# Spotr: GPS Spoofing Detection via Device Fingerprinting

Mahsa Foruhandeh, Abdullah Z. Mohammed, Gregor Kildow, Paul Berges, and Ryan Gerdes
Virginia Tech
{mfhd,abdullahzubair,gregor,paulberges,rgerdes}@vt.edu

## ABSTRACT

As the world's predominant navigation system, GPS is critical to modern life, finding applications in diverse areas like information security, healthcare, marketing, and power and water grid management. Unfortunately this diversification has only served to underscore the insecurity of GPS and the critical need to harden this system against manipulation and exploitation. A wide variety of attacks against GPS have already been documented, both in academia and industry. Several defenses have been proposed to combat these attacks, but they are ultimately insufficient due to scope, expense, complexity, or robustness. With this in mind, we present our own solution: *fingerprinting* of GPS satellites. We assert that it is possible to create signatures, or fingerprints, of the satellites (more specifically their transmissions) that allow one to determine nearly instantly whether a received GPS transmission is authentic or not. Furthermore, in this paper we demonstrate that this solution detects all known spoofing attacks, that it does so while being fast, cheap, and simpler than previous solutions, and that it is highly robust with respect to environmental factors.

## CCS CONCEPTS

• **Security and privacy** → **Spoofing attacks**; **Hardware attacks and countermeasures**; • **Hardware** → **Signal integrity and noise analysis**; *Digital signal processing*;

## KEYWORDS

Signal Fingerprinting, Spoofing Detection, GPS

**ACM Reference Format:**
Mahsa Foruhandeh, Abdullah Z. Mohammed, Gregor Kildow, Paul Berges, and Ryan Gerdes. 2020. Spotr: GPS Spoofing Detection via Device Fingerprinting. In *13th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec '20), July 8–10, 2020, Linz (Virtual Event), Austria.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3395351.3399353

## 1 INTRODUCTION

GPS, in its original role as a navigational aid, can be found in everything from smartphones to unmanned aerial vehicles (UAVs). This alone would merit the attention of security researchers. Today trackers use it to trace the movement of entities, games and streaming services use it for localizing advertisements, secure facilities use it to "geo-fence" their equipment, and power grids use it for phase synchronization. Put simply, there are very few aspects of modern life that are not deeply dependent upon GPS.

In order to spur the growth of GPS and ease its adoption by the private sector, all aspects of the design and implementation of the so-called L1 C/A ("Coarse Acquisition") signal were made public. GPS—"...the most popular unauthenticated protocol in the world."[17]—has since become the primary Global Navigation Satellite System (GNSS). Allowing anyone to acquire a detailed understanding of GPS with very little effort, combined with the relatively trusting nature of GPS transmitter/receiver interaction, makes the system ripe for manipulation and catastrophic exploitation. As such, we must harden GPS as a matter of national security.

Several methods that attempt to provide post hoc security for GPS exist in the literature [31]. Our approach uses physical-layer identification (PLI), aka *device fingerprinting*, and falls within the category of correlation profile anomaly detection. There is a long history in using device fingerprinting for securing transceivers [14] and satellites are an example of such transceivers. We introduce Spotr, GP**S** s**p**o**o**fing detec**t**ion via device fingerp**r**inting that is able to determine the authenticity of GPS signals based on their physical-layer similarity to signals that are known to have originated from GPS satellites. We extract strong features from the outputs of the complex correlators common to all GPS receivers and use them to generate templates for the genuine satellite signals, which we call *fingerprints*. Finding a strong fingerprint for genuine GPS signals is non-trivial, since a spoofing adversary always tries to mimic the characteristics of the authentic signal to the best of their capability. We introduce simple (but difficult to spoof) features here, yet we are able to detect the most powerful spoofing attacks with high accuracy and in a timely fashion. Our main contributions are:

- We detect all the existing spoofing attacks on different dimensions of GPS receivers in literature (time and/or position) using a single channel/receiver.
- We are able to track a genuine satellite over different days, environmental conditions, and locations, such as urban and rural settings, with and without multipath propagation effects. Our method does not require channel modelling, which reduces its complexity. We are only limited to having a good observation of the signal to perform device fingerprinting.
- Since we use fingerprinting, we do not impose a lower-bound on the range of spoofing detection. This is an improvement to the state-of-the-art spoofing detectors which fail to find attacks up to the range of 1000 m.

Our spoofing detector functions properly regardless of the number of devices that the spoofer is utilizing. This enables us to detect coordinated attacks on multilateration systems (estimation of distance based on time of arrival of waveforms travelling at a known speed [21]). Our approach is purely passive with no modifications required to the existing GPS protocol, satellite orbits or uplink/downlink

*WiSec '20, July 8–10, 2020, Linz (Virtual Event), Austria*

ACM ISBN 978-1-4503-8006-5/20/07...$15.00
https://doi.org/10.1145/3395351.3399353

communication channels. We also make our dataset of genuine and spoofed GPS signals available for the community.

## 1.1 Related Work

The public availability of civil GPS implementation details has made it trivial to accurately and reliably (re)create navigational data intelligible to any GPS receiver. What's more, these receivers are not only inherently trusting and accept any GPS signal they can demodulate, but the automatic gain control (AGC) system in the standard GPS receiver is designed to favor (lock onto) the strongest signals. While this behavior may seem intuitive, it also assumes – dangerously – a benign environment which means that a malicious attacker can easily assert control over a receiver simply by using a closer, more energetic signal to overpower the weaker, authentic signals. This is known as the *overpower attack* and it is the focus of most detection and mitigation strategies. Numerous lines of defense are given in [17, 31] which can be broadly categorized as cryptographic, hardware-specific or signal processing-based solutions.

Non-predictable modulation techniques are proposed to overcome the problem of open signal structure in [30, 38]. These are mostly cryptography-based methods that, while valid, depend on modifications to the space segment. As such, the costs involved with redevelopment or even modification of satellites in situ render these solutions impractical despite their effectiveness.

Jamming, often seen as a threat to GPS, can also be a standalone, non-cryptographic line of defense against GPS spoofing. One such approach uses jamming-to-noise (J/N) sensors that are inexpensive and easy to build. The sensors mark activity as malicious if the energy of the in-band signal exceeds a given threshold, which forces the spoofer to limit its power and makes the overpower attack far more difficult to maintain. This defense is probabalistic, however, as a carefully crafted spoofer could still employ more nuanced power adjustments and it would be unwise for one to assume anything less than maximum determination in an adversary [17, 37].

Hardware-based solutions include those that augment GPS receivers with additional antennae or inertial sensors. The single/ multiple antenna methods are fast and reliable, but are expensive and require modifications to current receivers and redesign of future receivers [20, 27, 35]. For their own part, the inertial sensors create a new accuracy concern as their measurements are temperature sensitive and thus subject to drift depending on environmental factors [10].

Another set of defense methods implement spread spectrum security code (SSSC) or navigation message authentication (NMA) on wide area augmentation system (WAAS) [11]. WAAS is an air navigation aid developed by the Federal Aviation Administration (FAA) to augment GPS in order to increase its accuracy, integrity, and availability. With no modifications to the space segment of the GPS, this is strong enough to stop the spoofer because SSSC is a strong high rate security code; however, it cannot authenticate a full three-dimensional navigation solution. Long delay is another drawback of SSSC if used in aviation. NMAs, easier to implement compared to SSSC, are slightly less secure with equal delay [23].

A recent alternative solution is the Multi-System Multi-Frequency defense. Secondary to the U.S. Air Forces's ongoing GPS modernization project, civilian GPS signals are now being transmitted on other bands such as L2 and L5 in addition to the legacy L1. The signals at L2 and L5 can be used for consistency check of the signal at L1, which makes spoofing more challenging [25].

Correlation profile anomaly detection is a different approach which mainly relies on the difficulty of suppressing the genuine GPS signals and the fact that even the strongest spoofer fails at exactly mimicking the authentic GPS signal's behaviour. This helps to find anomalies and use them to detect spoofing activities. It is a low cost software solution which eliminates the need for additional hardware. This method is known to function efficiently for stationary receivers, while its performance is influenced by propagation effects of the wireless channel such as multipath and fading [19]. In a similar way, a PLI-based method for spoofing attacks is given in [28] where the frequency offset and transient phase noise of the attacker's radio are the main features for detection, which are weaker than the correlator outputs used in the present work as they are more susceptible to spoofing.

Some techniques attempt to secure the GPS C/A signals using the existing military signals which are not cost-efficient [32].

Numerous attacks have already been demonstrated on the GPS system, the strongest of which is a seamless lock takeover attack [35] where the attacker takes over the target receiver gradually, by avoiding abrupt changes which might lead to detection. SPREE is introduced in [33] which is a spoofing resistant GPS receiver, able to limit the state-of-the-art spoofing attacks up until 2016 including the seamless lock takeover attack [35]. It introduces a spoofing detection technique named auxiliary peak tracking (APT) which operates next to the second module of SPREE, named navigation message inspector (NMI). Unlike ordinary GPS receivers, SPREE acquires and tracks all of the signals (the strongest and the weaker ones), and uses NMI to look for discrepancies in the content of the navigation messages to detect possible attacks. It contributes to the security of GPS systems by limiting the range of spoofing attacks on position to the radius of 1000 m. This is accomplished at the cost of using more than one channel to acquire, track and decode each satellite's signal, which also requires modifications to the GPS receiver. The main challenge for SPREE is to distinguish the auxiliary peak of a spoofing signal from that of a multipath component of a genuine GPS signal. This makes it difficult for SPREE to cancel the spoofing signal due to uncertainties at identifying the source of the signal. SPREE also relies on the presence of the authentic signals (assuming that these signals are not already suppressed by the attacker) which makes it vulnerable to attacks which nullify the authentic signals.

In a similar way, Vestigial Signal Defense (VSD) [39] attempts to detect spoofing based on analyzing the distortions present in the output of the receiver's complex correlation function. VSD faces challenges to distinguish these distortions from legitimate multipath components. Other similar detection strategies which rely on inherent spatial characteristics of the received signal (e.g. angle of arrival [26]) face the same challenges. Simpler spoofing detection strategies that look for anomalies in the physical-layer characteristics of the received signal such as abrupt changes in power level of the received signal or the AGC value [1] are usually not able to detect a spoofer which has proper control over its signal.

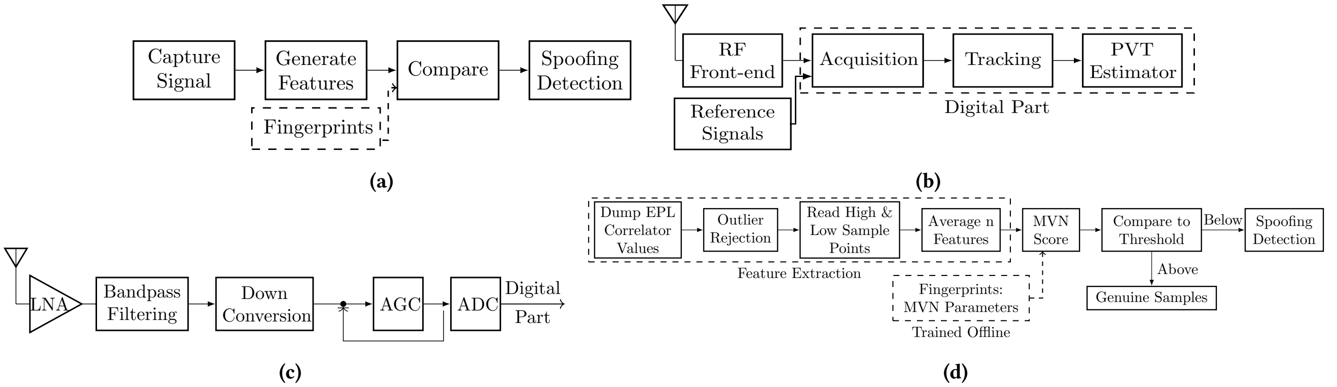


**Figure 1: (a) Overview of satellite fingerprinting, spoofing detection. (b) Block diagram of GPS receiver. (c) RF front-end of GPS receiver. (d) Spotr overview of defense.**

### 1.2 Paper Structure

Section 2 covers the types of attacks that we address. Sec. 3 provides a background on GPS and describes our defense methodology where we introduce Spotr with device fingerprinting for detecting spoofing attacks. The same section covers the feature extraction and the real-time spoofing detection process. Data collection is explained in Sec. 4. Experimental validation of Spotr is discussed in Sec. 5, where we also do a feature stability analysis to examine the consistency on different conditions before we conclude in Sec. 6.

## 2 ATTACK MODEL

We envision a physical-layer attack where the objective of the attacker is to induce a different position, velocity, and/or time (PVT) solution in a targeted civilian GPS receiver. A spoofing attacker is able to craft digitally valid GPS signals indistinguishable from genuine ones, inject them into a wireless channel, and produce a valid PVT at the victim. A replay attacker, on the other hand, captures analog data from genuine satellites and replays it to the victim. The latter is a stronger attack since it replicates the subtle intrinsic features of a genuine signal. The attacker is aware of the location of the target receiver. In both cases, if a receiver locks to the fake satellite (signal) then the attacker can choose a desired/altered PVT. We note that GPS signal generators that would enable this are readily available. We do not place any artificial restrictions on the attacker's ability to generate and inject signals, the spoofed signal may be greater in power than the genuine GPS signals ("over-powered"), at a slightly higher power level ("matched-powered"), at a lower power level ("under-powered") in the case of a near-instantaneous switch from an exclusively authentic stream into an exclusively spoofed one, and/or phase aligned to enable "seamless" takeovers [18, 19]. This maximizes the chances of an attacker to remain undetected while enabling all of the attacks discussed in [33]. Finally, a replay attacker is able to adjust power levels and create signals with correlation outputs as close as possible to genuine ones.

## 3 SATELLITE SPOOFING DETECTION

Here, we will describe our physical-layer based spoofing detection scheme, Spotr. Feature extraction is the most essential step in designing a PLI system. We start this out by finding simple, yet strong features that make the detection possible while cutting down on the complexity. Next, we use an offline training-testing scheme to generate thresholds and use them for real-time detection of spoofing or tracking of genuine GPS signals. Fig. 1a shows an overview of a PLI system in the context of satellite spoofing detection, detailed below.

### 3.1 GPS Receiver

Fig. 1b illustrates a block diagram of a GPS receiver. The main components of a GPS receiver are (i) RF front-end, (ii) acquisition module, (iii) tracking module, and (iv) PVT estimator module. The RF front-end investigates if satellite signals are available. The acquisition module searches for the delayed versions of any satellite's pseudo random noise (PRN) sequence while estimating the Doppler shift and compensating for it. PRNs are codes used to differentiate the signals generated from different satellites in a code division multiple access (CDMA) system. The role of a tracking block is to follow the evolution of the signal synchronization parameters: code phase, Doppler shift and carrier phase. As a main component of the tracking block, the VOLK-Library [12] from gnss-sdr, is responsible for running the delay locked loop (DLL) and phase locked loop (PLL) at the GPS receiver. It runs at a varying speed in a feedback system based on the quality of tracking in that instance. The PVT estimator uses this to report a solution on position, time and velocity. The most common method to do so is called multilateration, where an accurate estimate of location coordinates of a GPS transmitter plus time requires data from at least four satellites [22].

### 3.2 Feature Selection Rationale

GNSS receivers are composed of an analog RF front-end and a digital part. Fig. 1c illustrates the block diagram of the RF front-end with an automatic gain control (AGC) component right before the Analog-to-Digital converter (ADC). AGC is an adaptive feedback loop system that uses a variable-gain amplifier (VGA) in order to provide consistent inputs to the ADC [3]. This consistency makes correlators a good observation point for feature extraction. A potential optimal detector, to determine the presence of a valid signal, is a matched-filter, which maximizes the signal-to-noise ratio of a known input signal in additive white Gaussian noise (AWGN) [7].

Earlier work in fingerprinting satellites uses weaker features, such as modulation imperfections or AGC parameters, that are known to be vulnerable to low-cost spoofing attacks [6, 8]. Correlator outputs [1], however, require better resourced attackers with arbitrary waveform generators (AWGs) [15]. In an effort to mimic such an attacker we perform matched-powered replay attacks in Sec. 5.2. Their security, together with the theory of operation of the AGC, make the correlation outputs reliable features (Fig. 2a).

## 3.3 Feature Extraction

Based on documentation of GPS, we have samples after I-Q demodulation at the RF front-end which contain information on the P(Y) and C/A code, the military and civilian codes, respectively. The output of the matched-filter correlation, $\mathrm{M_o}$ would be

$$\begin{aligned} \mathrm{M_o} &= (I + \mathrm{j}\,Q) \times (I_{\text{rep}} - \mathrm{j}\,Q_{\text{rep}}) \\ &= (I \times I_{\text{rep}} + Q \times Q_{\text{rep}}) + \mathrm{j}\,(Q \times I_{\text{rep}} - Q_{\text{rep}} \times I) \end{aligned} \tag{1}$$

where rep shows the replica of the PRN code in the receiver. Given the right-hand side of the equation, if there is a match, the real part will hold a large enough value and the imaginary part will be close to zero which will result in a lock at the tracking block. There are three such correlators shown in Fig. 2a, named early, prompt, and late correlators called, E, P and L. A sample of the real and imaginary part of the P correlator is given in Fig. 2b where we will focus on the real part which holds most of the information from the satellite. The ideal value of the imaginary part is zero [24].

The gnss-sdr receiver uses two metrics to validate the signal quality to generate a lock, named the code-lock-detector (CN0) and carrier-lock-detector (CLT). The CN0 test is defined as

$$\widehat{C/N}_{0_{dB-Hz}} \underset{\text{no lock}}{\overset{\text{lock}}{\gtrless}} \gamma_{code}, \tag{2}$$

where $\widehat{C/N}_0$ is an estimate of $C/N_0 = \frac{C}{\frac{N}{BW}}$ ($C$ is the carrier power, $N$ is the noise power and $BW$ is the bandwidth of the observation) which is the carrier-to-noise density ratio and refers to the ratio of the carrier power and the noise power per unit of bandwidth. The threshold $\gamma_{code}$ is set to 32 dB-Hz here. The the lock detector test for the carrier tracking loop is defined as,

$$\cos(2\widehat{\Delta\phi}) \underset{\text{no lock}}{\overset{\text{lock}}{\gtrless}} \gamma_{carrier}, \tag{3}$$

where $\Delta\phi = \phi - \hat{\phi}$ is the carrier phase error. If the estimate of the cosine of twice the carrier phase error is above a certain threshold, the loop is declared in lock. The threshold $\gamma_{carrier}$ is set to 0.5 radians. This is referred to as CLT in our data collection software at Sec. 4. Fig. 3a shows the histograms of the CN0 and CLT, which are used for deciding on the thresholds.

After filtering out the sample points that fail the above lock tests, we look into the real values of the EPL correlator outputs to generate strong features for each satellite. For each correlator, we separate the points above zero and the ones below zero as high/low sample points and we will have features of six dimensions which are high and low sample points of EPL correlator outputs. Feature one is high of E correlator, feature two is low of E correlator, feature three is high of P correlator, feature four is low of P correlator, feature five is high of L correlator, and feature six is low of L correlator. A histogram of features 5 and 6 is shown in Fig. 3b.

## 3.4 Multivariate Normal Distribution

As discussed in Sec. 3.3 we are dealing with high dimensional data. In order to describe the characteristics of this data we apply a multivariate normal distribution (MVN) [4] to each dataset. For a set of $\{\mu, \Sigma\}$ that we assign, the MVN score is calculated by Eq. 4. The Gaussian distribution fits the closest to our features, one example of which is plotted in Fig. 3b. We hypothesize that the features are correlated and we intend to capture this relation between them; that is the reason we choose MVN as the scoring metric. It is the main metric for measuring similarity in our work and translates to how close each observation from a dataset is to a specific distribution.

$$f_x\,(x_1, x_2, \cdots, x_k) = \frac{\exp\left(-\frac{1}{2}\,(x-\mu)^T\,\Sigma^{-1}\,(x-\mu)\right)}{\sqrt{(2\,\pi)^k\,|\Sigma|}}. \tag{4}$$

To guarantee a clear separation between the genuine GPS signals and the spoofed ones in different weather conditions or locations, we use the average MVN score from multiple sample points. Fig. 3c shows the effect of using a single sample point for comparison versus using the average of 100 sample points. The lower graph shows a better separation between the genuine and spoofed signals. Hence, in the experimental evaluations in Sec. 5 we will use multiple sample points (denoted as $n$) and average their MVN scores.

## 3.5 Real-time Detection

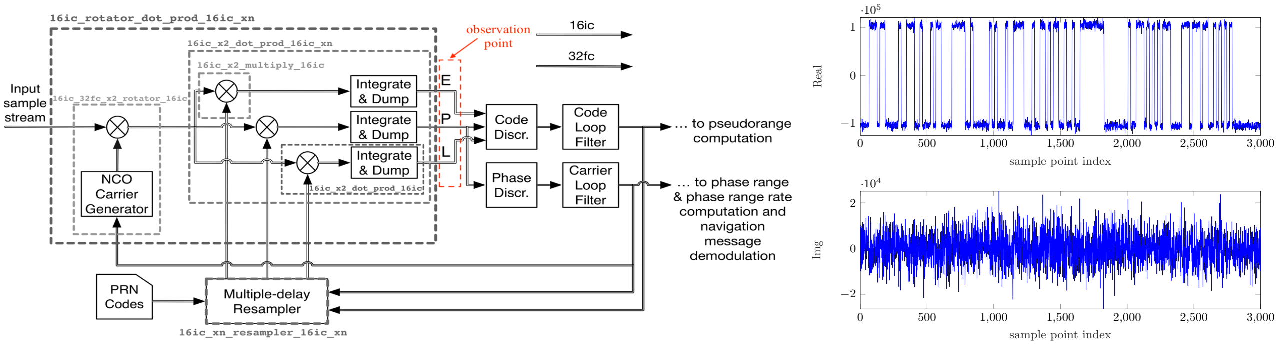
We design our PLI-based spoofing detector in two main steps. First, we do a training and testing procedure which results in thresholds for identification of genuine satellite signals from the spoofed ones. Next, we train the detector to trigger once an anomaly is observed using these thresholds in a real-time manner.

**Training:** In this step we use 400 sample points observed from a spoof-free dataset and generate a template set of features calculated according to Sec. 3.3. We use these training features to fit an MVN distribution for the genuine GPS signals. We assume training is secure, which is accomplished by collecting data at known locations.

**Testing:** We calculate the MVN score for each observation from the spoofed GPS dataset. A low MVN score is an indication of spoofing in the GPS signal. To define a threshold and finalize the spoofing detection process, we conduct a binary search algorithm to find a threshold that corresponds to equal error rates (EER). EER is a measure of performance for bio-metric systems which indicates

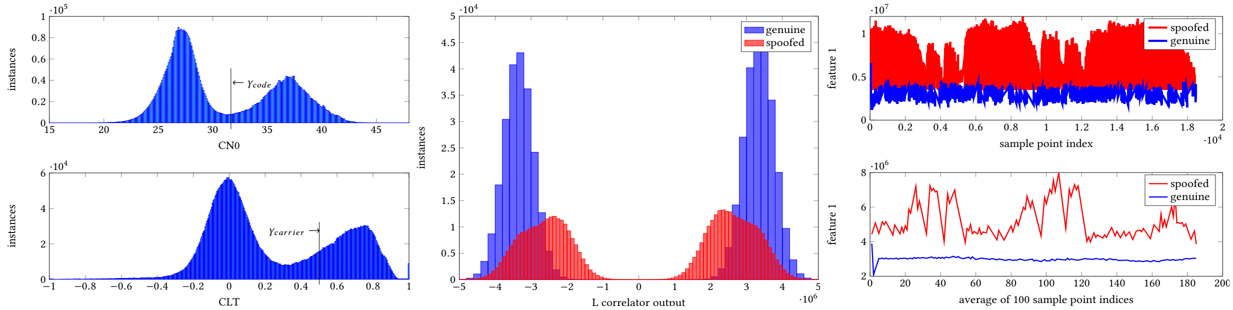
```
Algorithm 1: Feature Extraction
  Generating a feature template F for each satellite
  for Each PRN $i > 1$ do
      Collect the outputs of EPL correlators
      $F_i \leftarrow$ high and low of EPL outputs
      Apply code-lock-detector test (Eq.2)
      Apply carrier-lock-detector test (Eq.3)
  $F \leftarrow F \cup F_i$
```

[1] GPS uses direct sequence spread spectrum (DSSS) modulation. For the purpose of demodulation and de-spreading, correlators are required in the receiver design. Therefore, they are an inherent part of any GPS receiver.



**(a)** **(b)**

**Figure 2: (a) EPL correlator outputs for GPS digital receiver [12], the observation point of Spotr. (b) Output of P correlator.**



**(a)** **(b)** **(c)**

**Figure 3: (a) Histogram of CN0 and CLT for SatGrid:S10 (see Tables 3 and 4 for details on the datasets). (b) Late correlator output for PRN23 of TexBat:S8. (c) Using average of multiple features to create separation between spoofed and genuine signals.**

a condition in which the false positive rates (FPR) equal to false negative rates (FNR). An ideal value for EER would be zero [5].

**K-fold Cross Validation:** Using the training-testing procedure explained above, we define an extended threshold which identifies all the genuine GPS signals from the spoofed ones for all but a specific dataset. Next, we validate this threshold by calculating the MVN scores for that dataset, and perform the detection process using this threshold. Then, we look into the FPR and FNR to evaluate how well this threshold functions on this dataset which was not included during the training phase. This procedure is repeated for all K-1 datasets, and is called K-fold cross validation [4].

**Real-time Detection:** The offline defined thresholds in the previous steps are used to trigger the spoofing detector. The receiver generates the features for each sample point and the MVN score of these features based on the distribution of genuine satellite signals. If this score falls below the threshold, the sample will be dropped; otherwise it will be passed over to the next GPS receiver blocks.

Fig. 1d gives an overview of defense for Spotr, and Algorithm 2 explains the details of the steps mentioned above.

## 3.6 Time Complexity Analysis

We use a very simple feature extraction algorithm with all of the operations performed in the time domain which eliminates computational complexity. The feature extraction given at Alg. 1 imposes time complexity of $O(1)$. Training an MVN model using the training sample points requires covariance matrix calculations and matrix inversion operations with time complexity of $O\left(n^3\right)$. However all this can be done offline. This cuts down the complexity of our algorithm to fitting a single sample point into an already trained MVN model after feature extraction to $O(1)$.

# 4 EXPERIMENTAL SETUP

In this section we briefly discuss the organization of GPS navigational messages, followed by the details of our experimental setup and method of data collection. Our datasets, summarized in Tables 3 and 4, encompass variations over many months and environmental conditions. The tables contain specifications of the genuine and spoofed datasets, respectively.

## 4.1 GPS Data Segmentation

A GPS transmission is broken into 5 sub-frames that update at varying intervals with an average duration of 6 s each. The first sub-frame contains information on the health and accuracy of the satellite, GPS timing information, and any clock corrections. The second and third sub-frames contain ephemeris data (orbital measurements) for the transmitting satellite. The fourth sub-frame contains abbreviated almanac data for satellites 25 through 32, as well as ionospheric and UTC data, satellite configuration, and any special messages. The fifth and final sub-frame contains abbreviated almanac data for satellites 1 through 24 and the time and week number of the almanac itself. Note that sub-frames four and five require approximately 12.5 mins to complete, which means sub-frames one, two, and three will update many times during this period. For the attacker to spoof a successful PVT solution, continuous spoofing of between 18 s and 30 s is required. See the GPS governing agency [36] or [9, 16, 21] for more details.

## 4.2 Experimental Hardware

We have live data and replay data collections. Our receivers are the B210, the B205mini, and the X300 software-defined radios (SDRs) from Ettus Research. For live signal collection we use the X300, which includes a GPS-Disciplined Oscillator (GPSDO) and a UX-160 daughterboard. Two inexpensive ($\tilde{\$}$10) GPS antennae are used, one connected to the RX2 port of the daughterboard and one connected directly to the GPSDO. For our replay (spoofing) sessions we use the B210 (USB connection) as the transmitter and the X300 (gigabit Ethernet connection) as the receiver. The two radios are connected (See Fig.4) via SMA-terminated coaxial cabling that begin at the TX/RX port of the B210, pass through a varying attenuator, and end at the RX2 port of the X300. Wired connections like this offer an ideal transmission for the attacker because the channel propagation effects do not influence the signaling of the attacker's radio.

**Algorithm 2:** Overview of Proposed Approach: Training, Testing, and Real-time Detection/Tracking

```
Training:
for dataset i do
    $F_i$ ← feature extraction Alg. 1
    $\mathfrak{F} \leftarrow \mathfrak{F} \cup F_i$ trained
$\mathrm{MVN}(\mu, \Sigma)$ ← fit MVN on $x\%$ of $\mathfrak{F}$
$\mathrm{sc}_{\mathrm{MVN}}$ ← the density values for the remaining $\mathfrak{F}$ on $\mathrm{MVN}(\mu, \Sigma)$
threshold ← EER analysis on $\mathrm{sc}_{\mathrm{MVN}}$
Testing:
$\mathrm{MVN}(\mu, \Sigma)$ ← Training
for sample point i do
    $F_i$ ← feature extraction Alg. 1
    $\mathrm{sc}^{\mathrm{i}}_{\mathrm{MVN}}$ ← density values of $F_i$ on $\mathrm{MVN}(\mu, \Sigma)$
K-fold Cross Validation:
$F_i$ ← feature extraction Alg. 1
for each dataset i do
    for The remaining datasets j(j!=i) do
        threshold ← EER analysis at Training
        $\mathrm{MVN}(\mu, \Sigma)$ ← EER analysis at Training
$\mathrm{sc}^{\mathrm{i}}_{\mathrm{MVN}}$ ← probability of $F_i$ matching to $\mathrm{MVN}(\mu, \Sigma)$
FPR, FNR ← apply threshold to $\mathrm{sc}^{\mathrm{i}}_{\mathrm{MVN}}$
Real-time Detection: decision making procedure:
$\mathscr{F}$ ← extract average features for $n$ sample points, Alg. 1
$\mathrm{MVN}(\mu, \Sigma)$ ← Training
threshold ← Training
$\mathrm{sc}_{\mathrm{MVN}}$ ← density values of $\mathscr{F}$ if generated from $\mathrm{MVN}(\mu, \Sigma)$
if $\mathrm{sc}_{\mathrm{MVN}} >$ threshold then
    Authentic sample point
else
    Malicious sample point
```

## 4.3 Experimental Software

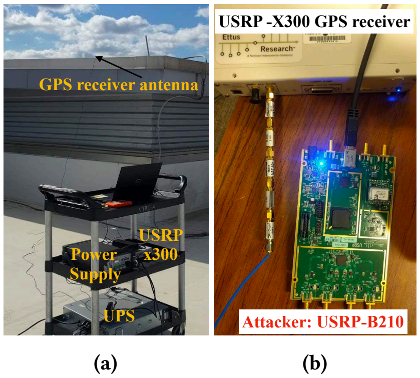
Our data collection software is GNSS-SDR [12], an open source application for GNSS research using SDRs. Signal parameters are set via a configuration file which allows us to specify parameters such as carrier frequency, sampling rate, and data type. Most importantly, we can arrange for each of the eight available channels to lock only with a specified PRN, ensuring that we can single out each satellite for individual tracking. PRNs are selected based on optimum viewing angle (e.g. as close to directly overhead as possible) and their positions are correlated between an online tracking website (https://in-the-sky.org) and an Android app (GPS Test) [2].

GPS-SDR-SIM [29], another open source application, is our primary means of both generating rudimentary sets of spoofed data and for transmitting said data over the wired connection. This application allows one to present a GPS ephemeris file (via a repository maintained by NASA), specify time, date, location, and duration variables, and then generate authentic GPS binary data ready for transmission. One issue we encountered is that while the software does generate viable GPS NAV messages and we can successfully spoof a location with them, the software is not meant to address PRN authenticity. As such, we modify this application to allow for greater specificity in the data generated resulting in a spoofed signal accurate in time, location, and PRNs received. Our modifications will be shared for use by the security community.

A summary of the genuine and spoofed GPS data (SatGrid) that we collected using our setup is given in Tables 3 and 4, respectively. The collection date for the spoofed data corresponds to the date that the genuine data was acquired, not when it was actually replayed. We discuss this in more detail in Sec. 5.

## 4.4 Second-Party Data: UT-Austin TexBat Repository

Several works in the field of GPS security are tested against the de-facto standard of a publicly available repository of GPS signal spoofing traces called Texas Spoofing Battery (TexBat) [19]. The dataset is provided by the Radionavigation Lab at Univ. of Texas-Austin, including 10 rounds of data collection with a duration of 400 s each with the first 100 s being spoof-free, sampled at 25 Msps with a 16 bit resolution for complex values. TexBat includes two



(a) (b)

**Figure 4: (a) Genuine data collection setup (b) Spoofing setup. An Ettus X300 USRP served as the GPS device fingerprinter, and a B210 USRP for generated the spoofing data.**

rounds of separate spoof-free data, which are categorized based on mobility of the platform as *clean static* and *clean dynamic*. Clean static data is collected from a reference antenna located in a building, while the clean dynamic data is recorded from an antenna mounted atop a vehicle traveling in Austin, TX. The rest of the datasets are spoofed data with different attacks in time and/or position [19] which have as low as meter level alignment with the genuine signals. TexBat induces a 600 m position offset, and 2 µs error in time.

In this work we test Spotr on TexBat data in addition to our dataset (SatGrid) for two reasons: First, because it provides a fair comparison platform for security researchers and helps them go beyond statistical analysis methods for testing their solutions. The detection methods which were limited to verify the null hypothesis (no attack) before, were then able to test alternate hypothesis thanks to the spoofing data of TexBat. Second, the UT spoofer that generates the TexBat data (Table 4), is a strong attacker which is diverse in terms of dimensions of the genuine GPS signals that it targets to attack.

## 5 EXPERIMENTAL RESULTS

In this section we evaluate Spotr on different available attack scenarios operating at different conditions of time, location, multipath richness, and/or hardware. For reference, Tables 3 and 4 summarize all genuine and spoofed data respectively, both our own live recordings and the Univ. of Texas-Austin "TexBat" [19] repository.

We follow two main objectives in fingerprinting the GPS signals using the features that we introduced in Sec. 3.3: First, to identify a genuine satellite signal from a spoofed one, referred to as *detection*; Second, to prove the consistency of these features by tracking the signals from a genuine satellite on different conditions, called *tracking*. We evaluate the detection and tracking capability of Spotr on different types of attacks on GPS signals listed in Table 4. First we evaluate a spoofing attacker over different environmental conditions in Sec. 5.1. Next, Spotr is evaluated against the stronger replay attacker with more insights on the success rate of the attacker in inducing falsified PVT solutions in Sec. 5.2. We cannot compare the error rates with state-of-the-art, because they report results using higher level metrics, such as the maximum location offset [6, 34]. See Sec. 1.1 for details on SPREE which limits the range of spoofing attacks on position to the radius of 1000 m, whereas we report maximum continuous spoofing time of 47.3 s in Sec.5.2. Also, our low error rates are indicative of the fact that circuitry of a satellite is different from an SDR (D1) or non-SDR (D2,D3) spoofer.

For the scenarios in Table 1 the Train and Test columns include the general dataset information, followed by the genuine datasets from device $m$ (Dm) labeled by Dm:Gx and spoofed datasets collected from Dm labeled by Dm:Sx, with x being the dataset index in the Tables 3 and 4. For the cases that only one dataset is reported (e.g. TexBat:S5) the first 100 s is spoof-free.

Note that the low values on the Y-axis of all the figures are due to the high dimensionality of the data (six). The MVN density values close to the maximum achievable MVN score are interpreted as high scores. For example, the maximum achievable MVN score of Fig. 5b is 6e-27 which is calculated by using a given Σ of the training data, $k = 6$ and $x = \mu$ with Eq. 4.

### 5.1 Evaluation and Discussion

We evaluate Spotr against a spoofing attacker over different times, locations, multipath conditions, and the fingerprinter device being used in the data collection setup, in the following.

*5.1.1 Detection and Tracking over Time.* After generating features based on Sec. 3.3, we randomly select 40,000 samples from SatGrid:G1 and SatGrid:S1 (Spoofing-Attacker) on Sep 24, 2018 in Blacksburg and fit an MVN model using Eq. 4 to train a $\mu$ and $\Sigma$. Next, we use the remaining data on the same day to do EER analysis on the MVN scores, and train an identification/detection threshold. Note that, as many sample points as needed (shown by $n$) to achieve an EER of zero (the ideal value for EER) are used in all of the cases, which is reported in the X-axis label of the graphs if required.

Next, we use the trained MVN model and the threshold to test the genuine and spoofed data collected on different days using the same spoofer at Blacksburg in consecutive days right after the training, also after one year on Sep 10, 2019 at Arlington (SatGrid:G23). Fig. 5b shows the MVN scores associated with this analysis. We observe that without any averaging on the sample points, all of the genuine data have MVN scores above the trained threshold while the MVN scores of spoofed data are mostly zero (which is the reason they are not printed in the graph with logarithmic scale on Y-axis). This allows us to conclude that we are able to track a genuine signal over the period of one year which validates the stability of our features with time. Scenario 2 of the Table 1 illustrates the error rates (FNR and FPR) as well as the required number of averaged sample points (n) to achieve the reported error rates for all the data of Blacksburg and Arlington collected on Sep 2018 and 2019. Table 2a shows the results of a more general training-testing procedure on SatGrid data collected on Sep 2018 using K-fold cross validation.

Similar analysis is performed at Scenario 7 of Table 1 on a selection of TexBat data to validate the stability of our features over time. Unlike Scenario 2, both the data used for training and testing in this case are collected in rich-multipath environment. Fig. 5f shows the MVN scores of the test data, TexBat:S6, where the MVN model and the threshold are trained on TexBat:S5. Presence of strong multipath is the reason for the need to average multiple sample points (35000 in this case), before making a decision on their authenticity.

*5.1.2 Detection and Tracking at Different Locations.* Following the same training process in Sec. 5.1.1, we test the detection and tracking of Spotr on the data collected from different locations using a spoofing attacker given as Scenario 1 in Table 1. Fig. 5a shows MVN scores of the data collected in Missouri, while the original MVN model and threshold are trained based off of Blacksburg. Using a detection threshold of 1.77e-92, for all of the satellites that appeared on different days of data collection listed in Table 3, leads to EERs of zero, which allows us to conclude that we can distinguish the genuine signals from spoofed ones regardless of the location.

*5.1.3 Detection and Tracking in the Presence of Multipath.* The most common challenge in spoofing detection of GPS systems is the multipath effect which makes it difficult to distinguish a genuine multipath component of a GPS signal from a malicious one. This problem is elaborated by Wesson et. al [39], and stated as a limitation at [26, 33], however a solution has not been proposed yet. To investigate the performance of Spotr in the presence of multipath we fit an MVN model to the multipath-free training data, and test it on rich-multipath data and vice versa for both SatGrid and TexBat (Scenarios 3, 7, 8 and 9 of Table 1). This round of analysis can help us understand the extent to which genuine data with multipath effect can be confused with spoofing data.

The initial set of TexBat data is collected on 2012 using their first fingerprinting hardware (D1) on a static platform with line of sight among which TexBat:S1, S2, S3, and D4 are the multipath-free ones. Another set from the TexBat repository is collected in a rich-multipath environment with the same fingerprinter mounted on a vehicle and driven through a populated area of Texas for 3 miles. TexBat:S5, and S6 are collected from this "dynamic" platform. Fig. 5h illustrates the MVN scores of genuine and spoofed data where the model is trained on this rich-multipath data and tested on the earlier mentioned multipath-free datasets from TexBat D1. Scenario 9 of Table 1 reports the error rates of this analysis with a FNR of 5.23% which means there is spoofing activity which remains undetected for this case after using 1000 sample points for a final decision making on their authenticity. As mentioned earlier in Sec. 3.1 the VOLK-Library [12] from gnss-sdr that is responsible for running the dll-pll loops at the GPS receiver, runs at a varying speed in a feedback system. This is why we cannot translate the number of undetected malicious sample points to the notion of time and report the maximum continuous time that a spoofing activity will remain undetected by Spotr without access to accurate timestamps. We elaborate on this timing problem in Sec. 5.2 by analyzing the worst case scenario (strongest attacker: matched-powered replay attacker on SatGrid) with timestamps.

To further evaluate the robustness of Spotr for all possible cases, we perform training and testing on the rich-multipath data from TexBat in Scenario 7 of Table 1, where using multiple sample points helps us to overcome the influence of multipath on the detection/tracking process shown in Fig. 5f. We also evaluate Spotr on SatGrid at Scenarios 3 and 8, where training is done on a multipath-free (Sep 10, 2019) and rich-multipath data (Aug 23, 2019), respectively. We are able to perform highly accurate detection/tracking with only one sample point as demonstrated in Fig. 5c and Fig. 5g.

*5.1.4 Detection and Tracking against Different Attacks.* Table 2b shows the results of a more general training-testing procedure on TexBat data collected using D1 2012 using K-fold cross validation where the spoofing types vary from single time or position, to simultaneous time and position detailed in Table 4. This analysis is performed on a mix of multipath-free data (Fig. 5d and Fig. 5e) and data collected in a rich multipath (Fig. 5h) at the same time. The error rates in the table indicates high FPR and FNR values on some occasions, caused by using the multipath affected data at the training phase. This shows that genuine data with multipath effect can be confused with spoofing data if this propagation effect is not taken into consideration at the training stage. To compensate for the reduced accuracy in the detection/tracking process, we increase the number of observation sample points and report new rates in Table 2c.

*5.1.5 Detection and Tracking using Different Hardware Platforms as a Fingerprinter.* The main idea behind our proposed spoofing detection is hardware fingerprinting of the transmitters. This allows us to hypothesize that the features that we exploit in our algorithm should change with the signal acquisition platform. In this section, we validate this by training a model on the data from SatGrid and testing it on the data from TexBat. Fig. 5i shows that we are not able to track the genuine data from TexBat fingerprinting hardware when training a model with SatGrid (the MVN density values of the genuine data from TexBat all hold a value of zero, which is why they are not printed in logarithmic scale on the Y-axis). This is a general limitation of fingerprinting approaches (even if the platforms are identical [13]) and can be overcome by training each fingerprinter on genuine satellite data before deployment.

## 5.2 Security Analysis: Feature Replay

In this section, we look into the capabilities of our strongest attacker, where we give the attacker the exact same samples as that of a genuine satellite. Table 3 includes two rounds of SatGrid GPS data collected on Nov 8, 2019, which unlike all other listed datasets includes high fidelity timestamps. Data from a matched-power replay attack associated with this data is also given in Table 4. This is the strongest attacker amongst the ones listed in Table 4 for several reasons: First, the attacker is capable of estimating and matching its power to the power level of the genuine GPS signal at the target receiver in a real-time manner. This represents the best case (unrealistic) scenario for the attacker as it hides the spoofing activities from not only in-band power based spoofing detectors but also precludes anomalies at the complex correlator output tabs of the receiver caused by the struggle between the genuine and the spoofed signals aiming to take control of the tracking block [18]. Second, because of our attack model and data collection setup, some inevitable amount of delay is inherent to the attacker, which we have removed. In the literature of device fingerprinting [8], this is the strongest attacker.

For any spoofing attack to take place successfully, it is compulsory for the attacker to spoof four GPS channels at the receiver successfully and simultaneously. This is because the PVT solution solves the linear problem for four unknowns of X, Y, and Z coordinates plus time, and to do so, relies on the information it collects from at least four channels at the tracking block. According to Table 3, SatGrid:G25 collected on Nov 8, 2019 at a rooftop of Arlington includes data from eight PRNs. Hence we look into all possible combinations of these PRNs that could generate a successful PVT fix in the receiver. These combinations are called PRN-sets here, which include 70 cases for eight satellites.

Fig. 6b shows the average of the "maximum continuous spoofing time" for all the PRN-sets when Spotr exploits multiple sample points in order to perform the detection/tracking process. The X-axis shows the number of these sample points, denoted by n. The average of "maximum continuous spoofing time" in this case reduces from 100 s to 47.3 s if 100,000 sample points are used by Spotr. The figure also reports the "overall continuous spoofing time" for all of the PRN-sets in the same graph, which is also reduced from 266.6 s to 101.2 s if n=100,000. This shows Spotr's ability to detect spoofing activities in 47.3 s in the presence of the strongest attacker.

The navigation message consists of 30 s frames which are 1,500 bits long. That is why the number of 30 s locks for which a spoofer is able to remain undetected by Spotr is a more accurate metric

**Table 1: This table demonstrates the efficacy of Spotr for detection/tracking of spoofed/genuine signals (listed in Tables 3 and 4) across locations, times, in the presence or absence of multipath (MP), and for different attacks using different hardware platforms (TexBat:D1 and D2 or SatGrid:D3). The table reports thresholds for obtaining equal error rates (EER) of zero at the training phase, as well as false positives (FPR) and false negatives (FNR) with the number of required sample point observations (n) for attaining the reported FPR and FNR at the testing phase.** ***key*****:** ***The Train and Test columns include the general dataset information, followed by the genuine datasets from device m (Dm) labeled by Dm:Gx and spoofed datasets collected from Dm labeled by Dm:Sx, with x being the dataset index in Tables 3 and 4. For the cases that only one dataset is reported (TexBat, D1 2012) the first 100 s of the data is spoof-free. For example, D3:G1,G2,G3,G4 shows the genuine data collected from SatGrid, indexed by G1,G2,G3 and G4 in Table 3.***

| Scenario | Train | Threshold | Test | FPR | FNR | n | Plot |
|---|---|---|---|---|---|---|---|
| 1. Location | SatGrid (Blacksburg,~MP)<br>D3:G1,G2,G3,G4<br>D3:S1,S2,S3,S4 | 1.177e-192 | SatGrid (Missouri,~MP)<br>D3:G5,G8<br>D3:S5,S6 | 0% | 0% | 1 | Fig. 5a |
| 2. Date | SatGrid (Blacksburg,~MP)<br>D3:G1<br>D3:S1 | 5.15e-186 | SatGrid (Blacksburg,~MP)<br>D3:G2,G3,G4<br>D3:S2,S3,S4 | 0% | 0% | 1 | Fig. 5b |
| 3. MP | SatGrid (Arlington,~MP)<br>D3:G23<br>D3:S8 | 1.86e-182 | SatGrid (Arlington,MP)<br>D3:G22<br>D3:S7 | 0% | 0% | 1 | Fig. 5c |
| 4. Attack (D1) | TexBat (Texas,~MP)<br>D1:S1,S2,S3,S4 | 4.25e-36 | TexBat (Texas,~MP)<br>D1:S1,S2,S3,S4 | 0% | 0.96% | 500 | Fig. 5d |
| 5. Attack (D2) | TexBat (Texas,~MP)<br>D2:Clean1<br>D2:S7,S8 | 2.56e-38 | TexBat (Texas,~MP)<br>D2:Clean1<br>D2:S8 | 0% | 0% | 60000 | Fig. 5e |
| 6. Hardware | SatGrid (Missouri)<br>D3:G5,G8<br>D3:S5,S6 | 2.6e-33 | TexBat (Texas,~MP)<br>D1:S1,S2,S3,S4 | 100% | 0% | 1 | Fig. 5i |
| 7. Date | TexBat (Texas, MP)<br>D1:S5 | 7.7e-38 | TexBat (Texas, MP)<br>D1:S6 | 0% | 0% | 35000 | Fig. 5f |
| 8. ~MP | SatGrid (Arlington,MP)<br>D3:G22<br>D3:S7 | 1e-80 | SatGrid (Arlington,~MP)<br>D3:G23<br>D3:S8 | 0% | 0% | 1 | Fig. 5g |
| 9. Attack (D1) | TexBat (Texas, MP)<br>D1:S5,S6 | 1.6e-37 | TexBat (Texas,~MP)<br>D1:S1,S2,S3,S4 | 0% | 5.23% | 1000 | Fig. 5h |

**Table 2: (a) Evaluation of the detection process for the genuine (G1-G4) and spoofed (S1-S4) data generated by SatGrid (D3) using FPR and FNR values. (b) 6-fold cross validation on TexBat spoofing scenarios (S1-S6) including both multipath/non-multipath data for the 2012 fingerprinter (D1) based on one sample point of observation. (c) 6-fold cross validation on TexBat spoofing scenarios (S2-S6) based on 40,000 sample points of observations.**

**(a)**

| Validation Dataset | FPR | FNR | EER threshold |
|---|---|---|---|
| SatGrid : S1&G1 | 0 % | 0 % | $0.6829 \times 10^{-31}$ |
| SatGrid : S2&G2 | 0 % | 0 % | $0.5126 \times 10^{-31}$ |
| SatGrid : S3&G3 | 0 % | 0 % | $0.0138 \times 10^{-31}$ |
| SatGrid : S4&G4 | 0 % | 0 % | $0.8071 \times 10^{-31}$ |

**(b)**

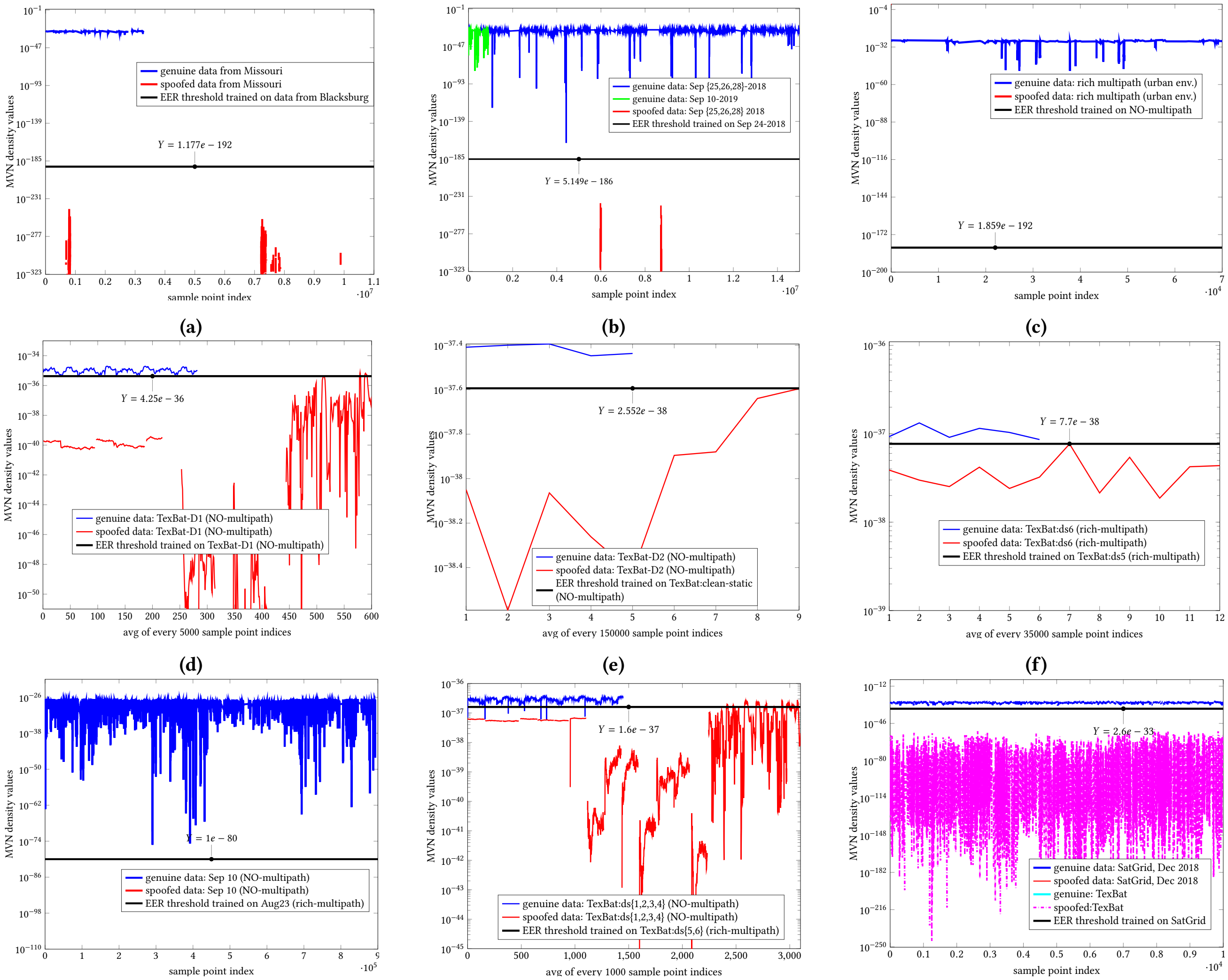
| Validation Dataset | FPR | FNR | EER Threshold |
|---|---|---|---|
| TexBat : S1 | 0% | 0% | $3.6370 \times 10^{-36}$ |
| TexBat : S2 | 0% | 0% | $2.4183 \times 10^{-36}$ |
| TexBat : S3 | 0.38% | 3.35% | $9.4367 \times 10^{-36}$ |
| TexBat : S4 | 0.18% | 46.8% | $9.8347 \times 10^{-37}$ |
| TexBat : S5 | 26.29% | 0% | $2.9695 \times 10^{-35}$ |
| TexBat : S6 | 24.48% | 0% | $2.5882 \times 10^{-35}$ |

**(c)**

| Validation Dataset | FPR | FNR | EER Threshold |
|---|---|---|---|
| TexBat : S1 | 0% | 0% | $3.544 \times 10^{-36}$ |
| TexBat : S2 | 0% | 0% | $2.4183 \times 10^{-36}$ |
| TexBat : S3 | 0% | 0% | $2.6983 \times 10^{-35}$ |
| TexBat : S4 | 0% | 12.5% | $1.8412 \times 10^{-35}$ |
| TexBat : S5 | 0% | 0% | $5.519\,76 \times 10^{-35}$ |
| TexBat : S6 | 0% | 0% | $3 \times 10^{-35}$ |

for performance evaluation. Fig. 6c shows the number of 30 s locks that the spoofer is able to generate for each PRN-set. Spotr is able to reduce the number of undetected locks from 360 occurrences when n=1, down to 131 when n=100,000. The number of locks for the genuine data is given as a bench mark here (plotted as blue bars) where there are 361 locks for all the PRN-sets. This gives a high level understanding of the performance of Spotr and does not translate directly to the continuous spoofing capabilities of the attacker. The attacker would be detected within the interval of 30 s locks shown in Fig. 6c.

So far, we can conclude that the number of sample points to be averaged by Spotr, n, is a critical parameter that directly influences the detection/tracking performance if the attacker is capable of adjusting its power levels. Fig. 6a shows how this number changes with the SatGrid replay attacker's power level, when the EER holds the ideal value of zero for all the power levels. It increases as the under-powered attacker increases its strength to the matched-powered level with the genuine data, and reduces again as the power of the attacker becomes far more than that of the genuine signals.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

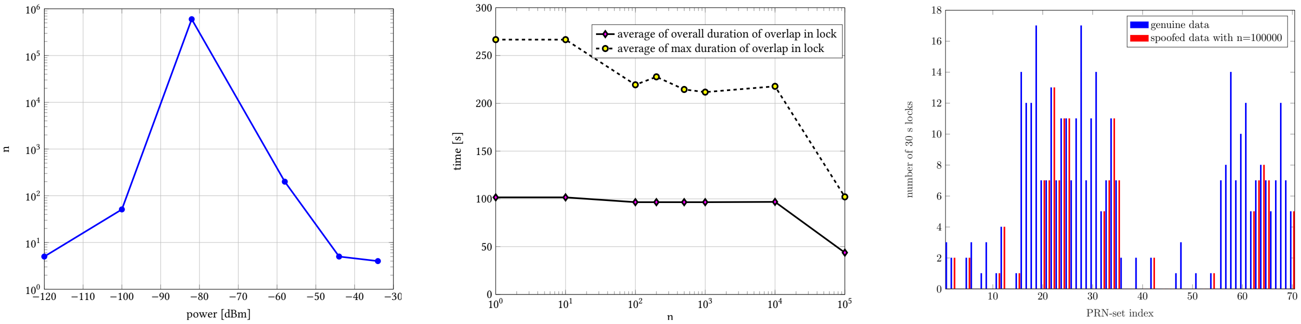
**Figure 5: The results of using Spotr for detection/tracking of spoofed/genuine signals (a) across locations (b)-(f) cross time (c)-(g) in the presence or absence of multipath (d)-(e)-(h) for spoofing attacks of TexBat using different hardware platforms (i). In Figures (a)-(b)-(c) and (g), most of the MVN scores for spoofed signals (depicted by red line) are exact zero values, hence not printed in the graph with logarithmic scale on Y-axis. Logarithmic scale is used for better visualization.**

## 6 CONCLUSION

In this work, we introduced a physical-layer identification based spoofing detector for GPS satellites. Even though the spoofed signals are as closely phase aligned as possible with their authentic counterparts, they are never exactly the same as the genuine GPS signals. In fact, a perfect carrier-phase alignment is impossible for a spoofer. We take advantage of anomalies that are introduced into the complex correlator outputs of a standard GPS receiver during a spoofing attack, without depending on the digital information or the GPS receiver's solution for position, velocity and time. Hence, our algorithm detects spoofing attacks launched on civil GPS signals, and prevents the receiver from locking to the spoofed GPS signals in as few as 47.3 s in the worst case scenario. We validated our method by testing it on the de-facto standard of a publicly available repository of GPS signal spoofing traces called the Texas Spoofing Battery (TexBat), as well as our data (SatGrid) collected over several months at multiple locations in United States. More specifically, we are able to detect spoofing activities and track genuine signals over different times and locations and propagation effects related to environmental conditions.

## ACKNOWLEDGMENTS

This work was supported in part through funding from NSF IUCRC S2ERC (Award Number 1650540) and CACI International Inc. The authors wish to thank Dr. Ramin Baseri, Mr. John Lee, and Mr. Andrew Lee of CACI and Drs. Jeffrey Reed and Robert W. McGwier of Virginia Tech for their guidance on, and support of, the work.



**(a)** **(b)** **(c)**

**Figure 6: (a) The number of required sample point observations (n) for Spotr to get EER of zero at different power levels of data collected on Sep 10, 2019 data at Arlington. (b)-(c) This analysis is performed on the matched-powered replay attack data, generated based off the genuine SatGrid:G25 Nov 8, 2019 data collected from the rooftop of Arlington.**

# REFERENCES

[1] Dennis M Akos. 2012. Who's afraid of the spoofer? GPS/GNSS spoofing detection via automatic gain control (AGC). *Navigation: Journal of the Institute of Navigation* 59, 4 (2012), 281–290.

[2] Android App. 2020. *GPS test.* https://play.google.com/apps/testing/com.android.gpstest

[3] Frederic Bastide, Dennis Akos, Christophe Macabiau, and Benoit Roturier. 2003. Automatic gain control (AGC) as an interference assessment tool.

[4] Christopher M Bishop. 2006. *Pattern recognition and machine learning.* springer.

[5] Ruud M Bolle, Jonathan H Connell, Sharath Pankanti, Nalini K Ratha, and Andrew W Senior. 2013. *Guide to biometrics.* Springer Science & Business Media.

[6] Ali Broumandan, Ali Jafarnia-Jahromi, and Gérard Lachapelle. 2015. Spoofing detection, classification and cancelation (SDCC) receiver architecture for a moving GNSS receiver. *Gps Solutions* 19, 3 (2015), 475–487.

[7] Leon W Couch, Huaizong Shao, Xiaofeng Li, and Lianfu Liu. 1997. *Digital and analog communication systems.* Vol. 6. Citeseer.

[8] Boris Danev, Heinrich Luecken, Srdjan Capkun, and Karim El Defrawy. 2010. Attacks on physical-layer identification. In *Proceedings of the third ACM conference on Wireless network security.* 89–98.

[9] Fabio Dovis (Ed.). 2015. *GNSS Interference Threats and Countermeasures* (1st ed.). Artech House, Norwood. 213 pages.

[10] M El-Diasty, A El-Rabbany, and S Pagiatakis. 2007. Temperature variation effects on stochastic characteristics for low-cost MEMS-based inertial sensor error. *Measurement Science and Technology* 18, 11 (2007), 3321.

[11] Per Enge, Todd Walter, Sam Pullen, Changdon Kee, Yi-Chung Chao, and Yeou-Jyh Tsai. 1996. Wide area augmentation of the global positioning system. *Proc. IEEE* 84, 8 (1996), 1063–1088.

[12] Carles Fernández-Prades. 2019. *GNSS-SDR.* https://www.gnss-sdr.org

[13] Mahsa Foruhandeh, Yanmao Man, Ryan Gerdes, Ming Li, and Thidapat Chantem. 2019. SIMPLE: single-frame based physical layer identification for intrusion detection and prevention on in-vehicle networks. In *Proceedings of the 35th Annual Computer Security Applications Conference.* 229–244.

[14] Ryan M Gerdes, T. E Daniels, M. Mina, and S. Russell. 2006. Device Identification via Analog Signal Fingerprinting: A Matched Filter Approach.. In *NDSS.*

[15] Ryan M Gerdes, Mani Mina, Steve F Russell, and Thomas E Daniels. 2012. Physical-layer identification of wired Ethernet devices. *IEEE Transactions on Information Forensics and Security* 7, 4 (2012), 1339–1353.

[16] Paul D. Groves. 2013. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems* (2nd ed.). Artech House, Boston. 776 pages.

[17] Todd Humphreys. 2012. Statement on the vulnerability of civil unmanned aerial vehicles and other systems to civil GPS spoofing. *University of Texas at Austin (July 18, 2012)* (2012), 1–16.

[18] TE Humphreys. 2016. *TEXBAT data sets 7 and 8.* Technical Report. Technical Report. Available online: http://radionavlab. ae. utexas. edu.

[19] Todd E Humphreys, Jahshan A Bhatti, Daniel Shepard, and Kyle Wesson. 2012. The Texas spoofing test battery: Toward a standard for evaluating GPS signal authentication techniques. In *Radionavigation Laboratory Conference Proceedings.*

[20] Kai Jansen, Nils Ole Tippenhauer, and Christina Pöpper. 2016. Multi-receiver GPS spoofing detection: error models and realization. In *Proceedings of the 32nd Annual Conference on Computer Security Applications.* ACM, 237–250.

[21] Elliott D. Kaplan and Christopher J. Hegarty (Eds.). 2017. *Understanding GPS/GNSS Principles and Applications* (3rd ed.). Artech House, Boston.

[22] Keysight Technologies. 2014. Keysight Technologies GNSS Technologies and Receiver Testing (application note).

[23] Ulrich Kröner and Franc Dimc. 2010. Hardening of civilian GNSS trackers. In *Proceedings of the 3rd GNSS Vulnerabilities and Solutions Conference.*

[24] Pratap Misra and Per Enge. 2006. Global Positioning System: signals, measurements and performance. *Massachusetts: Ganga-Jamuna Press* (2006).

[25] Oliver Montenbruck, Peter Steigenberger, Lars Prange, Zhiguo Deng, Qile Zhao, Felix Perosanz, Ignacio Romero, Carey Noll, Andrea Stürze, Georg Weber, et al. 2017. The Multi-GNSS Experiment (MGEX) of the International GNSS Service (IGS)–achievements, prospects and challenges. *Advances in Space Research* 59, 7 (2017), 1671–1697.

[26] Paul Y Montgomery. 2011. Receiver-autonomous spoofing detection: Experimental results of a multi-antenna receiver defense against a portable civil GPS spoofer. In *Radionavigation Laboratory Conference Proceedings.*

[27] Paul Y Montgomery, Todd E Humphreys, and Brent M Ledvina. 2009. A multi-antenna defense: Receiver-autonomous GPS spoofing detection. *Inside GNSS* 4, 2 (2009), 40–46.

[28] Daniel Moser, Patrick Leu, Vincent Lenders, Aanjhan Ranganathan, Fabio Ricciato, and Srdjan Capkun. 2016. Investigation of multi-device location spoofing attacks on air traffic control and possible countermeasures. In *Proceedings of the 22nd Annual Int. Conf. on Mobile Computing and Networking.* ACM, 375–386.

[29] OSQZSS. 2018. GPS-SDR-SIM. https://github.com/osqzss/gps-sdr-sim/

[30] Oscar Pozzobon. 2011. Keeping the spoofs out: Signal authentication services for future GNSS. *Inside GNSS* 6, 3 (2011), 48–55.

[31] Mark L Psiaki and Todd E Humphreys. 2016. GNSS spoofing and detection. *Proc. IEEE* 104, 6 (2016), 1258–1270.

[32] Mark L Psiaki, Brady W O'Hanlon, Jahshan A Bhatti, Daniel P Shepard, and Todd E Humphreys. 2013. GPS spoofing detection via dual-receiver correlation of military signals. *IEEE Trans. Aerospace Electron. Systems* 49, 4 (2013), 2250–2267.

[33] Aanjhan Ranganathan, Hildur Ólafsdóttir, and Srdjan Capkun. 2016. SPREE: a spoofing resistant GPS receiver. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking.* ACM, 348–360.

[34] Aanjhan Ranganathan, Hildur Ólafsdóttir, and Srdjan Capkun. 2016. SPREE: A Spoofing Resistant GPS Receiver. *Proceeding MobiCom '16 Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking* (2016). https://doi.org/10.1145/2973750.2973753

[35] Nils Ole Tippenhauer, Christina Pöpper, Kasper Bonne Rasmussen, and Srdjan Capkun. 2011. On the requirements for successful GPS spoofing attacks. In *Proceedings of the 18th ACM conference on Computer and communications security.* ACM, 75–86.

[36] U.S.A.F. 2019. GPS: The Global Positioning System. https://www.gps.gov

[37] Pai Wang, Yongqing Wang, Ediz Cetin, Andrew Graham Dempster, and Siliang Wu. 2019. GNSS Jamming Mitigation Using Adaptive-Partitioned Subspace Projection Technique. *IEEE Trans. Aerospace Electron. Systems* 55, 1 (2019).

[38] Kyle Wesson, Mark Rothlisberger, and Todd Humphreys. 2012. Practical cryptographic civil GPS signal authentication. *Navigation: Journal of The Institute of Navigation* 59, 3 (2012), 177–193.

[39] Kyle D Wesson, Daniel P Shepard, Jahshan A Bhatti, and Todd E Humphreys. 2011. An evaluation of the vestigial signal defense for civil GPS anti-spoofing. In *Radionavigation Laboratory Conference Proceedings.*

# A APPENDIX

Table 3 details the authentic satellite data collection, and Table 4 details the spoofed data collection for Sec. 4.

**Table 3: Genuine datasets. SatGrid is the data that we collected at Blacksburg and Arlington, and TexBat is the data provided by Radionavigation Lab in UT Austin detailed in Sec. 4.4 collected by their 2015 hardware platform (D2).**

| Dataset | collection date | multipath | duration | start time | location | PRNs |
|---|---|---|---|---|---|---|
| **TexBat:clean** | July 2015 | No | 420 s | – | Texas | {3,6,7,13,16,19,23} |
| **TexBat:clean** | July 2015 | Yes | 420 s | – | Texas | {9,15,18,22} |
| **SatGrid:G1** | Sep 24, 2018 | No | 60 min | 8:11 am | Blacksburg | {1,7,8,9,11,18,27,28} |
| **SatGrid:G2** | Sep 25, 2018 | No | 60 min | – | Blacksburg | {1,7,8,9,11,18,27,28} |
| **SatGrid:G3** | Sep 26, 2018 | No | 60 min | – | Blacksburg | {1,7,8,9,11,18,27,28} |
| **SatGrid:G4** | Sep 28, 2018 | No | 60 min | 4:10 pm | Blacksburg | {1,7,8,9,11,18,27,28} |
| **SatGrid:G5** | Dec 15, 2018 | No | 60 min | 12 pm | Missouri | {10,14,20,21,32} |
| **SatGrid:G6** | Dec 15, 2018 | No | 60 min | 2 pm | Missouri | {14,22,25,31,32} |
| **SatGrid:G7** | Dec 15, 2018 | No | 60 min | 8 pm | Missouri | {7,8,9,27,30} |
| **SatGrid:G8** | Dec 16, 2018 | No | 60 min | 2 am | Missouri | {2,6,12,17,19} |
| **SatGrid:G9** | Dec 16, 2018 | No | 60 min | 3:30 pm | Missouri | {3,14,22,26,31} |
| **SatGrid:G10** | Dec 16, 2018 | No | 60 min | 7 pm | Missouri | {7,8,9,23,27} |
| **SatGrid:G11** | Dec 16, 2018 | No | 60 min | 9 pm | Missouri | {1,7,8,11,30} |
| **SatGrid:G12** | Dec 16, 2018 | No | 60 min | 11 pm | Missouri | {1,17,19,28,30} |
| **SatGrid:G13** | Dec 17, 2018 | No | 60 min | 6 am | Missouri | {7,11,17,28,30} |
| **SatGrid:G14** | Dec 17, 2018 | No | 60 min | 12 pm | Missouri | {10,14,20,31,32 } |
| **SatGrid:G15** | Dec 17, 2018 | No | 60 min | 10 pm | Missouri | {2,5,13,15,29} |
| **SatGrid:G16** | Dec 18, 2018 | No | 60 min | 4 am | Missouri | {2,5,6,12,25} |
| **SatGrid:G17** | Dec 18, 2018 | No | 60 min | 11 am | Missouri | {10,14,18,20,32} |
| **SatGrid:G18** | Dec 18, 2018 | No | 60 min | 2 pm | Missouri | {3,14,22,31,32} |
| **SatGrid:G19** | Dec 18, 2018 | No | 60 min | 11 pm | Missouri | {1,11,17,19,28} |
| **SatGrid:G20** | Dec 19, 2018 | No | 60 min | 9 am | Missouri | {10,15,20,21,24} |
| **SatGrid:G21** | Dec 20, 2018 | No | 60 min | 9 pm | Missouri | {1,11,13,18,28} |
| **SatGrid:G22** | Aug 23, 2019 | yes | 10 min | 4 pm | Arlington (urban) | {2,13,15,21,29} |
| **SatGrid:G23** | Sep 10, 2019 | No | 45 min | 3 pm | Arlington (football field) | {2,5,8,15,17,21,24,29} |
| **SatGrid:G24** [2] | Nov 8, 2019 | No | 45 min | 10 am | Arlington (rooftop) | {2,24,13,15,20,5,21,29} |
| **SatGrid:G25** | Nov 8, 2019 | No | 50 min | 11 am | Arlington (rooftop) | {10,24,13,15,20,5,21,29} |

**Table 4: Spoofed datasets. There are different hardware platforms (fingerprinters) that are used for generating them. The fingerprinter of TexBat on 2012 (D1) is different from TexBat 2015 (D2). SatGrid attacker also has a separate hardware (D3).**

| Dataset | collection date | spoofing type | threat model | spoofing power status | multipath | duration | location |
|---|---|---|---|---|---|---|---|
| **TexBat:S1** | Sep 2012 | both | Replay | under-powered | No | 420 s | Texas |
| **TexBat:S2** | Sep 2012 | time | Replay | over-powered | No | 420 s | Texas |
| **TexBat:S3** | Sep 2012 | time | Replay | matched-powered | No | 420 s | Texas |
| **TexBat:S4** | Sep 2012 | position | Replay | matched-powered | No | 420 s | Texas |
| **TexBat:S5** | Sep 2012 | time | Replay | over-powered | Yes | 420 s | Texas |
| **TexBat:S6** | Sep 2012 | position | Replay | matched-powered | Yes | 420 s | Texas |
| **TexBat:S7** | July 2015 | time | Replay | matched-powered | No | 420 s | Texas |
| **TexBat:S8** | July 2015 | time | Replay | matched-powered | No | 420 s | Texas |
| **SatGrid:S1** | Sep 24, 2018 | both | Spoofing | over-powered | No | 60 min | Blacksburg |
| **SatGrid:S2** | Sep 25, 2018 | both | Spoofing | over-powered | No | 60 min | Blacksburg |
| **SatGrid:S3** | Sep 26, 2018 | both | Spoofing | over-powered | No | 60 min | Blacksburg |
| **SatGrid:S4** | Sep 28, 2018 | both | Spoofing | over-powered | No | 60 min | Blacksburg |
| **SatGrid:S5** | Dec 15, 2018 | both | Spoofing | over-powered | No | 60 min | Missouri |
| **SatGrid:S6** | Dec 16, 2018 | both | Spoofing | over-powered | No | 60 min | Missouri |
| **SatGrid:S7** | Aug 23, 2019 | both | Replay | adjusted-power | No | 10 min | Arlington |
| **SatGrid:S8** | Sep 10, 2019 | both | Replay | adjusted-power | No | 45 min | Arlington |
| **SatGrid:S9** | Nov 8, 2019 | both | Replay | adjusted-power | No | 50 min | Arlington |
| **SatGrid:S10** | Nov 8, 2019 | both | Replay | adjusted-power | No | 50 min | Arlington |

[2] SatGrid:G24 and SatGrid:G25 have high fidelity timestamps.