Capture of UAVs Through GPS Spoofing Using Low-Cost SDR Platforms

João Gaspar · Renato Ferreira · Pedro Sebastião · Nuno Souto

Published online: 12 March 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract
The increased use of unmanned aerial vehicles (UAVs), better known as drones, by civilians has grown exponentially and their autonomous flight control systems have improved significantly, which has resulted in a greater number of accidents and dangerous situations. To help resolve this problem, in this paper, we address the use of low-cost Software Defined Radio (SDR) platforms for simulating a global navigation satellite system (GNSS), more specifically the global positioning system (GPS), in order to transmit false signals and induce a location error on the targeted GPS receiver. Using this approach, a defensive system can be implemented which can divert, or even take control of unauthorized UAVs whose flight path depends on the information obtained by the GPS system.

Keywords Global navigation satellite system · GPS spoofing · Software defined radio · Unmanned aerial vehicles

1 Introduction
Due to the difficult problem of dealing with unauthorized operations of unmanned aerial vehicles (UAVs) and the growing occurrence of incidents, especially involving flights in areas close to airports, military areas, restricted areas, or dangerous areas, some solutions have started to emerge. These include jammers [1], firearms and hawks trained for the purpose of “hunting” the UAVs [2]. Some products already appeared on the market like the KNOX [3] developed by MyDefence, but only have a jamming solution approach. All these methods have limitations since they can lead damage to the device itself, injure the animal
responsible for the “hunting”, as well as putting at risk the personal safety of the citizens present on the site where the intercepted device may land uncontrollably. Another possible solution for UAV control in zones or situations described previously is the spoofing of the UAV command, regardless of whether it is controlled by satellite navigation, WiFi or other radio waves. However, communication protocols can vary between different brands and models of UAVs which makes it more complicated to implement spoofing for all of them. It is, therefore, simpler to perform the spoofing of satellite navigation signals since the GPS signal is broadly used by most UAVs, especially for autonomous operations. There have been some studies and investigations in this area of spoofing GPS against UAVs. In [4], the authors studied and explored the vulnerabilities of GPS systems in drones in order to divert or gain control over the aircraft. Also, another example of exploiting the GPS vulnerabilities is the UnicornTeam, a team of security researchers whose main focus is the security of systems using radio technologies. This is a team that has been part of the DEF group with 23 vendors and proven with various approaches that it is possible to spoof a GPS receiver in [5].

Due to its flexibility for multiple applications, there has been a growing interest in the use of Software Defined Radio (SDR) for implementing and testing radio systems. An SDR platform, such as the bladeRF used in the tests presented in this paper, is a radio communication system where components that were traditionally implemented in hardware (e.g. mixers, filters, modulators/demodulators) can be developed in software using the right frameworks for each different brand and model of SDR.

In this paper, we describe the development of a mobile spoofing system that integrates low-cost SDR platforms and a software GPS signal simulator combined with a set of sensors to determine the unauthorized UAV location. The implemented system is capable of transmitting false GPS signals to redirect or even gain control of the vehicle flying over protected areas. For evaluating the behavior of the system’s operation, several types of GPS receivers were tested as targets for the spoofed signal in different scenarios.

The remainder of the paper is organized as follows: Sect. 2 presents some related works. Section 3 introduces the global satellite navigation systems with a focus on the GPS system. Section 4 describes what is spoofing and presents some techniques used in the spoofing of GPS signals. Regarding the developed spoofing system, the description of its operation and its architecture are presented in Sect. 5. In Sect. 6 describes the experimental tests using the system developed for spoofing different GPS receivers. In Sect. 7 the conclusions are drawn, and finally, acknowledgments are presented.

2 Related Works

Daniel P. Shepard, Jahshan A. Bhatti, and Todd E. Humphreys of the University of Austin Texas Aerospace Engineering Department [4] have studied and exploited the vulnerabilities of GPS drone systems to deflect or gain control over the aircraft.

The attacker or spoofer generates false GPS signals for all authentic signals it can receive. False signals received by the drone receiver arrive in line with true GPS signals, considering the delay times and phases, as illustrated in Fig. 1 [4].

The same authors of the previously described study demonstrated [6] that it is possible to perform civil GPS spoofing to unmanned aerial vehicles. The system used in the tests was an improved version of the original system referenced in [7], developed by the University of Austin Texas Radio Navigation Laboratory Fig. 2.
Initially, the system receives the authentic GPS signals and uses them to adapt their false signals to the information coming from the true signals.

As it can be seen from the block diagram in Fig. 3, the control module reads the components of GPS signals (code phase, carrier phase, and Doppler effect) through a receiver. These components are modified using linear measurement models and used to create false GPS signals.

To demonstrate the functioning of the spoofing system the victim was an unmanned aerial vehicle Hornet Mini UAV (Fig. 4), owned by the University. A vehicle with civil satellite navigation capability. The scheme used to perform the spoofing tests were as
shown in Fig. 5 with a distance between spoofer and receiver (Hornet Mini) of approximately 650 m.

The Hornet Mini was manually controlled to the 650 m position and raised to an altitude of approximately 12 m. After being in place, the command for satellite navigation was changed and to stay in the same position. Then they started sending false GPS signals making sure that the code phases were aligned with the original signals and quickly gained control over the aircraft [6]. The only downside is that the spoofer needs eye contact with the aircraft to avoid uncontrolled landing or deflection.

3 Global Navigation Satellite System

GNSS systems have a high level of complexity because they comprise various subsystems working together. While the satellites are the more “visible” part of the system, terrestrial infrastructures are crucial for correct operation supporting necessary maintenance tasks of the satellites orbits. Users only have access to a radio link in the system, the downlink transmission from the satellites of the constellation. Since the downlink signal is transmitted in broadcast, there is no limit on the number of users of the system [8].
3.1 Global Positioning System

The GPS system is the only one explored in this paper since it was the first system to come into operation and, currently it is the most commonly used system.

Its architecture is divided into several segments: user segment, which consists of all types of GNSS GPS signal receivers, a space segment, which brings together all satellites constituting the constellation, and a ground segment, which is responsible for monitoring, controlling and updating stations [9, 10].

3.1.1 Ground Segment

Main functions are:

- Monitor the satellites;
- Define the orbits for each satellite to predict the ephemeris and almanac data;
- Determine the altitude and location of each satellite and send the corrections to the satellites so they remain in the correct orbit [9, 10].

3.1.2 Space Segment

Constellation base containing 24 satellites, consisting of six almost circular orbits with a slope of 55° referenced with the equatorial plane at an altitude of 20183 km. Each satellite can make a circle around the planet in exactly 11 h 57 min and 58 s. This makes it possible to have four satellites in line of view in any position on the planet thus,
enabling localization, under normal atmospheric conditions. The constellation was officially declared operational in 1995.

Main functions are:

- Receive from the ground segment the corrections of the orbits apply them;
- Transmit the GNSS signals [9, 10].

### 3.1.3 User segment

It consists of a wide variety of receivers, including military, mass-produced receivers for civil use and even for scientific purposes. Its main functions are:

- Receive signals corresponding to GNSS systems and evaluate their status;
- Perform measurements of propagation time;
- Perform measurements due to the Doppler effect;
- Calculate the location of the receiver;
- Calculate the speed of the terminal and provide time measurements [9, 10].

### 3.2 GPS Frequencies, Codes and Modulations

The GPS system has 3 frequency ranges, L1, L2, and L5 being L2 and mainly L5 frequencies still with some development [11]. In this paper, only the L1 frequency range was addressed.

L1 is the most commonly used worldwide GPS frequency range. It operates at a 1.57542 GHz frequency and the access to it is by CDMA (more details are provided in Table 1. It contains three different signals: Coarse/Acquisition (C/A) code, P code, and M code.

* C/A code became the most adopted and important code, intended for civil use, and many solutions developed in the market to use the GPS system rely on this signal. It has a millisecond length at a chipping rate of 1023 Mbps [12].

* P code precision code intended only for military applications. The P (Y) code is often employed in place of the P code when using anti-spoofing systems. The code features a 7 day long length, with a chipping rate of 10.23 Mbps and guarantees confidentiality and authentication [12].

* M Code designed exclusively for military use and may eventually replace the P and P (Y) code. It has better features to resist jamming and guarantees better performance and more flexibility than the P (Y) code [12].

| Table 1 | GPS L1 technical features [11, 12] |
|---------|-----------------------------------|
| Service | C/A | P(Y) code | M code |
| Modulation | BPSK(1) | BPSK(10) | BOC\textsubscript{sin}(10,5) |
| Code length | 1023 | P: 6.19 $\times$ 10\textsuperscript{12} | Cryptographically generated |
| | Y: Cryptographically generated | | |

 Springer
It can be concluded that GPS, in the L1 frequency range, is divided into two main types of transmitted GPS signals:

- Open signals for civil use;
- More robust and more accurate signals, for military use.

For this purpose, the codes described above remain in use and the localization is divided into two services: Standard Positioning Service (SPS) and Precise Positioning Service (PPS), which correspond respectively to the two different types of signals.

SPS is a service that can be accessed by any normal (civil) user. It is based on the C/A code sequence. The PPS service is only accessible to authorized users (military) and it not only uses C/A code but also P-code. This allows greater accuracy of the location on the globe.

4 Spoofing

Spoofing, in general, is a fraudulent or malicious practice in which communication is sent from an unknown source, disguised as a source known to the receiver. The use of spoofing is more common in mechanisms and communication networks that do not have a high level of security. In the civil Global Positioning System (GPS) signal there is not any type of encryption or authentication to protect or to prove that the signal comes from a reliable source or the non-occurrence of repudiation of the signal. So, to accomplish spoofing and deceive a GPS receiver, it is necessary to simulate GPS signals as if they came from real satellites.

All devices that use radio frequency for communication have the vulnerability that the information transmitted is available to everyone within the range of the transmission. And most of the systems using GPS signal receivers are vulnerable to the spoofing techniques described below:

*Simple spoofing* technique capable of generating false global navigation satellite system (GNSS) signals. It can be put into practice using:

- Low-cost hardware to receive and reproduce GNSS signals. Custom signal simulators can be inserted into the configuration to control and modify some of the transmission parameters [13];
- Commercial hardware that is usually expensive and more complex to manipulate, but often with greater capacities for the processing and transmission of electromagnetic signals [13].

*Intermediate spoofing* in this case, the attacker synchronously generates false signals while simultaneously attempting to attack each channel of the target receiver by performing code phase alignment between false and genuine incoming signals [14].

*Spoofing with multiple transmitting antennas* advanced technique, used mainly against multiple antenna receivers, in which each transmitting antenna of the attacker combines with a corresponding receiving antenna in the victim [15].

*Spoofing with high gain antennas* enhanced attack based on the use of antennas with enough gain to separate GNSS signals from noise, including, for example, unknown or encrypted code chips [16].
Sophisticated spoofing can be performed by a set of coordinated and synchronized attackers, capable of attacking the victim’s receiver in an organized manner. In addition, they have three-dimensional position information about the phase centers of their antennas and the phase center of the victim’s antenna, thus overcoming complex countermeasures, such as those based on the estimation of the angle of arrival [14].

5 Mobile Spoofing System

The spoofing system developed and described in this paper operates using an open hardware electronics prototyping platform, sensors, an SDR module and a System on a Chip (SoC) as the central processor of the system. This system adopts a simple spoofing technique, which can generate and transmit false GPS signals. However, it takes into account the current location of the UAV and employs a directive antenna for focusing the transmission on the intended target, making the system more sophisticated and more difficult to detect by the UAV control station.

5.1 UAV Location Determination

To determine the location of the UAV, the implemented system integrates three sensors and a receiver, namely:

- Lidar sensor Lidar Lite v3 [17] (measurement of UAV distance);
- 3D accelerometer MPU6050 [18] (tilt/pitch measurements);
- e-compass LSM303D [19] (measuring system orientation);
- GPS signal receiver [20] (detection the system positioning).

By associating the measurements of the sensors with the angles defined in Figs. 6 and 7 the location is determined as follows:

UAV distance in line of sight: Sensor Lidar measurement = h;
The tilt angle with the horizontal plane: 3D Accelerometer measurement = β;
System orientation angle: Magnetometer measurement = θ;
System location: GPS receiver = X.

After obtaining the values of the sensors and receiver, ‘a’ and ‘b’ are calculated using simple trigonometric equations as in (1) and (2)

---

Fig. 6 Side plane view analytical calculations

---

\[ \begin{align*}
\beta & \text{- Angle of inclination} \\
h & \text{- Distance in line of view} \\
b & \text{- Drone height} \\
a & \text{- Distance on the plane} \\
X & \text{- Position of the system} \\
Y & \text{- Drone position}
\end{align*} \]
With the result of the UAV distance in the horizontal plane (variable “a”) associated with the value exported from the magnetometer (variable ‘θ’), the mean value of the radius of planet Earth [21] and knowing the location of the system itself (variable ‘X’), the location of the UAV’s (variable ‘Y’) latitude and longitude can be determined using [22].

\[
a = \cos(\beta) \times h
\]

\[
b = \cos(\beta) \times h.
\]

With the result of the UAV distance in the horizontal plane (variable “a”) associated with the value exported from the magnetometer (variable ‘θ’), the mean value of the radius of planet Earth [21] and knowing the location of the system itself (variable ‘X’), the location of the UAV’s (variable ‘Y’) latitude and longitude can be determined using [22].

\[
LatY = \sin^{-1}\left(\sin(LatX) \times \cos\left(\frac{a}{1000 \ R_{\text{earth}}}\right) + \cos(LatX) \times \sin\left(\frac{a}{1000 \ R_{\text{earth}}}\right) \times \cos(\theta)\right)
\]

\[
LongY = LongX + \tan^{-1}\left(\frac{\cos\left(\frac{a}{1000 \ R_{\text{earth}}}\right) - \sin(LatX) \times \sin(LatY)}{\sin(\theta) \times \sin\left(\frac{a}{1000 \ R_{\text{earth}}}\right) \times \cos(LatX)}\right)
\]

where the following definitions are adopted

LatX—Latitude value of system location;
LongX—Longitude value of system location;
LatY—Latitude value of UAV location: LatY;
LongY—Longitude value of UAV location: LongY;
Rearth—Planet Earth radius (≈6378.137 km).

Note that the value of ‘a’ and ‘Rearth’ need to be on the same scale (kilometers) and all angles must be in radians, not degrees.

5.2 Deviation of UAV

After acquiring the location of the UAV through the process previously described, the next step of the GPS spoofing system is reached, namely the generation of the fake signal. For the transmission of false GPS signals, we used the bladeRF x40 SDR platform. It presents
a basic radio architecture, but it is capable of encompassing modulation techniques and
basic telecommunications coding schemes. It has USB 3.0 communication capability and
a fully programmable FPGA chip for faster system development [23]. The choice of the
bladeRF was made taking into account its low energy consumption, versatility, can tune
from 300 MHz to 3.8 GHz, can be configured to operate as a custom RF modem, a GSM
and LTE picocell, a GPS receiver, an ATSC transmitter, or a combination Bluetooth/WiFi
client, without the need for any expansion cards, and all of the bladeRF host software,
firmware, and HDL is an open source, and available on GitHub [24]. To implement the
spoofing step, we adopted the free available online software, bladeGPS1 simulator, which
was developed by OSQZSS in Japan and is capable of constructing and simulating real
GPS signals. Looking at the different functionalities available in the bladeGPS simulator,
there is one that can be easily exploited for implementing a spoofing system. In fact, one
of the functions provided by the bladeGPS simulator is the ability to use NMEA messages,
marked in red in Fig. 8, for the dynamic simulation of GPS signals. This enables the gen-
eration of not only static localizations but also trajectories and allows a simple way to con-
struct GPS messages equal to the real ones, thus, spoofing the UAV current location.

The idea of messages in NMEA format is to send a data line called a sentence that is
totally independent of the previous and posterior lines. The information in the sentence is
formatted according to the category of device that will receive them, indicated by a two-
letter prefix. In the case of GPS receivers, the prefix is Global Positioning (GP) [25, 26]
and for GPS fix location it is completed with another three-letters prefix, GGA, meaning
Global Positioning System Fix Data. NMEA messages were developed by the National
Marine Electronics Association, which develops specifications that define the interface
between various marine systems and electronic equipment. The communication for GPS
signal receivers is defined in these specifications [17]. Most computer programs that pro-
vide real-time positioning information understand and expect to receive information in
NMEA format.

Each sentence starts with the character ‘$’, ending with ‘*’ and the value of the check-
sum (represented by two hexadecimal numbers). The checksum is calculated with an XOR
operation of all characters between ‘$’ and ‘*’. All information is contained in a single line
with the various data separated by commas and represented in ASCII text. It can never
exceed 80 characters per sentence. The first data consists of a code name that defines the
type of data found in the sentence. Each data type has its own interpretation and is defined
in the NMEA standard. The GGA sentence provides GPS correction data [27]. An example
of a GPGGA sentence with the definition of each data field is shown in Fig. 9.
In this way, as we already know how the NMEA messages are created and the current location of the UAV, the process of simulating the GPS signals for the deviation of the UAV position can be synthesized in the elaboration of an NMEA message with a sentence sequence that indicates a false current location to the vehicle. This can cause...
to correct its present position and thus change its actual position to a position outside the restricted area, as in Fig. 10.

The sentence formation was also constructed dynamically, i.e., simulating a moving location, causing the UAV to correct its course in continuous displacement, but with a route that will direct it to the landing area. To do this, it is necessary to determine two consecutive locations of the UAV through the measurements of the sensors and receiver of the system, as previously explained in Figs. 6 and 7. With these two locations (red markings in Fig. 11) and the time interval between them, the system can determine not only the direction of the UAV path as well as its speed.

To determine the UAV simulated route that will lead him to the landing area Fig. 12 it was used the Maps JavaScript API as the updated map source which allows to use existing functions and develop a graphical interface that presents the results of the system and its current state. Using the API, it can be calculated and traced the “Simulated Route” necessary to drift the UAV to the landing area, with the following steps:

- **Distance Computation:** calculate the distance between the second location of the UAV, obtained from the sensors in the system, and the landing area. It is exemplified as the “Final route” in Fig. 11;
- **Heading Determination:** determine the angles $\alpha_1$ and $\alpha_2$ shown in Fig. 11, namely, the angle of the UAV’s original course line (“Current route” in Fig. 11) and the angle of the line between the UAV and the landing area (“Final route” in Fig. 11). These angles are measured using the north bearing as a reference;
- **Offset Determination:** Given angle $\alpha_3$ (in Fig. 11), the location of origin (last location of the UAV through the sensors of the system), and the distance to the Landing Area it is possible to compute the “Simulated route” as in Fig. 11. Note that angle $\alpha_3$ is easily obtained as in (5).
\[ \alpha_3 = (\alpha_1 - \alpha_2) + \alpha_1 \] (5)

### 5.3 Architecture

A general scheme of the whole mobile spoofing system is represented in Fig. 13, for easier perception of its operation. It is divided into different blocks which we describe next. The corresponding implementation adopted for the experimental tests are shown in Fig. 14.
• (a) A Raspberry Pi used as the central controller of the system. It relied on a Linux Ubuntu distribution install with all scripts and software saved in memory. It distributes the commands and tasks to other subsystems and scripts;
• (b) In this block there are a set of scripts responsible to download the maps and construct the NMEA messages (developed in JavaScript and implemented with NodeJS). It uses the framework Electron of NodeJS to implement the system as a desktop application that runs on boot;
• (c) Comprises a script (developed and implemented in Python) responsible to redirect the sensors values to the NodeJS application and read the switches state to trigger the system to initiate or stop the transmission of the spoofing signals;
• (d) Corresponds to the sensing unit. It was implemented using an Arduino Uno board with all the sensors and receiver mentioned and described previously in order to estimate the location of the UAV (developed and implemented in C++). The communication with the sensors is established through i2c bus, with all the sensor data transferred to block a) using Universal Serial Bus (USB) interface;
• (e) Output block that is in charge of the transmission of the false GPS signals. It is supported by a bladeRF SDR board connected to a YAGI antenna with high transmission gain and directivity.

6 Experimental Results

Several tests were performed on the sensors used for estimating the UAV location in order to gauge their limitations and possible influences on the final results. Spoofing tests were also made using the overall system in an indoor and outdoor environment.

Three different types of GPS receivers were used as targets:
• smartphone;
• u-blox M8 GNSS Evaluation Kit;

Fig. 14 Spoofing system in a laboratory environment
6.1 Sensor Tests

Measurement accuracy tests of the lidar, magnetometer, and accelerometer were accomplished in a laboratory scenario. These tests were designed to individually evaluate each of the sensors in question, determining their possible read errors and help the integration into the overall system.

6.1.1 LIDAR Lite v3

The laboratory tests performed on the sensor were precision measurement tests. For the test, a measuring tape of 10 m was used and measurements were taken spaced by 1 m. As the distance increases, an increase in the measurement error also occurs, as seen in the test results in Table 2 and Fig. 15, which shows a linear upward trend line. Because the error values are not very high, they cannot significantly influence the final measurements for determining the location of the drone due to the original tolerance error in the real GPS system.

6.1.2 MPU6050

The precision test was developed using a smartphone and its gyroscope. Encapsulating the MPU6050 sensor to the smartphone, measurements were taken with a five-degree interval, in a range of 0º to 90º.

Table 2  Error values measured by the Lidar sensor

| Distance (m) | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  |
|------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Error (m)  | 0.026 | 0.031 | 0.017 | 0.02 | 0.08 | 0.12 | 0.1 | 0.14 | 0.11 | 0.13 |

Fig. 15  Precision measuring chart of the Lidar sensor

- u-blox MAX-7Q receiver.
Given the values indicated by the smartphone and the values measured by the MPU6050 sensor, we take 15 samples and calculated an average for every measurement angle. The difference between both values of the platforms was calculated and are presented in Table 3. The behavior of the error with the variation of the angle is also shown in the graph of Fig. 16.

It has been found that with increasing angle, there is also a slight increase in measurement error. This can be seen from the values in Table 3 and the chart of Fig. 16, which shows a linear upward trend line of very low slope. Therefore, these errors present a negligible influence on the final measurements to determine the location of the drone.

### 6.1.3 LSM303D

A precision test was developed using a smartphone and its magnetometer. Accommodating the LSM303D sensor to the smartphone, measurements were taken with a range of 10°, in a range of 0° to 360°.

Using the values provided by the smartphone and the values measured by the sensor LSM303D we take 15 samples and calculated an average for every measurement angle. The difference between the values of both platforms were calculated. The resulting error and its variations with the angle are presented in Table 4 and Fig. 17.

By varying the angle, there is no increase in the measurement error, as can be seen in the values in Table 4 and in the graph of Fig. 17. A horizontal linear trend line is shown, demonstrating that the measured errors vary, but with a certain coherence in the differences.

---

**Table 3** Error values measured by the MPU6050 sensor

| Angle (°) | 0  | 5  | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 | 55 |
|-----------|----|----|----|----|----|----|----|----|----|----|----|----|
| Error (°) | 0.02 | 0.37 | 0.49 | 0.1 | 0.12 | 0.28 | 0.41 | 0.25 | 0.52 | 0.32 | 0.48 | 0.42 |

| Angle (°) | 60 | 65 | 70 | 75 | 80 | 85 | 90 |
|-----------|----|----|----|----|----|----|----|
| Error (°) | 0.74 | 0.54 | 0.22 | 0.18 | 0.24 | 0.36 | 0.08 |

**Fig. 16** Precision measuring chart of the MPU6050 sensor
of values between platforms. The recorded error values can cause a larger influence than the previous sensors in the determination of the location of the drone.

### 6.2 Indoor Tests

The first phase of tests were carried out inside a building, that is, without the influence of actual GPS signals. It was not taken in account the distance between the spoofer and the targets, the spoofer transmitted the signals side by side from the targets tested, since the scope of this investigation was to demonstrate and prove that it is possible to implement a system capable of determining the location of a UAV and transmit GPS spoofing signals to divert it from the current route. This allows testing the response of the various target receivers to the spoofing system when they have no previously acquired location.

#### 6.2.1 Smartphone

The tests have been made with a smartphone to verify that many of the GPS receivers installed on these devices are vulnerable to spoofing attacks. The smartphone selected as a target for these tests was used as the LG L90 model. The GPSTest application [28], Fig. 18, was installed in the device, which allows real-time visualization of which satellites are visible, GPS signal power levels and location in the world map.
Using the system developed and described in Sect. 5 it was possible to simulate, with coordinates defined by the user, the location of the smartphone in relation to its real position. As shown in Fig. 19 on the left, several signals from satellites with good SNR are detected. On the center some of the simulated satellites are shown in the line of sight and, to the right, the false location is shown through the latitude, longitude and with the red marker on the world map. Note that the application estimated the location on Caracas Venezuela, even though the true location was Lisbon Portugal. With this test, it was verified that it is possible to deceive the location of a GPS receiver installed in a smartphone in an indoor environment.

6.2.2 u-blox M8 GNSS Evaluation Kit

for the second target was the u-blox receiver M8 GNSS Evaluation Kit, Fig. 20. The M8 u-blox evaluation kit allows a simple evaluation of positioning technologies. It features an integrated USB interface that provides power, eliminating the need for external
power supply while supporting high-speed data transfer. The receiver was used with a computer via the USB interface in conjunction with the u-center software, which is a powerful tool for evaluating, performing and configuring u-blox GNSS receivers.

It was possible to simulate, with user-defined coordinates, the location of the u-blox receiver M8 GNSS Evaluation Kit in relation to its actual position. As shown in the lower part of Fig. 21, several signals from GPS satellites with good signal–noise ratio (SNR) are detected in the line of sight in the constellation. At the top of the monitor, the simulated location is shown on the world map through a green marker. In this case, the location obtained by the u-center was Washington DC United States of America, but the true one was Lisbon Portugal. With this experience, it has been found that it is possible to mislead the location of a GPS receiver u-blox prepared to evaluate location systems in an indoor environment.
6.2.3 u-blox MAX-7Q

Considering that the u-blox MAX-7Q receiver, Fig. 22, is widely used in several types of terrestrial, aerial and aquatic drones for supporting autonomous missions, it was also tested as a target for the GPS spoofing. The receiver was used in conjunction with an Arduino, in order to communicate its location through the Arduino serial interface and enabling the visualization of the information.

In this experiment, this type of receiver did not present any type of resistance when receiving fake GPS signals. As can be seen in Fig. 23, the receiver identifies its location with the latitude and longitude values of Pyongyang Korea of the North, which do not correspond to the true ones 38.74673, -9.15274 (Lisbon Portugal). With this test, it was verified that in an indoor environment it is possible to easily deceive the location of a u-blox GPS receiver used in many drones.

6.3 Outdoor Tests

The following outdoor tests were carried out under the influence of real GPS signals. It was not taken into account the distance between the spoofer and the targets, the spoofer
transmitted the signals side by side from the targets tested. This allows the evaluation of the spoofing system performance in a scenario where the receiver already has a pre-acquired location through real GPS signals.

6.3.1 Smartphone

Outdoor smartphone tests were carried out to verify that many of the GPS receivers installed in these devices are vulnerable to spoofing attacks, even after they already have obtained a location through real GPS. Once again, the smartphone model used as the target was the LG L90.

For these tests, after the smartphone acquired true location, the spoofing system started to transmit the fake GPS signals. It was observed that after about 3 min the system lost the location, taking into account that it detected another GPS signal (false GPS signal transmitted by the bladeRF platform), and then accepted the false GPS signal possibly because it had better SNR. Under these conditions, and with the results described above, it was possible to simulate, with user-defined coordinates, the location of the smartphone in relation to its actual position. As shown in Fig. 24, on the left side several satellite signals with good SNR are detected. In the center are shown some of the simulated satellites and real satellites of the constellation. On the right, it is shown as a blue dot the wrongly induced location of the smartphone in the GoogleMaps application. The true position of the smartphone corresponds to the red dot. With this test, it was verified that it is possible to deceive the location of a GPS receiver installed in a smartphone in an outdoor environment, even under the influence of real GPS signals.

6.3.2 u-blox M8 GNSS Evaluation Kit

Following a similar test approach, after the u-blox receiver, M8 GNSS Evaluation Kit got its location, the transmission of false GPS signals was started. After about 2 min the
receiver had already detected the new GPS signals, but only after about 30 min, it accepts the GPS signals created by the bladeRF platform.

Treating itself as a receiver used to evaluate and analyze the performance of GNSS systems, it has features that make it less susceptible to spoofing and jamming attacks. Hence its behavior makes it more difficult to accept the false GPS signals transmitted.

As shown in Fig. 25, at the bottom of the monitor, are shown some of the simulated and real satellites in line of sight. On the right side of the monitor is presented the world map with the simulated location represented as a green marker, while the actual location of the receiver was the red dot.

With this experiment, it was observed that it is possible to mislead the location of a GPS receiver u-blox M8 in an outdoor environment already with the defined location, but for the purpose of the developed system, the time it takes to accept the signals would be a critical point for the spoofing system. Possibly using jamming techniques before beginning the spoofing transmission would be a good option to speed up the process of deceiving the receiver with false GPS signals.

### 6.3.3 u-blox MAX-7Q

Finally, the u-blox receiver MAX-7Q was also tested in an outdoor environment where it already had acquired location through real GPS signals. It was observed that after 2 to 3 min it did not show any resistance when receiving the false GPS signals and accepted the false induced location. As can be seen in Fig. 26, the receiver changes its location (marked in red latitude and longitude) according to the values entered in the bladeGPS software, Fig. 27, that is, its true location is not the one displayed by the second values of latitude and longitude in the Arduino serial, but by the first one, Fig. 26.

With this test, it was found that even in an outdoor environment it is possible to induce a wrong location on a u-blox GPS receiver which is widely used in drones.

![Fig. 25 Receiver u-blox M8 GNSS evaluation kit spoofing outdoor](image-url)
7 Conclusions

Using spoofing techniques, it is possible to recreate signals identical to the actual signals of existing systems, which makes it possible to elaborate advanced techniques of attacks that can be even capable of blocking the communications of a system.

In this paper, we described a possible elaboration of a portable system capable of diverting unauthorized UAVs using GPS spoofing techniques. The implemented system is based on flexible low-cost SDR equipment which is capable to transmit, receive, record and reproduce any radio communication systems. The development of the GPS spoofing system has proven that with a set of sensors, some analytic calculations and low-cost SDR equipment the GPS receivers do not have mechanisms protecting against spoofing and that it is possible to use a vulnerability of the GPS system to create something with practical applicability.

Comparing our approach to using this type of spoofing technology with the previously presented in Sect. 2. Ours not only features the ability to spoof various systems that use
GPS technology but also the ability to determine the current location of the UAV, making the spoofing technique less easily detectable.

Acknowledgements This work was funded by FCT/MEC through national funds and co-funded by FEDER—PT2020 partnership agreement under the project UID/EEA/50008/2019.

Compliance with Ethical Standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Kratky, M., & Minarik, V. (2017). The non-destructive methods of fight against UAVs. In ICMT 2017—6th international conference on military technologies (pp. 690–694).
2. The Guardian. (2016). Eagle-eyed: Dutch police to train birds to take down unauthorised drones | World news | The Guardian. https://www.theguardian.com/world/2016/feb/01/dutch-netherlands-police-birds-unauthorised-drones. Retrieved November 08, 2018.
3. My Defense. (2018). KNOX Anti-Drone Solution—MyDefence Communication.
4. Kerns, A. J., Shepard, D. P., Bhatti, J. A., & Humphreys, T. E. (2014). Unmanned aircraft capture and control via GPS spoofing. Journal of Field Robotics, 31(4), 617–636.
5. Lin, H., & Qing, Y. (2015). GPS SPOOFING Low-cost GPS simulator.
6. Shepard, D. P., Bhatti, J. A., Humphreys, T. E., & Fansler, A. A. (2012). Evaluation of smart grid and civilian UAV vulnerability to GPS spoofing attacks. Ion Gnss (pp. 3591–3605).
7. Humphreys, T. E., Ledvina, B. M., Tech, V., Psiaki, M. L., Hanlon, B. W. O., & Kintner, P. M. (2008). Assessing the spoofing threat: Development of a portable GPS civilian spoofer. In Proceedings of the 21st international technical meeting, Satellite Division, Institute of Navigation.
8. Oxley, A. (2017). Chapter 2 - Introduction to GPS (pp. 19–38). Kingdom of Bahrain: Kingdom of Bahrain.
10. Ferreira, R. (2018). Effective GPS Jamming Techniques for UAVs using low-cost SDR platforms. In 2018 Global Wireless Summit.
11. Samama, N. (Ed.) (2007). GNSS system descriptions. In Global positioning (pp. 131–161), Wiley.
12. Kaplan, E. D. (2006). Understanding GPS: Principles and applications (2nd ed.). Norwood: Artech House.
13. Dovis, F. (2015). GNSS interference threats and countermeasures. Norwood: Artech House.
14. Ledvina, B. M., Bencze, W. J., Galusha, B., Miller, I. (2010). An in-line anti-spoofing device for legacy civil GPS receivers. In Proceedings of the 2010 international technical meeting of the institute of navigation (pp. 689–712). San Diego, CA.
15. Psiaki, M. L., & Humphreys, T. E. (2016). GNSS spoofing and detection. Proceedings of the IEEE, 104(6), 1258–1270.
16. Kuhn, M. G. (2005). An asymmetric security mechanism for navigation signals, In International workshop on information hiding IH (Vol. 3200, pp. 239–252).
17. Lite, L., & Specifications, T. (2016). able of contents lidar lite v3 operation manual and technical specifications wiring harness, pp. 1–14.
18. Ave, B., Number, D., & Date, R. (2012) MPU-6000 and MPU-6050 product specification, Vol. 1, No. 408.
19. Ecopack, E. F., Lsm, T., Lsm, T., & Lsm, T. (2012) 3D accelerometer and 3D magnetometer module, no. June, pp. 1–54.
20. u-blox, “MAX-7 series | u-blox.”. https://www.u-blox.com/en/product/max-7-series. Retrieved July 30, 2018.
21. David, Dr., & Williams, R. (2017). Earth fact sheet, NASA. https://nssdc.gsfc.nasa.gov/planetary/factsheet/earthfact.html. Retrieved August 01, 2018.
22. Veness, C. (2017). Calculate distance and bearing between two Latitude/Longitude points using haversine formula in JavaScript. MIT License. https://www.movable-type.co.uk/scripts/latlong.html. Retrieved September 21, 2018.

Springer
Capture of UAVs Through GPS Spoofing Using Low-Cost SDR Platforms

23. bladeRF Naund, “bladeRF x40 | Nuand -.”. https://www.nuand.com/blog/product/bladerf-x40/. Retrieved July 31, 2018.
24. Nuand (2017). “bladeRF Power Consumption”. https://github.com/Nuand/bladerF/wiki/bladerF-Power-Consumption. Retrieved September 18, 2018.
25. Dale DePriest, “NMEA data”. https://www.gpsinformation.org/dale/nmea.htm. Retrieved September 17, 2018.
26. Park, B., Lee, J., Kim, Y., Yun, H., & Kee, C. (2016). DGPS enhancement to GPS NMEA output data: DGPS by correction projection to position-domain. The Journal of Navigation, 66(2), 249–264.
27. Page, G. P. S. C. (2011). GPS—NMEA sentence information. http://home.mira.net/~gnb/gps/nmea.html. Retrieved May 18, 2018.
28. GPS Test—Apps no Google Play. https://play.google.com/store/apps/details?id=com.chartcross.gpptest&hl=pt_BR. Retrieved August 30, 2018.

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

João Gaspar is currently a Ph.D. student in Information Science and Technology at ISCTE-IUL. He is developing his thesis in the field of telecommunications studying and evaluating new strategies to protect areas of unauthorized drones through communications spoofing and beamforming. Previously graduated in Electrical Engineering and Telecommunications in 2016, at the Superior School of Technologies of Castelo Branco, Portugal, and obtained his Master M.Sc. degree in Telecommunications and Computer Science Engineering in 2018, at ISCTE—University Institute of Lisbon, Portugal.

Renato Ferreira graduated in electrical engineering and telecommunications in 2016, at the Polytechnic Institute of Castelo Branco, Portugal, and obtained his Master MSc degree in telecommunications and computer engineering in 2018, at ISCTE—University Institute of Lisbon, Portugal. In Since 2018 he is a doctoral Ph.D. student at ISCTE-IUL in Information Science and Technology at ISCTE-IUL. He is developing his doctoral thesis in the field of telecommunications studying and evaluating efficient techniques for neutralization of communications and radionavigation of multiple unmanned vehicles.
Pedro Sebastião  Ph.D. in Electrical and Computer Engineering at IST, currently lecturer at ISCTE-IUL’s Information Science and Technology Department, is Board director of AUDAX-ISCTE—Entrepreneurship and Innovation Centre at ISCTE, responsible for the LABSLIS-BOA incubator and researcher at Instituto de Telecomunicações. His main researching interests are in monitoring, control and communications of drones, unmanned vehicles, planning tools, stochastic process (modelling and efficient simulations), internet of things, efficient communication systems, jamming and spoofing techniques and business models. He has oriented several master’s dissertations and doctoral theses. He is the author or co-author of more than two hundred scientific articles and he has been responsible for several national and international R&D&I projects. He has been an expert and evaluator of more than one hundred national and international Civil and Defense R&D projects. He has several scientific, engineering and pedagogical awards. Also, he has organized or co-organized more than fifty national and international scientific conferences. He planned and developed several postgraduate courses in technologies and management, entrepreneurship and innovation and transfer of technology and innovation. He has supported several projects involving technology transfer and creation of start-ups and spinoffs of value to the society and market. He developed his professional activity in the Portuguese Defense Industries, initially in the Office of Studies and later as board director of the Quality Department of the Production of New Products and Technologies. He was also responsible of wireless communication systems technologies in the Nokia-Siemens business Unity.

Nuno Souto  graduated in aerospace engineering—avionics branch, in 2000 in Instituto Superior Técnico, Lisbon, Portugal and received his Ph.D. in 2006. From November 2000 to January 2002 he worked as a researcher in the field of automatic speech recognition for Instituto de Engenharia e Sistemas de Computadores, Lisbon Portugal. He joined the ISCTE-Lisbon University Institute, as an assistant professor in 2006. He is a researcher at IT (Instituto de Telecomunicações), Portugal, since 2002 and has been involved in several international research projects and many national projects. His research interests include wireless networks, signal processing for communications, OFDM, single carrier transmission with frequency domain equalization, channel coding, modulation, channel estimation, synchronization, MIMO schemes, wireless sensor networks and unmanned aerial vehicles. He is a member of the IEEE Signal Processing Society.