A Low-Cost GPS Spoofing Detector Design for Internet of Things (IoT) Applications

Md Tanvir Arafin
ECE Department
University of Maryland
College Park, Maryland
marafin@umd.edu

Dhananjay Anand
National Institute of Standards and Technology
Gaithersburg, Maryland
dhananjay.anand@nist.gov

Gang Qu
ECE Department
University of Maryland
College Park, Maryland
gangqu@umd.edu

ABSTRACT

The civilian Global Positioning System (GPS) is widely used for precise positioning, timekeeping, and synchronization in embedded systems. As a result, emerging digital infrastructure such as the Internet of Things (IoT) are dependent on GPS to locate and synchronize *Things* in the network. From a security perspective, civilian GPS signals are vulnerable to malintent because they are not encrypted and can easily be spoofed. Several countermeasures have been proposed to detect GPS spoofing attacks, but most of them require extensive signal processing capabilities and additional electronic components to capture and analyze RF signals. These add-ons may not be available to IoT devices, and if present, they will affect the device's power budget significantly. Therefore, new techniques for spoofing detection and survival are required before integrating GPS receivers with IoT devices and other critical infrastructures where energy and computation power are limited. In this work, we propose a novel GPS spoofing detection scheme based on hardware oscillators. Our design depends on measuring the frequency drift and offset of a free-running crystal oscillator with respect to the GPS signals. In our secure GPS spoofing detector design the trust is intrinsic, *i.e.*, the receiver only trusts the on-board free running local oscillator. Intrinsic properties of these oscillators exhibit a strong correlation with the authentic GPS signals and any anomaly in this measurement will indicate potential attacks on the received GPS signals. This proposed design is cost-effective, secure, backward compatible with existing receivers, and does not require additional RF circuitry or network connection with other clocks for detecting attacks.

Keywords

GPS Spoofing, Hardware Oscillator, Anomaly Detection, Hardware-Oriented Security and Trust.

1. INTRODUCTION

The progress towards an Internet of Things (IoTs) is highly dependent on the secure and successful integration of a trusted and robust geospatial localization and clock synchronization mechanism for *Things* across a large distributed network. Currently, both these functions are predominantly provided by the Global Positioning System (GPS). As a result, in today's world, GPS receivers have become a ubiquitous component in embedded systems: from the smart-phones and smart devices to automated vehicles and Phasor Measurement Units (PMUs) in the electric grid. However, this pervasive nature of GPS receivers in current and next-generation smart *Things* necessitates a thorough study of their security vulnerabilities and corresponding countermeasures.

Recently, there have been several demonstrations of weaknesses and vulnerabilities of GPS signals and GPS receivers [2, 7, 8, 11, 15, 13]. It has been argued that the future of navigation is crucially dependent on defending spoofing attacks on global navigation satellite system (GNSS) signals such as GPS [13]. Moreover, the common trend in IoT application designs is to use the civilian GPS signals, these signals are broadcast without encryption and several practical demonstrations of spoofing mechanisms have been documented in the literature [2, 7, 8, 15]. Most of the analysis on GPS spoofing is directed towards the spoofing of position data, however, GPS system is also used for large area clock synchronization, and therefore, attacks on GPS signals can impact networked infrastructure where accurate time keeping is important. For example, Phasor Measurement Units (PMUs) used in the electrical power system synchronize themselves using GPS signals, and an attack on this synchronization can induce failures in the power system across a wide area [9].

Along with research on potential attacks, several authentication and anti-spoofing techniques for GNSS signals have been developed in recent years. These techniques can be broadly categorized as signal and data-level authentication. For signal authentication, received signal characteristics of the civilian signal may be verified against encrypted GPS transmissions, additionally, the plane-of-polarization and angle-of-arrival can be measured to validate the signal as well. Data-level techniques are based on authentication of the received data with reference to a-priori knowledge of position or authentication using corroborative localization and timing sources. These methods have been shown to be effective; however, they are not usually used in the commercial GPS receivers due to implementation cost[11]. Moreover, the accuracy, computational and power efficiency, security against complex attacks and real-time performance of these approaches are still areas of active research.

In this paper, we present a data-level GPS spoofing detection mechanism that relies on intrinsic hardware properties of a free-running crystal oscillator. Since the free-running oscillator is located on the device and not externally synchronized, it presents a minimal attack surface while exhibiting a strong correlation with authentic GPS signals. It is our proposition that '*anomalies*' in the correlation index may indicate potential attacks on the received GPS data. Our

ACM acknowledges that this contribution was authored or co-authored by an employee, or contractor of the national government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only. Permission to make digital or hard copies for personal or classroom use is granted. Copies must bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. To copy otherwise, distribute, republish, or post, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

GLSVLSI '17, May 10-12, 2017, Banff, AB, Canada

© 2024 ACM. ISBN 978-1-4503-4972-7/17/05…$15.00

DOI: http://dx.doi.org/10.1145/3060403.3060455

approach is simple, fast and can perform at near real-time. Additionally, the design is low cost and can act as an add-on to virtually any GPS receiver.

2. PRELIMINARIES

2.1 The GPS System

The GPS system consists of satellite transmitters and (usually) terrestrial receivers. Each transmitter satellite broadcasts at two frequencies: 1575.42 MHz (L1) and 1227.6 MHz (L2). The L1 carrier messages are available for civilian purposes. These messages are not encrypted but modulated with pseudo-random noise (PRN) codes to distinguish each satellite. The L2 carrier is modulated by encrypted codes and reserved for military purposes. Message from each GPS satellite contains information about the position of the satellite and the time of the on-board atomic clock[12].

To calculate true receiver-to-satellite distance, the receiver requires the range (r_{true}) of a satellite at a given time. This can be calculated by the multiplying the signal propagation time (from the transmitter to the receiver) with the speed of light (c). Then, for a receiver located in (x_r, y_r, z_r) and a transmitter at (x_t, y_t, z_t) position, the range is given as:

$$r_{true} = c\, t_{propagation} = \sqrt{(x_t - x_r)^2 + (y_t - y_r)^2 + (z_t - z_r)^2} \quad (1)$$

To solve equation 1 for (x_r, y_r, z_r), one requires ephemerides for three satellites. However, since the clock on the GPS transmitter (t_{GPS}) and the clock on the receiver (t_{local}) are not perfectly synchronized, there exists an offset t_r between these two time-scales. Therefore, the satellite-to-receiver distance that a receiver perceives is a pseudo-range (r_{pseudo}), where:

$$r_{pseudo} = r_{true} - ct_r \quad (2)$$

Therefore, a GPS receiver needs to solve for four unknowns (x_r, y_r, z_r, t_r) for precise location and perfect synchronization. At least four satellite data is required for solving this system of equations. Using this solution, a GPS receiver updates its position, and synchronizes its local clock frequently to t_{sync} for keeping perfect synchronization with the universal coordinated time (UTC) where:

$$t_{sync} = t_{local} + t_r \quad (3)$$

In practice, this method provides an accuracy in the order of 10 meters in position and nearly $0.1\mu s$ in time [3]. As a result, GPS signals can be used not only for positioning of the receivers but also for precise clock synchronization of receivers across the globe.

2.2 GPS SPOOFING

Attacks on the GPS signal are usually performed by an adversary by either jamming or spoofing one or both radio channels. For the jamming attack, an adversary transmits overwhelming radio interference over the L1 and/or L2 band. For spoofing attacks, the adversary mimics real GNSS transmissions to intentionally alter data received by a victim receiver. Since civilian GPS signals are not encrypted and the structure of the signal is well known, spoofing attacks are relatively straightforward to execute using a commercial signal generator and RF transmitter.

2.3 Existing Spoofing Detection Techniques

As discussed in Section 1, signal-level detection methods can take on several forms including (1) check/validate the received RF signal strength against a level threshold, (2) compare L1 and L2 carrier signals, (3) justify the directional characteristics and polarization of the received signal [11]. The strength of a received GPS signal is typically less than -150dBW and presence of substantially stronger signals for a single satellite or over the entire frequency band can be a sign of attack in progress. However, it is possible for a spoofing attacker to adjust power levels to evade detection limiting the usefulness of the threshold detection mechanism in practice[11]. Furthermore, most consumer grade receivers only support a single RF band, and therefore, comparing L1 and L2 carriers would require complete replacement of the internal RF-circuitry. Also, in a case of a replay attack, an attacker can delay both L1 and L2 signals to avoid detection. Finally, using directional characteristics of the receiver antenna to cross-validate received signals from each satellite requires specialized phase tracking hardware to detect directional variations. Similarly, signal polarization characteristics have been shown to be an effective authentication aid [7], however, specialized receiver front ends and signal processing are required to effectively implement the approach.

Data-level spoofing detection uses demodulated GPS data to detect spoofing. GPS data can be validated with known position data or otherwise obtained time to detect attacks[6]. For example, a stationary receiver can check its known position with the position-solution of the receiver. Since trilateration position error can be ± 10 meters, the position solution is a weak measure of credibility. It has been recently demonstrated by Jiang *et al.* [9] that attackers can use this uncertainty to induce a phase angle error of 52 degrees in a PMU receiver by using simple optimization based evasion algorithms. Monitoring jumps in the time reported by the GPS signals is another possible GPS spoofing countermeasure. One can deploy accurate clocks to measure time deviation between the reported GPS clock and the on-board clock. However, precise clocks (such as atomic clocks *etc.*) are expensive and not used in practice for commercial purposes. For IoT components, one can also compare the GPS time with networked time protocols such as NTP, however, the approach can suffer when the network is down. Moreover, the resolution of attack detection is limited by the accuracy of the networked clocks.

While it is likely that signal threshold based authentication will be integrated into commercial GPS receivers used for critical applications, the solutions discussed in this section are currently cost prohibitive for consumer grade GPS receivers in the nascent IoT infrastructure. Hence, we propose a low-cost, computation and power-budget friendly data-level GPS spoofing detection mechanism in Section 3.

3. PROPOSED APPROACH

The key idea of this work is to cross-validate GPS time signals with intrinsic properties of a hardware oscillator to detect spoofing attacks. The spoofing detector will calculate frequency states of this hardware clock using the received GPS signal as a reference. Any attacks on the received GPS data will create anomalies on the internal frequency states of this clock. Once an attack is detected, the design will attempt to generate an approximated version of the correct GPS time t_{GPS} to holdover the timing system during an attack. This design would require two additional resources in addition to the GPS receiver: (1) a single (or multiple) free running oscillator(s), (2) additional data processing capabilities. The architecture of the system is shown in Figure 1.

Our approach depends on the internal frequency states (*i.e.,* frequency drifts and skew) of a hardware oscillator for spoofing detection. Hence, we provide details on hardware clock models in the next section to elucidate this approach.

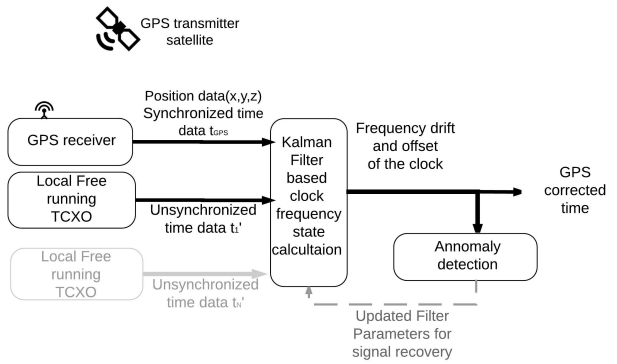

Figure 1: Design of the proposed secure GPS receiver with on-board spoofing detection. The receiver is equipped with a single (or multiple) free running temperature controlled oscillator(s). Kalman filter-based state estimation is used for calculating frequency states of this clock. In the case of multiple clocks, time from each free running oscillator can be used for generating a low noise stable virtual clock. Anomalies in the frequency drift and offset of this single clock (or the low noise virtual clock) calculated with respect to the GPS signal can reveal spoofing attacks on GPS signal. Updated filter parameters can help to reconstruct the approximated *true* time-offset during a spoofing attack.

3.1 Hardware Clocks

Clocks and oscillators in embedded systems are primarily used for time-keeping and synchronization purposes. In the majority of the embedded systems, crystal-based real-time clocks (RTCs) are used for precise time-keeping. These RTCs are far from perfect and they deviate from ideal time due to both systematic and random variations. These systematic variations arise from the imperfections in the physical realization of the clock and they are observed as time and frequency offsets and frequency drift. Therefore, at a given time t, the deviation of a clock from ideal time can be expressed as [1]:

$$x(t) = x_0 + y_0 t + \frac{1}{2} D t^2 + \epsilon(t) \tag{4}$$

where x_0, y_0 and D represents the time offset, frequency offset (also known as skew) and frequency drift. $\epsilon(t)$ represents non-deterministic random deviations. The frequency offsets and drifts of an RTC arise from the microscopic variations in the crystal used in these oscillators. The frequency offsets and drifts also vary with the dissimilarity in design, power supply, and environmental factors. These properties have been found to be different for similar oscillators working in a same operating condition. Therefore, we have the following assumptions regarding the frequency states of hardware clocks:

A1. Frequency drifts and skew of a clock with respect to a more precise reference are nearly constant and unique for a clock reference pair for a given duration.

A2. The states of a known free-running local oscillator are predictable for a given reference, and one can detect unusual activity in the reference by looking at the properties of the local oscillator.

A3. These properties vary uniquely for different clock pairs due to the random variations in their fabrication and are impossible to recreate without tampering the hardware of both the clocks.

Based on our assumption, the GPS induced internal states of a given free running oscillator are relatively constant, and therefore, can be used to detect spoofing attacks. This is the key concept for our approach. It should be noted that Khono *et al.* [10] first proposed the idea of remote device fingerprinting using the uniqueness of frequency offset of hardware clocks. Since the publication of Khono's work [10], there has been a significant development in this field of remote device fingerprinting using hardware oscillators. Our assumptions can be validated by Khono's work, the subsequent works in the literature and our experimental results and analysis presented in this work.

3.2 State-Space Model of Hardware Clocks

For precisely calculating hardware clock states, we use a stochastic model of the clocks where a clock-state is characterized by a column vector $x(t) = [x_1(t)\ y_1(t)\ D_1(t)]^T$. Here, $x_1(t), y_1(t)$ and $D_1(t)$ represents the time offset state, frequency offset state and frequency drift state respectively. The clock state follows the stochastic difference equations as given in [4]:

$$\frac{dx_1}{dt} = y_1 + w_1; \frac{dy_1}{dt} = D_1 + w_2; \frac{dD_1}{dt} = w_3; \tag{5}$$

where, $w_i(t)$ represents the associated zero mean white noise with spectral densities q_i. For an ensemble of q clocks the state vector can be written as $[x_1, y_1, D_1 \ldots x_q, y_q, D_q]^T$. Discrete-time equations for a system described by 5 can be written as [4]:

$$\mathbf{X_n} = \mathbf{F_n X_{n-1}} + \mathbf{W_n} \tag{6}$$

$$\xi_\mathbf{n} = \mathbf{H_n X_n} + \mathbf{V_n} \tag{7}$$

where, $n = 0, 1, 2, \ldots$ corresponds to discrete time t_n and measuring time interval $\Delta = t_n - t_{n-1}$.

For a single clock measurement, the $\mathbf{X_n} = [x_1, y_1, D_1]^T$ represents the state vector, and $\xi_\mathbf{n}$ denotes the observation vector. $\mathbf{F_n}$ is the state transition matrix which is calculated as:

$$\mathbf{F_n} = \begin{bmatrix} 1 & \Delta & \Delta^2/2 \\ 0 & 1 & \Delta \\ 0 & 0 & 1 \end{bmatrix} \tag{8}$$

The process noise $\mathbf{W_n}$ is considered to be zero mean additive white noise with covariance matrix $\mathbf{Q}$, where

$$\mathbf{Q} = \begin{bmatrix} q_1\Delta + q_2\frac{\Delta^3}{3} + q_3\frac{\Delta^5}{20} & q_2\frac{\Delta^2}{2} + q_3\frac{\Delta^4}{8} & q_3\frac{\Delta^3}{6} \\ q_2\frac{\Delta^2}{2} + q_3\frac{\Delta^4}{8} & q_2\Delta + q_3\frac{\Delta^3}{3} & q_3\frac{\Delta^2}{2} \\ q_3\frac{\Delta^3}{6} & q_3\frac{\Delta^2}{2} & q_3\Delta \end{bmatrix} \tag{9}$$

This state model for a clock ensemble is amenable to the design of optimal stochastic filters, which are broadly used for minimizing variance within a clock ensemble. In this work, we use this state space model for hardware oscillators and use an optimal filter (Kalman Formulation) to estimate these states for a single oscillator. It should be noted that if there are more than one on-board free running oscillators available, one could create a virtual time reference using an ensemble of clocks offering improved detection thresholds for spoofing attacks.

3.3 Kalman Filter Design for Spoofing Detection

We use discrete time state-model for developing a Kalman filter based spoofing detector. For our single clock experiment, we have the measurement matrix $\mathbf{H_n} = [1, 0, 0]$. $\mathbf{V_n}$

represents the zero mean measurement noise with covariance $\mathbf{R}$. For local measurements, we set $\mathbf{R} = \mathbf{0}$ to neglect the noise term. The algorithm for this linear Kalman filter [5] is given by the following equations:

Prediction Step:

$$\mathbf{m_{n|n-1}} = \mathbf{F_n m_{n-1|n-1}} \tag{10}$$

$$\mathbf{P_{n|n-1}} = \mathbf{F_n P_{n-1|n-1} F_n^T} + \mathbf{Q} \tag{11}$$

Update Step:

$$\mathbf{K_n} = \mathbf{P_{n|n-1} H_n^T} (\mathbf{H_n P_{n|n-1} H_n^T} + \mathbf{R})^{-1} \tag{12}$$

$$\mathbf{m_{n|n}} = \mathbf{m_{n|n-1}} + \mathbf{K_n}(\xi_\mathbf{n} - \mathbf{H_n m_{n|n-1}}) \tag{13}$$

$$\mathbf{P_{n|n}} = \mathbf{P_{n|n-1}} - \mathbf{K_n H_n P_{n|n-1}} \tag{14}$$

Here $\mathbf{m_{n|n}}$, $\mathbf{P_{n|n}}$ are the Gaussian posterior mean and covariance at n^{th} time-step, and $\mathbf{K_n}$ is the Kalman gain at that step. The clock states at n^{th} time-step is given by the components of $\mathbf{m_{n|n}}$, since $\mathbf{m_{n|n}}$ is the learned estimate of time offset, frequency offset and drift at that step. For our simulations, we assume the Gaussian posterior mean at the beginning is zero and the initial posterior covariance is $\mathbb{I}^{3\times3}$.

3.4 Anomaly Detection

Spoofing attacks induce variations in the GPS reference signal ranging from discrete step changes to slowly evolving changes in the demodulated GPS data. Our anomaly detection strategy classifies changes in the time offset, frequency drift, and frequency offset measurements in the received GPS data in relation to the hardware oscillator as anomalous when $\mathbf{X_n}$ lies outside the confidence interval of its predicted value $\mathbf{m_{n|n-1}} \pm \mathbf{S_{n-1}}$, where,

$$\mathbf{S_{n-1}} = \mathbf{H_n P_{n|n-1} H_n^T} + \mathbf{R} \tag{15}$$

is the predicted variance of the offset. This approach can be used for detecting simpler attacks inducing a step change in the time offset. This technique depends only on a single data point and an estimate, and therefore, leads to a large number of false positives. Moreover, an *advanced* attacker will self-consistently change the offset to avoid such detection.

A better approach is to use a windowed strategy that takes account of a number of recent measurements and find out the likelihood of a new measurement and estimate. For this approach, we calculate

$$p(m_{i,n}|m_{i,n-1}, \ldots m_{i,n-k}) = \frac{1}{\sqrt{2\pi\sigma_{i,n-1}^2}} e^{(-\frac{(m_{i,n}-\bar{m}_{i,n-1})^2}{\sigma_{i,n-1}^2})} \tag{16}$$

where, $i \in \mathbb{Z}^+$ for a 3×1 Gaussian posterior mean, k is the window size, $\sigma_{i,n-1}^2$ is the variance and $\bar{m}_{i,n-1}$ is the mean of the predicted values inside the window $(n-1)$ to $(n-k)$. By calculating a moving average of the log-likelihood z_n, we can detect an anomalous event when z_n crosses a predefined threshold. Here,

$$z_{i,n} = \alpha z_{i,n-1} + (1-\alpha)\ln(p(m_{i,n})) \tag{17}$$

with α as the smoothing factor.

4. EXPERIMENTAL SETUP AND RESULTS

For our experiments, we use a commercial GPS receiver and temperature-compensated crystal oscillators (TCXOs) to build an ensemble of free-running hardware oscillators. We use this design to validate the assumptions and formulation presented in Section 3. The measurement infrastructure is comprised of an ARM microprocessor-based data acquisition system with on-board data storage. Since this is a data-level spoofing detection technique, we inject attacks by altering the GPS time-offsets of the received GPS data and detect the attacks by using the free-running TCXO clock offsets measured with respect to this spoofed GPS data.

4.1 Adversary Model

The major goal of the adversary is to produce erroneous time or position measurements in the GPS receiver. We assume that the adversary has complete access to the RF channel during the attack, *i.e.,* he can replay, alter and/or replicate the RF carrier, spreading code and data bits of any or all of the visible satellites. We can divide the attackers in two categories: a *naive* and an *advanced* attacker. A *naive attacker* changes the time and/or the position bits in the GPS signal, which is reflected a sudden jump in the perceived time/location of the victim. An *advanced* attacker first induces the receiver to lock onto its spoofed signal by transmitting code, phase, and Doppler-matched signals with gradually increasing power, and then drags off the code and phase carrier in such a way that he avoids a discontinuous step change in time or the location estimates of the victim receiver. The adversary is time bound, *i.e.,* he has a limited time to spoof the receiver.

4.2 Attack Example and Spoofing Detection

To demonstrate the proposed spoofing detection approach, we consider an *advanced* attacker, who performs an attack on a GPS trained Phasor Measurement Unit (PMU) as shown in [14]. The attack involves the deployment of a simulated gradually increasing delay on GPS signals, which results in an anomalous exaggeration of signal transmission time, and in turn, induces an offset error in the GPS receiver. This attack described by the authors has been used by other experimental evaluations of fault detection algorithms and provides a well-documented baseline to study the effectiveness of our proposed approach. Figure 2(a) illustrates the evolution of the attack starting at 5130 seconds causing a gradual deviation of the GPS trained clock (solid line) against the true reference (dashed line).

To detect the attack, we modeled the TCXOs using the stochastic model presented in Section 3. Existing time offset based data level detection techniques can only detect spoofing if there is a discontinuous change in time, caused by a step change in the GPS reference. However, in this attack, the time is delayed slowly making the attack mostly undetectable. Since the process noise measurements of our TCXO were unknown, we used empirical values based on prior literature on clock jitter $q_1 = 10^{-3}, q_2 = 10^{-6}, q_3 = 10^{-9}$. We then design a state space model and use the Kalman filter formulation to detect anomalies in the received GPS signal.

From Figure 2, we can see that if we use the simultaneous negative values of averaged log-likelihood of frequency drift and offset as an indication of spoofing attack, the detector is able to detect the first anomaly at 5752s, (about 10 minutes after the start of the attack). Note that in this particular experiment the spoofing attack was detected when the cumulative error on the local clock was less than $4\mu s$. This is a relatively small error for some GPS-dependent systems.

5. ANALYSIS

Accuracy: The accuracy of the proposed detection technique depends on the noise margin and stability of the hardware clocks. For our experiments, we have used inexpensive temperature controlled crystal oscillators (TCXOs), which

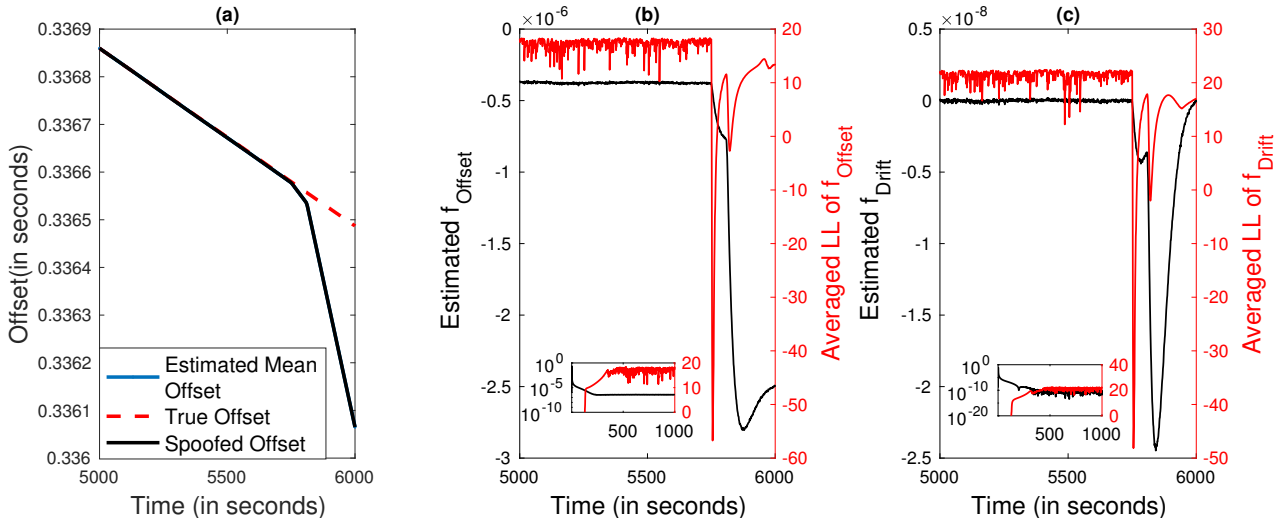

Figure 2: (a) Clock offset of a free running TCXO with respect to a GPS reference. A spoofing attack is initiated at 5130 seconds. The estimated time offset faithfully follows the spoofed signal as there is no sudden jump in the time offset;(b) Estimation of the frequency offset (black curve) and the averaged log-likelihood of the frequency offset(red curve) of the local clock with respect to the spoofed GPS signal during before and after the attack. The initial start-up and transition period for f_{offset} and the log-likelihood is shown in the inset. (c) Similar estimation of frequency drift (black curve) and the averaged log-likelihood of the frequency drift (red curve) of the local oscillator. The window size (k) for this computation is 128 data points.

provides a degree of immunity to temperature variations. One can use an ensemble of hardware oscillators to create a virtual clock using the system of equations as discussed in section 3.2 to reduce this noise. It should be noted that the accuracy and robustness of our detection mechanism requires a prior estimate of the measurement noise $\mathbf{Q}$. Without any prior estimate, for example, if we set the spectral densities q_1, q_2 and q_3 to 1, then this technique detects the attack at 5811 and 5813 seconds and reports a false positive at 3183 seconds.

Moreover, the detection mechanism is dependent on the window size (k) that provides a historical moving average. Larger windows provide better estimates; however, larger window size requires more memory and longer start-up time. For the experiment shown in Figure 2, $k = 8$ gives a false alarm rate of 66% which reduces to 20% for $k = 16$ and 0% for $k = 32$% and higher. Furthermore, there is a transient period during startup for the Kalman filter (as shown in the insets of Figure 2(b) and (c)) which requires a finite wait period before a consistent estimation of the log-likelihood. Therefore, the weakness of this proposed approach is the start-up period before effective detection is possible. For the example attack provided in this work, it takes about 600 seconds for system start-up.

Computation Cost: The cost of matrix-vector computations for a Kalman filter in the prediction and update step contains computation in the order of $O(\mathbf{D}^2)$, $O(\mathbf{MD})$, and $O(\mathbf{M}^3)$ complexity. The covariance matrices are symmetric, and therefore, Cholesky factorization can be used for maintaining $\mathbf{P}$ in a square-root form. Since the prediction and update step requires the knowledge of only current and previous steps, this construction has a very low memory complexity. The anomaly detection step has logarithmic complexity which can be simplified by approximating $p(m_{i,n}) \approx \mathbf{e}^{\left(-\frac{(m_{i,n}-\bar{m}_{i,n-1})^2}{\sigma^2_{i,n-1}}\right)}$. The averaging window has a fixed memory requirement which can be lowered by reducing the number of historical data points.

Hardware Overhead: GPS receivers already contain a hardware oscillator which is synchronized using the GPS signals. By turning off the synchronization it may be possible to convert this clock to a free running oscillator. The synchronization based timing corrections can be performed in software. Another approach is to add a hardware component with embedded free running oscillators to employ the proposed method without altering the GPS receiver design. The computation can be performed using on-board processors in IoT devices or by adding a low power microcontroller that takes GPS derived time as an input and provides the corrected time and spoofing detection capability to the system.

Power Constraints for IoT: It should be noted that using this spoofing detection technique for IoT devices may significantly increase the device's power consumption due to the need for continuous monitoring of the GPS signal. In order to save power by turning off monitoring periodically, the detector should compute the current frequency states as quickly as possible. Since there is a start-up time for this approach to detect spoofing, one can store the historic values for the Gaussian mean and variance along with the noise measurements to reduce the startup time. For low power GPS receivers, if the receiver is in sleep mode and starts with a spoofed GPS data from an attacker then this detection technique may be ineffective due to convergence transients during startup.

Signal Recovery: This detection technique has a-priori knowledge of the correct clock states, and one can calculate an approximated value of the true time offset using the historical clock states. For example, to recover from the example spoofing attack, once the attack is detected, the *approximately* correct value of the time offset can be calculated with the Gaussian posterior mean $(\mathbf{m_{n|n-1}})$ and the historical mean of the predicted values $\bar{\mathbf{m}}_{\mathbf{n,n-1}}$ using:

$$m_{1,n|n} = m_{1,n-1} + \Delta m_{2,n|n-1} + \frac{1}{2}\Delta^2 m_{3,n-1|n} \qquad (18)$$

$$m_{2,n|n} = \bar{m}_{2,n-1} \tag{19}$$

$$m_{3,n|n} = \bar{m}_{3,n-1} \tag{20}$$

and then using $m_{1,n|n}$ as the *approximate* value of the true time offset at a time step n.

Hardware Intrinsic Security: Our approach does not depend on networked components for synchronization or attack detection. The hardware oscillator(s), as well as the computational and measurement components, are located on the receiver and algorithm updates may be introduced using existing firmware update infrastructure. Since dependency on other networked components can make the system vulnerable to network attacks, the proposed method provides better security and reliability.

Tamper Resistant Design: Tampering with the onboard clock(s) would detectably affect and change the clock frequency states. As the changes are hardware dependent and unique, the design is inherently tamper-resistant.

Comparison with Existing Approaches As discussed earlier in this paper, attack detection via comparison of time-offset between the GPS-clock and a local clock is not always reliable and self-consistent attacks can easily spoof such detection techniques. Furthermore, offset comparison requires time from a single local clock to be synchronized with a trusted clock periodically to keep the readings accurate. As a result, current spoofing detection solutions in the literature on comparing time with a known entity requires a notion of trust to a third party time provider. To avoid these shortcomings, our proposed design measures the clock states of a hardware oscillator with respect to the GPS-time to verify the authenticity of the GPS signal.

It should be noted that commercially available low-cost GPS receivers do not use any of the countermeasures described in the previous section [11]. Most of the measures are only implemented as prototypes in the lab environment where software-defined radio platforms are used in most of the tracking, analysis, and detection approaches. Implementation cost for practical deployment of these prototypes significantly limits their widespread use. The spoofing detection method presented in this paper does not require additional RF circuitry or antennas and uses low-cost TCXOs and data analysis to detect attacks. Furthermore, since our method is a data-level detection mechanism, no change to the receiver architecture is required to integrate our countermeasure as an add-on to existing systems.

6. CONCLUSIONS

In this work, we present a design for integrating data-level spoofing detection with an existing GPS-based timing system. The design uses single (or multiple) free running oscillators to detect anomalies in the GPS-derived frequency drift and the offset. We demonstrate that this approach is able to provide fast and accurate detection of GPS spoofing attacks published in the literature. Since GPS spoofing attacks pose a significant threat to IoT systems, there is value in including spoofing detection methods such as the method presented in this paper in future GPS receiver designs.

7. DISCLAIMER

Official contribution of the National Institute of Standards and Technology; not subject to copyright in the United States. Certain commercial equipment, instruments, or materials are identified in this paper in order to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

Parts of this paper may have been presented in technical seminars and included in government publications. Recorded versions of those seminars and copyright free versions of publications are available through the National Institute of Standards and Technology.

8. REFERENCES

[1] D. W. Allan. Time and frequency(time-domain) characterization, estimation, and prediction of precision clocks and oscillators. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 34(6):647–654, 1987.

[2] C. Bonebrake and L. R. O'Neil. Attacks on GPS time reliability. *Security & Privacy, IEEE*, 12(3):82–84, 2014.

[3] P. H. Dana and B. M. Penrod. The role of GPS in precise time and frequency dissemination. *GPS World*, pages 38–43, 1990.

[4] C. A. Greenhall. A review of reduced Kalman filters for clock ensembles. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 59(3):491–496, 2012.

[5] S. S. Haykin. *Adaptive filter theory*. Pearson Education India, 2008.

[6] T. E. Humphreys. Detection strategy for cryptographic GNSS anti-spoofing. *IEEE Transactions on Aerospace and Electronic Systems*, 49(2):1073–1090, 2013.

[7] T. E. Humphreys, B. M. Ledvina, M. L. Psiaki, B. W. OHanlon, and P. M. Kintner Jr. Assessing the spoofing threat: Development of a portable GPS civilian spoofer.

[8] A. Jafarnia-Jahromi, A. Broumandan, J. Nielsen, and G. Lachapelle. GPS vulnerability to spoofing threats and a review of antispoofing techniques. *International Journal of Navigation and Observation*, 2012, 2012.

[9] X. Jiang, J. Zhang, B. J. Harding, J. J. Makela, and A. D. Dominguez-Garcia. Spoofing GPS receiver clock offset of Phasor measurement units. *IEEE Transactions on Power Systems*, 28(3):3253–3262, 2013.

[10] T. Kohno, A. Broido, and K. C. Claffy. Remote physical device fingerprinting. *IEEE Transactions on Dependable and Secure Computing*, 2(2):93–108, 2005.

[11] M. G. Kuhn. Signal authentication in trusted satellite navigation receivers. In *Towards Hardware-Intrinsic Security*, pages 331–348. Springer, 2010.

[12] P. Misra and P. Enge. *Global Positioning System: Signals, Measurements and Performance Second Edition*. Lincoln, MA: Ganga-Jamuna Press, 2006.

[13] M. L. Psiaki and T. E. Humphreys. Protecting GPS from spoofers is critical to the future of navigation. *IEEE Spectrum*, 2016.

[14] D. P. Shepard, T. E. Humphreys, and A. A. Fansler. Evaluation of the vulnerability of phasor measurement units to GPS spoofing attacks. *International Journal of Critical Infrastructure Protection*, 5(34):146 – 153, 2012.

[15] N. O. Tippenhauer, C. Pöpper, K. B. Rasmussen, and S. Capkun. On the requirements for successful GPS spoofing attacks. In *Proceedings of the 18th ACM Conference on Computer and Communications Security*, pages 75–86. ACM, 2011.