A Machine Learning Based Smartphone App for GPS Spoofing Detection

Javier Campos[1], Kristen Johnson[1], Jonathan Neeley[1], Staci Roesch[1], Farha Jahan[2], Quamar Niyaz[1(✉)], and Khair Al Shamaileh[1]

[1] Purdue University Northwest, Hammond, IN 46323, USA
{jicampos,john1954,neeleyj,sroesch,qniyaz,kalshama}@pnw.edu
[2] The University of Toledo, Toledo, OH 43607, USA
farha.jahan@utoledo.edu

Abstract. With affordable open-source software-defined radio (SDR) devices, the security of civilian Global Position System (GPS) is at risk of spoofing attacks. Spoofed GPS signals from SDR devices have indicated that spoofed signals have higher values of signal-to-noise ratios (SNRs). Utilizing these values along with other parameters, we propose a machine learning (ML) based GPS spoofing detection system for classifying spoofed signals. To build our detection system, we launch spoofing attacks on a GPS receiver using a low-cost SDR device, LimeSDR, and apply ML algorithms on SNR values and the number of tracked and viewed satellites. A performance comparison between different ML algorithms shows that Random Forest (RF) and Support Vector Machine (SVM) achieve 99.5% accuracy, followed by K-Nearest Neighbors (KNN) (99.4%). To demonstrate easy integration of the algorithm with GPS enabled devices, we develop an Android-based smartphone app that successfully notifies the user about the spoofing signals.

Keywords: GPS spoofing · Machine learning · Security · Smartphone app

1 Introduction

From mobile phones to aviation and autonomous vehicles, the use of Global Positioning System (GPS) for navigation and timing has become ubiquitous. As more people and devices rely on GPS, the threat of spoofing attacks increases [2,5]. GPS signals are vulnerable to being spoofed, thus displaying incorrect/inaccurate locations to the user. Civilian GPS signals are not encrypted, and their receiving devices lack effective defense mechanisms, thereby posing a higher risk of a spoofing attack. A GPS defense system must at least detect a spoofed signal and notify the user.

In this work, a machine learning (ML) based GPS spoofing detection mechanism that uses parsed information from the National Marine Electronics Association (NMEA) sentences is proposed. These sentences are standard data format

N. Park et al. (Eds.): SecureComm 2020, LNICST 336, pp. 235–241, 2020.
https://doi.org/10.1007/978-3-030-63095-9_13

supported by most GPS modules. The defense mechanism extracts signal-to-noise ratio (SNR) values and the number of tracked and viewed satellites from NMEA sentences to classify GPS signals. To demonstrate effectiveness of the defense mechanism and its integration flexibility with GPS enabled devices, we implement it as an Android app that notifies the user when a false signal is received and stops updating the spoofed location. Detecting GPS spoofing attacks based on ML techniques have been previously explored in many works [7,9,10]. A work similar to ours demonstrated GPS Spoofing attack on mobile phones, external GPS modules, and car navigation system through RINEX files which provide only raw satellite data [4]. Our work differs in a way that we use ML algorithms for spoofing detection with the help of features extracted from NMEA sentences, which are supported by most GPS modules. Besides, we develop a smartphone app that can use the ML model to detect spoofed GPS signals. A GPS Anti-Spoof app is available on the Google Play that uses celestial navigation instead of analyzing any GPS signals or NMEA sentences [1].

2 Methodology

To design and implement the defense mechanism against a GPS spoofing attack, we setup the necessary hardware and software to generate and collect authentic and spoofed GPS signals. A software-defined radio (SDR) kit, LimeSDR [6], is used. The kit is connected to a PC via a USB cable and integrated with GNU-Radio and GPS-SDR-SIM software. Ephemeris data downloaded from NASA's Archive of Space Geodesy on its Crustal Dynamics Data Information System (CDDIS) is used to launch the spoofing attack. The data provides information on current and predicted location, timing, and health of GPS satellites [8]. The downloaded ephemeris files for the desired date and time are in compressed format. They are first decompressed and converted into binary files of GPS baseband signal data streams using GPS-SDR-SIM. GNURadio converts these binary files into radio frequency signals and transmits such signals to a GPS receiver via LimeSDR. The transmission successfully spoofs the GPS receiver to a false location. Spoofing attacks are also launched using the setup on a smartphone equipped with a built-in GPS module to validate the defense mechanism of our developed app against the attack.

An Arduino microcontroller interfaced with a u-blox NEO-6M GPS module receiver is chosen to collect and parse GPS data. The output of the GPS module is NMEA sentences shown in Fig. 1. These sentences include all the information provided by GPS signals such as latitude, longitude, number of satellites being tracked, and SNR. The two most important NMEA sentences are "GPGGA" and "GPGSV". GPGGA sentences contain location information (latitude, longitude, altitude, and the number of satellites). The GPGSV sentences contain information on the satellites within the view. One of its parameters also describes the SNR of each satellite that played a key role in our ML implementation [3].

```
$GPGSA,A,1,,,,,,,,,,,,,99.99,99.99,99.99*30
$GPGSV,2,1,05,01,,,20,02,,,21,07,,,23,12,,,22*78
$GPGSV,2,2,05,13,,,21*7D
$GPGLL,,,,,,V,N*64
$GPRMC,,V,,,,,,,,,,N*53
$GPVTG,,,,,,,,,N*30
$GPGGA,,,,,,0,00,99.99,,,,,,*48
$GPGSA,A,1,,,,,,,,,,,,,99.99,99.99,99.99*30
```

Fig. 1. Sample NMEA sentences captured in Arduino interfaced with a GPS receiver.

2.1 Machine Learning Based Defense Mechanism

The stages that are involved to develop the prototype for the defense mechanism against a GPS spoofing attack are as follows:

Capturing GPS Data. The defense mechanism detects a spoof according to the parsed information from NMEA sentences. Thus, NMEA sentences are collected from the GPS module in two scenarios, one of authentic locations and other for spoofed locations. For the former locations, the GPS module is set to operate without interference from the LimeSDR, and NMEA sentences are collected for 20 min each at ten different true geographical locations. After the data for true locations is collected, the LimeSDR is set up about 40 ft from the GPS receiver. False versions of the true locations are generated with the corresponding ephemeris data and transmitted from the LimeSDR. With any distance greater than 40 ft, the GPS receiver was unable to receive signals. NMEA sentences for these false locations were collected for 20 min as well. Later, these NMEA sentences for authentic and false locations are used to prepare the dataset for the development of an ML model.

Detecting Spoofing Attacks Using Machine Learning. The ML algorithm classifies locations as authentic or spoofed based on a given set of input features. These features are extracted from collected NMEA sentences. The information about each satellite is listed in GPGSA and GPGSV sentences including the number of satellites within range and their SNR values. It is observed that a valid GPS signal has lower SNR as compared to spoofed signals by the LimeSDR device. Therefore, for each position in the parsed data, the average SNR value and standard deviation are considered as features to identify a location as spoofed or authentic. Other information from NMEA sentences, such as the number of satellites in view and tracked and horizontal dilution is analyzed. After a close examination, there is not enough evidence to conclude that horizontal dilution could help differentiate between authentic and spoofed locations.

However, the number of satellites in view and being tracked did show promising results. Hence, we created a dataset that consists of one output class for a location to be genuine or spoofed and four input features: i) average SNR, ii) standard deviation in SNR, iii) number of satellites in view, and iv) number of satellites being tracked.

Table 1. Accuracy, precision, recall, and f-measure for each machine learning algorithm

	Accuracy (%)	Precision (%)	Recall (%)	F-measure (%)
KNN	99.46 ± 0.001	99.62	99.57	99.60
RF	99.53 ± 0.002	99.70	99.70	99.67
SVM	99.55 ± 0.002	99.65	99.59	99.62
LR	99.10 ± 0.007	98.46	98.27	98.35
NB	97.83 ± 0.008	98.29	97.97	98.10

From the collected NMEA sentences, 19,925 records are extracted for the dataset. These records are split into training, validation, and testing using a 60-20-20 split, respectively. We considered five traditional ML classification algorithms for the detection mechanism: K-Nearest Neighbors (KNN), Random Forest (RF), Support Vector Machines (SVM), Logistic Regression (LR), and Naïve Bayes (NB). Table 1 displays the performance of each ML algorithm for well-known performance metrics for classification including accuracy, precision, recall, and f-measure. Results are recorded after using cross-validation of five-fold for each classifier using data from the test set. It is found that KNN, RF, and SVM models performed better than LR and NB models in terms of accuracy and f-measure. The first three algorithms achieved an accuracy of around 99.5% and f-measure of 99.6%.

3 Smartphone App Implementation

To implement the ML based detection mechanism in a device, we developed an Android app and installed it on a smartphone. The predefined `LocationListener` interface from Android API detects any location change and updates the smartphone's position (longitude and latitude) regularly through the `onLocationChange()` method. We use Google Maps for a visual display of the location. Once the app is ready to read the GPS data, we broadcast spoofed signals, and the map shows the spoofed location. For the ML algorithm, we used KNN due to its easy implementation and comparable performance with RF and SVM. The KNN algorithm for detection is written in Java as a separate module for integration with the app, and the initial collected data used by the algorithm are stored as a text file. Since the algorithm uses parsed information from NMEA sentences, they are extracted using the `LocationManager` class, which can read

the NMEA sentences (code snippet shown in Fig. 2). The app parses each string starting with GPGSA and GPGSV, and extracts information. It then sends the data to the algorithm to predict whether the received signal is faulty or not. An alert pop-up indicates that the GPS signal is spoofed.

```
if(s.startsWith("$GPGGA")) {
    String[] splitSentence = s.split(",");
    if (!splitSentence[7].isEmpty())
        satellitesTracked = Integer.parseInt(splitSentence[7]);
    else
        satellitesTracked = 0;
}

if(s.startsWith("$GPGSV")) {
    String[] splitSentence = s.split(",");
    if(!splitSentence[3].isEmpty())
        satellitesInView = Integer.parseInt(splitSentence[3]);
    else
        satellitesInView = 0;
    if(NmeaSentenceCount == 0) {
        NmeaSentenceCount = Integer.parseInt(splitSentence[1]);
        NmeaSnr = new ArrayList<Integer>();
    }
    int i = 7;
    while(i < splitSentence.length) {
        if(!splitSentence[i].isEmpty()) {
            Log.d("GPGSV", "String: " + splitSentence[i]);
            NmeaSnr.add(Integer.parseInt(splitSentence[i]));
        }
        i += 4;
    }
    if(Integer.parseInt(splitSentence[2]) == NmeaSentenceCount) {
        checkLocation(NmeaSnr, satellitesInView, satellitesTracked, knn);
        NmeaSentenceCount = 0;
    }
}
```

Fig. 2. Code snippet of the Android app

We evaluated the Android app on a Samsung Galaxy S9 smartphone at one of the author's residence. We downloaded the ephemeris data and followed the steps discussed in Sect. 2 to spoof the location to other place. The spoofed radio frequency signals were received by the smartphone. The Android app successfully identified the spoofed signal and displayed a warning to the user and stopped further location updates. Figure 3 shows the app interfaces before and after the GPS spoofing attack. Airplane mode was turned on to prevent the smartphone from using Wi-Fi and mobile data to refine location estimations.

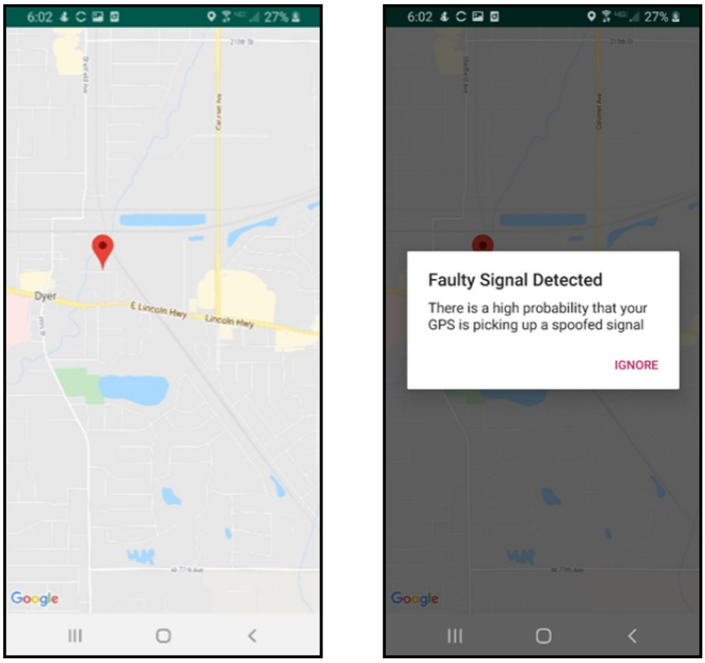

Fig. 3. Android app graphical interfaces for genuine and spoofed locations

4 Conclusion and Future Work

In this work, we implemented an ML based defense mechanism against GPS spoofing attack and embedded it in an Android app. We found that ML algorithms such as KNN, RF, and SVM can detect such attacks with an accuracy of around 99.5%, which predominately utilizes SNR values, the number of satellites viewed, and being tracked from NMEA sentences to categorize spoofed signals. We implemented the KNN algorithm in an Android app that would notify the user of a spoofed signal and prevent the map from displaying the spoofed location.

Our experiments and defense mechanism are setup keeping amateur attackers in mind who can launch attacks by using low-cost SDR kits and following the available online resources. The defense mechanism can defend against attacks by those attackers who are not familiar with manipulating hardware and software of SDRs to mimic authentic signals more closely. In future, we will investigate the impact of manipulating SNRs and the number of satellites in view on the ML based defense mechanism to make it robust and widely applicable. Integrating an accelerometer and a gyroscope with the smartphone app can be used to cross-reference measured data from GPS satellites. Furthermore, the app will be profiled to optimize memory and CPU usage for better user experience.

References

1. GPS Anti Spoof. https://play.google.com/store/apps/details?id=com.clockwk.GPSAntiSpoof
2. Woodford, C.: Satellite navigation (2019). https://tinyurl.com/y8wss3wt
3. DePriest, D.: NMEA Data (2019). https://tinyurl.com/b7jvw
4. Goavec-Merou, G., Friedt, J., Meyer, F.: GPS spoofing using software defined radio (2019)
5. Jahan, F., Javaid, A.Y., Sun, W., Alam, M.: GNSSim: an open source GNSS/GPS framework for unmanned aerial vehicular network simulation. ICST Trans. Mob. Commun. Appl. **2**(6), e2 (2015)
6. Lime Microsystems: LimeSDR (2019). https://limemicro.com/products/boards/limesdr/
7. Manesh, M.R., Kenney, J., Hu, W.C., Devabhaktuni, V.K., Kaabouch, N.: Detection of GPS spoofing attacks on unmanned aerial systems. In: 2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC), pp. 1–6. IEEE (2019)
8. Noll, C.E.: The crustal dynamics data information system: a resource to support scientific analysis using space geodesy. Adv. Space Res. **45**(12), 1421–1440 (2010)
9. Panice, G., et al.: A SVM-based detection approach for GPS spoofing attacks to UAV. In: 2017 23rd International Conference on Automation and Computing (ICAC), pp. 1–11 (2017)
10. Shafiee, E., Mosavi, M., Moazedi, M.: Detection of spoofing attack using machine learning based on multi-layer neural network in single-frequency GPS receivers. J. Navigat. **71**(1), 169–188 (2018)