Evaluation of DoS attacks on Commercial Wi-Fi-Based UAVs

Gabriel Vasconcelos[1], Rodrigo S. Miani[1], Vitor C. Guizilini[2] and Jefferson R. Souza[1]

[1]Faculty of Computing, Federal University of Uberlandia, Brazil
[2]School of Information Technologies, University of Sydney, Australia

***Abstract*:** One of the biggest challenges for the use of Unmanned Aerial Vehicles (UAVs) in large-scale real-world applications is security. However, most of research projects related to robotics does not discuss security issues, moving on directly to studying classical problems (i.e., perception, control, planning). This paper evaluates the effects of availability issues (Denial of Service attacks) in two commonly used commercially available UAVs (AR.Drone 2.0 and 3DR SOLO). Denial of Service (DoS) attacks are made while the vehicles are navigating, simulating common conditions found both by the general public and in a research scenario. Experiments show how effective such attacks are and demonstrate actual security breaches that create specific vulnerabilities. The results indicate that both studied UAVs are susceptible to several types of DoS attacks which can critically influence the performance of UAVs during navigation, including a decrease in camera functionality, drops in telemetry feedback and lack of response to remote control commands. We also present a tool that can be used as a failsafe mechanism to alert the user when a drone is reaching out a determined flight limit range, avoiding availability issues.

***Keywords*:** UAVs, Network security, Denial of service attacks, Wi-Fi.

1. Introduction

Unmanned aerial vehicles (UAVs) are experiencing significant and quick progress, with large companies such as DJI, 3D Robotics, and Google successfully employing them in many tasks. These include farming [1], monitoring [2], search & saving [3] and mapping [4]. In all these situations, the aerial vehicle should operate, either remotely piloted by an expert or autonomously, for continued periods of time in unexplored dynamic areas.

A Drone is defined as a UAV that can be controlled either by control or computers, which are capable of producing autonomous behaviors up to different degrees of complexity [5]. While piloting a commercial drone is a relatively easy task, most people are not aware of the information being constantly streamed back to its base, which might include camera feed, laser or radar range data, inertial measurements, and global coordinates from GPS. Because of this constant data sharing, which might be sensitive, information security becomes critical, especially when autonomous aerial vehicles are involved [6]. A robot with no security protocols in place can be brought down immediately by attackers, or its information can be stolen for nefarious purposes, including video footage or its GPS log history. An attacker could even take full control of the robot and perform unexpected activities remotely with impunity, such as colliding with objects or people in the area.

In this paper, we propose an empirical analysis of different availability attacks on the AR.Drone 2.0 and 3DR SOLO, two widely popular commercial drones. Availability is a computer security principle which refers to the ability of a user to access information or resources in a system. One example of availability attack is known as Denial of Service (DoS), where the attacks seek to make a computer system unavailable to its intended users. The goal is to measure how effective these attacks are in both accessing the data contained in each drone and also hindering the communication between the drone and controller while maintaining a safe environment for testing. Three different DoS tools are considered here: Low Orbit Ion Cannon (LOIC) [7], Netwox [8], and Hping3 [9], each with different characteristics that might be more suitable to exploit certain types of vulnerabilities.

This work is an extended version of the conference paper [10]. The main differences of this work to the previous one are: presenting, to the best of our knowledge, the first evaluation of DoS attacks in the 3DR SOLO drone; performing a comparison between the vulnerabilities found in the AR. Drone 2.0 and 3DR SOLO drone and providing a tool that runs inside the ROS (Robot Operating System) platform and could be used to mitigate availability issues such as losing the control of the drone to another user (fly-away attacks).

Our proposed methodology involves: 1) the delimitation of all related variables (i.e., controlling pilot, drone being monitored and the third-party attacker); 2) the explanation and discussion of the specific properties of each drone; 3) how they might affect possible security breaches; and 4) the step-by-step description of each attack performed, to facilitate the reproduction of results. The methodology can be easily extended to include other type of DoS attacks and drones, and the detailed instructions to reproduce these attacks are available in[1]. Similarly, the code to the system that alerts the pilot about likely communication issues between the controlling device (laptop or smart-phone) and the UAV is available in [2].

Evaluating the impact of attacks and consequently identifying new threats to drones would help create a discussion between the research community and vendors about what security vulnerabilities should be taken into consideration when designing new products.

The remainder of this paper is divided as follows: Section 2 provides an overview of computer security, including theoretical background and different reconnaissance and DoS attacks that are used in this work. A review of the current state-of-the-art in various applications of drone security can be found in Section 3. Section 4 introduces the proposed

[1] https://github.com/jrsouza/dos_attacks

[2] https://github.com/jrsouza/alert_system.git

methodology, describing in detail the circumstances in which each experiment took place, how it was validated and the specific configurations for each DoS attack tool and drone. Section 5 presents and discusses the proposed experiments, including a detailed analysis of results and their meaning. Finally, we conclude the paper in Section 6 and delineate some directions for future work.

2. Background

Here we provide an overview of the theoretical background of our experiments. First, we discuss computer security concepts, then the notion of reconnaissance attacks, and finally the fundamentals of denial of service and fly-away attacks.

2.1 Computer Security Principles

The ubiquitous use of computational networks, not to mention the broader aspect of technological systems, have caused a massive change in how our society currently functions, ranging from the quick dissemination of smartphones to the mass use of online transactions and cloud computing. Given this new reality, the need to to protect data related to civilians, companies and even governmental bodies or the military is more important than ever.

The field of computer security [11] relates to the task of making computational systems more robust to third-party abuse. One example of such violence is the activity known as cyber-attack. Cyber-attacks can be defined as a set of malicious activities to disrupt, deny, degrade or destroy information and service in computer systems [12]. Any action taken to undermine the functions of a computer network or device can be viewed as a cyber-attack [13]. Ponemon Institute, in a recent survey conducted in 237 separate companies [14], showed that the mean annualized cost for protecting and dealing with cyber-attacks, for a U.S. organization, is around $17 million per year.

Examples of cyber-attacks include [12]: viruses attached to emails, probing of a system to collect information, malicious code that replicates itself in order to spread to other computers (computer worms), unauthorized usage of a system, flooding a targeted computer resource with superfluous requests in an attempt to overload the system (DoS attack), or exploiting a bug in software to modify system data. Some approaches that attackers can use to gain access to a system or limit the availability of that system include social engineering, masquerading, exploiting a vulnerability, and abuse of functionality.

Three different aspects decompose computer security: confidentiality, integrity, and availability. A cyber- attack can be executed through the data stream on networks and aims to compromise each one of these aspects.

Jonsson and Pirzadeh [15] define each aspect as follows:

– **Confidentiality:** Ability to prevent and/or hinder information disclosure to third-parties that should not have such access.

– **Integrity:** Ability to protect against the improper modification and/or destruction of information.

– **Availability:** Ability of the called-upon service to deliver the relevant information.

The scope of the paper is studying availability attacks on drones or harming the availability of the communication channel between the pilot and the UAV. An attacker, for example, could try to exhaust computer resources from the drone by sending multiple requisitions to the drone so it cannot communicate with the pilot. Such attacks are commonly referred to as Denial of Service (DoS). This paper focuses on availability attacks that may occur in two commercially available drones: the AR.Drone 2.0 and the 3DR SOLO.

2.2 Reconnaissance and Scanning

When a cyber-attack is being planned, the initial step usually consists of gathering information about the net- work that will be targeted. This is formally known as reconnaissance and includes social engineering and automated tools to extract as much knowledge of the tar- get as possible, for example, IP addresses and uniform resource locators (URLs) [16].

In the next step, the attacker uses information gathered in the reconnaissance phase to discover active hosts on the network and information about the hosts, such as operating system, active ports, services, and applications. This phase is called scanning. Running a port scan on a given IP address is one of the most used automated scanning attacks. A port scan is used to check for open or closed network ports and used or unused services. The services may or may not have a vulnerability that the attacker could exploit [17]. An Internet Control Message Protocol (ICMP) port scan is used to check the availability of a target device and the fingerprint of the target operating system.

In this work, we use Nmap [18] to scan the UAVs. Nmap is a free open source network scanning utility, that is available under the GNU General Public License as published by the Free Software Foundation [19]. It runs on most operating systems including Linux, Windows, and MacOSX and IT security professionals widely use it. Nmap is the most commonly used network scanner, and many third-party tools integrate with Nmap, such as Linux distributions

2.3 Denial of Service Attacks

A DoS attack is represented by an effort of an attacker to intercept legal users of a service from using the coveted resources [20]. A common variant of this attack is called Distributed Denial of Service (DDoS) and consists of multiple devices targeting the same computing resource.

Reports from the annual Verizon Data Breach Investigation Report [21], [22] pointed out a rising trend in DoS attacks, in particular those related to DDoS incidents. One example is an attack that exploited a large number of insecure Internet of Things (IoT) devices. In September 2016, an IoT network built from the Mirai malware was responsible for a massive attack targeting a security blog. Months later other Mirai-based attacks were reported at the French web host OVH and the DNS service provider Dyn. According to [23], Mirai's strategy is straightforward: it uses a list of common usernames and passwords to locate under-secured IoT devices (home-routers, network-enabled cameras, and digital video recorders) that could be remotely accessed (e.g., admin/admin).

One type of DoS attack is known as resource attack [24]. Such attacks overwhelm the victim's computer or network resources by sending continuous streams of illegitimate packets. Since there is no simple way to differentiate the

valid packets from the malicious pack- ets, it can be hard to defend against this type of attack. SYN Flood is a resource attack that exploits a flaw in the TCP three-way handshake. The attacker sends several SYN requests to the target and does not answer to the server's SYN-ACK response. The server continues to wait for an ACK packet for each one of these requests, saving resources for each of the requests and eventually preventing the establishment of new connections.

The fast productization, increased demand, and adoption of UAVs devices might make it hard for the industry to evaluate critical aspects of security. For this reason, we believe that the AR. Drone and 3DR SOLO might also be vulnerable to similar security issues as previously described. In fact, several works analyze security aspects of UAVs [25], [26]. In Section 3 we will provide an overview of them. In this paper, we are interested in analyzing the impact produced by DoS resource attacks on the AR.Drone 2.0 and 3DR SOLO.

2.4 Fly-away attacks

Due to the lack of proper authentication mechanisms, drones could be stolen by anyone within the Wi-Fi signal range of the UAV. This attack is also called fly-away attacks or hijacking. As discussed in [27], a simple fly- away attack scenario consists of an attacker driving a car near to the drone. The attacker might hijack the drone and runs away in the getaway car.

This type of attack can also be seen as a availability issue since the owner of the drone might not be able to establish a connection with it during the attack. [28] presents the steps for an attacker performing a fly-away attack on the AR.Drone. The attack exploits the pairing mechanism of the drone that is based solely on using the MAC address of a wireless network adapter. Another example of a fly-away attack is showed in [29]. The author developed a tool that can be used together with a drone to scan, exploit, and wirelessly take over (fly-away) other drones within Wi-Fi distance. Our idea is to create a tool that alerts the user when the drone is out of a predefined limit flight range. This could be used as an alert to prevent such attacks.

3. Related Work

There are several designs for UAV platforms. The primary distinction concerning their capability and ease of operation is their physical size and power, which limits their payload carrying capacity, operating altitude, and range [30]. Regarding scope, UAV platforms can be divided into Large (500km operating range and 200kg of payload size), Medium (500km operating range and 50kg of payload size), Small and mini (10km operating range and less than 30kg of payload size), and Micro and Nano (less than 10km operating range and less than 5kg of payload size). Their size and power also define the applications that can be supported by each class of UAV. For example the AR.Drone is an example of a Micro and Nano UAV that can be used for trajectory tracking in indoor environments [31].

Vattapparamban et al. [26] and Altawy and Youseef [25] examines the primary security, privacy, and safety aspects associated with the use of civilian drones. Vat- tapparamban et al. [26] review several different security challenges related to using of drones and also provided results on deauthentication attacks, GPS spoofing attacks and drone hijacking using the Wi-Fi Pineapple, a well-known rogue access point. Altawy and Youseef [25] claim that the design of a UAV system should in- corporate mitigation techniques that address the possible security threats. Disclosing the values of real-time data, gaining a prior knowledge of the system parameters and interrupt the regular operation of the system are some of the risks that can be carried out against on UAVs. The authors also identify the following security requirements for a secure UAV operation: authorized access, availability, information confidentiality, information integrity, system integrity and accountability of actions. Availability, for instance, is a property that should guarantee that the UAV performs their required functions without disruption during its operational period.

Regarding the state-of-the-art attacks on drones, several models were investigated by the Federal Trade Commission [32]: Cheerson CX-10W, Parrot AR.Drone Elite, and Hawkeye II from DBPOWER. Some of the security flaws found by the researchers included: unencrypted data traffic and open access points causing at least two of them to fall from the sky. Reference [33] investigated vulnerabilities in DJI Phantom drones. In earlier versions (before version 3), it was possible to change the SSID of the access point, causing the Drone to disconnect from the controller. Newer versions of DJI Phantom drones have open ftp, telnet, and even the ssh service running. However, all these services are password protected by default. At last, [27] investigate the family of Discovery U818A drones. The authors were able to perform fly-away attacks (running away with the Drone), lock-out attacks (preventing the legitimate owner of the device from connecting to it) and stealing user data. Table 1 summarize previous work that provide empirical evaluation of drone security. Most of the references propose experimental attacks. However, none of them compare the same attacks performed in different drone models. Since the AR.Drone is one of the most evaluated drones, our idea is to analyze the behavior of another device, 3DR SOLO (which was not studied yet), when receiving the same attacks as the AR.Drone.

The security of Parrot AR.Drone 2.0 quad-copter was studied in the following papers: [28], [34], [35]. Some of the vulnerabilities studied in these papers include the presence of unencrypted connections which could reveal confidential data and might lead to hijacking the drone and some DoS attacks using the Hping3 tool. In our previous work [10], several important issues that were left behind in the previously mentioned works are discussed. Some of them include: analyzing network delay caused by the attack, investigating other DoS attack types, such as TCP SYN (that can be performed using LOIC and Netwox) and evaluating the impact of the attack on the drone functionalities.

Our goal here is to replicate the same attacks previously performed in the AR. Drone 2.0 in a different UAV model named 3DR Solo. We will establish a comparison with the attacks shown in both models and also develop and test a tool to assist users from both of the platforms during fly-away attacks.

4. Methodology

Our goal is to check UAV's behavior during availability attacks. First we will evaluate DoS flood attacks using three

different tools. Next we will propose a experiment to validate a tool that can be used during situations where the pilot might lose the communication with the drone, for instance, in a hijacking attempt. We use two UAVs platforms to conduct experiments: AR.Drone 2.0 and 3DR SOLO UAVs.

4.1 DoS attacks using Specific Tools

The methodology consists of five steps, which are based on our previous work [10], and comprises an indoor scenario and two actors: a pilot and an attacker who launches reconnaissance and DoS attacks on the UAV:

1. Establish a connection between the pilot and UAVs (AR.Drone and SOLO);
2. Pilot sends a set of commands to UAVs (taking off, short flights and landing) to understand its behavior in normal conditions (no attackers);
3. Establish a connection between attacker and UAVs;
4. The attacker makes reconnaissance attacks on UAVs using port scan tool;
5. While pilot is sending a series of commands to UAVs, an attacker uses information obtained in step 4 to launch a DoS attack towards UAVs.

The components involved in our experiments are depicted in Figure 1, and below we give a brief explanation of each one:

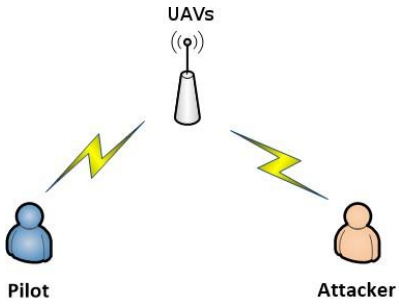


Figure 1. Main components of the proposed experiment in an indoor scenario [10].

– **Pilot:** It is a person that connects to a wireless network of the AR.Drone 2.0 or SOLO UAVs using a laptop. Also, this human uses a USB joystick plugged into the laptop to control the UAVs. We use ROS[3] to interface laptop and UAVs;

– **AR.Drone 2.0:** It is a quad-rotor helicopter that can be guided either by a mobile device (iOS or Android systems) or a laptop (our case). This drone has some features: Wi-Fi b/g, MEMS 3-axis accelerometer, 2-axis gyroscope, propellers, four brushless motors, lithium-polymer battery, front and vertical cameras, and an ultrasonic sensor;

– **SOLO:** It is a quad-rotor helicopter that can be piloted by a mobile device on the iOS or Android systems. This drone has some features: Wi-Fi, LSM303D integrated accelerometer, L3GD20 gyro, propellers, four motors, lithium-polymer battery, and GoPro HERO3 camera;

– **Attacker:** The person connecting to the drone's network using a Wi-Fi connection. He or she uses software tools to scan the UAV ports before applying different DoS attacks (section 4.1.2).

4.1.1 Reconnaissance

Since the chosen UAVs are Wi-Fi-based devices, reconnaissance attacks, in our context, consists of using the Nmap tool [18] to perform port scans at the drones. Nmap could also link a port to its state (open, closed or filtered) and respective service name and even which operating system the targeted system is running. According to Nmap manual [18], an open port means that an application on the target machine is listening for connections/packets on that port. Filtered means that a firewall or another network obstacle is blocking the port so that Nmap cannot tell whether it is open or closed. Closed ports have no application listening on them.

4.1.2 DoS Attacks Tools

The next step involves performing the DoS attacks on the drone. The Hping3 tool was an obvious choice to do that, since it was already used to perform DoS attacks in previous works [10], [34]. However, since Hping3 is a very simple tool, we decided to conduct DoS attacks using two other tools. Our first choice was LOIC (Low Orbit Ion Cannon) which is a powerful DoS tool that is used by the hacker group Anonymous [37]. We also picked a popular open source network tool-set named Netwox. Among other features, this tool can also be used to execute DoS flood attacks. Next, we provide a brief description of each tool.

LOIC was developed by Praetox Technologies as a tool for software stress-testing [7]. The original code is available on the Praetox website, that is no longer maintained. However, it has been modified and updated by the public, and the Anonymous group has used it as a tool for DDoS attacks [37]. It is straightforward to use, with the Windows version requiring a target address before clicking the "IMMA CHARGIN MAH LAZER" button. Possible options include different packet types (HTTP, UDP or TCP), port numbers and several others. Here we focus on the "TCP" packet type and set a specific port that will be used to launch the TCP SYN resource attacks.

Netwox can be used to perform multiple network tests and also some attacks. We will use the Netwox tool number 76 to launch the SYN flood attack.

Hping3 is a packet generator and analyzer for the TCP/IP protocol (Internet Protocol - IP). The new version of the Hping3 is programmable using the TCL language, human-readable description of TCP/IP packets that the programmer can write scripts related to low- level TCP/IP packet manipulation. We will use Hping3 to launch a resource DoS attack by sending multiple spurious packets to the UAVs (AR.Drone and SOLO) (–fast –flood option).

4.2 Fly-away attack - drone losing the signal from the controlling device

Our idea here is to develop a mechanism that can be used to alert the pilot during situations where the drone is suffering availability issues due to its distance from the controlling device (the laptop, in our case). A fly-away attack is an example of a situation where our method can be employed.

The methodology employed to evaluate the proposed method consists of four steps:

1 Establish a connection between pilot and AR.Drone 2.0;
2. Pilot sends a set of commands to AR.Drone 2.0 (taking off, short flights);

[3] Robot Operating System: http://wiki.ros.org/

3.Initialize the ORB-SLAM2[4] package using the monocular camera of the AR.Drone;
4 Initialize the alert system to alarm the pilot that UAV can lose the signal.
The main components of this outdoor experiment are the pilot, AR.Drone 2.0, ORB-SLAM2 and the Alert System itself. Some details of the ORB-SLAM2 and Alert System are presented above.
– **ORB-SLAM2:** It is a real-time SLAM (Simultaneous Localization and Mapping) library for Monocular, Stereo and RGB-D cameras that computes the camera trajectory and a sparse 3D reconstruction. It can detect loops and re-localize the camera in real time. We use ORB SLAM2 [38] to locate the AR.Drone 2.0 in the outdoor scenario with its front camera, so this package provides us a current position of the AR.Drone 2.0. Figure 2 presents two images, the first shows the outdoor scenario of the experiment captured by a video camera (Sony N50). The second image shows the front image of AR.Drone 2.0 using ORB SLAM2, after the AR.Drone 2.0 find yourself in the scenario; see the features in the outdoor environment.

Figure 2. The first image shows the outdoor scenario of the experiment captured by a video camera. Second image shows the front image of AR.Drone 2.0 using ORB SLAM2 package.

– **Alert System:** We developed a code in the programming language C/C++ that uses a topic generated by the ORB-SLAM 2.0 package from ROS (position). We calculated the Euclidean distance between the starting position (position on AR.Drone 2.0 takeoff) and the current position of AR.Drone 2.0. In this way, it is possible to calculate the distance in meters between the two points. After the AR.Drone 2.0 moves away from the starting point and when it reaches more than 10 meters, a base station (laptop) emits a beep sound (we use the canberra-gtk-play library[5]), and then the pilot could return the AR.Drone 2.0 to a safer area and the beep is interrupted. It is important to note that the meters value was chosen after several outdoor experiments with the AR. Drone 2.0. This value could be altered by the user.
Figure 3 shows when the audible alert is issued after reaching more than 10 meters of the distance the AR.Drone 2.0 from the initial takeoff. The left image shows in meters the distance that the AR.Drone 2.0 is from the starting point, and when it exceeds 10 meters it is displayed on the base station screen ”Watch out!” and it will emit one beep to the pilot (person).

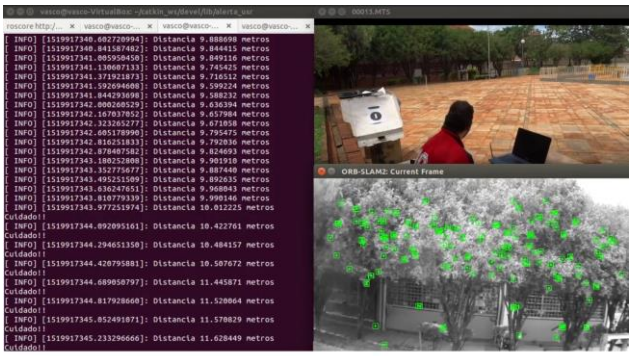


Figure 3. The left image shows in meters the distance that AR.Drone 2.0 is from starting point, and when it exceeds 10 meters it is displayed on base station screen to ”Watch out!”.

5. Experimental Results

Using the previously described methodology, we conducted three types of DoS attack on two robots - AR.Drone (Figure 4) and Solo (Figure 5).

5.1 Experiment 1 - Evaluation of the DoS attack tools

We assessed the aerial robots inside the university in a real-time fashion using a standard laptop with the Ubuntu 14.04.
Table 2 shows the comparison of the technical specifications between the UAVs (AR.Drone 2.0 and 3DR SOLO) such as camera, processor, Wi-Fi, flight time and control type. It can be seen that the SOLO has an advantage in the camera question because use a gimbal to stabilize the camera in the flight. The SOLO processor is twice the processing capacity in comparison to AR.Drone 2.0. The capacity of SOLO’s battery is more significant than Ar.Drone, but SOLO ends up losing in the weight category because it is much more substantial, so the autonomy of flight two ends up being very similar. The platforms supported to control the UAVs are different but run on the same system (IOS or Android).
The first step is to establish a connection between the pilot and drone. This procedure is easy since the AR.Drone 2.0 and 3DR SOLO (UAVs) works as an Access Point, creating a wireless network under the name “ardrone2-044078” and ”Solo 0342”. These networks have no security capabilities which means that any device equipped with a wireless network card and within range may be able to establish a connection with the UAVs. A wireless network with no protection abilities signifies a severe issue for assuring confidentiality and integrity between pilot and UAV [39].

[4] ORB-SLAM2: https://github.com/raulmur/ORB SLAM2

[5] www.systutorials.com/docs/linux/man/1-canberra-gtk- play/

Figure 4. Parrot AR.Drone 2.0 robotic platform used in the experiments.

Figure 5. 3DR SOLO robotic platform used in the experiments.

Table 2. Comparison of UAVs technical specifications.

	UAVs	
	AR.Drone 2.0	**SOLO**
Camera	HD Front Camera - 720p	Gopro hero 3 - 1080p
Processor	ARM Cortex A8 of 1GHz	two ARM Cortex A9 of 1GHz
Wi-Fi	Wi-Fi 802.11 b/g/n	3DR Link secure Wi-Fi
Battery	1500 mah	2500 mah
Weight	0.380 kg	1.5 kg
Flight Time	18 minutes	20 minutes
Control	IOS or Android	IOS or Android

After that, instructions were sent to the UAVs to understand its behavior in normal conditions, or with no attackers. We estimated the network latency by calculating the Round Time Trip (RTT) (time required for a packet to travel from a source to a target and back again) between the pilot and UAVs for 5 (five) minutes with the ICMP ping. The average network latency for this period was 20.92 ms (AR.Drone 2.0) and 9.21 ms (SOLO), as shown in Tables 3 and 4. As expected, due to better technical specifications, a network packet ex- changed between the laptop and the AR. Drone 2.0 could take two times more to reach the destination when compared to the 3DR Solo.

Table 3. Average latency for the DoS attack tools (AR.Drone 2.0 Parrot).

Regular conditions	**Hping3**	**LOIC**	**Netwox**
20.92ms (ICMP)	**455.82ms**	-	-
24.41ms (TCP Port 21)	-	**188.03ms**	**260.58ms**
57.60ms (TCP Port 23)	-	90.54ms	212.90ms
81.97ms (TCP Port 5555)	-	-	110.82ms

Table 4. Average latency for the DoS attack tools (3DR SOLO).

Regular conditions	**Hping3**	**LOIC**	**Netwox**
9.21ms (ICMP)	**247.25ms**	-	-
12.03ms (TCP Port 22)	-	171.13ms	189.15ms
7.12ms (TCP Port 53)	-	121.38ms	235.17ms
8.44ms (TCP Port 14560)	-	**181.47ms**	**324.89ms**

Next, the attacker creates a connection with both UAVs (AR.Drone and SOLO) and perform reconnaissance attacks. As previously stated, any device equipped with a wireless network card will be able to connect to AR.Drone and SOLO. Using a standard laptop, the attacker readily joined the AR.Drone and SOLO wireless network. The attacker can assume that the AR.Drone's IP address is "192.168.1.1", and for SOLO, the IP address is "10.1.1.10", after using the Nmap scanning on ports (Figure 6). Using this information, the attacker could launch port scan attacks using Nmap tool.

Figure 6 shows the results produced by the Nmap. We can see the IP address of the AR.Drone "192.168.1.1" and 3DR SOLO "10.1.1.10". Also, the ports number and protocol, service name and state. Three TCP ports, representing three different services of the AR.Drone were available: 21 (FTP), 23 (Telnet) and 5555 (Freeciv - AR.Drone video camera streaming). For the SOLO, two different available services were found: 22 (The Secure Shell (SSH) Protocol) and 53 (Domain Name Server). Both ports 21 and 23 provide direct access to the AR.Drone 2.0 through the following shell commands: "ftp 192.168.1.1" and "telnet 192.168.1.1". None of these services are password protected. An attacker might use telnet to get a root shell and be able to execute malicious remote commands, for example, a complete shutdown of the system. At last, the port 22 also provides direct access to the

3DR SOLO using the shell command: "ssh 10.1.1.10". However, this mechanism is password protected.

(a) AR.Drone 2.0

(b) 3DR SOLO

Figure 6. Nmap scanning on ports.

The attacker now has a good knowledge of the system and may be able to launch DoS attacks on the UAVs. Nevertheless, the DoS attacks will be targeted in TCP ports; we also require to include the network latency in regular conditions for every found TCP port (21, 23 and 5555) for the AR.Drone 2.0 and TCP port (22, 53) for the 3DR SOLO. The average network latency for each case is depicted in Tables 3 and 4. The IP address of the AR.Drone 2.0 is "192.168.1.1" and the IP address of the 3DR SOLO is "10.1.1.10" The following DoS commands were executed from the attacker computer based on [10].

- "netwox 76 –dst-ip IP –dst-port 21 ", "netwox 76 –dst-ip IP –dst-port 23 " and "netwox 76 –dst-ip IP dst-port 5555 "
- "hping3 –fast –flood IP " ([34])
- LOIC was carried via GUI with parameters: IP address, Method TCP and ports 21 and 23 (LOIC does not support sending packets to port 5555).

The goal of all commands is to execute a Flood DoS attack on a certain target. LOIC and Netwox enable to choose a specific port as a target to the attack. The instructions for the DoS attack are shown in Figures 7, 8 and 9. Table 3 and 4 presents the values achieved by the DoS tools in each one of the attack rounds. Table 3 and 4 presents an increase in the network latency in milliseconds during the DoS attacks for the tools. Higher values of network latency are an indicator of connectivities issues between two devices. This means that sending illegitimate network packets to the AR.Drone 2.0 and 3DR SOLO caused a direct impact on its network resources, validating our experiment. The less powerful processor embedded in the Drones (especially for the AR.Drone 2.0) could be one of the reasons behind the success of the DoS resource attack. Since we performed flood attacks, analyzing how the TCP stack was implemented in each UAV could also provide some answers about the reasons behind the attack. Besides, the absence of basic security configurations (open wireless network and WPA, for instance) is also a factor that contributes to gain knowledge about the system and, consequently, allowing some attacks.

Tables 3 and 4 shows that the highest value of the average latency of the network obtained by the three DoS attack tools was the Hping3 tool with 455.82ms for the AR.Drone 2.0 and the Netwox with 324.89ms for the 3DR SOLO. Tables 5 and 6 shows the latency increase rate produced by three DoS attack tools. Example, 21.788 is the result of the division between 455.82 (AR.Drone 2.0 on attack) by 20.92 (regular network conditions) and 38.494 is the result of the division between 324.89 (3DR SOLO on attack) by 8.44 (regular network conditions) as seen in the Tables 3 and 4.

Table 5. Latency increase rate of the DoS attack tools (AR.Drone 2.0 Parrot).

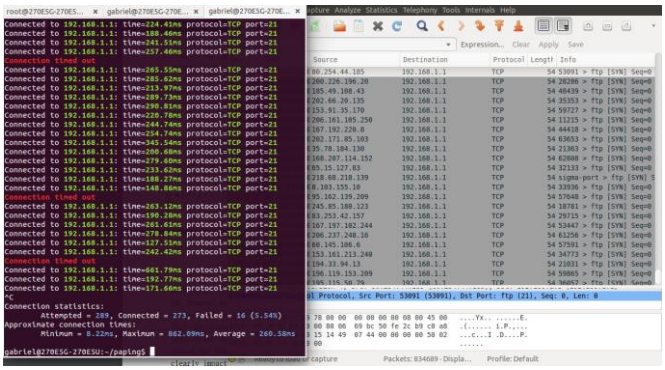
	Hping3	**LOIC**	**Netwox**
ICMP	**21.788**	-	-
TCP Port 21	-	7.702	10.675
TCP Port 23	-	1.571	3.696
TCP Port 5555	-	-	1.351

We can see in Table 5 that the Hping3 DoS attack tool produced the highest value of latency increase rate compared to the other two DoS attack tools (LOIC and Netwox) for the AR.Drone 2.0. Furthermore, Table 6 indicates that the Netwox DoS attack tool produced the highest value of latency increase rate compared to the other two DoS attack tools (LOIC and Hping3) for the 3DR SOLO. Deligne [34] also launched a DoS attack using Hping3. In his paper, the behavior of the drone is haphazard and gets out of control, either by hitting an obstacle or shutting down the system board in less than a second. We were not able to reproduce this behavior, even with a five-minute attack. We believe that the company (Parrot and 3DR) might have upgraded the firmware to deal with a high number of network packets sent to the drone. However, Hping3 can still be considered a serious threat to the AR.Drone as the flood attack implemented in the Netwox tool for the 3DR SOLO. In other words, the latency increase rate for both studied UAVs during an event of flood attack could pose serious risks to the pilot. Next we will present an impact of such attack.

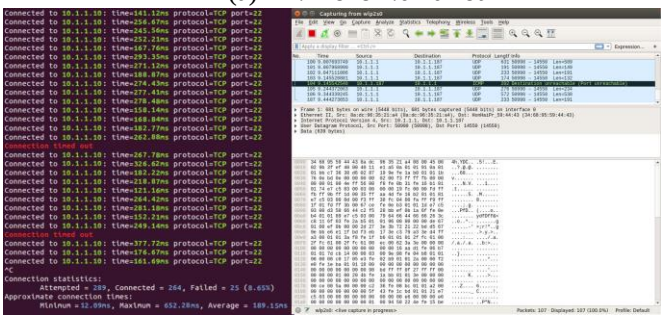
The impact of DoS on the studied UAVs can be noticed by running a video streaming application (port 5555 for AR.Drone and port 14560 for the 3DR Solo) while both drones are under attack.

Table 6. Latency increase rate of the DoS attack tools (3DR SOLO).

	Hping3	**LOIC**	**Netwox**
ICMP	26.701	-	-
TCP Port 22	-	14.225	15.045
TCP Port 53	-	16.873	33.116
TCP Port 14560	-	21.619	**38.494**

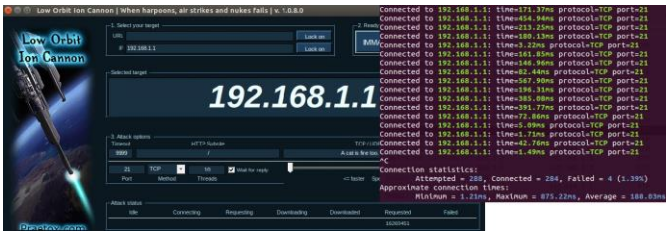


(a) AR.Drone 2.0 Parrot

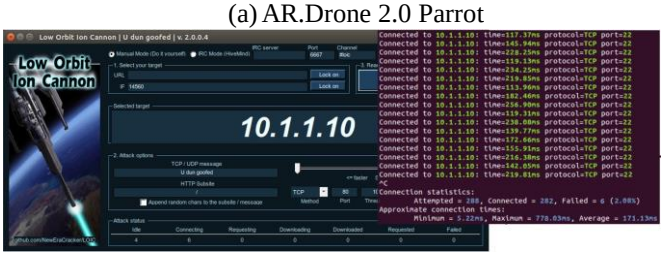


(b) 3DR SOLO

Figure 7. The Netwox can be seen on the left side. The Wireshark tool is executed and shown on the right side (a) - AR.Drone 2.0 and (b) - 3DR SOLO.

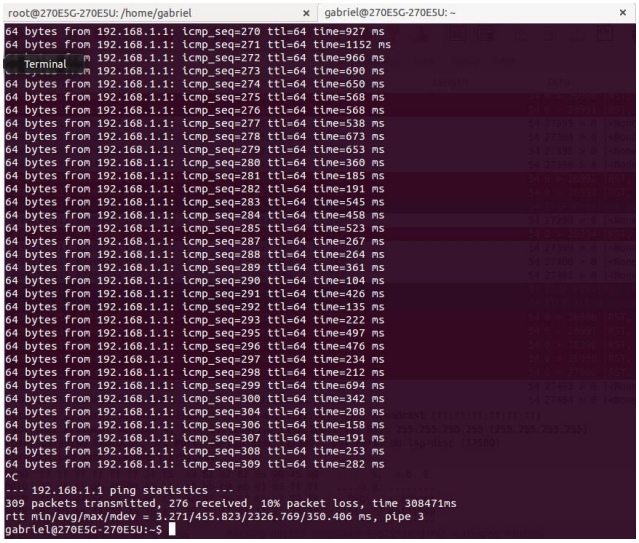


(a) AR.Drone 2.0 Parrot

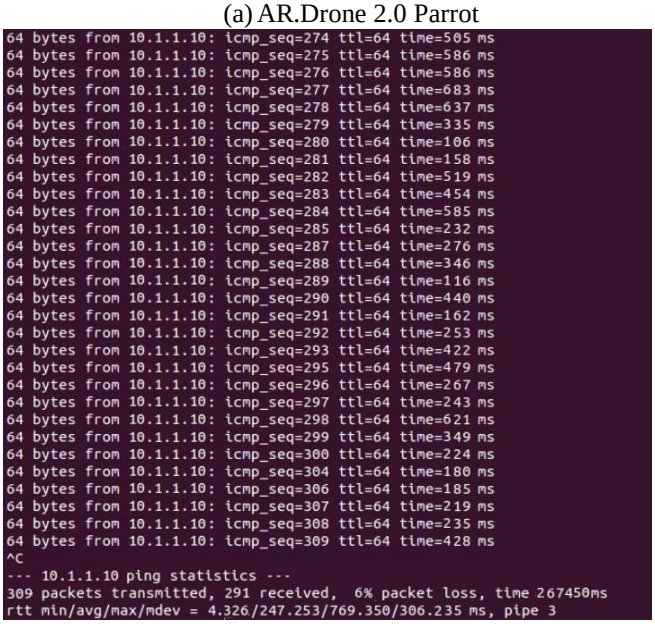


(b) 3DR SOLO

Figure 8. The GUI for the LOIC on port 21 can be seen on the left side. The average latency results by LOIC are shown on the right side (a) - AR.Drone 2.0 and (b) - 3DR SOLO.



(a) AR.Drone 2.0 Parrot



(b) 3DR SOLO

Figure 9. Hping3 is run to show results of average latency.

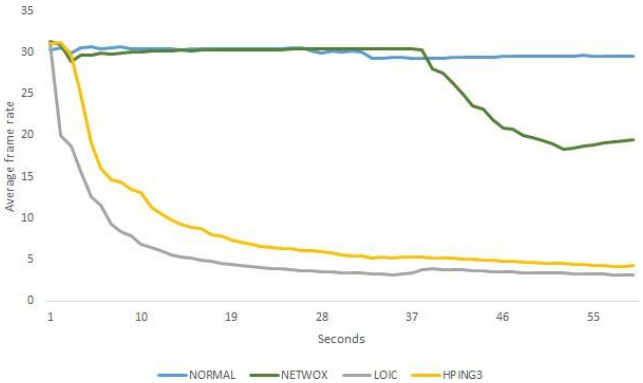


Figure 10. AR.Drone 2.0 camera average frame rate per seconds.

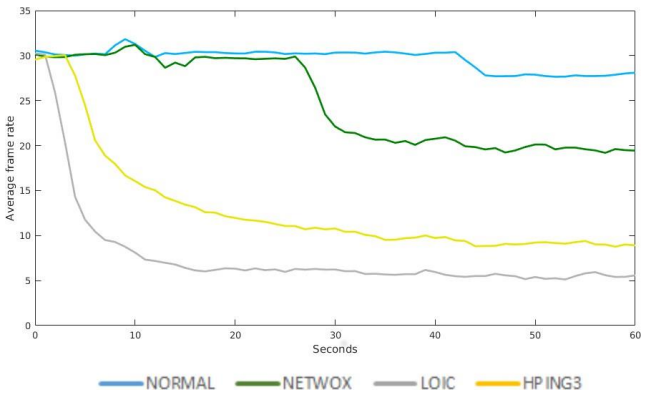


Figure 11. 3DR SOLO camera average frame rate per seconds.

Using the command "rostopic hz" in ROS we were able to obtain the video frame rate transferred between the pilot and the drone. Higher values indicate a good video quality. Figures 10 and 11 shows the average frame rate per second transferred between the pilot and both drones (AR.Drone 2.0 and 3DR SOLO) during the attack performed by the three tools.

It is possible to see that the Frame Rate (FR) is constant when there is no DoS attack in progress for both drones. However, during an attack, the FR dramatically decreases, which may have an impact on video quality. The tools that most influenced the average FR of the UAVs cameras were the LOIC and Hping3 tools for both drones (Figures 10 and 11). We have produced a video showing the attack tools that affected the average network latency (LOIC on port 21, Netwox on port 21 and Hping3). The influence of the DoS attack tools can be shown in the Video Streaming Application[6].

5.1.1 Comparison between AR.Drone 2.0 and 3DR SOLO

AR.Drone 2.0 is a simple and very cheap quadcopter compared to 3DR SOLO. On average, AR.Drone 2.0 costs around US$100, while the mean cost of 3DR SOLO (Bundle with Gimbal, Backpack, Battery, and 8 Propellers) is around US$700. The AR.Drone 2.0 was launched in 2012 while the 3DR SOLO was launched in 2015.

Thus, for experimental evaluation purposes, we expected 3DR SOLO to behave better than AR.Drone 2.0 regard- ing the DoS attacks. However, the experimental results showed that both UAVs had similar behavior when facing flood DoS attacks. Figures 10 and 11, show that the effects of DoS attacks performed by the three tools on the camera average frame rate is practically the same for both drones.

As for the impact of the attacks on network latencies, the UAVs presented different results. For AR.Drone 2.0 the Hping3 attack presented the highest value for the ICMP port, whereas for the 3DR SOLO the Netwox attack presented the highest value for port 14560. The latency of the 3DR SOLO network was higher compared to the AR. Drone 2.0 (Tables 5 and 6). Thus, even with 3DR SOLO costing more, it presents the same network availability vulnerabilities. So, a more robust and more expensive drone could suffer from the same problem of an already outdated competitor. This result leads to relevant questions, that are not new in the security community [40], but requires attention from the IT industry in general: are project designers of IoT (UAVs, sensors and other smart-objects) products thinking about security? Or do they want to put their products on the market as soon as they can? What is the role of the vendors in this environment?

5.2 Experiment 2 - Fly-away attack

We tested the maximum distance of the drone for the alert on Campus University to validate the experiment, AR.Drone 2.0 fly to a limit and emit a warning beep to the user. The idea here is to simulate a kind of availability issue know as hijack or a fly-away attack were the pilots could lose the control of the UAV. The tests were done in a real field with external environment characteristics, such as the light pole, trees, and other features. Video of this experiment is available in https://youtu.be/uhW4UIaSAao. Figure 12 shows the real field of the maximum distance of the drone for the alert.

Figure 12. The external field using the AR.Drone 2.0.

A standard laptop with Ubuntu 14.04 OS was used as equipment, with ROS indigo installed along with the AR.Drone 2.0 drivers for ROS. This laptop was used to guide the drone during the tests.

To calculate the initial and current distance of AR.Drone 2.0, we developed a C/C++ code in the format of a ROS package named as Alert System. So that it works in real time during the flight. This system reads values from a topic ROS generated by the ORB SLAM2 package called "/orbslam2/pose", which provides several benefits such as position: x, y, and z. Therefore, with this information, it is possible to calculate the distance between the AR.Drone 2.0 and the object that is highlighted or visible by the front camera of the AR.Drone 2.0. Then, if the ORB SLAM2 package cannot identify the features in the environment, it will not be possible to calculate the current position of the AR.Drone 2.0. Figure 13 shows the alert system working, which the user can see the distance of the AR.Drone 2.0 from the initial position (taking off of the AR.Drone 2.0) to the current position that the drone is in motion.

Figure 14 shows that the ORB SLAM2 can identify the visual features of the external environment. We ran the test as follows, started the drivers of the AR.Drone 2.0 on the ROS, then started the ORB SLAM2 package and then the Alert system package. After initiating the necessary software, the pilot took off the AR.Drone 2.0 with altitude approximately to 2 meters. Then it was moved forward until reaching 10 meters. Upon entering this mark, a visual and audible alert ("beep") is issued, then while the AR.Drone 2.0 returning from the 10 meters, the alert system is interrupted.

[6] Video Streaming Application: https://youtu.be/6QlGMn3_9XQ

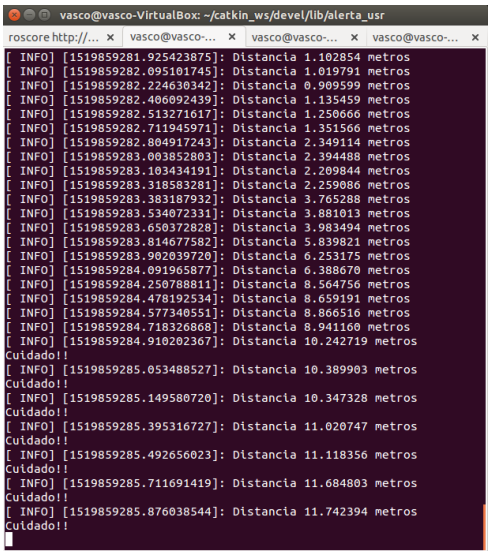


Figure 13. The alert system package of the ROS in working.

In the end, all tests of this the proposed methodology were applied and executed successfully, validating the experiment. It is important to note that the 10 meters mark can be changed by the pilot.

The user alert system was performed in an experimental test for AR.Drone 2.0 specifically, as it was verified in the experiments with real scenarios; which the same showed when a distance was near 15 meters, the UAV lost the connection with the base station. Thus, this experiment was analyzed to prevent pilot from losing contact with the UAV. However, the most modern UAVs present a considerable distance with their base station, since they are piloted by radio frequency; un- like the AR.Drone 2.0 that was piloted with the USB joystick within a Wi-Fi distance. Thus, we understand that the test performed can be integrated with other UAVs for other distances, since the current applications for Drones can vary from medical services to delivery of services among other uses.

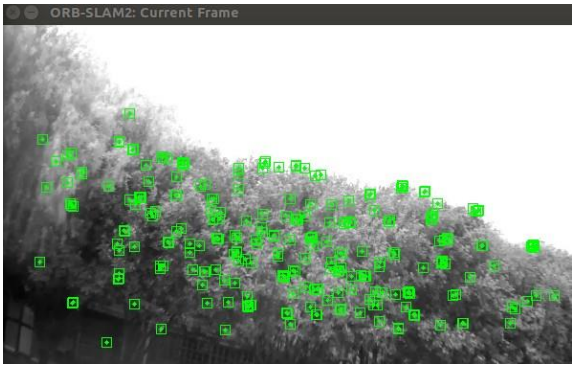


Figure 14. ORB SLAM2 package during the outdoor test.

6. Conclusions

We evaluated availability issues of two popular UAVs: AR.Drone 2.0 and 3DR SOLO. Two attacks against the availability of both drones were proposed: DoS attacks and fly-away attacks (UAV losing signal from the base station). DoS tools (LOIC, Netwox, and Hping3) were executed on an external computer, simulating an attacker attempting to bring down the drone. After a series of comparisons, the Hping3 and Netwox tools showed the high impact on the UAV and resulted in the lowest average frame rate of the AR.Drone 2.0 and 3DR SOLO camera respectively. The results indicate that even with better hardware and software configurations, the performance of the 3DR SOLO during the DoS attacks was similar to the AR.Drone 2.0. This result indicates that UAVs companies still need to design and release products focusing on information security.

We understand that the contributions done in this article are a fundamental step towards enforcing bet- ter information security policies for UAVs. Besides con- ducting newer experiments to evaluate the security of such devices, it would be necessary to create security regulations for UAVs and develop mechanisms to detect and mitigate attacks that could be embedded in UAVs, similar to home routers that are usually equipped with firewalls and access control lists.

As future work, we will develop new experiments to conduct Distributed Denial of Service (DDoS) at- tacks. We understand that a DDoS attack may produce an even more meaningful impact on the investigated UAVs. Furthermore, we intend to study the anomaly detection technique, which could be applied for detecting DoS attacks. Finally, we will test another aerial robotics platform, known as Phantom 4 Pro, DJI company. These can help to build a complete experiment and a better understanding of the security vulnerabilities in UAVs. Finally, the solution provided for the fly- away attack is not optimal and can affect the efficiency of the UAVs. Moreover, we investigate concerning authentication failure or how someone else can hijack and take away the Drone.

7. Acknowledgment

The authors would like to acknowledge the support granted by CNPq (process 400699/2016-8), UFU, FACOM and Faculty of Engineering & Info. Technologies, The University of Sydney, under the Faculty Research Cluster Program.

References

[1] S. Efron, The Use of Unmanned Aerial Systems for Agriculture in Africa. Doctoral dissertation, The Pardee RAND Graduate School, 2015.

[2] S. D'Oleire-Oltmanns, I. Marzolff, K. D. Peter, and J. B. Ries, "Unmanned aerial vehicle (UAV) for monitoring soil erosion in Morocco," Remote Sensing, vol. 4, no. 11, pp. 3390–3416, 2012.

[3] V. B. Hammerseth, "Autonomous unmanned aerial vehicle in search and rescue," Master's thesis, Institutt for teknisk kybernetikk, 2013.

[4] H. Eisenbeiss, "The Potential of Unmanned Aerial Vehicles for Mapping," Photogrammetrische Woche 2011, pp. 135–145, 2011.

[5] R. L. Wilson, "Ethical issues with use of Drone aircraft," 2014 IEEE International Symposium on Ethics in Science, Technology and Engineering, ETHICS 2014, 2014.

[6] R. Reshma, T. K. Ramesh, and P. S. Kumar, "Security incident management in ground transportation system using uavs," in Computational Intelligence and Computing Research (ICCIC), 2015 IEEE International Conference on. IEEE, 2015, pp. 1–7.

[7] P. Farina, E. Cambiaso, G. Papaleo, and M. Aiello, "Understanding DDoS Attacks from Mobile Devices," Int.

Conf. on Future Internet of Things and Cloud, pp. 614– 619, 2015.

[8] "Netwox," 2016. [Online]. Available: http://ntwox.sourceforge.net/

[9] "Hping3," 2016. [Online]. Available: http://www.hping.org/hping3.html

[10] G. Vasconcelos, G. Carrijo, R. Miani, J. Souza, and V. Guizilini, "The impact of dos attacks on the ar.drone 2.0," in XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR), 2016, pp. 127–132.

[11] J. Bau and J. C. Mitchell, "Security modeling and analysis," IEEE Security and Privacy, vol. 9, no. 3, pp. 18–25, 2011.

[12] A. A. Ghorbani, W. Lu, and M. Tavallaee, Network intrusion detection and prevention: concepts and techniques. Springer Science & Business Media, 2009, vol. 47.

[13] A. Hathaway, R. Crootof, P. Levitz, H. Nix, A. Nowlan, W. Perdue, and J. Spiegel, "The law of cyber- attack," California Law Review, vol. 100, no. 4, pp. 817– 885, 2012.

[14] Ponemon Institute, "2016 Cost of Cyber Crime Study and the Risk of Business Innovation," Tech. Rep., 2016.

[15] E. Jonsson and L. Pirzadeh, "A Framework for Sec. Metrics Based on Operational System Attributes," in Int. Work. on Security Measurements and Metrics, 2011, pp. 58–65.

[16] P. Engebretson, The basics of hacking and penetration testing: ethical hacking and penetration testing made easy. Elsevier, 2013.

[17] S. Panjwani, S. Tan, K. Jarrin, and M. Cukier, "An Experiment. Evaluation to Determine if Port Scans are Precursors to an Attack," in 2005 International Conference on Dependable Systems and Networks (DSN'05).

[18] "Network Mapper," 2016. [Online]. Available: https://nmap.org/

[19] A. Orebaugh and B. Pinkard, Nmap in the enterprise: your guide to network scanning. Syngress, 2011.

[20] F. Lau, S. Rubin, M. Smith, and L. Trajkovic, "Distributed denial of service attacks," International Conf. on Systems, Man and Cybernetics, vol. 3, pp. 2275–2280, 2000.

[21] V. E. Solutions, "Data breach investigations report," Verizon, Report, 2015.

[22] ——, "Data breach investigations report," Verizon, Report, 2016.

[23] E. Bertino and N. Islam, "Botnets and internet of things security," Computer, vol. 50, no. 2, pp. 76–79, Feb 2017.

[24] D. Moore, C. Shannon, D. J. Brown, G. M. Voelker, and S. Savage, "Inferring internet denial-of-service activity," ACM Transactions on Computer Systems (TOCS), vol. 24, no. 2, pp. 115–139, 2006.

[25] R. Altawy and A. M. Youssef, "Security, privacy, and safety aspects of civilian drones: A survey," ACM Transactions on Cyber-Physical Systems, vol. 1, no. 2, p. 7, 2017.

[26] E. Vattapparamban, I˙. Gu¨ven¸c, A. I˙. Yurekli, K. Akkaya, and S. Uluag˘a¸c, "Drones for smart cities: Issues in cyber-security, privacy, and public safety," in Wireless Communications and Mobile Computing Conference (IWCMC), 2016 International. IEEE, 2016, pp. 216–221.

[27] J. Valente and A. A. Cardenas, "Understanding security threats in consumer drones through the lens of the discovery quadcopter family," in Proceedings of the 2017 Workshop on Internet of Things Security and Privacy. ACM, 2017, pp. 31–36.

[28] F. Samland, J. Fruth, M. Hildebrandt, T. Hoppe, and J. Dittmann, "AR.Drone: security threat analysis and exemplary attack to track persons," The International Society for Optical Engineering, vol. 8301, 2012.

[29] S. Kamkar, "Skyjack: autonomous drone hacking," On- line. http://samy.pl/skyjack, 2013.

[30] K. Anderson and K. J. Gaston, "Lightweight unmanned aerial vehicles will revolutionize spatial ecology," Frontiers in Ecology and the Environment, vol. 11, no. 3, pp.138–146, 2013.

[31] L. V. Santana, A. S. Brando, M. Sarcinelli-Filho, and R.Carelli,"A trajectory tracking and 3d positioning controller for the ar.drone quadrotor," in 2014 International Conference on Unmanned Aircraft Systems (ICUAS),May 2014, pp.756–767.

[32] A. Glaser, "The U.S. government showed just how easy it is to hack drones made by Parrot, DBPower and Cheer-son," 2017. [Online]. Available: https://www.recode.net/2016/11/18/13655042/fulltranscript-april-glaser-drones-regulation-too-embarrassed-to-ask

[33] F. Trujano, B. Chan, G. Beams, and R. Rivera, "Security analysis of dji phantom 3 standard," Massachusetts Institute of Technology, May, 2016.

[34] E. Deligne, "ARDrone corruption," Journal in Computer Virology, vol. 8, no. 1-2, pp. 15–27, 2012.

[35] J.-S. Pleban, R. Band, and R. Creutzburg, "Hacking and securing the AR.Drone 2.0 quadcopter: Investigations for improving the security of a toy," The International Society for Optical Engineering, vol. 9030, 2014.

[36] M. Hooper, Y. Tian, R. Zhou, B. Cao, A. P. Lauf, L. Watkins, W. H. Robinson, and W. Alexis, "Secur- ing commercial wifi-based uavs from common security attacks," in Military Communications Conference, MIL-COM 2016-2016 IEEE. IEEE, 2016, pp.1213–1218.

[37] S. Mansfield-Devine, "Anonymous: Serious threat or mere annoyance?" Network Security, no. 1, pp. 4–10,2011.

[38] M. J. M. M. Mur Artal, Rau´l and J. D. Tard´os, "ORB-SLAM: a versatile and accurate monocular SLAM system," IEEE Transactions on Robotics, vol. 31, no. 5, pp. 1147–1163, 2015.

[39] T. Olufon, C. Campbell, S. Hole, K. Radhakrishnan, and A. Sedigh. "Mitigating External Threats in Wireless Local Area Networks". International Journal of Communication Networks and Information Security (IJCNIS) , 2014. 6, no. 3

[40] A. Rehman, S. Rehman, I. Khan, M. Moiz and S. Hasan,"Security and privacy issues in IoT," International Journal of Communication Networks and Information Security (IJCNIS), Vol. 8, No. 3, 2016.

Table 1 Previous work on evaluating drone security

Reference	Drone model	Methodology	Results
Deligne [34]	AR.Drone	Experimental attacks	The following attacks were performed: DoS (Hping), hijacking the drone and hijacking the video capture.
Samland et al. [28]	AR.Drone	Experimental attacks	The following attacks were performed: hijacking the drone, hijacking the video capture and tracking of persons using GPS.
Pleban et al. [35]	AR.Drone 2.0	Theoretical investigation of feasible attacks	Securing the Wi-Fi connection by implementing WPA functionalities.
Trujano et al. [33]	DJI Phantom 3	Experimental attacks	The following attacks were performed: disconnecting clients, password brute-forcing and network mapping.
Vasconcelos et al. [10]	AR.Drone 2.0	Experimental attacks	The following attacks were performed: DoS (Hping, LOIC and Netwox) and take-down flying drone. An analysis of the impact of DoS attacks on the camera frame rate were also presented.
Hooper et al. [36]	Parrot Bebop	Experimental attacks	The following attacks were performed: buffer-overflow, DoS (Small Replayed JSON Record) and ARP cache poison.
Valente and Cardenas [27]	Discovery U818A	Experimental attacks	The following attacks were performed: hijacking the drone, hijacking the video capture, take-down flying drone and overwrite root password.