GPS Spoofing even more limited.

By downloading publicly available ephemeris data of the GPS satellites I was able to use existing software to calculate the constellation map of how the satellites should be aligned at a certain locations. I was then able to construct a binary file based on the ping data the hypothetically visible GPS satellites would say. I then used a Software-Defined Radio to transmit the data and trick various devices such as my own cellphone to think that it was in a different location than it really was in. This works because even though the GPS satellites are still working normally, my transmitter at close range is able to transmit a signal much louder to the nearby device than the satellites many thousands of miles away can. In essence the GPS satellites' signals are drowned by the sheer loudness of my signal. I also tested this against the drone at various distances. I spoofed three different locations. The first location I spoofed was the coordinate (0 N, 0 E). This coordinate is referred to as Null Island since in many GPS-based errors devices will default to sending back zeros as coordinates. In response to this, the drone shut off its GPS receiver and changed mode back manual directional control. This is a great defense against GPS spoof, and to test its limits I spoofed other locations. The next location I spoofed was Miami, Florida, a location still over a thousand miles away. The drone once again detected this massive "teleportation" and returned control once again to manual instead of automatic. The final spoof was a much smaller teleportation. I spoofed a location 500 feet away from the drone. This time the drone did not detect this teleportation as absurd and it kept control to the GPS sensor. Later tests could not be performed physically because after the last test, the GPS receiver broke. This may have been due to the signal strength being too much for the drone to handle. Thus I continued with simulated attacks against a digital drone.

This drone is not completely defenseless against GPS spoofing attacks. It detected whether or not a GPS signal made the drone jump too far in location and shut down GPS based control when the discrepancy was too high. This works to an extent, but more sophisticated attacks could spoof a location within the drone's range of acceptability and then "drift" the signal until the coordinates received drastically changes the drone's location. Several detection methods have been proposed to counteract this such as detecting signal strength, minimizing use of gps, and using an encrypted GPS standard. Detecting if signal strength is too loud runs into much of the same problems as detecting if the teleportation distance is too far. Both methods are unreliable since in both cases there will be at least some variation in signal strength and some teleportation corrections due to noise in measurement and signals bouncing off of surrounding landscape and clouds. Using an Encrypted GPS standard would require deploying decryption keys to every device that uses GPS. This would make the decryption useless as everyone would use the same keys.

There is however, a way to counteract this. Many manned aerial vehicles (MAV) not only use GPS, but they also have other forms of measurement to approximate actual location when GPS is unreliable. This fix is viable for UAVs that must carry some sort of payload as their capacity to carry velocity-detecting equipment is much higher than for small hobbyist drones that need to keep weight to a strict minimum.

For this fix I used a Kalman filter to fuse the kinematic data from the velocity sensor and the GPS sensor into a single estimate for the drone's actual position. I used a Kalman filter because it returns a weighted average of the sensors based on variance. Essentially as the drone flies the Kalman filter learns which sensors are acting erratically and weighs them less since the variance of the sensor is high and therefore has less trust. The results of the simulation can be seen in the graph.

4 Future Work and Conclusion

After conducting the simulation it becomes clear that using a Kalman filter is not adequate protection against a GPS spoofed signal. The simulation shows that the Kalman filter basically takes the perfect average of the two sensors since the variance in the individual data each sensor gives is equal. In reality, this may actually be worse in the hacked signal's favor since a hacked signal can eliminate much measurement data since the signal does not have to travel to space and back and can therefore have less variance than the velocity sensor's kinematic position measurements. This task may be better suited to a machine learning model that can detect a variety of GPS drift techniques. The one used in the simulation was a linear drift, but many different curves could be used in theory in order to evade detection. Future work would have to focus on creating a machine learning model that can detect whether or not a GPS signal is being spoofed in a variety of conditions, positions, times, and weather patterns.

According to my research about the hardening of UAVs, it seems that UAVs have indeed been made more secure. When it comes to taking control of a UAV, WiFi based attacks are out of the question. They can still be used to gather information and perform spying on what the user is looking at. This still needs fixing, but is out of the scope of this paper since it does not involve the control of a drone itself. GPS spoofing is still a serious problem that can only be remedied by using a uniquely encrypted protocol like each major military force does which costs several

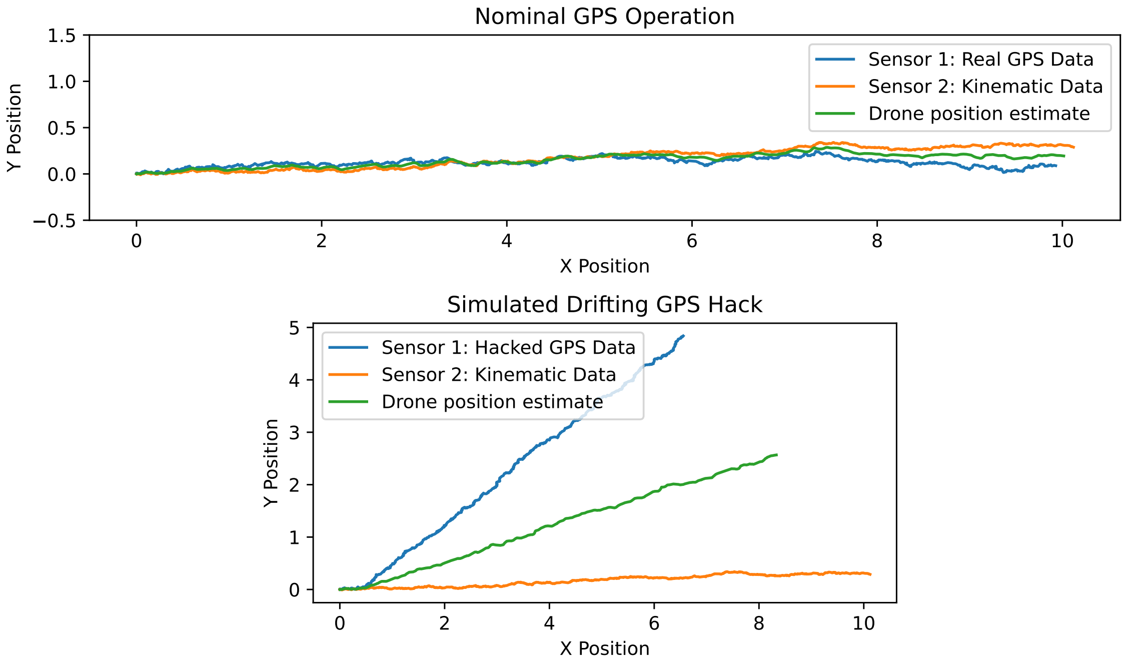

Figure 1: Drone position estimates under normal operation and under a spoofed signal.

million dollars in infrastructure, or in comparing the GPS signal to another source of positional data. GPS is still widely used today for its high accuracy, global coverage, and quick response time. [1] Therefore, they can't be completely replaced with a secondary source of positional data, but it still can be used as a verification for the data.

References

[1] Gpredict: Real-time satellite tracking and orbit prediction application. `http://gpredict.oz9aec.net/`. Version 2.2.1.

[2] Communication act of 1934. U.S. Statutes at Large, 1934. 47 U.S.C. §333.

[3] Aatif Khan. Hacking the drones. In *OWASP*. OWASP, 2016.

[4] Elliot Nesbo. Can drones be hacked?, 2022. Accessed: September 16, 2023.

[5] Nils Rodday. Hacking a professional drone. In *Blackhat Asia 2016*. Blackhat, 2016.

[6] Waqas. Us border patrol drones hacked by drug cartels, 2016. Accessed: October 23, 2023.