Protecting the Unmanned Aerial Vehicle from Cyberattacks

Jesus Nunez\textsuperscript{1}, Vincent Tran\textsuperscript{1}, and Ajay Katangur\textsuperscript{2}

\textsuperscript{1}Department of Computer Science, California State University Dominguez Hills, Carson, California
\textsuperscript{2}Department of Computer Science, Missouri State University
\textsuperscript{1}esusnunez1343@gmail.com,\textsuperscript{2}vincentvtran17@gmail.com,\textsuperscript{2}ajaykatangur@missouristate.edu

Abstract—The increased use in drone technology has made them a popular option for companies to use when preforming certain tasks. Due to this increase in popularity, security analysis has become crucial. In this paper, attacks preformed and observations associated with security vulnerabilities in the AR parrot 2.0, 3DR Solo, and the DJI Phantom 4 Pro drones will be presented. The current auto-pilot systems and security protocols will also be examined for vulnerabilities and cyberattacks that are common in network systems. Currently the AR parrot 2.0 drone communicates through an open Wi-Fi connection making it vulnerable to multiple forms of attacks. The 3DR Solo works through a password protected Wi-Fi signal, however, it is possible to obtain such password with the use of specific tools discussed later in the paper. This poses a potential threat to the system leaving it open to intrusion. Although the DJI has improved security compared to previous models, GPS spoofing still remains a viable form of attack making it a possible vulnerability in the system. Other forms of attacks will also be explored to see if there is more beyond tampering with the network communications of the drones that might pose a threat to the system.

Keywords—drones, UAV, cyberattack, vulnerability

I. INTRODUCTION

Unmanned Aerial Vehicles (also known as drones) technology is rapidly growing in popularity [1]. Primarily used by the defense department, UAV's are now utilized for non-military tasks putting them at high demand by both average consumers or hobbyist, and commercial businesses like Google and Amazon. Companies are already utilizing drones for specific tasks such as delivery services, agriculture, infrastructure development, aerial surveying, and tasks that might be dangerous for a human to preform [2]. As a result, UAV traffic is going to increase in the coming decade, and with it the possibilities of cyberattacks will rise. UAV's can be hijacked, their paths can be changed, and they can be made to collide with other UAV's or objects. Furthermore, if they are equipped weapons system, they can be maliciously utilized to fire in non-hostile situations. Commercial aircraft, although not a UAV, is still an aerial vehicle and it currently does not have a solid defense against a cyberattacks. In [3] the Department of Homeland Security (DHS) state that commercial aviation backbone is built upon a network of trust. This is not very reassuring considering peoples lives are only protected by trust when in the air. Boeing estimates a 20 year plus service life for its current aircraft which means 15 to 20 more years of cyber vulnerability [3].

A. Motivation

UAV's are typically controlled from the ground by using a remote control which uses radio waves for communication. For a hacker, it is not difficult to jam the radio waves, thereby effectively making the person lose control of the UAV. Each of the drones examined in this paper function differently so the approach taken in looking for vulnerabilities will vary. As long as the Wi-Fi signal the drone communicates in is able to be picked up by the hacker, the drone is vulnerable to an attack. There already exists tools that facilitate an attack to drones. Icarus [4], a tool used to grant an attacker complete control over a target drone, allows the attacker to steer, accelerate, brake and even crash them. SkyJack, an open source drone-hijacking program, can also be used to seek out and wirelessly take over other drones within range essentially allowing the attacker to create an army of zombie drones all under their control. For these reasons, it is critical to analyze and address the security of aerial vehicles especially those whom are controlled autonomously through a wireless connection.

B. Security Vulnerabilities in Unmanned Aerial Systems (UAV's)

The AR Parrot 2.0, 3DR Solo, and DJI Phantom 4 Pro drones are analyzed for vulnerabilities in this paper. Attempts to exploit open connections of the AR parrot drone, revising the application program interface (API) of the 3DR Solo to find potential weaknesses that might allow the tampering of flight plans and configurations, and exploring other options of attacks besides GPS spoofing of the DJI Phantom 4 Pro will also be tested.

Wi-Fi issues: The AR parrot drone communicates through an open Wi-Fi Protected Access (WPA) connection making it possible for multiple users to connect simultaneously, and as a result making it impossible to determine who is the valid original user. The 3DR Solo has more security by requiring the use of a password to access the connection between the controller and the drone. If the attacker is able to obtain the password, it is possible to have multiple users connected and without the original user knowing, is able to change settings and deauthenticate the original user by either changing the password or using tools like aircrack-ng[5].

GPS based attack: Higher end drones use GPS signals for navigation. The GPS signals used by civilian drones are not
encrypted making them vulnerable to spoofing and other GPS based attacks. A more detailed example of the steps required to spoof the GPS and take over a UAV are presented in [6]. Currently the DJI Phantom 4 Pro has no public access to the drones API for security reasons. This means that the only way an attacker can cause harm to the drone is through spoofing of its GPS signals while airborne.

The rest of the paper is organized as followed. Section-II provides some related work and details on the influence it has had on this research. Section-III offers insight on the experiments that have been preformed on the specific drones. The results of these experiments are also provided in this section. Section-IV is the conclusion along with future work.

II. RELATED WORK

The research on the vulnerabilities of Unmanned Aerial Systems done in Ben-Gurion University of the Negev in Israel confirmed the vulnerabilities discovered in the AR Parrot drone. Attacks pertaining to the drones open Wi-Fi, deauthentication of the original user, exploiting the open file transfer port (FTP) to manipulate data, and snooping into the Wi-Fi and packet capturing were all preformed in this paper. The attacks done on the DJI Phantom 4 by this research team have provided interesting insight to theoretical attacks that might be possible. Considering however, that most of the attacks done involve manipulation of software through direct access of the drones hardware, they are not efficient since the supposed hacker will never have direct access to either the software or hardware of the drone. These attacks, although they suggest a potential vulnerability, might not actually be possible to preform by an attacker. In an ideal scenario, the hacker has direct access to the drone and its software, however, the chances of that occurring are not high. For this, their research done on the DJI Phantom 4 only provides options to explore if one had direct access to the system. The only possible attack preformed that proves to be a vulnerability in the DJI Phantom 4 is the spoofing of its GPS signals using specialized tools. More specifics on this research in [7].

III. EXPERIMENTS AND RESULTS

This section will be broken down into 3 subsections, each describing what tests have been preformed on the specific drone along with the results of each test.

A. AR Parrot 2.0 Drone

The AR Parrot 2.0 drone, shown in figure 1, is an easy to fly drone. A mobile device along with the "Freeflight Pro" application are required for control and communication of the drone. The AR Parrot 2.0 drone is a Linux-Based quad-rotor that contains a HD 720p camera, and a Wi-Fi access point. The attacks preformed on this drone are focused on the open Wi-Fi connection. This specific model does not offer the ability for the user to add a password to the connection to increase the security of the drone. Due to this, the overall security of the drone is minimal and its vulnerability to an attack is high.

1) Open Wi-Fi: The AR Parrot 2.0 drone uses an IEEE Wi-Fi 802.11n signal. The IP address of the drone is 192.168.1.1, and since the connection is not protected by a password, all communications are unencrypted. The drone allows for multiple users to connect at once and with no way to validate who the original user is, anyone can control the drone. This is a massive vulnerability in the system and unless the user is given the option to protect the drones connection with a password, the drone is prone to an attack.

2) Deauthenticating Owner: The AR parrot 2.0 drone is flown using the Free Flight mobile application. Using tools such as aircrack-ng [5], it is possible to scan for current Wi-Fi networks in the vicinity and then attack a specific one using its unique mac address. For the parrot drone, it is not hard to break into the connection since it is not password protected. Once the handshake between the computer and the drone is established, the deauthentication procedure takes place. This disconnects the original user making it possible for the attacker to take over using a separate mobile device.

B. 3DR Solo

The 3DR Solo consists of a controller and drone components as seen in figure 2. A mobile phone can be used as part of the set up for video streaming, however it is not required to fly the drone. Both the 3DR Solo and the controller contain
an embedded Linux system. The drone specifically contains a camera and gimbal, as well as GPS. The controller has an access point that connects to the Solo and a mobile device via Wi-Fi. The connection is a password protected connection so breaching into the connection is not as easy like with the parrot drone. The test performed required the use of aircrack-ng, and a dictionary file with possible passwords to facilitate the attack. Theoretically, the attack should not require a dictionary file, but for this test it is utilized to test for a possible vulnerability.

1) Deauthentication of Owner: For the 3DR Solo, the controller is the root of all the communications with the drone and mobile devices. The procedure for this attack involves first obtaining the drones password, and then utilizing aircrack-ng to deauthenticate the original user. The result of this attack causes the drone not to crash, but to return home, the initial location from where it took off. After it lands, the connection between the drone and the controller is disrupted until the attack is ceased.

2) Force Landing Using Mobile Device: The 3DR Solo Drone controller works as a router for the connection communication to the drone. The connection is password protected, the default password being "sololink". The assumption is that most users will not change the password and keep the default therefore, in most cases, the 3DR Solo's password will be the default. With this assumption, the method of attack is the following, using another mobile device, it can be connected to the 3DR Solo network. This is an easy task due to the fact that the password is the default. Once the connection is established, two mobile devices are now connected to the one controller, from here using the attacking device, the password can be changed and this would force the drone to land and the original mobile device to disconnect. This method allows the attacker to force land the drone and disconnect the original user from the network. This however, can only be performed if the initial assumption is true, the user will not change the password from the default on their device.

3) Using Two Drones to Spoof Controller: For this procedure 2 drones and one controller are utilized. The assumption here is that by deauthenticating the original drone, using the second drone to send a pairing signal, the controller will be spoofed and connect to the other drone rather than the original drone. The second assumption is the following, once the original drone is successfully deauthenticating, the pairing signal is sent from the other drone. The end result being a prolonged connection time between the pairing of the controller and the original drone. This would allow an attacker to gain much more time for an attack should the need for it be required.

4) Analyzing 3DR Solo APK: By first obtaining the 3DR Solo APK (Android Package Kit) file, the .dex files can be converted to .jar which can then be used to see the source code of the application. Once the .dex class files have been converted to .jar files, the code can be seen using a Java Decompiler tool. All tools and steps to do the following are in [8]. The Java Decompiler can only be used to see all the source code of the application, but not change it. The code that allows the linking between the drone and the controller is shown which provides a better understanding of the process undergone when the drone is trying to communicate with the controller. The purpose of analyzing the source code is to see if there might be any vulnerabilities in the way the application handles errors and exceptions regarding the connection link between the drone and the controller.

C. DJI Phantom 4 Pro

The DJI Phantom 4 Pro, shown in figure 3, contains a controller and mobile device, along with the drone. The mobile device uses the DJI GO 4 application which acts as the interface to all the drones controls and settings also providing display to the live video feed.

1) GPS Spoofing: DJI Phantom drones are GPS based drones. GPS allows for better navigation, and due to the fact that the drone functions under civilian GPS signals, the lack of encryption makes it easier to spoof the GPS. To spoof a GPS receiver, a transmitter is used to send false GPS signals, forcing the GPS on board the drone to synchronize with the attackers signals. If done successfully, this enables the attacker to hijack the drone and place it at a desired location.

IV. CONCLUSION AND FUTURE WORK

In this paper, the AR Parrot 2.0, 3DR Solo, and the DJI Phantom 4 Pro drones have been analyzed for security vulnerabilities. The AR Parrot 2.0 drone is the most vulnerable of the 3 making it highly prone to an attack. Unless the manufactures allow for the user to add a password to the network the drone communicates on, this drone is too risky for any personal or commercial use. The 3DR Solo drone proved to be a far more secure drone than the AR Parrot 2.0. The ability to add a unique password to the network the drone communicates on adds a layer to the security for any user. In this paper, the lack of specific equipment limited the ability to further analyze other communication protocols in the 3DR Solo and DJI Phantom 4 Pro. In future work, radio frequency analysis will help better understand these protocols, and the use of SDR equipment, such as HackRF One [9], will facilitate this task.
ACKNOWLEDGEMENT

This work was supported in part by the National Science Foundation under grant no. 1757940.

REFERENCES


