Understanding Security Threats in Consumer Drones Through the Lens of the Discovery Quadcopter Family

Junia Valente, Alvaro A. Cardenas
Erik Jonsson School of Engineering & Computer Science
The University of Texas at Dallas
{juniavalente, alvaro.cardenas}@utdallas.edu

Abstract

In this paper we identify new threats to drones in an effort to have a better public discussion of realistic attacks that vendors need to take into consideration when designing their products. In particular we study in detail the security of a new drone family (U818A) released in 2016, which is quickly becoming a best-selling brand, and is re-purposed and sold by a variety of drone vendors. We implemented and tested several attacks and considered privacy issues (e.g., remotely accessing someone else's drone to take video or images of a private setting), security issues (e.g., stealing a drone mid-flight), and safety issues (e.g., taking down a drone operated by someone else). We finish the paper by recommending basic steps to improve the security of drones.

1 Introduction

While drones and Unmanned Aerial Vehicles (UAVs) have been historically used in surveillance and military contexts, nowadays drones have become widely accessible to the general public, and in turn, they are raising new societal security and privacy considerations. From the point of view of privacy, drones can let users spy on neighbors [30, 32], and enable literal *helicopter parenting* [39]. Safety and security are also other concerns; drones can be used by cyber-attackers to reach wireless networks that were otherwise unreachable [28], and a physical attacker can use them to fire weapons remotely [42]. Accidents can also cause safety concerns as showcased by the city of Seattle's first *mishandling a drone in public* charge [9].

In this article we analyze security and privacy threats in drones through a detailed study of the security practices of the family of Discovery U818A quadcopters—mainly on the U818A quadcopter by the U.K. company DBPOWER[1] and on the Force1 quadcopters[2] sold by the U.S. company USA Toyz. We also analyze the Parrot AR.Drone Elite drone [1]; however, since the vulnerabilities we observed in the Parrot drone have been previously reported by other researchers, we focus our findings to the Discovery drones. The Discovery devices were released to the market in 2016, and rebranded under different company names.

In particular, we discuss the vulnerabilities we found for the Discovery drones (based on our disclosure to US-CERT in Note VU#334207 [4] and assigned CVE-2017-3209 [2]) and show how these vulnerabilities can be used by attackers to achieve different goals, including:

- Fly-away attacks: running away with the drone.
- Take-down a flying drone: intentionally causing accidents like the one reported in Seattle [9].
- Lock-out attack: preventing the legitimate owner of the device from connecting to it.
- Steal user data: getting access to the video feed of the drone.
- Use the drone to take unauthorized (private) video or pictures.

This paper is organized as follows: we first present a discussion of drones within the bigger context of security and privacy, and include a brief survey of common vulnerabilities and attacks on consumer drones. Then we present the threat model, and attacks against Discovery drones. We conclude with additional findings, and discussions on the impact of the vulnerabilities and steps to prevent attacks.

2 Growing Concerns over Consumer Drones

The growing use of drones for law enforcement and personal use is raising privacy concerns. For example, surveillance drones fly at heights beyond range of sight [15], and often, individuals being monitored are unaware when the surveillance takes place [16]. Similarly consumer drones have been used to spy on unsuspecting victims [24, 30, 32, 40]. In the U.S., the supreme court decision of *Florida v. Riley* concluded that citizens do not have a reasonable expectation of privacy that their activities will not be observed from the air, as helicopters and airplanes often fly over private properties, so flying a drone above a certain altitude would be permitted in the U.S. For flying a drone below this minimum altitude, Voss [37] proposes that in some cases, instead of having the FAA control the airspace below this minimum altitude, the control should be shifted to landowners: small UAVs should not operate near ground without explicit permission of landowners. They hope that this would reduce regulatory burdens to fly small UAVs while increasing public acceptance of unmanned aircrafts. Further, others believe that privacy concerns around unmanned aircrafts will improve only by a combination of adopting voluntary policies by law enforcement, and through new federal, state, and local legislations [36].

Other countries have taken tougher rules for drones. For instance, Sweden banned the use of camera drones, unless the owner can prove the benefit of using the device outweighs "public's right to privacy" [25, 34]. Using camera drones for journalism, taking nature photographies, or filming weddings are not enough justification and strictly prohibited there. One argument, is that camera drones allow the pilot to *see* places and have access to spaces (even when they are public) otherwise out of sight of the pilot.

These privacy concerns have motivated researchers to propose techniques to detect nearby drones automatically [6–8].

[1] DBPOWER Discovery UDI U818A WiFi FPV Quadcopter Drone
[2] Force1 Discovery Wi-Fi U818A FPV Virtual Reality Upgrade

IoTS&P'17, November 3, 2017, Dallas, TX, USA

ACM ISBN 978-1-4503-5396-0/17/11…$15.00
https://doi.org/10.1145/3139937.3139943

In addition to privacy, there is a growing concern about safety issues as drones have been used to smuggle items to prisoners in London [5], they have been used as physical weapons [41, 42], and even as cyber-weapons (e.g., a remote pilot flying a drone around a city to spread malware) [28, 29].

To mitigate these threats governments have been proposing new rules so some drones have a built-in geofencing features to avoid no-fly zones. For instance, as of 2015, DJI drones provide users with airspace live information (e.g., temporary flight restrictions due to natural disaster) and also the device restricts the pilot's ability to fly around no-fly zones (e.g., power plants, airports, national-security locations such as Washington, D.C.) [14]. However, it has been possible to modify particular drones to remove all restrictions to fly over no-fly zones due to code glitches or security vulnerabilities [3].

2.1 State-of-the-Art Attacks on Drones

Research on cyber-security practices of drone manufacturers have been receiving increasing attention. For example, researchers from the Federal Trade Commission presented security flaws on three off-the-shelf drones: oneCase Cheerson CX-10W ($39.99), Parrot AR.Drone Elite ($150), and Hawkeye II 2nd FPV Quadcopter from DBPOWER ($325). They showed that anyone can connect and watch the drone video feed (without the legitimate user noticing) because data traffic was unencrypted, and it was possible to cause at least two of them to fall from the sky—due to open access points that are not password protected on the devices [22].

There has been a lot of attention from the security community to the Parrot AR.Drone 2.0 quadcopter [11, 19, 22, 23, 27, 31, 35]. To start with, the Parrot AR.Drone 2.0 ($179.99) has both the `ftp` and `telnet` services not only *open* but also *not* password protected. So by default anyone near the drone can access the drone operating system as `root` [27] (this is not possible to do on the Discovery family of drones by default—however, as we show later, the new vulnerabilities we found allow us to exploit a misconfigured `ftp` server to gain `root` access and launch similar attacks as in the Parrot drone). Samland et al. [31] further show how an attacker may take advantage of unencrypted communications with the drone to hijack the device, eavesdrop on video streams, and even track people (by using the GPS receiver to determine their position). To gain control of the drone, they use the `root` access to modify the SSID of the wireless network of the drone. This automatically disconnects the legitimate controller communication with the drone, and allows an attacker to immediately establish a connection to the drone. Another popular way is to deauthenticate the true control using the `Aircrack-ng` suite as used in the SkyJack project [23] to take control of a flying drone (either Parrot AR.Drone 2.0 or version 1).

In earlier versions of the DJI Phantom drones (before version 3), it was also possible change the SSID of the access point mid air, causing the drone to disconnect from the controller, and allowing another device to take control of the drone. According to [17], an improvement of the DJI Phantom 3 over other drones is that it prevents another controller from taking control of the drone (e.g., when the legitimate controller is lost due to a de-auth attack). The drone device itself keeps a history of all controllers connected through its Wi-Fi access point, and accepts control commands only of the first device connected (as most drones do). However, the difference is that if that controller stops sending commands to the drone (e.g., because it was knocked off the access point), then the drone will not accept commands from another controller, until the legitimate controller is the one that disconnects itself. It is unclear happens if the drone is disconnected while flying.

Another study [17] shows that the DJI Phantom 3 ($464.99) has similar vulnerabilities to the Parrot drones and previous versions of Phantom drones: they have open `ftp`, `telnet`, and even the `ssh` service running—although unlike some other models, all these services are password protected by default. One difference from the DJI drone over other popular devices, is that the controller device itself also creates the Wi-Fi access point. This is because the DJI system is composed of three subcomponents: the drone, a camera module attached to the drone, and the controller. The mobile app sends control commands to the controller, and the controller relay the commands to the drone via a radio signal. Researchers have found the `root` password for the `ftp` service (for all subcomponents) by reverse engineering the DJI mobile app. Although, they were not successful in finding the `root` password for the other services [17]. Drone-enthusiasts further found that it is possible to use the `ftp` access in the DJI Phantom 3 to turn on and off `telnet`, and consequently gain access using same password as the `ftp` server [18].

Further, it has been well known that in the DJI drones, it is possible to bypass the device's geofencing restrictions [14]. Recently, the concern has shifted to the fact that the Chinese company intentionally collects data from the flight and stores in their servers both in the U.S. and in China. DJI drones have been used by the U.S. Army in various operations, and a recent study concluded there are "operational risks" involved in using DJI drones [20]. The growing concerns on their security [3] have prompted the U.S. Army to ban its personnel from further using any DJI drones on August/2017 [12]. As the author in [20] explains, it may be possible that they found that an "adversary could hijack a control session through a bug in DJI's protocol, or obtain telemetry, audio and video covertly." Since then, DJI has requested that the U.S. Army work with them over the cybersecurity concerns.

Researchers have also shown that it is possible to take down a drone by launching attacks against the gyroscopes of the device. Son et al. [33] demonstrate that it is possible to disrupt the control of DIY drones by using intentional sound noise against vulnerable MEMS gyroscopes; and Wang et al. further show "sonic attacks" against the gyroscope sensors of DJI Phantom drones to disrupt the performance of the device [21].

For a comprehensive list of additional vulnerabilities found on drones as well as attack tools and methodologies, refer to [38].

3 Threat Model

In this paper, we consider an attacker that is within Wi-Fi range of the drone but otherwise has no other direct physical access to the device nor any authentication token for the device. This attacker is a representative threat model for several drones which have Wi-Fi access points to support network connectivity for their devices. In fact, embedded devices with a Wi-Fi access point that is used to communicate with an app in a smartphone is a very common pattern in IoT. Fig. 1 shows an illustration of this pattern, where the owner of the IoT device uses an app in a smartphone to talk via Wi-Fi to a device (e.g., a drone, an NVR, an IP camera, or a smart light bulb). In this scenario the adversary is within the Wi-Fi range of the device, but has no other physical or cyber-access.

4 Security Analysis of Discovery Drones

We present attacks we tested against the Discovery U818A drones family. These drones are very popular, however the "Discovery" brand is not a household name because these drones are rebranded and sold under a variety of different names and companies, including UDI RC, Holy Stone, Kolibri, Hero RC, Force1, USA Toyz, and

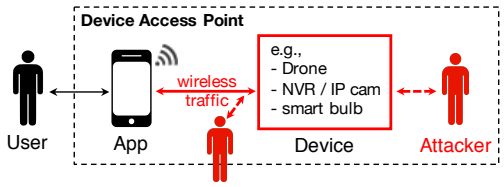

Figure 1. Attack points: attack on device (via insecure network services like `ftp`) or on the network (by eavesdropping the traffic).

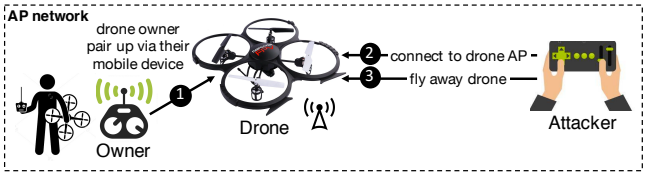

Figure 2. Attack 1—a near-by attacker can hijack the drone and fly it away using one of many drone apps available for mobile devices.

DBPOWER. The FCC ID in these drones indicates that the actual manufacturer of all these devices is a company by the name: Udirc Toys Co., Ltd.

We bought two devices of this family, the U818A quadcopter by the U.K. company DBPOWER and on the Force1 quadcopter sold by the U.S. company USA Toyz. These drones are new in the market (they were released in 2016), and both are popular on Amazon.com: for example, the DBPOWER drone ($109.99) is currently listed as the top #1 best seller (under the airplanes & jets hobby toys category on Amazon). The USA Toyz drone ($149.95) is an upgrade that is compatible with a virtual reality (VR) headset and can autonomously fly along a desired flight path (drawn by the pilot with the mobile app).

While some of the attacks we present may have been successful in other popular drones, to the best of our knowledge we are the first ones to exploit *incorrect filesystem permissions* to gain `root` access to consumer drones; and to analyze Discovery drones to confirm that at least two vendors are affected by these security flaws.

We now describe some of the potential attack vectors we identified (and then tested with success) on our drones. A video illustrating one of our successful attacks is available on YouTube[3]. While some attacks may be specific to our drones, there are several general attack-patterns that are relevant for a variety of drones. Our motivation is that by presenting these potential threats we can improve the discussion on possible drone vulnerabilities that manufacturers need to be aware of when they design their systems.

4.1 Fly-away Attack

One of the common security problems in IoT devices is the lack of authentication. While personal computers and smartphones now require users to authenticate themselves before accessing the device, many IoT devices have weak or no user/owner authentication, and in this case, it leads to completely new threat scenarios. In particular, our drones could be easily stolen by anyone within the Wi-Fi signal range of the device. In a simple theft scenario, the attacker could be driving a car near the a user of a drone. The attacker then accesses the device and runs away in the getaway car while controlling the drone. Getting access to the drone is different if the drone is on the ground and idle, or if a user is currently controlling it.

Drone on the ground: In a trivial attack, a near-by attacker can simply use the drone app—which does not require any type of user authentication—to fly the drone away, when the device is turned on but not flying. Fig. 2 shows this attack with the following steps:

Step 1: The legitimate `drone` pilot pairs up mobile device to drone.
Step 2: The attacker connects mobile phone to drone access point.
Step 3: While drone is turned on and idle on the ground, the attacker can use a proprietary drone app to fly the drone away.

Drone in the air: An attacker can launch a de-auth attack against the legitimate mobile device controlling the drone, and as soon as the legitimate user is kicked off from the drone access point, the attacker can immediately take control with their mobile device. There is not even a need for an attacker to write their own script because they can simply download the drone app from the Google Play or Apple's app store, and use that during the attack—these apps do not require user authentication or device pairing verification. However, if the attacker writes their own exploit script, it is possible to launch more sophisticated attacks like the SkyJack project [23] that mounts to a drone, a piece of hardware running exploits, to autonomously take control, mid-air, over other Parrot drones [10].

Observations: Since the network connection between the device and the mobile app is *not* encrypted, an attacker can easily map different `udp` packet patterns being sent to the Discovery drone *to* core drone functionalities (e.g., control throttle, yaw, pitch, or roll movements; take a picture; record a video). This information can be used to control the drone programmatically (as in the SkyJack project). For Discovery drones the commands (over `udp`) are undocumented and not found on forum websites, but for other drones, this can be a feature (e.g., the Parrot AR.Drone 2.0 [1] provides well-documented APIs for creating mobile applications to communicate with their drones). As we present later, we found a large number of mobile apps that implement exactly the same commands (sent over `udp`) that we found on the proprietary app for DBPOWER and USA Toyz drones. It remains unclear whether this is a design feature or an example of copy-and-paste software practice in the wild.

4.2 Exploiting Incorrect Filesystem Permissions Attack

Several drones have `ftp` servers so users can retrieve photos and videos taken during flight. Our two drones from DBPower and USA Toyz had anonymous `ftp` users (i.e., they accept any strings as a password [13]). Besides the obvious privacy threat of an adversary reading our captured photos and videos, another possible threat an attacker can do with an anonymous `ftp` is to explore the file system looking for interesting files.

In both our drones we searched and found `/etc/shadow`. Since our drones also had a `telnet` server, having access to the `root` password would give our attacker complete control of the device. We downloaded the password file and tried to crack it with *Hashcat* and *John the Ripper*, but were unsuccessful in cracking it. However, when looking for file permissions we found that the password file was a symbolic link to another file that had read and write access by any user in the system! As such we were able to replace `/etc/shadow` with our favorite hashed password and obtain complete control of the device (the fact that a nearby user can replace the password file was a vulnerability we reported to US CERT [4]

[3] https://youtu.be/bAxPBNDeCmM

and to the manufacturers of the drones). We now explain in more detail the attacks.

4.2.1 Steal User Data Attack

This attack takes advantage that the `ftp` server allows not only an anonymous user to login, but provides full filesystem read/write permissions to the anonymous user [4]. Any remote user within range can read arbitrary files included images and videos.

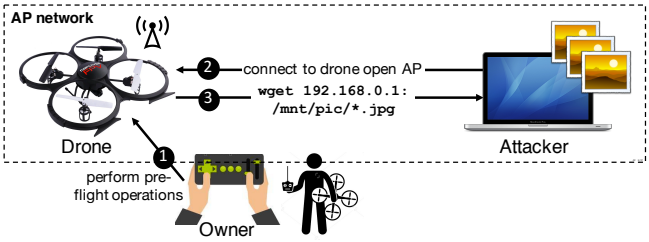

Figure 3. Attack 2—an attacker can steal media captured by the `drone` without the `drone` owner flying the device noticing.

Attack 2 - Near-by unauthorized users can access images and videos recorded by Discovery quadcopters. *Anonymous* `ftp` *access exposes* `drone` *files to unauthorized users* as shown in Fig. 3:

Step 1: The legitimate `drone` user performs pre-flight operations.

Then, when the pilot starts flying the drone, taking photos or recording videos, an attacker can download media without being detected:

Step 2: The attacker connects their machine to the access point.

Step 3: The attacker may run a command like: `$wget 192.168.0.1:/mnt/pic/*.jpg` to download the photos captured, and `$wget 192.168.0.1:/mnt/Video/*.mp4` to download the videos.

These steps assume the drone pilot has previously recorded videos or taken pictures. However, as we show later in Section 5, an attacker can use unknown ports (other than `ftp`) to send commands directly to the drone to request it to take photos, and transfer files between the drone and the attacker's machine. In our experiments, we use Mac OS or Linux to test the attacks.

4.2.2 Overwrite Root Password Attack

Remember that with the `ftp` access we can read `/etc/shadow`. We tried to recover the `root` password, but it was not possible even after letting cracking tools run for a few days. However, we found inconsistencies with the permissions of the password file: although the password file `/etc/shadow` has read-only permission, the **shadow** file is a symbolic link to another file `/etc/jffs2/shadow` which has not only `read` but also `write` permissions by all users including the anonymous `ftp` user. As a result, a remote attacker can overwrite the password file on the drone and gain `root` access.

Attack 3 - Anyone near-by a Discovery drone can modify sensitive files (via anonymous `ftp` login) to gain `root` access to the device. *We found that anyone can access the drone* `ftp` *server and* ***modify*** *the* `shadow` *password file that holds password hashes! We found inconsistencies in the permissions of the password file. Needless to say, an attacker with this knowledge can take over the* `root` *access of the device.* Fig. 4 shows the attack workflow as follows:

Step 1: The attacker connects to the `drone` access point.

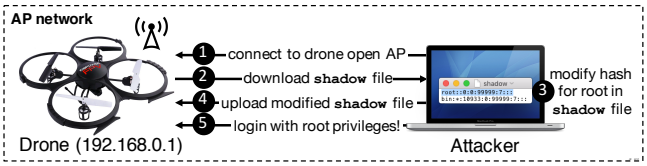

Figure 4. Attack 3—an attacker can use the anonymous `ftp` login as a backdoor to gain full `root` access to the device via `telnet`.

Step 2: The attacker downloads the password hash file **shadow** to its local machine (e.g., `$wget 192.168.0.1:/etc/jffs2/shadow`). No password is required because of the anonymous `ftp`.

Step 3: The attacker removes the password hash for the `root` user in the **shadow**, e.g., replace `root` hash with: `root::0:0:99999:7:::`.

Step 4: The attacker uploads the modified **shadow** file to the drone: `curl -T shadow ftp://192.168.0.1:21/etc/jffs2/shadow`.

Step 5: Now that the attacker has removed the hash for the `root` user, the `telnet` server is not protected by a password and anyone can `telnet` to the device with `root` privileges.

When the user attempts to establish a `telnet` connection, they will see a login prompt "`anyka login`". The user can then type "`root`" and press enter to gain `root` access to Discovery drones.

The `telnet` access allows unauthorized users with full view of which programs are running, what devices are connected to the drone access point, see active network connections between the drone and other devices (including the pilot's mobile device), run programs on the device (like BusyBox utilities in Discovery drones), and take down the device when it is flying. As we tested, it also possible to use the `telnet` access to block network traffic to disrupt the first-person-view experience (using a VR headset compatible with Discovery drones) while interrupting the video stream.

4.3 Take-down Flying Drone Attack

A few drones such as the Parrot Bebop drone and the AR.Drone 2.0 are known to be vulnerable to this attack by default because unlike the Discovery drones, they contain a `telnet` wide open by default. As researchers have shown before, once an attacker can `telnet` to the device, a common proof-of-concept attack is to take-down the drone—where the most popular demonstration of this attack was presented at `DEFCON` 23 [26] and is summarized as follows:

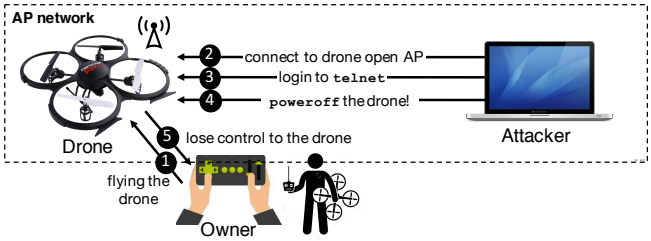

Figure 5. Attack 4—any near-by user can take down a flying drone by leveraging the `telnet` access gained by exploiting the `ftp` server.

Attack 4 - Unauthorized users are able to take down a flying drone when they gain `telnet` access. *Discovery drones do not have* `telnet` *open by default, but as we show, inconsistencies in the drone filesystem allows to disable the password for the* `root` *user.* We

can then take down a flying device as shown in Fig. 5:

Step 1: The drone owner is flying the drone.
Step 2 and 3: The attacker connects to the drone access point, and remote accesses the open `telnet` (which we disabled the `telnet` password).
Step 4: The attacker can send a command to power-off the device and take down the drone in the air.
Step 5: At this point, the pilot's mobile device loses connection to the drone, and the drone abruptly falls to the ground or gracefully lands depending on the device.

Besides sending a `poweroff` command, it is also possible to use the `kill` command to terminate critical processes (namely, the `daemon` and `lewei_cam` processes for Discovery drones). It is also possible to modify configuration files (like the access point for example) to possibly brick the device or to permanently disable core functionalities (like the video streaming, access point, etc).

5 Additional Findings on Discovery Drones

In this section we provide additional findings. In particular, we found that Discovery devices communicate over commands called `lewei_cmd`. All commands sent back and forth between the drone is sent via `lewei_cmd` over unknown ports. This includes control commands, video stream, the file transfer process, etc.

5.1 Undocumented drone ports

One of the attacks we described (where an attacker can steal media content created by the drone) assumes that the drone owner has previously recorded the videos or taken pictures. However, an attacker can use other open ports—namely, undocumented ports such as `7060` and `8060` that can be found by looking at the inbound-/outbound connections to the drone (e.g., by logging in via `telnet` and running the `netstat` command on the drone) or by analyzing network packets between the mobile app and the drone—to send commands directly to the drone. By analyzing the traffic between the drone app and the drone, we found that the traffic is not encrypted and further that the app sends commands to the drone in the form of `lewei_cmd` commands. An attacker with this information, is able to easily map different `lewei_cmd` commands to drone functionalities (e.g., take photo, record video, retrieve file names from the SD card mounted to the drone, transfer files off from the drone to their machine). For example, an unauthenticated user can use the command below (in Listing 1) to request the drone to take a photo and return a photo denoted as `photo_file`:

Listing 1. We found that anyone can send a `lewei_cmd` command to port `8060` to request the drone (`192.168.0.1`) to take and return a photo to their computer (connected to the drone access point).

```
{ echo lewei_cmd; echo -e "\x00\x13" "\x00"$s{1..35}; } | tr -d "
      \n" | nc 192.168.0.1 8060 > photo_file
```

This raises privacy concerns. Consider a scenario where a student has recently bought this drone and has the device at their dorm apartment. The minute the student turns on the drone—anyone within Wi-Fi range is able to connect to the drone open access point, and send commands to request the drone to take photos. Or more simply, anyone that has a compatible drone app on their phone is able to open the app and view a video live stream from the drone.

In addition, we also found that we can use additional `lewei_cmd` commands (via port 8060) to read the drone file system and transfer files off the drone device to a machine.

Listing 2. We found that users can send the following `lewei_cmd` command to retrieve file names from drone directory `/mnt/Video`.

```
{ echo lewei_cmd; echo -e "\x0\x8" "\x0"$s{1..11} "\x14" "\x0"$s
      {1..31} "\x5" "\x0"$s{1..6} "\x7f" "\x0"$s {1..4}; } | tr -
      d ' \n' | nc 192.168.0.1 8060
```

This command can be customized: e.g., by using `"\x5"` in the command we specify that the response should return up to five file names (where default seems to be `"\xff"`). Moreover, we can use a similar `lewei_cmd` to download each specific file. Note that this is done over port 8060 and not via `ftp`. In fact, it is not clear why the `ftp` access is open because it appears to *never* be used.

5.2 Reuse of drone apps

After analyzing the Android mobile applications for both Discovery drones, we noticed something interesting: we can use the `DBPOWER/RC` app (for DBPOWER drones) to control and receive video stream from USA Toyz drones; and similarly, we can use the `Flyingsee` app for USA Toyz drones to control DBPOWER drones. The reason might be because both the `DBPOWER/RC` and `Flyingsee` apps are developed by the same vendor: Udir Toys Industrial Co., Ltd[4]. However, we found 15 other apps in the Google Play Store (from the same or different vendors) that can also fly the Discovery drones and successfully view their video stream.

In total we found that at least 17 Android apps are compatible with the drones we analyzed. In essence, even when the apps are from a vendor other than the drone vendor, we can use any of these apps to control our drones and stream video. After performing deep packet inspection (on packets sent by the app), we concluded this is possible because these 17 apps work under similar assumptions:

(1) The apps assume the drone IP address is `192.168.0.1`.
(2) The apps continuously send control commands (i.e., slight variations of `66808000808080800c0c99`) to `udp` port `50000`.
(3) The apps send `lewei_cmd` commands to `tcp` ports `8060`, `7090`, and `9060` for video streaming, taking photo, and transferring video off the drone.

Therefore, we argue that it is possible that more vendors (besides DBPOWER and USA Toyz) might be also affected with misconfigured `ftp` servers. It is not clear whether we can verify this by simply looking at the apps and without having physical access to each device. However, because we found that there are several mobile apps that communicate in exactly the same way (e.g., the apps conform to the same commands `lewei_cam` commands for controlling and streaming video in the two drones we analyzed), we assume it is possible that these apps correspond to drones that use the same underlying firmware as the one currently found in the DBPOWER and USA Toyz devices which are vulnerable to CVE-2017-3209.

6 Conclusions

Improving drone security: The vulnerabilities described in this paper (specially on Discovery U818A drones) can be mitigated by:

- Securing drone access point with a strong password, and WPA2.
- Limiting the number of devices allowed to connect to the access point. Also, enforcing user authentication, and denying income and outgoing traffic from and to unauthorized devices.
- Disabling `ftp` and `telnet`. We found that none of these services are needed for the normal operation of the Discovery drones. But if there must be an anonymous `ftp` user, then the device should not allow read and write access to the entire `root` directory.

[4] http://www.udirc.com/

- Sending network packets between app and drone over a secure channel.
- Upgrading the software running in the device. The Discovery quadcopters (released in 2016) use BusyBox 1.20.2 which was released in 2012. Since then, there has been 18 software updates to BusyBox, and these devices may be vulnerable to other known BusyBox vulnerabilities [4]. (Unfortunately, it is not possible for users to update the firmware in Discovery drones).

While it may be impossible to determine which other drones have misconfigured `ftp` servers without having physical access to each device, when we calculate the number of downloads between all the 17 apps that implements `lewei_cam` commands, the number exceeds 200,000 for Android devices. By including iPhone apps, the number of apps could reach around half a million. Perhaps, this number of downloaded apps corresponds to the number of drones affected by these vulnerabilities in the wild.

Our concerns over safety (taking down a drone) and privacy (taking unauthorized pictures) show that even if the drone is purchased as a toy, attacks can have dangerous consequences.

Vulnerability Disclosure: We disclosed the vulnerabilities on the DBPOWER drone to US-CERT on February 23, 2017. Under their 45-day policy, they made the vulnerabilities public under Note VU#334207 [4] and CVE-2017-3209 [2]. We later verified that different vendors were affected and we directly disclosed this information to affected vendors (Force1, UDIRC, USA Toyz). We did not hear back from them. On April 24 US-CERT updated their Note to reflect the new findings.

Acknowledgments

We thank Paul Murley and Travis Neyland for discussions on the security of the DBPower drone. This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-17-1-0135 and by NSF under award number CNS 1553683.

References

[1] [n. d.]. Parrot AR.Drone 2.0 Elite Edition. https://www.parrot.com/us/drones/parrot-ardrone-20-elite-edition. ([n. d.]).
[2] 2017. CVE-2017-3209 - Incorrect Default Permissions. Available from MITRE. (April 2017). http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-3209
[3] 2017. DJI Firmware Hacking Removes Drone Flight Restrictions. https://www.darknet.org.uk/2017/07/dji-firmware-hacking-removes-drone-flight-restrictions/. (July 2017).
[4] 2017. Vulnerability Note VU#334207 - DBPOWER U818A WIFI quadcopter drone allows full filesystem permissions to anonymous FTP. Available from US-CERT. (April 2017). https://www.kb.cert.org/vuls/id/334207
[5] BBC. 2017. Prisons drone-delivery drugs plot: Eleven charged. https://www.bbc.com/news/uk-39616399. (May 2017).
[6] Simon Birnbach, Richard Baker, and Ivan Martinovic. 2017. Wi-Fly?: Detecting Privacy Invasion Attacks by Consumer Drones. (2017).
[7] Joël Busset, Florian Perrodin, Peter Wellig, Beat Ott, Kurt Heutschi, Torben Rühl, and Thomas Nussbaumer. 2015. Detection and tracking of drones using advanced acoustic cameras. In *SPIE Security+ Defence*. International Society for Optics and Photonics, 96470F–96470F.
[8] Ellen E Case, Anne M Zelnio, and Brian D Rigling. 2008. Low-cost acoustic array for small UAV detection and tracking. In *Aerospace and Electronics Conference, 2008. NAECON 2008. IEEE National*. IEEE, 110–113.
[9] City of Seattle. 2017. City Attorney's Office prevails in drone case. https://news.seattle.gov/2017/01/13/city-attorneys-office-prevails. (Jan. 2017).
[10] Jordan Crook. 2013. Infamous Hacker Creates SkyJack To Hunt, Hack, And Control Other Drones. https://techcrunch.com/2013/12/04/infamous-hacker-creates-skyjack-to-hunt-hack-and-control-other-drones/. (Dec. 2013).
[11] Drew Davidson, Hao Wu, Robert Jellinek, Vikas Singh, and Thomas Ristenpart. 2016. Controlling UAVs with Sensor Input Spoofing Attacks.. In *WOOT*.
[12] Dani Deahl. 2017. The US Army is reportedly banning all drones from China's DJI. https://qz.com/1046724/the-us-army-is-reportedly-banning-all-drones-made-by-chinas-dji-over-security-concerns/. (Aug. 2017).
[13] P Deutsch, A Emtage, and A Marine. 1994. *How to use Anonymous FTP*. Technical Report.
[14] DJI. 2015. DJI Introduces New Geofencing System for its Drones. https://www.dji.com/newsroom/news/dji-fly-safe-system. (Nov. 2015).
[15] "Electronic Frontier Foundation". [n. d.]. Surveillance Drones. https://www.eff.org/issues/surveillance-drones. ([n. d.]).
[16] "Electronic Privacy Information Center". [n. d.]. Domestic Unmanned Aerial Vehicles (UAVs) and Drones. https://epic.org/privacy/drones/. ([n. d.]).
[17] Greg Beams Reece Rivera Fernando Trujano, Benjamin Chan. 2016. *Security Analysis of DJI Phantom 3 Standard*. Technical Report.
[18] DJI forum. 2016. Connecting Phantom 3 with your computer. http://forum.dji.com/thread-55122-1-1.html. (June 2016).
[19] Thomas Fox-Brewster. 2015. Maldrone: Watch Malware That Wants To Spread Its Wings Kill A Drone Mid-Flight. https://www.forbes.com/sites/thomasbrewster/2015/01/27/malware-takes-down-drone/#54feef744c92. (Jan. 2015).
[20] Sean Gallagher. 2017. Army tells troops to stop using DJI drones immediately. https://arstechnica.com/gadgets/2017/08/army-tells-troops-to-stop-using-dji-drones-immediately-because-cyber/. (Aug. 2017).
[21] Sean Gallagher. 2017. Researchers demonstrate "sonic gun" threat against smart devices. https://arstechnica.com/gadgets/2017/07/sounds-bad. (July 2017).
[22] April Glaser. 2017. The U.S. government showed just how easy it is to hack drones made by Parrot, DBPower and Cheerson. https://www.recode.net/2017/1/4/14062654/drones-hacking-security-ftc-parrot-dbpower-cheerson. (Jan. 2017).
[23] Samy Kamkar. 2013. SkyJack. https://github.com/samyk/skyjack. (Dec. 2013).
[24] Charlotte Krol. 2015. Drone catches man sunbathing on wind turbine in Rhode Island, US. http://www.telegraph.co.uk/news/newsvideo/viral-video/11829995/Drone-catches-man-sunbathing-on-wind-turbine-in-Rhode-Island-US.html. (Aug. 2015).
[25] Tom Mendelsohn. 2016. Sweden's highest court bans drones with cameras. https://arstechnica.com/tech-policy/2016/10/camera-spy-drones. (Oct. 2016).
[26] Michael Robinson. [n. d.]. Knocking my neighbors kids cruddy drone offline. https://www.youtube.com/watch?v=sW7KrZKfmvg. ([n. d.]).
[27] Johann-Sebastian Pleban, Ricardo Band, and Reiner Creutzburg. 2014. Hacking and securing the AR. Drone 2.0 quadcopter: investigations for improving the security of a toy. In *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 90300L–90300L.
[28] Thomas Ricker. 2016. Watch a drone hack a room full of smart lightbulbs. https://www.theverge.com/2016/11/3/13507126/iot-drone-hack. (Nov. 2016).
[29] E. Ronen, A. Shamir, A. O. Weingarten, and C. O'Flynn. 2017. IoT Goes Nuclear: Creating a ZigBee Chain Reaction. In *2017 IEEE Symposium on Security and Privacy (SP)*. 195–212.
[30] Rebecca J. Rosen. 2013. So This Is How It Begins: Guy Refuses to Stop Drone-Spying on Seattle Woman. https://www.theatlantic.com/technology/archive/2013/05/so-this-is-how-it-begins-guy-refuses-to-stop-drone-spying-on-seattle-woman/275769/. (May 2013).
[31] Fred Samland, Jana Fruth, Mario Hildebrandt, Tobias Hoppe, and Jana Dittmann. 2012. AR. Drone: security threat analysis and exemplary attack to track persons. In *Proceedings of the SPIE*, Vol. 8301.
[32] Bruce Schneier. 2015. Is it OK to shoot down a drone over your backyard? https://www.cnn.com/2015/09/09/opinions/schneier-shoot-down-drones/. (Nov. 2015).
[33] Yunmok Son, Hocheol Shin, Dongkwan Kim, Young-Seok Park, Juhwan Noh, Kibum Choi, Jungwoo Choi, Yongdae Kim, et al. 2015. Rocking Drones with Intentional Sound Noise on Gyroscopic Sensors.. In *USENIX Security Symposium*. 881–896.
[34] Lisa Vaas. 2016. Sweden bans cameras on drones, deeming it illegal surveillance. https://nakedsecurity.sophos.com/2016/10/27/sweden-bans-cameras-on-drones-deeming-it-illegal-surveillance/amp/. (Oct. 2016).
[35] Gabriel Vasconcelos, Gabriel Carrijo, Rodrigo Miani, Jefferson Souza, and Vitor Guizilini. 2016. The impact of DoS attacks on the AR. Drone 2.0. In *Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR), 2016 XIII Latin American*. IEEE, 127–132.
[36] John Villasenor. 2013. Observations from above: unmanned aircraft systems and privacy. *Harv. JL & Pub. Pol'y* 36 (2013), 457.
[37] Paul B Voss. 2013. Rethinking the regulatory framework for small unmanned aircraft: the case for protecting privacy and property rights in the lowermost reaches of the atmosphere. In *Unmanned Aircraft Systems (ICUAS), 2013 International Conference on*. IEEE, 173–178.
[38] Sander Walters. 2016. How Can Drones Be Hacked? The updated list of vulnerable drones & attack tools. https://medium.com/swalters/how-can-drones-be-hacked-the-updated-list-of-vulnerable-drones-attack-tools-dd2e006d6809. (Oct. 2016).
[39] Olivia B. Waxman. 2015. World's Most Embarrassing Dad Has Drone Follow Daughter to School. https://time.com/3831431/dad-daughter-drone-school/. (April 2015).
[40] Sadie Whitelocks. 2015. Real helicopter parenting! Concerned father uses a DRONE to follow his eight-year-old daughter to school. http://www.dailymail.co.uk/news/article-3052784/Real-helicopter-parenting-Concerned-father-uses-DRONE-follow-eight-year-old-daughter-school.html. (April 2015).
[41] John Zorabedian. 2015. Teen cooks a turkey with flame-shooting drone. https://nakedsecurity.sophos.com/2015/12/11/teen-cooks-a-turkey-with-flame-shooting-drone/. (Dec. 2015).
[42] John Zorabedian. 2016. This state wants to ban gun-toting, flame-shooting, gas-spraying drones. https://nakedsecurity.sophos.com/2016/03/04/this-state-wants-to-ban-gun-toting-flame-shooting-gas-spraying-drones/. (March 2016).