GPS Spoofing for Android and iOS Mobile Systems

Michael Nelson
Computer Science, Iona College
New Rochelle, USA
email: mnelson4@gaels.iona.edu

Paolina Centonze
Computer Science, Iona College
New Rochelle, USA
e-mail: pcentonze@iona.edu

***Abstract*— Global Positioning Systems (GPS) are incredibly commonplace in daily life. Examples include fishing vessels, aircraft, cars, and even mobile phones. Currently, civilian GPS signals are transmitted in plain text on known frequencies. Due to the lack of security and obfuscation, there have been many experiments proving the ease and dangers of GPS spoofing and jamming. However, most of the analysis in the past few years has focused on vehicles rather than the devices everybody has and uses daily: smartphones and tablet devices. Spoofing these devices through satellite defined radio as well as applications will show the biggest vulnerabilities in these systems. As of now, there is no clear evidence of one Operating System (OS) being more secure than another or what types of applications are most susceptible to hacking. In this paper, we analyze the effectiveness of different spoofing methods on various commonly used applications. This analysis will determine whether iOS or Android is easier to spoof by comparing the ease and effectiveness of these spoofing methods.**

Keywords- GPS; spoof; mobile; iOS; Android.

I. INTRODUCTION

Several famous cases of GPS spoofing and jamming have come out in the past decade. Vehicles such as military drones and mega yachts [1] alike have been hacked into going to an unintended location using vulnerabilities in their navigation systems. With an increase of devices using GPS and location-based services, the severity of false locations will only become more widespread. While it is easy to see why a person would spoof another, whether it be to draw them to a remote location, land their equipment on another's property, etc., what many people overlook is why someone would spoof their own signal. Fishing vessels do it to fish in restricted waters, drone users do it to fly in unauthorized air space, and media consumers do it to access region-blocked content. Whatever the reason for spoofing, the intent is malicious. Right now, GPS messages for the public are called Civilian Navigation (CNAV) messages [2]. These civilian navigation messages have a warning that they should not be used for safety-of-life or other critical purposes until they are declared safe by the government. Even the government sees clear flaws and vulnerabilities in the current system. They are producing new types of messages regularly to improve GPS systems, however, for something that is used so much in many devices, it should be a top priority to make secure and robust. Many sectors of the economy and life rely on GPS [3]. A forced landing of a US drone by Iran is discussed in detail in "Susceptibility of GPS-Dependent Complex Systems to Spoofing" [3] that was performed despite the encryption and protection of military GPS. Others have tested spoofing in different types of scenarios, such as with car GPS devices in "A Practical GPS Location Spoofing Attack in Road Navigation Scenario" [6]. This analysis has a different set of challenges to overcome. Unlike a boat, that has a wide open sea, cars will notice if their GPS directs them to drive off the road. For this setup, a car has to actively follow another car and make small changes to the route as it travels. While unique, it is just one of many examples of practical GPS spoofing that can affect an average person. Unlike the analyses of the papers just mentioned, our work goes more in-depth into mobile OS spoofing. Multiple spoofing methods, as well as applications, will be compared to prove where the biggest security flaws are.

The rest of the paper is structured as follows. Section 2 will go over the technical aspects of how location and time are calculated from GPS signals as well as the process of spoofing a GPS receiver. Section 3 will discuss similar works that act as a proof of concept for mobile GPS spoofing. Next, Section 4 will go over how the experiment and analysis will be performed. It will also include details on the software and hardware that will be used. Section 5 will present some preliminary results from what has been done so far. Finally, Section 6 will list some conclusions that can be drawn at this stage and how future experiments can build off of this work.

II. TECHNICAL BACKGROUND

The basic concept of how GPS spoofing works is simple. Normally, a receiver is connected to several satellites, as shown in Figure 1. The spoofer sends out a weak signal that becomes more and more powerful until it overtakes the signals from the satellites. Once locked to the receiver, false signals can be sent to the device to make it think it needs to change course to correct itself. The light blue line in Figure 2 shows the course the device or vehicle was supposed to take, and the solid line shows where it actually ends up. Whether it be a drone, boat, car, or phone, this concept is the same.

A basic understanding of how the GPS signals are sent and received is also needed to understand how spoofing works. Figure 1 does not have a random number of satellites. At any given time, a receiver is using signals from four different satellites, as shown. Each of these signals is processed with a delay for the time it takes for a signal to get from the satellite to a receiver. The delay is calculated with the individual signals to coordinate time and position. The calculations are as follows:

 ISBN: 978-1-61208-729-0

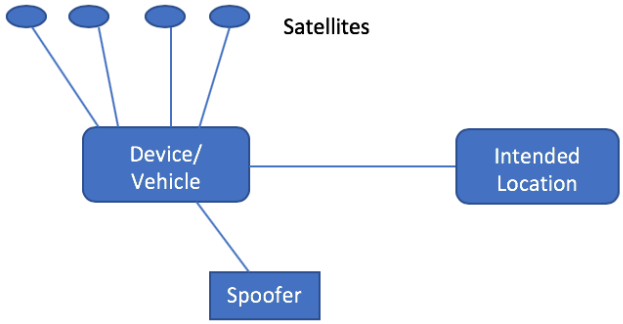

Figure 1. Phase 1.

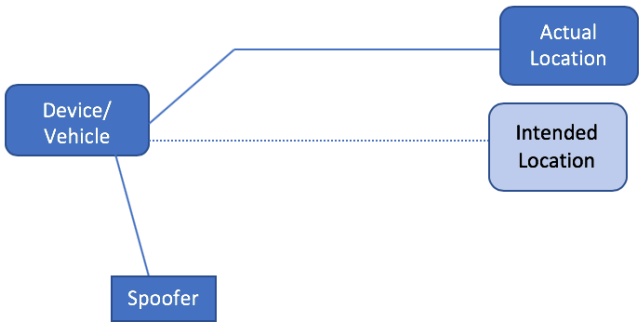

Figure 2. Phase 2.

$$Delay_i * C = Path_i \quad (1)$$

$$(T_i - T_0) * C = Position(sat_i) - position(RX) \quad (2)$$

$$(T + D_1 - T_0) * C = Pos(x_1, y_1, z_1) - Pos(x, y, z) \quad (3)$$

$$(T + D_2 - T_0) * C = Pos(x_2, y_2, z_2) - Pos(x, y, z) \quad (4)$$

$$(T + D_3 - T_0) * C = Pos(x_3, y_3, z_3) - Pos(x, y, z) \quad (5)$$

$$(T + D_4 - T_0) * C = Pos(x_4, y_4, z_4) - Pos(x, y, z) \quad (6)$$

The x, y and z variables with number subscripts represent the coordinates of the satellites, while the x, y and z without subscripts are the receiver coordinates. Modifying these signals with the expected delays is what allows a GPS device to be spoofed. The delays are all the D numbered variables in the equations. T0 is the initial time of the request while T is the current time. All that is needed to know about C is that it is a constant. As pointed out by [4], the proper delays are the most difficult part of GPS spoofing. Producing GPS signals is not that hard, but getting them to transmit with the proper delays to represent both the movement of the satellites and rotation of the earth is the key to a successful spoof. The four equations (3)-(6) are combined to get an accurate reading of the receivers location and time. All equations are taken from [4].

III. RELATED WORK

Some groups have tried and succeeded in spoofing cell phones and other mobile devices after overcoming the signal delay problem. [4] failed at this at first citing the problem of the Doppler effect. Once this obstacle is overcome and the mobile device is receiving the spoofed GPS signals is where this analysis starts to differ. Instead of testing Satellite Defined Radio (SDR) spoofing on several devices, it will be tested on two devices of similar specifications alongside a different type of location spoofing. These two different spoofing methods will be tested on a variety of applications including Snapchat, Tinder, Poke´mon Go, and more. While there is published research about spoofing the location of a user in Poke´mon Go [5], the paper does not delve into how these methods work on other location-based applications. There still remains the question of whether other applications are already able to detect this spoofing and if so what makes them different. Also, the methods they try do not include SDR spoofing. The authors fail to mention whether this was out of the scope of their analysis or seemed too difficult to implement, but it is a valid method of location spoofing all the same.

IV. METHODOLOGY

Many mobile applications use the location of their users for some purpose. The objective of this analysis is to see where these location-based applications are most vulnerable and on what platform. The two spoofing methods used will be SDR, where a signal will be transmitted to the tablet, and spoofing applications that alter the location on the tablet. Most researchers until now have focused on falsifying signals and transmitting them to phones, small drones, GPS receivers, etc. So, while there is a proof of concept for spoofing mobile devices the results of this analysis should provide a deeper understanding of how they affect the applications that people use and could disrupt their daily lives. The process by which the analysis will go is as follows:

1. First, the spoofing device will be set up, ready to transmit a signal and the spoofing applications chosen
2. It will then be confirmed the devices are receiving the false GPS signal from the spoofing device
3. The testing applications will be set up and used while the spoofing is taking place
4. All results will be recorded and analyzed

A. Devices

Testing will be done on two tablets using false GPS data. One is a Nvidia Shield K1 while the other is a 32GB iPad with Wi-Fi. Having two unique and distinct operating systems, iOS and Android, will allow for comparisons on which one, in general, is easier to spoof through applications and/or SDR or if they are similar. It is possible the architecture of one OS has a key vulnerability, in the way they receive and calculate GPS signals which the results should determine. The device generating the signal for the spoof is a HackRF, one capable of operating frequencies from 1 to 600 MHz. It is USB powered, plugging directly into a computer. As seen in Figure 3, the left side is used to attach an antenna for better reception and transmission while the right side has connector options for

 ISBN: 978-1-61208-729-0

Figure 3. HackRF One.

clock input and output. It is also capable of receiving data and could be used to record and replay data it receives to another device. We use predetermined coordinates instead of receiving and transmitting them live or through a replay.

B. *Software*

A simulator called "gps-sdr-sim" is available on GitHub [7]. There are several simulators available online, some free and others paid. However, the cost of paid simulators is very expensive and out of the scope for the average person. One of the arguments we are trying to make is anyone can hack GPS signals, so the technology would need to be easily available to all. The software chosen is compatible with the HackRF and has several options available. One is pulling data from a file that can be prepared before the run. If other SDR devices were used, a live signal could be recorded and rebroadcast at a different time and place. Another way is to just run it with a set of coordinates and a time and it broadcasts a steady signal, as shown in Figure 4. Only the coordinates at end of the terminal command are important to us as they are the coordinates of the intended fake location.

This is the simplest way but the most effective for our intentions. The tablets only need to spoof a steady location rather than actively move around, as pointed out in [6]. An active spoof is particularly difficult because, if at any point the spoofed signal is interrupted, the GPS device will try to reconnect and possibly connect back to satellites providing real data. Each device also uses a different spoofing application as there are different online marketplaces for each OS. Android will use an application called Floater. This application is free on the Google Play Store and fairly simple to set up. On iOS, an application called iSpoofer will be tested. It costs about thirteen dollars for a three-month subscription and requires a windows computer. However, there is a free trial available which will be used to not give an unfair advantage to one application. While not as convenient as the Android application, it is still accessible and usable by the average person.

C. *Performing the Spoof*

All the necessary applications will be installed on both devices with either the same or as close to the same version as possible. Otherwise, while unlikely, it is possible that security

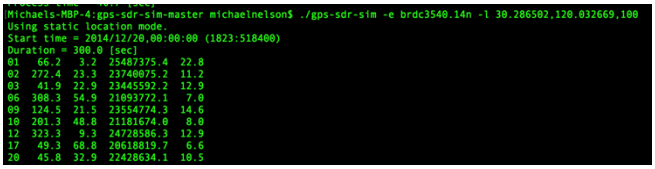

Figure 4. GPS Simulator Run.

holes we are exploring could have been patched in that time. Also to ensure fairness, the spoofing will be done at the same location with the same fake location. Otherwise, it would be possible that a certain area has stronger GPS signals, compromising the results.

V. EXPECTED RESULTS

Once the devices are being spoofed, several applications will be tested from categories including: streaming, social media, gaming, and possibly more. Their version number and whether they could be spoofed will all be recorded for analysis four times, once for each OS with both spoofing methods. Depending on the application, there will be different indications of a successful spoof. For example, some media streaming applications will show another countries library because of the location, but will not stream the content because it can tell that it is not the actual region of the device through other information like IP or DNS. This would mean the spoof was unsuccessful. Four tables similar to Tables 1 and 2 will be created for Android and iOS, respectively. Two tables will be for the application spoofing, and two tables will be for the SDR spoofing. These include some preliminary results for the application spoofing.

As expected, the SDR spoofing is a bigger challenge than the application spoofing due to improper delays. This is most likely due to a bad timing crystal, which is common in a simple HackRF device. Having a separate device that could offer a clock input for the HackRF could fix this. As for the results, Tables 1 and 2 have preliminary results from the application spoof. The application version numbers were recorded, which sometimes differ greatly because of the different release of updates on either OS. The "yes" or "no" in the app spoof rows signify whether the spoof was or was not successful. The applications tested can be further broken up into three categories: streaming, social media, or gaming. Poke´mon Go, Geocaching, and Turf Wars are gaming applications. Netflix and YouTube are for streaming. Snapchat and Tinder would be considered social media applications. By analyzing the results of the different categories, it is possible to see more vulnerabilities in one because of the method they extract their location data or rely on GPS. Conclusions could be drawn from the results received so far, however it is better to not make too many assumptions on incomplete data. As of now, it seems that streaming applications are less susceptible to spoofing, probably due to gathering locations from alternate methods than GPS. However, many applications were able to be spoofed on both operating systems, which displays big vulnerabilities in GPS

 ISBN: 978-1-61208-729-0

TABLE I. ANDROID SPOOF.

Application	Pokémon Go	Geocaching	Turf Wars	Netflix	Youtube	Snapchat	Tinder
Version	0.138.3	7.10.1	1.47	7.6.0	14.15.53	10.55.0.0	10.13.0
App Spoof	yes	yes	yes	no	no	yes	yes

TABLE II. IOS SPOOF.

Application	Pokémon Go	Geocaching	Turf Wars	Netflix	Youtube	Snapchat	Tinder
Version	1.109.0	7.10.0	2.981	11.31.2	14.16	10.56.0.23	10.12.1
App Spoof	yes	yes	yes	no	no	yes	yes

or in both iOS and Android operating systems. Final results should be able to give a more accurate and detailed analysis than possible right now.

VI. CONCLUSIONS AND FUTURE WORK

This analysis has shown the vulnerabilities that currently exist in mobile OSs and possibly GPS receivers. Further research could be done into why some applications can detect or block spoofing and others cannot. Once specific flaws are found, solutions and patches can be developed to ensure the security of location-dependent applications. As the users location is used in more and more systems, accuracy and security are paramount. Further analysis could be done with more spoofing applications, other SDR devices, or a different set of user applications. Multiple devices would be able to receive and transmit information in real-time, which may have a better effect than the static signals we used. There are a variety of methods and strategies that could be implemented to achieve the same result. What is important, is with the rise in the use of GPS in so many facets of life, advancement in security needs to follow or match it.

ACKNOWLEDGMENT

We are thankful for the computer science department at Iona College based in New Rochelle, New York for providing the tablets for testing as well as structuring their curriculum to encourage research.

REFERENCES

[1] M. L. Psiaki and T. E. Humphreys. “Protecting GPS From Spoofers Is Critical to the Future of Navigation.” IEEE Spectrum: Technology, Engineering, and Science News, IEEE Spectrum, 29 July, 2016.

[2] Official U.S. government information about the Global Positioning System (GPS) and related topics. [Online]. [retrieved: April, 2019] Available From: https://www.gps.gov/.

[3] L. Faria, C. A. Silvestre, M. A. Correia, and N. A. Roso. “Susceptibility of GPS-Dependent Complex Systems to Spoofing”. Journal of Aerospace Technology and Management, 10, e0218. Epub January 15, 2018 [retrieved: June, 2019] doi:10.5028/jatm. v10.839

[4] L. Huang and Q. Yang. “Low-cost GPS simulator – GPS spoofing by SDR”. Qihoo 360 Technology Co. Ltd. Defcon 23, Las Vegas. [PowerPoint slides] 2015

[5] B. Zhao and Q. Chen. “Location Spoofing in a Location-Based Game: A Case Study of Poke mon Go”. Advances in Cartography and GIScience (ICACI ’17), Lecture Notes in Geoinformation and Cartography. Springer, Cham. pp. 21-32. [retrieved: May, 2019] doi:10.1007/978-3-319-57336-6 2

[6] K. Zeng, Y. Shu, S. Liu, Y. Dou, and Y. Yang. “A Practical GPS Location Spoofing Attack in Road Navigation Scenario”. Proceedings of the 18th International Workshop on Mobile Computing Systems and Applications. (HotMobile ’17). Feb. 2017. pp. 85-90. [retrieved: May, 2019] doi:10.1145/3032970.3032983

[7] Simulator Link. [retrieved: May, 2019] https://github.com/osqzss/gps- sdr- sim

[8] Device Site Link. [retrieved: June, 2019] https://greatscottgadgets.com/hackrf/one/

[9] YouTube Application Link. [retrieved: June, 2019] https://www.youtube.com/

[10] Poke´mon Go [retrieved: May, 2019] Application Link. https://www.pokemongo.com/en-us/

[11] Netflix Application Link. [retrieved: June, 2019] https://www.netflix.com/browse

[12] Tinder Application Link. [retrieved: June, 2019] https://tinder.com/app/recs

[13] Snapchat Application Link. [retrieved: June, 2019] https://www.snapchat.com/

[14] Geocaching Application Link. [retrieved: June, 2019] https://www.geocaching.com/play/mobile

[15] Turf Wars Application Link. [retrieved: June, 2019] https://turfwarsapp.com/

 ISBN: 978-1-61208-729-0