sensors

MDPI

Article

Authenticated Timing Protocol Based on Galileo ACAS

Francesco Ardizzon [1,*,†], Laura Crosara [1,†], Nicola Laurenti [1], Stefano Tomasin [1] and Nicola Montini [2]

[1] Department of Information Engineering, University of Padova, Via Gradenigo 6/A, 35131 Padova, Italy
[2] Qascom, Via Marinali 87, 36061 Bassano del Grappa, Italy
* Correspondence: francesco.ardizzon@unipd.it
† These authors contributed equally to this work.

Abstract: Global navigation satellite systems (GNSSs) provide accurate positioning and timing services in a large gamut of sectors, including financial institutions, Industry 4.0, and Internet of things (IoT). Any industrial system involving multiple devices interacting and/or coordinating their functionalities needs accurate, dependable, and trustworthy time synchronization, which can be obtained by using authenticated GNSS signals. However, GNSS vulnerabilities to time-spoofing attacks may cause security issues for their applications. Galileo is currently developing new services aimed at providing increased security and robustness against attacks, such as the open service navigation message authentication (OS-NMA) and commercial authentication service (CAS). In this paper, we propose a robust and secure timing protocol that is independent of external time sources, and solely relies on assisted commercial authentication service (ACAS) and OS-NMA features. We analyze the performance of the proposed timing protocol and discuss its security level in relation to malicious attacks. Lastly, experimental tests were conducted to validate the proposed protocol.

Keywords: GNSS; CAS; OSNMA; timing; security

Citation: Ardizzon, F.; Crosara, L.; Laurenti, N.; Tomasin, S.; Montini, N. Authenticated Timing Protocol Based on Galileo ACAS. *Sensors* **2022**, *22*, 6298. https://doi.org/10.3390/s22166298

Academic Editor: Jari Nurmi

Received: 30 July 2022
Accepted: 18 August 2022
Published: 21 August 2022

1. Introduction

Timing and synchronization are two key services provided by global navigation satellite systems (GNSSs). GNSS receivers use ranging signals and satellite-reported time information to obtain a position, velocity, and time (PVT) solution, providing time with nanosecond-level accuracy [1]. Moreover, synchronization between receivers at different locations can be established and maintained using GNSS reference time, such ascoordinated universal time (UTC) or the Galileo system time (GST). Thanks to this level of accuracy, several sectors rely on GNSS for synchronization operations, from financial institutions that use GNSS to timestamp transactions to Industry 4.0 and Internet of things (IoT) applications. The main standards for the dissemination of time and frequency over digital networks are the network time protocol (NTP) and the precision time protocol (PTP). The accuracy of NTP is usually within tens of milliseconds over the Internet, and it can be less than 1 ms in local area networks (LANs) with ideal network conditions [2], while PTP provides better accuracy, from hundreds of nanoseconds to microseconds [3]. Consequently, when nanosecond-level accuracy is required, a good solution is to adopt a local time server that relies on GNSSs.

For many of the mentioned applications and others that will be considered, accurate, reliable, and trustworthy time synchronization is required, and it can be obtained by relying only on authenticated GNSS signals. Indeed, the authentication feature provides trustfulness as it incorporates specific features that cannot be predicted or falsified into the broadcast GNSS signals, and an authentication-enabled receiver can interpret these characteristics to distinguish authentic signals from forgeries. The authentication can take place at two complementary levels: at the data level, i.e., on navigation messages, and at the ranging level, on pseudoranges between the satellite and receiver. The combination of data and code authentication enables the calculation of a secure PVT solution.

Navigation message authentication (NMA) techniques aim to ensure the authenticity of the content of the navigation messages, providing the user with the integrity protection of data. Open service navigation message authentication (OS-NMA) is a data authentication function for public Galileo E1B signals [4] in which the message transmitted by the satellites is interleaved with authentication data generated through broadcast authentication protocol timed-efficient stream loss-tolerant authentication (TESLA) [5], suitably adapted for optimal transmission via Galileo [1,6]. The TESLA protocol employs a one-way chain shared by Galileo satellites with a public root key. The keys in the chain are used in reverse order to generate message authentication codes (MACs). Keys are then shared (always in reverse order) in broadcast mode with a delay of a few seconds. The receiver can verify the MACs as soon as it becomes aware of the key.

Securing the pseudorange measurements computed by the receiver means authenticating the signal's source and the time that it takes for the signal to reach the receiver. Spreading code encryption (SCE) techniques are the most reliable option to limit access to GNSS signals, as they render the spreading code unpredictable. Some SCE-type solutions in the literature are the P(Y) code for GPS and the commercial authentication service (CAS) for Galileo, which complement OS-NMA by offering spreading code level authentication in the E6 band. The assisted commercial authentication service (ACAS), recently presented in [7,8], provides a code authentication method that is based on navigation data received and authenticated by OS-NMA, including the key to generate the digital signature. This is part of Galileo commercial service (CS). A change in the SCE approach for public GNSS signals was proposed in [9], where a spreading code authentication (SCA) technique was proposed that authenticates a transmitted signal by watermarking the public spreading code with unpredictable sequences. A similar SCA technique was proposed in [10], where short sequences called spread spectrum security codes (SSSCs) were interleaved with the public spreading code. This approach was refined in [11,12], where the authentication scheme called chips-message robust authentication (CHIMERA) was introduced, which aims at jointly authenticating the navigation data and the spreading code of GPS signals for civil usage. This scheme replaces a small part of the spreading code with a secret, cryptographically generated sequence that can subsequently be reproduced by the receivers when they become aware of the key. In this context, a way to optimize trade-offs between security level and signal availability to receivers that do not know the modified code was derived in [13]. In the following, we focus on the combination of OS-NMA and ACAS.

In this paper, we introduce a secure timing protocol that relies solely on E6C authentication features and OS-NMA authenticated messages. We used E6C ACAS to build a clock model that is both robust and thus able to compute reliable time corrections, and secure since it could detect signal tampering. Our approach comprised two consecutive steps: first, the receiver processes the E6C measurements to estimate the receiver clock bias and drift; second, the receiver combines the obtained measurements to estimate the current clock bias by either using a Kalman filter, or fitting a linear or quadratic least squares model. Moreover, we propose strategies for timing attack detection in which we check the consistency of each new measurement with the model that had been calculated. We look at two approaches for this task: clock monitoring and innovation testing. We model a time-push attack to validate the performance of the proposed security checks. Moreover, we evaluate the proposed protocol on both simulated and experimental data collected with a professional GNSS receiver in nominal conditions and under-attack scenarios.

The rest of the paper is organized as follows. Section 2 briefly reviews the main concepts of the ACAS mode; then, the scenario for our analysis is described in Section 3. The main contribution of this paper is provided in Section 4, where we describe our proposed approach for secure ACAS-based timing, while the attack and its detection are described is Section 5. Simulation and experimental results are discussed in Section 6. Lastly, Section 7 draws the conclusions of the paper.

2. Review of ACAS

CAS is the Galileo's SCE service aiming at providing signal authentication without modifications to Galileo first-generation core infrastructure and signals, and requiring only minimal changes to both the system and the receiver. CAS is currently under development but expected to be established by 2024: in particular, a proposal known as ACAS was presented in [7,8]. In ACAS, the E6C pseudo-random noise (PRN) spreading codes are neither short nor periodic sequences, but are generated by the system as a stream known as encrypted code sequence (ECS). Part of the ECS is re-encrypted using the TESLA keys employed by the OS-NMA protocol, and disseminated with the E1 open signal, generating the re-encrypted code sequence (RECS). The RECS are stored and published at predefined times on servers accessible to the public, such as the GNSS service center (GSC). Together with RECS, the server also publishes additional useful files for PVT computation, such as the broadcast group delay (BGD) for the E1–E6 bands. Once the RECS are retrieved from the server, the user can decrypt them by using the corresponding TESLA key, obtaining the related ECS. Lastly, the ECS is tested against previously stored samples received from the E6C signal, allowing for the user to verify the authenticity of the received signals. The TESLA key related to one (or more) RECS is revealed within the public Galileo E1B signal with a few seconds of delay compared to the transmission of the latter by the satellites.

This approach enables the receiving user to operate in standalone mode for the validity period of the predownloaded data (i.e., the RECS files) and without storing any secret cryptographic key. The RECS lengths are defined by the number of chips in these sequences, which is one of the key parameters in ACAS design as it determines the duration of the signal fragment used in correlation during the acquisition phase. Together with the size of the bins used for the Doppler frequency search, they define the acquisition search space and thereby the ability to find correlation peaks from which the pseudoranges and the authenticated PVT solution are computed. Another key parameter in ACAS is the distance between two consecutive RECS sequences, which determines how often the receiver can compute an authenticated solution. However, with ACAS, users assess the authenticity of the signals by checking the consistency between E6 and E1, which is not authenticated at the ranging level. In this work, we propose an authenticated timing protocol that relies only on ACAS and the navigation messages, which are both authenticated.

The PVT solution calculated via ACAS may also be useful for initializing the time synchronization required by OS-NMA, as RECSs files are designed to include the transmission time associated with the corresponding ECS of the keystream E6C, which can be used to resynchronize the receiver. The default ACAS operating mode is snapshot mode, since no navigation message and thus no ephemeris data are transmitted on E6.

3. System Model

We consider a scenario where a master clock is responsible for the synchronization of a network, composed of several devices or sensors connected via LAN. We assumed that this network was isolated; therefore, no attacker could influence the time dissemination process. The master clock is connected to a GNSS receiver, for instance, by being placed on the roof of a building with clear view of the sky. For this reason, we may assume that the received signals are transmitted by satellites mostly in line-of-sight (LOS), and that the effects of the multipath are minimal. The antenna position was fixed and known. We examined the case of a single-antenna receiver. Multiple antennas may still be employed, for example, to enhance the performance or security of the scheme by, e.g., checking the angle of arrival of a GNSS signal [14,15]. A representation of the considered scenario is depicted in Figure 1.

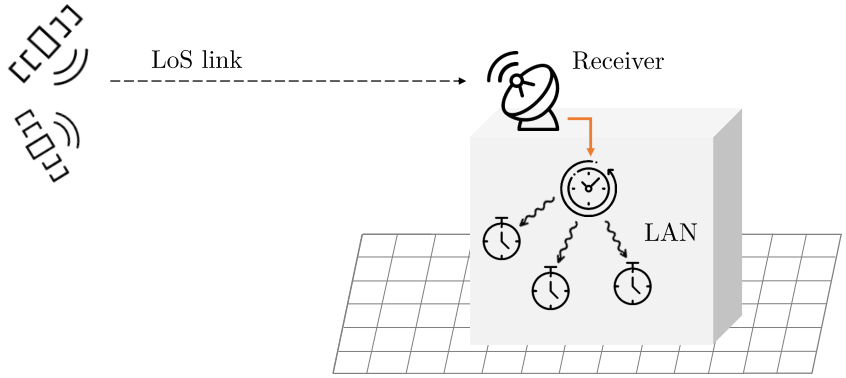


Figure 1. Pictorial representation of the considered scenario.

We considered a multifrequency receiver enabled to acquire and track Galileo signals (at least) in bands E1 ($f_c = 1575.42$ MHz) and E6 ($f_c = 1278.75$ MHz). Moreover, the receiver exploits both Galileo OS-NMA and ACAS. As briefly described in Section 2 and depicted in Figure 2, once the RECS files are published in the server and the TESLA key is received, the receiver decrypts the RECSs by using the corresponding key to obtain a local replica of the ECSs. Next, for the subset of Galileo satellites in view $\mathcal{S} \subseteq \{1, 2, \ldots 24\}$, it correlates the local replica with the prerecorded Galileo E6C signal samples and, from the correlation peaks, it computes code delay $u_i^{(s_i)}$ and the Doppler frequency $f_{\mathrm{D},i}^{(s_i)}$, measured by the receiver on signal on band E6, transmitted by satellite $s_i \in \mathcal{S}$ and received at time t_i.

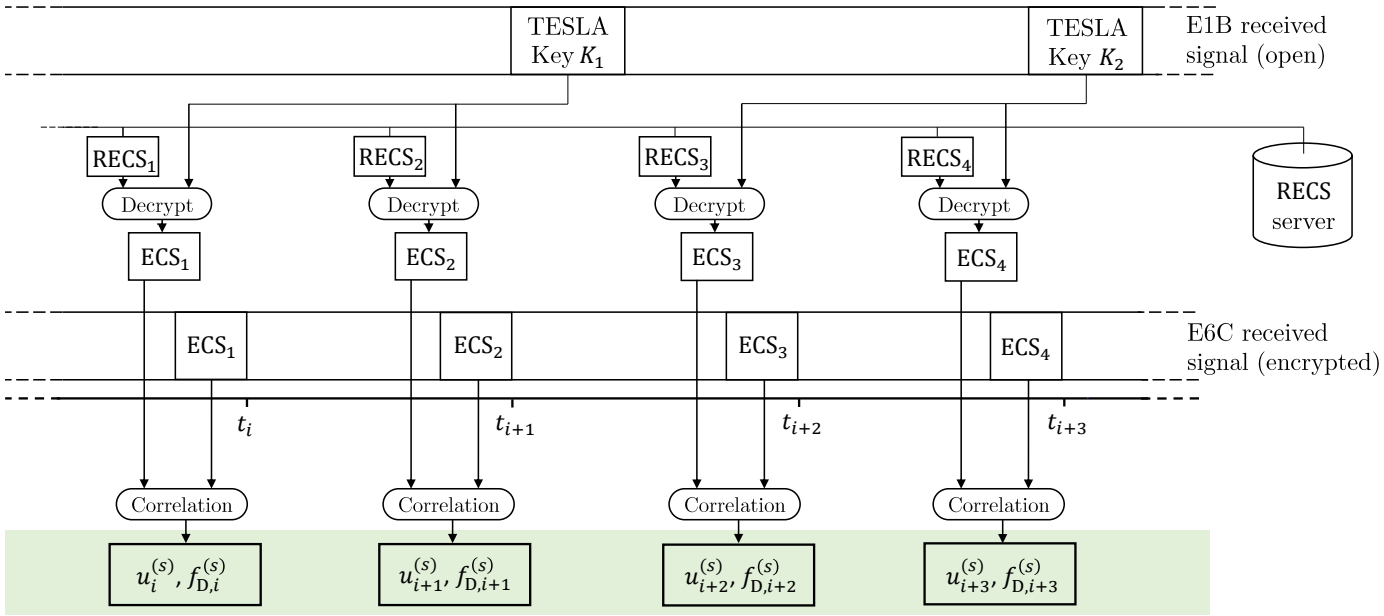


Figure 2. Summary of ACAS operations at the receiver side for signal transmitted by satellite s.

After collecting M observables, the aim is to estimate the current master clock bias. In detail, at time t_0, we used M measurements collected from the satellites in $\mathcal{S}$ at times $t_1, \ldots, t_M$, with $t_{i+1} \geq t_i$ and $t_0 > t_M$. Unlike the PVT computation, the proposed protocol provides timing even with fewer than four satellites in view.

4. Proposed Approach

In this section we propose a protocol that relies only on the observables authenticated by ACAS and the message, authenticated instead by OS-NMA, to compute the master clock correction. An ACAS observation is a 4-ple $\boldsymbol{O}_i = \{t_i, s_i, u_i, f_{\mathrm{D},i}\}$, where t_i is the observation

time, $s_i \in \mathcal{S}$ is the satellite ID, u_i is the observed code delay, and $f_{\mathrm{D},i}$ is the observed frequency offset (Doppler shift). We define the set of observables $\mathcal{O}$ as

$$\mathcal{O} = \{\boldsymbol{O}_i : \ i = 1, \ldots, M\} = \{(t_i, s_i, u_i, f_{\mathrm{D},i}) : \ i = 1, \ldots, M\}, \tag{1}$$

with $|\mathcal{O}| = M$, where all the measurements are obtained from the E6 signal. In the *preprocessing phase*, from observation in $\mathcal{O}$, we derived $\hat{T}_{\mathrm{b},i}$, estimated the clock bias at time t_i on the basis of observation $\boldsymbol{O}_i$, and $\hat{T}_{\mathrm{d},i}$, and estimated the clock drift at time t_i on the basis of observation $\boldsymbol{O}_i$. So, the output of the preprocessing phase is the set

$$\mathcal{T} = \{(\hat{T}_{\mathrm{b},i}, \hat{T}_{\mathrm{d},i}) : \ i = 1, \ldots, M\}, \tag{2}$$

which had the same cardinality as $\mathcal{O}$. Each measurement in $\mathcal{T}$, indexed by $i = 1, \ldots, M$, may be acquired by a different satellite. Next, the *current-state estimation* phase follows where measurements in $\mathcal{T}$ are used to compute the master clock correction, at time t_0, $\hat{T}_{\mathrm{b},0}$. Figure 3 summarizes both the two phases.

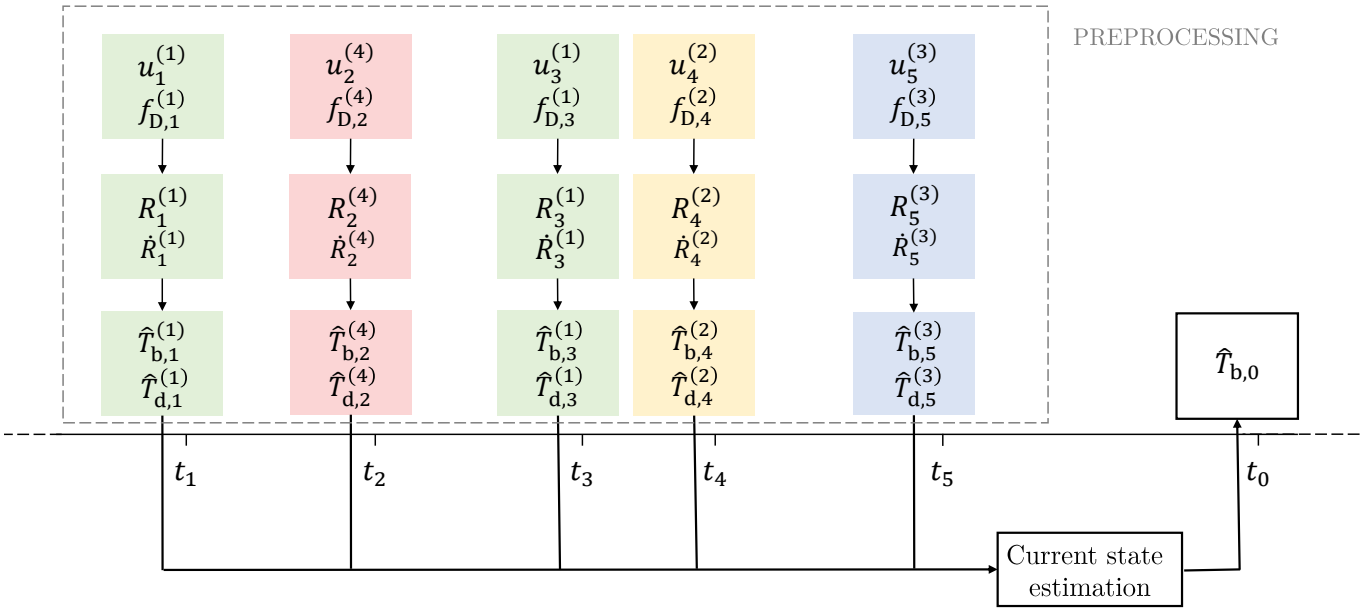


Figure 3. Schematic representation of preprocessing and current-state estimation phases.

The last phase concerns *security checks*, where we tried to detect anomalous estimates $\hat{T}_{\mathrm{b},0}$ of the clock bias; we considered the *clock monitoring* and *innovation test* as solutions for this task.

4.1. Preprocessing

Starting from each code delay measurement $u_i \in \mathcal{O}$, following the procedure described in [16], we computed pseudorange R_i at time t_i associated with satellite s_i. As indicated in Section 3, receiver position $\boldsymbol{P}_{\mathrm{rx}}(t)$ is fixed and publicly known. Satellite position $\boldsymbol{P}_{\mathrm{sat}}(t)$ and clock bias $T_{\mathrm{b,sat}}(t)$ can be retrieved from the authenticated OS-NMA message; thus, both can be considered to be authenticated at any time t. The computed pseudorange can be decomposed as

$$R_i = r_i + c(T_{\mathrm{b},i} - T_{\mathrm{b,sat}}^{(s_i)}(t_i)) + D_i + \eta_i, \tag{3}$$

where $r_i = ||\boldsymbol{P}_{\mathrm{sat}}^{(s_i)}(t_i) - \boldsymbol{P}_{\mathrm{rx}}(t_i)||$ is the geometric range, $T_{\mathrm{b,sat}}^{(s_i)}(t_i)$ is the satellite s_i clock bias, $D_i = D_{\mathrm{iono}}^{(s_i)}(t_i) + D_{\mathrm{tropo}}^{(s_i)}(t_i)$ is the sum of ionospheric and the tropospheric delays, η_i is

the additional noise due to the signal processing errors and multipath, and c is the speed of light.

The pseudoranges computed for E1 are corrected by using the estimations of ionosphere and troposphere delays transmitted in the E1B navigation message. In this case, we worked with E6 pseudoranges: however, since the troposphere is a nondispersive medium, the corrections for the tropospheric delay of band E1, $\hat{D}^{(s)}_{\text{tropo,E1}}(t)$, and E6, $\hat{D}^{(s)}_{\text{tropo,E6}}(t)$ were identical for all $s \in \mathcal{S}$. On the other hand, if the ionosphere is instead a dispersive medium, given the correction for E1, the correction for E6 is [17]

$$\hat{D}^{(s)}_{\text{iono,E6}}(t) = \hat{D}^{(s)}_{\text{iono,E1}}(t)\frac{f^2_{\text{E1}}}{f^2_{\text{E6}}}, \tag{4}$$

for all $s \in \mathcal{S}$ and for every time instant t. Correction $\hat{D}^{(s)}_{\text{iono,E1}}(t)$ must be obtained through a proper ionospheric correction model such as the Klobuchar model [18], or more precise models, such as Galileo NeQuick [19] or the IRI-P 2017 [20]. Only the measurements from E6 were actually authenticated; therefore, we could not exploit the measurements from another band (e.g., E1 or E5) to remove the ionospheric delay contribution, as it is typically performed in multifrequency GNSS receivers; instead, we had to use the model computed by using the parameters in the authenticated navigation message.

The receiver clock bias estimation at time t_i is then calculated from (3) and (4) as

$$\hat{T}_{\text{b},i} \triangleq \frac{1}{c}\left(R_i - r_i - \hat{D}_i\right) + T^{(s_i)}_{\text{b,sat}}(t_i) = T_{\text{b},i} + \xi_{\text{b},i}, \tag{5}$$

where $T_{\text{b},i}$ is the real receiver clock bias at time t_i, and $\xi_{\text{b},i}$ is the clock bias estimation error taking into account the error residuals due to the nonperfect atmospheric delays estimation and the additional noise component η_i.

Next, we compute the *pseudorange rate* $\dot{R}_i$ at time t_i as

$$\dot{R}_i = -\lambda f_{\text{D},i}, \tag{6}$$

where $f_{\text{D},i}$ belongs to the authenticated observables set $\mathcal{O}$ and λ is the wavelength of E6. From (3), the pseudorange rate can then be decomposed as

$$\dot{R}_i = \dot{r}_i + c\left(T_{\text{d},i} - T^{(s_i)}_{\text{d,sat}}(t_i)\right) + \gamma_i + \dot{\eta}_i, \tag{7}$$

where

$$\gamma_i = \gamma^{(s_i)}(t_i), \qquad \gamma^{(s)}(t) \triangleq \frac{\partial}{\partial t}\left[D^{(s)}_{\text{iono}}(t) + D^{(s)}_{\text{tropo}}(t)\right] \tag{8}$$

is a term modeling both the time derivatives of the the atmospheric delays and the signal processing errors. Moreover, the geometric range derivative $\dot{r}^{(s)}(t)$ is given by

$$\begin{aligned}\dot{r}^{(s)}(t) = \frac{\partial}{\partial t}||\boldsymbol{P}^{(s)}_{\text{sat}}(t) - \boldsymbol{P}_{\text{rx}}(t)|| &= (\boldsymbol{v}^{(s)}_{\text{sat}}(t) - \boldsymbol{v}_{\text{rx}}(t))^T \frac{\boldsymbol{P}^{(s)}_{\text{sat}}(t) - \boldsymbol{P}_{\text{rx}}(t)}{||\boldsymbol{P}^{(s)}_{\text{sat}}(t) - \boldsymbol{P}_{\text{rx}}(t)||} \\ &= (\boldsymbol{v}^{(s)}_{\text{sat}}(t) - \boldsymbol{v}_{\text{rx}}(t))^T \boldsymbol{e}^{(s)}(t) = v^{(s)}_{\text{LOS}}(t),\end{aligned} \tag{9}$$

where $\boldsymbol{e}^{(s)}(t)$ is the unit vector that points to the receiver antenna from the satellite, so $v^{(s)}_{\text{LOS}}(t)$ is the velocity projected into the LOS direction. Moreover, $\boldsymbol{v}_{\text{rx}}(t) = 0\ \forall t$, since the position of the GNSS receiver is fixed. Thus, term $\dot{r}_i$ appearing in (7) is obtained as

$$\dot{r}_i = \dot{r}^{(s_i)}(t_i) = v^{(s_i)}_{\text{LOS}}(t_i). \tag{10}$$

Analogously to (5), we compute

$$\hat{T}_{\mathrm{d},i} \triangleq \frac{1}{c}\left(\dot{R}_i - v_{\mathrm{LOS}}^{(s_i)}(t_i)\right) + T_{\mathrm{d,sat}}^{(s_i)}(t_i) = T_{\mathrm{d},i} + \xi_{\mathrm{d},i}, \tag{11}$$

where $T_{\mathrm{d},i}$ is the real receiver clock drift at time t_i and $\xi_{\mathrm{d},i}$ is the clock drift estimation error. Repeating this procedure for $i = 1, \ldots, M$, we obtain the set $\mathcal{T}$.

It is possible to statistically model both $\xi_{\mathrm{b},i}$ and $\xi_{\mathrm{d},i}$. A partial model for the first term is provided in [7,21,22]; however, the second-order descriptions of $\xi_{\mathrm{b},i}$ and $\xi_{\mathrm{d},i}$ are sufficient for the analysis in this paper.

4.2. Current-State Estimation

In the previous section, we showed how to derive measurements in $\mathcal{T}$ starting from the authenticated observables in $\mathcal{O}$. These estimates are exploited to compute the actual receiver clock bias that is used to correct the master clock. The design of a specific algorithm for this task is justified, since the clock bias and drift estimations are relative to time $t_i, i = 1, \ldots, M$; therefore, we need a model that exploits the past measurements to compute the current one. Moreover, past measurements are affected by noise, modeled by $\xi_{\mathrm{b},i}$ and $\xi_{\mathrm{d},i}$. We analyzed three different approaches to this task: a least squares (LS) quadratic model, a LS linear model, and a Kalman filter.

4.2.1. LS-Quadratic and Linear Model

The first two solutions leverage the idea that clock bias increases (or decreases) over time following a parabola, where the quadratic term, with coefficient *drift rate*, is expected to have a low impact. For instance, considering the *time of ephemeris* t_{oe}, the Galileo satellite clock bias is computed as follows [23]

$$T_{\mathrm{b,sat}}^{(s)}(t) = a_0^{(s)} + a_1^{(s)}(t - t_{\mathrm{oe}}) + a_2^{(s)}(t - t_{\mathrm{oe}})^2, \tag{12}$$

where $a_0^{(s)}$, $a_1^{(s)}$, and $a_2^{(s)}$ represent the satellite clock bias, clock drift, and clock drift rate measured at time t_{oe}, respectively. Typically the drift rate is transmitted to as $a_2^{(s)} = 0$, leading to a de facto linear model. Thus, we consider both a quadratic and a linear model.

Analogously to (12), calling $\tau_i = t_0 - t_i$ the time difference between the current time at which we want to compute the clock bias estimation and the time associated to the measurements, we can write

$$\hat{T}_{\mathrm{b},i} = a_0 + a_1\tau_i + a_2\tau_i^2 + \epsilon_{\mathrm{b},i}, \tag{13}$$

$$\hat{T}_{\mathrm{d},i} = a_1 + 2a_2\tau_i + \epsilon_{\mathrm{d},i}, \tag{14}$$

where a_0, a_1 and a_2 are now the parameters modeling the receiver clock behavior, $\hat{T}_{\mathrm{b},i}$ and $\hat{T}_{\mathrm{d},i}$ are the measurements in $\mathcal{T}$ computed in the preprocessing phase, $\epsilon_{\mathrm{b},i}$ and $\epsilon_{\mathrm{d},i}$ are the estimation errors related to the i-th measurement. Equivalently to (13) and (14), in matrix form, we have

$$\begin{pmatrix} \hat{T}_{\mathrm{b},i} \\ \hat{T}_{\mathrm{d},i} \end{pmatrix} = \begin{pmatrix} 1 & \tau_i & \tau_i^2 \\ 0 & 1 & 2\tau_i \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} + \begin{pmatrix} \epsilon_{\mathrm{b},i} \\ \epsilon_{\mathrm{d},i} \end{pmatrix} = \begin{pmatrix} \boldsymbol{E}_{\mathrm{b},i} \\ \boldsymbol{E}_{\mathrm{d},i} \end{pmatrix} \boldsymbol{a} + \boldsymbol{\epsilon_i}, \tag{15}$$

where $\boldsymbol{a} = [a_0\ a_1\ a_2]^T$ is the vector of parameters we aim to estimate. Next, considering all the measurements in $\mathcal{T}$, we stack the matrices, obtaining

$$\boldsymbol{y} = \begin{pmatrix} \boldsymbol{y}_{\mathrm{b}} \\ \boldsymbol{y}_{\mathrm{d}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{E}_{\mathrm{b}} \\ \boldsymbol{E}_{\mathrm{d}} \end{pmatrix} \boldsymbol{a} + \begin{pmatrix} \boldsymbol{\epsilon}_{\mathrm{b}} \\ \boldsymbol{\epsilon}_{\mathrm{d}} \end{pmatrix} = \boldsymbol{E}\boldsymbol{a} + \boldsymbol{\epsilon}, \tag{16}$$

where $\boldsymbol{y}_\mathrm{b}$ and $\boldsymbol{y}_\mathrm{d}$ are the columns vectors collecting the M bias and drift measurements, respectively, in $\mathcal{T}$, $\boldsymbol{E}_\mathrm{b} = [\boldsymbol{E}_{\mathrm{b},1}^T, \ldots, \boldsymbol{E}_{\mathrm{b},M}^T]^T$ and $\boldsymbol{E}_\mathrm{d} = [\boldsymbol{E}_{\mathrm{d},1}^T, \ldots, \boldsymbol{E}_{\mathrm{d},M}^T]^T$ contain the time difference terms associated to each measurement in $\boldsymbol{y}_\mathrm{b}$ and $\boldsymbol{y}_\mathrm{d}$, respectively, and $\boldsymbol{\epsilon} = [\boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_M]^T$. In order to minimize the mean square error (MSE), we performed the estimation by using the pseudoinverse

$$\hat{\boldsymbol{a}} = (\boldsymbol{E}^T\boldsymbol{E})^{-1}\boldsymbol{E}^T\boldsymbol{y}, \tag{17}$$

and we obtained the estimations of clock bias and drift at time t_0 as

$$\hat{T}_{\mathrm{b},0} = \hat{a}_0, \tag{18}$$

$$\hat{T}_{\mathrm{d},0} = \hat{a}_1. \tag{19}$$

An analogous derivation can be performed starting from a linear model, replacing (15) with

$$\begin{pmatrix} \hat{T}_{\mathrm{b},i} \\ \hat{T}_{\mathrm{d},i} \end{pmatrix} = \begin{pmatrix} 1 & \tau_i \\ 0 & 1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} + \begin{pmatrix} \epsilon_{\mathrm{b},i} \\ \epsilon_{\mathrm{d},i} \end{pmatrix}. \tag{20}$$

4.2.2. Kalman Filter

In this section, we investigate the use of a Kalman filter to estimate the bias. In particular, every time a new estimate $\{\hat{T}_{\mathrm{b},i}, \hat{T}_{\mathrm{d},i}\}$ was available, we updated the model and perform a new prediction; moreover, even when no new measurement was available, we exploited the previously trained model to estimate the current clock correction. A more detailed description of the Kalman filter can be found in [24].

The procedure was divided into two phases, *prediction* and *model update*. We call $\boldsymbol{x}_i$ the *true state* at time t_i, and $\boldsymbol{z}_i$ the *input* at time t_i, that is,

$$\boldsymbol{x}_i = \begin{pmatrix} T_{\mathrm{b},i} \\ T_{\mathrm{d},i} \\ \dot{T}_{\mathrm{d},i} \end{pmatrix}, \quad \boldsymbol{z}_i = \begin{pmatrix} \hat{T}_{\mathrm{b},i} \\ \hat{T}_{\mathrm{d},i} \end{pmatrix}, \tag{21}$$

where $\dot{T}_{\mathrm{d},i}$ represents the clock drift rate, which we did not measure directly. Then, the *state-transition matrix* and the *observation matrix* are given by

$$\boldsymbol{F}_i = \begin{pmatrix} 1 & t_i - t_{i-1} & (t_i - t_{i-1})^2 \\ 0 & 1 & 2(t_i - t_{i-1}) \\ 0 & 0 & 1 \end{pmatrix}, \quad \boldsymbol{H}_i = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}. \tag{22}$$

Differently from the general model for the Kalman filter, we had no control input. In the prediction step, we computed a priori state estimate $\hat{\boldsymbol{x}}_{i|i-1}$ and its covariance matrix $\boldsymbol{P}_{i|i-1}$:

$$\hat{\boldsymbol{x}}_{i|i-1} = \boldsymbol{F}_i\hat{\boldsymbol{x}}_{i-1|i-1} \tag{23}$$

$$\boldsymbol{P}_{i|i-1} = \boldsymbol{F}_i\boldsymbol{P}_{i-1|i-1}\boldsymbol{F}_i^\mathrm{T}. \tag{24}$$

Calling $\boldsymbol{R}_i$ the measurement noise covariance, during the update step, we computed

$$\boldsymbol{y}_i = \boldsymbol{z}_i - \boldsymbol{H}_i\hat{\boldsymbol{x}}_{i|i-1} \tag{25}$$

$$\boldsymbol{B}_i = \boldsymbol{H}_i\boldsymbol{P}_{i|i-1}^{-1}\boldsymbol{H}_i^\mathrm{T} + \boldsymbol{R}_i \tag{26}$$

$$\boldsymbol{K}_i = \boldsymbol{P}_{i|i-1}\boldsymbol{H}_i^\mathrm{T}\boldsymbol{B}_i^{-1} \tag{27}$$

$$\hat{\boldsymbol{x}}_{i|i} = \hat{\boldsymbol{x}}_{i|i-1} + \boldsymbol{K}_i\boldsymbol{y}_i \tag{28}$$

$$\boldsymbol{P}_{i|i} = (\boldsymbol{I}_2 - \boldsymbol{K}_i\boldsymbol{H}_i)\boldsymbol{P}_{i|i-1}. \tag{29}$$

We call $\hat{\boldsymbol{x}}_{i|i}$ and its covariance $\boldsymbol{P}_{i|i}$ the updated a posteriori estimate of the state. Term $\boldsymbol{y}_i$ is called *innovation* and is used together with its covariance $\boldsymbol{B}_i$ during the innovation check in the security steps. Repeating this procedure for every measure in $\mathcal{T}$, we obtained the M-th estimation $\boldsymbol{x}_M$. Then, from (23), we computed the a posteriori estimation at time t_0 as $\hat{\boldsymbol{x}}_{0|M} = \boldsymbol{F}_0 \hat{\boldsymbol{x}}_{M|M}$, where

$$\boldsymbol{F}_0 = \begin{pmatrix} 1 & t_0 - t_M & (t_0 - t_M)^2 \\ 0 & 1 & 2(t_0 - t_M) \\ 0 & 0 & 1 \end{pmatrix}. \tag{30}$$

Lastly, $\hat{T}_\mathrm{b}$ is the first element of $\hat{\boldsymbol{x}}_{0|M}$.

5. Timing Attack and Detection

In the system model of Section 3, we assumed that the position of the GNSS receiver was fixed and publicly known. Therefore, the receiver was assumed to perform a consistency check on the received signal, such that, if the receiver PVT computation yielded a position much different from the expected one or a significant velocity, an alarm would be raised. Moreover, since the satellites' position was known, the receiver could reject any signal coming from satellites that should not be in view: thus, the attacker is also forced to generate signals corresponding only to satellites actually in view by the legitimate receiver. Hence, the attacker knows that (1) all the attacks causing a relevant change in the victim's computed position or velocity are detected, and (2) signals transmitted by satellites that should not be in view by a legitimate receiver are neglected.

For these reasons, we consider an attacker performing a *time-push* attack: this is a *meaconing* attack where the receiver records signals and retransmits them with additional delays, adding an equal bias in all pseudoranges, which results in error in the time calculation of the PVT solution by the receiver, while the computed position does not change, as is proven in Section 6. Moreover, this attack may indeed target ACAS, where the signal cannot be tracked since the receiver operates in snapshot mode: this grants the attacker a time window to record the signal and perform a time-push attack. Sudden changes in the estimated clock bias may alert the receiver: thus, the attacker performs a time push in a smoothly progressive manner, gradually increasing the delay. However, to be effective, the attacker must be close to the victim's antenna to have the same satellites in view of the legitimate receiver.

A possible countermeasure to prevent this attack would be to render the area around the receiver inaccessible by, for instance, installing surveillance cameras and/or surrounding the building with a fence. Still, we considered a worst-case scenario where the attacker managed to approach close enough to the receiver antenna and isolate the legitimate receiver, ensuring that only fake signals are received to perform the time-push attack.

To detect the presence of false measurements among the obtained corrections, we considered *clock-monitoring* and *innovation-testing* [25,26] methods. Formally, we frame this problem as hypothesis testing: considering null-hypothesis $\mathcal{H}_0$ as the nominal condition where the signals are transmitted by the legitimate transmitter, the receiver observes a test statistic, β, and decides whether β is compatible with $\mathcal{H}_0$ or not.

5.1. Clock Monitoring

As discussed in Section 4.2, the receiver clock bias is typically assumed to have either linear or quadratic behavior over time: we can then analyze the clock bias corrections over time and if anomalous discontinuities are detected we raise an alarm. This is the idea behind clock-monitoring techniques. Given the clock model $\hat{\boldsymbol{a}}'$ estimated through either

(15) or (20) at time $t_i - \delta$, i.e., the previous epoch, it is possible to compute a prediction $\{\widetilde{T}_{\mathrm{b},i}, \widetilde{T}_{\mathrm{d},i}\}$ of the measurements at time t_i, as

$$\begin{pmatrix} \widetilde{T}_{\mathrm{b},i} \\ \widetilde{T}_{\mathrm{d},i} \end{pmatrix} = \begin{pmatrix} 1 & \delta & \delta^2 \\ 0 & 1 & 2\delta \end{pmatrix} \hat{\boldsymbol{a}}'. \tag{31}$$

Hence, for bias and drift, we adopted as the test statistic the quantities

$$\beta_{\mathrm{b},i} \triangleq \widetilde{T}_{\mathrm{b},i} - \hat{T}_{\mathrm{b},i}, \tag{32}$$

$$\beta_{\mathrm{d},i} \triangleq \widetilde{T}_{\mathrm{d},i} - \hat{T}_{\mathrm{d},i}, \tag{33}$$

and test

$$\hat{\mathcal{H}}_i = \begin{cases} \mathcal{H}_0 & \text{if} \quad |\beta_{\mathrm{b},i}| < \lambda_{\mathrm{b}} \text{ and } |\beta_{\mathrm{d},i}| < \lambda_{\mathrm{d}}, \\ \mathcal{H}_1 & \text{otherwise,} \end{cases} . \tag{34}$$

where thresholds λ_{b} and λ_{d} are chosen a priori by the user as a predefined false alarm (FA) probability. When a specific attack model is available, it may be possible to instead set the thresholds on the missed detection (MD) probability. More in detail, considering, for instance, drift threshold λ_{d}, it may be worth taking into account the actual clock specifications, thus evaluating a bound of the clock drift in nominal conditions [27].

If the distribution of the tests statistics $\beta_{\mathrm{b},i}$ and $\beta_{\mathrm{d},i}$ were known, it would be possible to replace (34) with two generalized likelihood ratio tests (GLRTs); however, the statistical characterization of such quantities is out of the scope of this work and is left to future works. Lastly, while we show the effectiveness of the clock monitoring only in relation to the LS models, such techniques may also be employed with the Kalman filter.

5.2. Innovation Testing

While using the Kalman filter, during the update step, each prediction is corrected by innovation term (25) that, in steady-state conditions, has mean and covariance

$$E[\boldsymbol{y}_i] = 0 \tag{35}$$

$$\mathrm{COV}(\boldsymbol{y}_i) = \boldsymbol{B}_i. \tag{36}$$

We can then use the normalized innovation as a test statistic, computed as follows:

$$\beta_{\mathrm{K},i} = \boldsymbol{y}_i^{\mathrm{T}} \boldsymbol{B}_i \boldsymbol{y}_i. \tag{37}$$

In nominal conditions, $\beta_{\mathrm{K},i}$ is assumed to have chi-squared distribution [26] with as many degrees of freedom as the size of the measurement z_i, $\beta_{\mathrm{K},i} \sim \chi^2$. Thus, to assess the authenticity of the measurement, we could use the GLRT test against a uniform distribution

$$\hat{\mathcal{H}}_i = \begin{cases} \mathcal{H}_0 & \text{if} \quad p(\beta_{\mathrm{K},i}|\mathcal{H}_0) \geq \lambda_{\mathrm{k}} \\ \mathcal{H}_1 & \text{otherwise} \end{cases} , \tag{38}$$

where λ_{k} is chosen by the user to match a predefined FA probability.

6. Results and Discussion

In this section, first, we validate the proposed approach; next, we show that the time-push attack described in Section 5 is successful even if a legitimate receiver knows its actual position, highlighting the need for additional security checks.

We collected experimental data to build the set of authenticated observables $\mathcal{O}$ serving as input for the preprocessing phase. The detection capabilities of the methods proposed in Sections 5.1 and 5.2 were tested against a simulated time-push attack.

6.1. Validation Using Experimental Data

To validate the proposed approach described in Section 4 we performed experimental tests collecting signals from an open-sky environment with a Septentrio PolarRx5 receiver connected to a A42 Hemisphere antenna. The experimental setup is depicted in Figure 4.

Figure 4. Setup used for the experimental dataset collection: Septentrio PolarRx5 receiver connected to an A42 Hemisphere antenna.

The output of the receiver was logged using the Septentrio binary format (SBF) standard and postprocessed after the experiments, obtaining a dataset of measurements from different constellations and frequency bands, summarized in Table 1.

Table 1. Constellations and central frequencies of the measurements collected in the experimental dataset.

	Central Frequency, f_c [MHz]								
	1176.45	**1207.14**	**1227.60**	**1245.5**	**1278.75**	**1268.52**	**1561.098**	**1575.42**	**1601.5**
Galileo	E5a	E5b			E6			E1BC	
GPS	L5		L2 C/A L2 P(Y)					L1 C/A L1 P(Y)	
Beidou	B2a	B2I				B3I	B1I	B1C	
GLONASS				L2 C/A					L1 C/A

We considered only measurements from two Galileo satellites that were visible during the whole experiment. As *ground truth* T_b that was later used to evaluate the goodness of our estimates $\hat{T}_b$, we used the clock bias measurements calculated from the PVT solution computed by the receiver using the whole set of measurements available in the dataset: on average, the PVT was computed by the receiver using the signal coming from 16 satellites. The Septentrio PolaRx5 is equipped with a voltage-controlled and temperature-controlled crystal oscillator (VCTCXO). Since only E6C ranging measurements were authenticated, we set the receiver to use the Klobuchar ionospheric correction model, which is the one typically used for GNSS receivers, estimating the ionospheric delay as in (4). More precise sophisticated models as Galileo NeQuick [19] and IRI-P 2017 [20] can be employed. For the sake of simplicity, we show that even the simpler Klobuchar model is enough to obtain satisfactory results, showing our method's robustness. Next, we extracted set $\mathcal{O}$ from our dataset considering only the measurements from E6C.

Figure 5 shows the master clock bias estimation error as the difference between the ground truth and the clock estimations, $\Delta\hat{T}_b$, obtained using the LS quadratic, LS linear

estimation methods and the Kalman filter in Figure 5. The LS methods described in Section 4.2.1 were used to compute one clock bias estimation $\hat{T}_b$ every 2 s using the 4 most recent available measurements, so that $M = 4$. The Kalman filter computed one new estimate $\hat{T}_b$ every second. All the tested methods were effective, achieving an error limited to less than 50 ns, obtaining precise timing with fewer than four satellites in view.

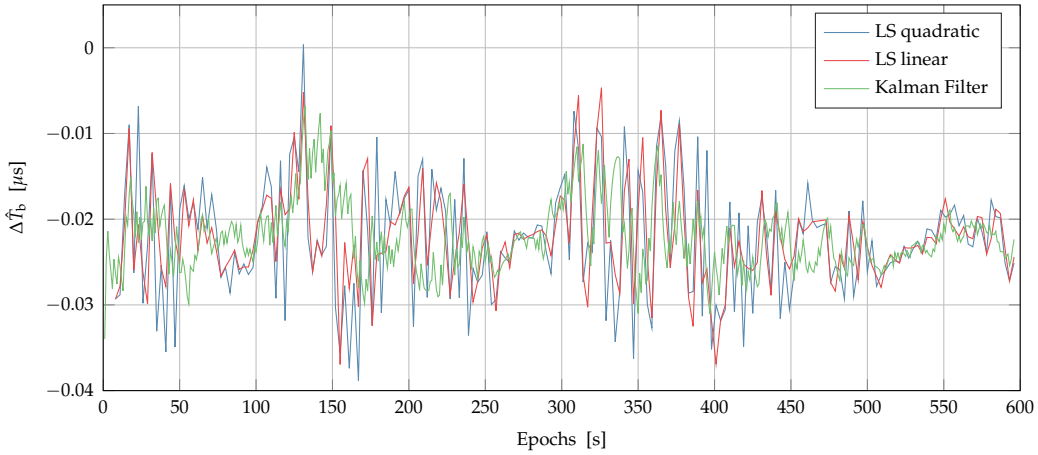


Figure 5. Difference between the ground truth and the clock estimations, $\hat{\Delta}T_b$, obtained by using the LS quadratic, LS linear and the Kalman filter on the experimental data.

6.2. Numerical Results and Attack Detection

To simulate the attacks, we used our signal generator and software receiver developed for the MORE Galileo open service signal integrity protection (MORE GOSSIP) project, funded by the European Space Agency (ESA) (see also [28]). We simulated the Galileo E6 baseband signal (the carrier frequency still influenced the Doppler frequency), generating both data (E6B) and pilot (E6C) components as in Galileo specifications [23], modulated with a BPSK(5), i.e., with code frequency $f_{\text{code}} = 5.115$ MHz. We considered an additional linear (deterministic) clock drift of 0.5 parts per million (ppm). We modeled a noiseless scenario with RECS duration equal to the PRN code length on E6, i.e., 5115 chips. Concerning CAS, we assumed that one new RECS would be disclosed every second. We generated 5 channels, i.e., 5 signals from five different satellites with 16 bit quantization. The sampling frequency was set to $f_s = 2f_{\text{code}} = 10.23$ MHz, and each simulation scenario lasted for 100 s. On the receiver side, the acquisition was performed by using the same sampling frequency, and the Doppler bin size was set to 75 Hz. The receiver collected measurements $\{\hat{T}_{b,i}, \hat{T}_{d,i}\}$ with a frequency of 1 Hz; as indicated before, since we assumed that the one RECS was made public every 60 s, we used only one of the measurements of the satellite in view per acquisition round as input for the model.

6.2.1. Nominal Scenario

We start by considering legitimate dataset $\mathcal{H}_0$. Only one RECS is disclosed at every epoch; thus, only one signal every epoch can be used to update the state.

Figure 6 shows the results obtained for the current-state estimation phase described in Section 4.2. In particular, we show $\Delta\hat{T}_b$, i.e., the difference between ground truth and clock estimations obtained by using the LS quadratic, LS linear, and the Kalman filter: all the methods were effective, achieving maximal deviation lower than 200 ns and a zero mean even using only one (new) measurement per epoch (i.e., per minute). Thus, all the methods could be employed for this task.

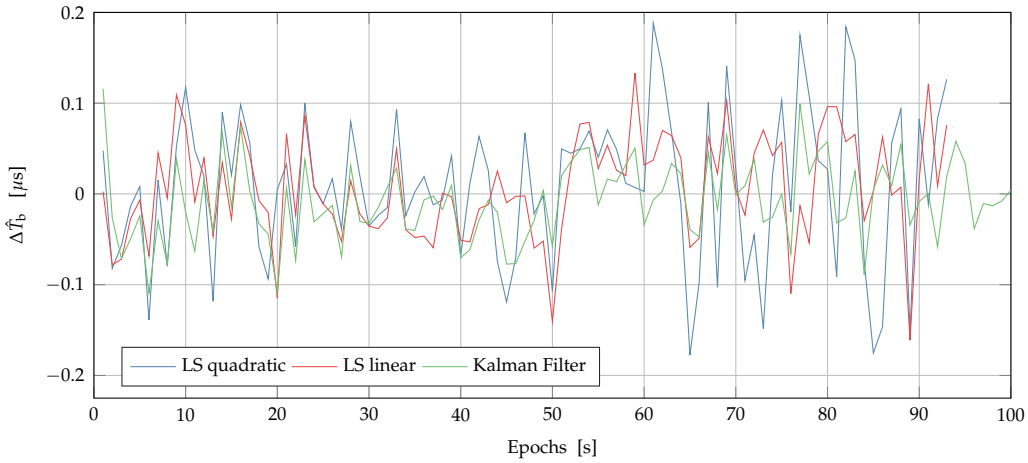


Figure 6. Difference between the ground truth and the clock estimations, $\Delta\hat{T}_b$, obtained by using the LS quadratic, LS linear and the Kalman filter on the simulated data.

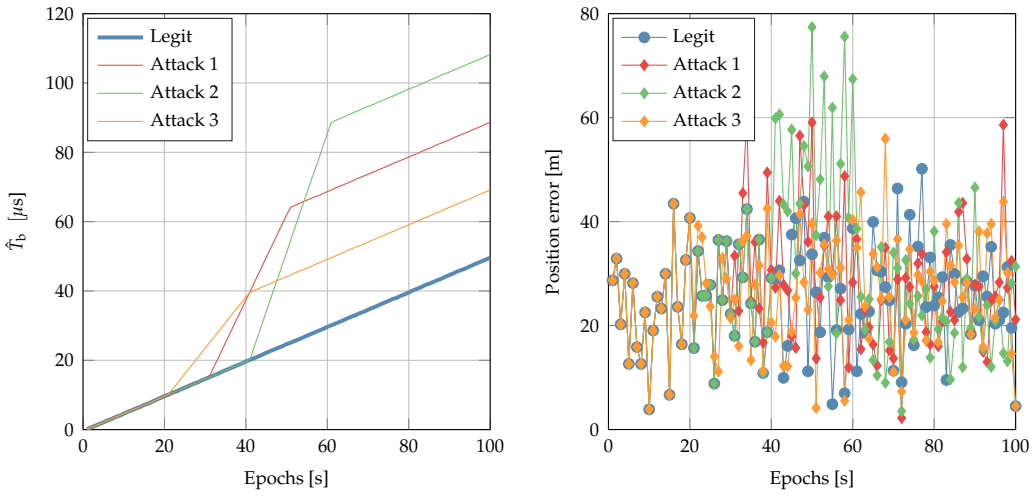
6.2.2. Attack Scenario

In this section, we evaluate under-attack scenarios, such as the ones described in Section 5.

In the first part of this section, we show the impact of a time-push attack, proving that such attacks cannot be detected just by the check on the receiver position. In the second part, we discuss the performance of the clock-monitoring and innovation-check methods, showing the different behaviors of the test statistics β_b, β_d, and β_K in the legitimate and under-attack scenarios, i.e., $\mathcal{H}_0$ and $\mathcal{H}_1$.

As indicated in Section 5, a sudden spike in the estimated clock bias may alert the receiver; thus, the attacker introduces the delays in a ramplike fashion. We modeled a scenario where the attacker managed to isolate the victim receiver and acquired only the forged E6 signals.

Figure 7 reports the results: while the positioning error statistic was indeed indistinguishable in $\mathcal{H}_0$ and $\mathcal{H}_1$, the impact on the clock bias is clear. This confirms that we cannot trust the timing obtained on a PVT that passes by the naive position check. Hence, we suggest dedicated algorithm and strategies specifically designed for secure timing.

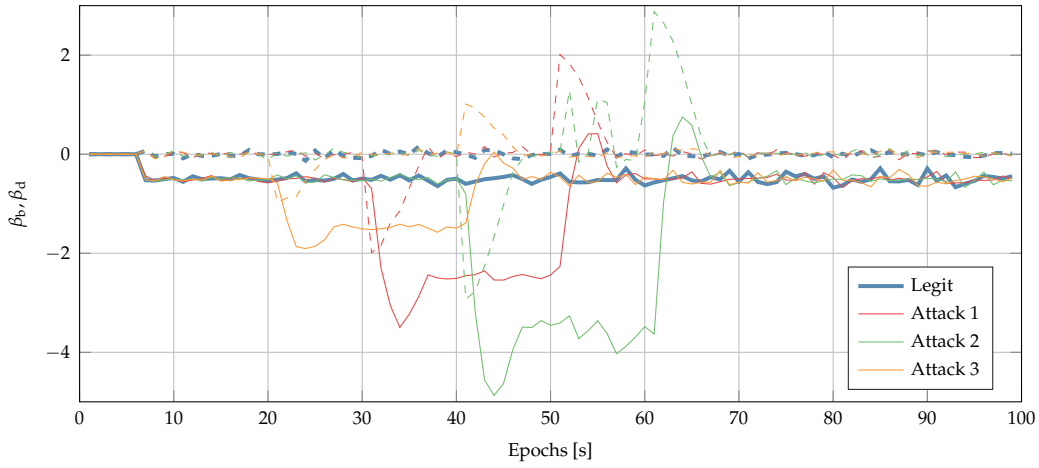


(**a**) (**b**)

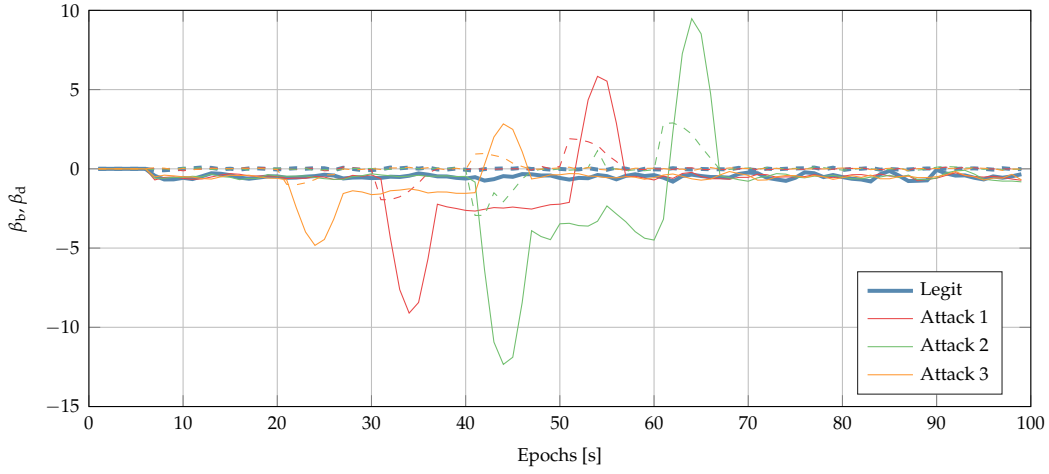
Figure 7. Comparison of legitimate and under-attack scenarios for (**a**) clock bias and (**b**) positioning error obtained using the simulated dataset.

Next, we validate the security checks described in Section 5 considering a legitimate scenario and three attack scenarios. Each attack lasted 20 s with a constant drift of 1, 2 and 3 ppm, and achieved a final delay of 20, 40, and 60 µs, respectively. Each attack started at a different time.

Figure 8 shows the test statistic obtained via clock monitoring in nominal conditions and an under-attack scenario: both β_b and β_d presented spikes associated to the start and end of the attack, which had a magnitude much greater than the standard deviation of the same test statistic in the nominal conditions. This test was, thus, indeed effective in detecting time-push attacks, since it is easy for the user to set a threshold to distinguish legitimate from under-attack scenarios. Moreover, performing more tests, it could be possible for the user to fine-tune the threshold by observing the receiver operating characteristic (ROC) curves.



(**a**)



(**b**)

Figure 8. Test statistics, β_b (continuous lines), and β_d (dashed lines) used by clock monitoring: comparison of legitimate (thick blue) and under-attack scenarios for the (**a**) linear and (**b**) quadratic LS models.

Figure 9 shows the test statistic β_K used for the innovation testing and described in Section 5.2. A jump is presented when the attacker starts (and ends) the time-push attack. Therefore, this technique is also successful at detecting time-push attacks.

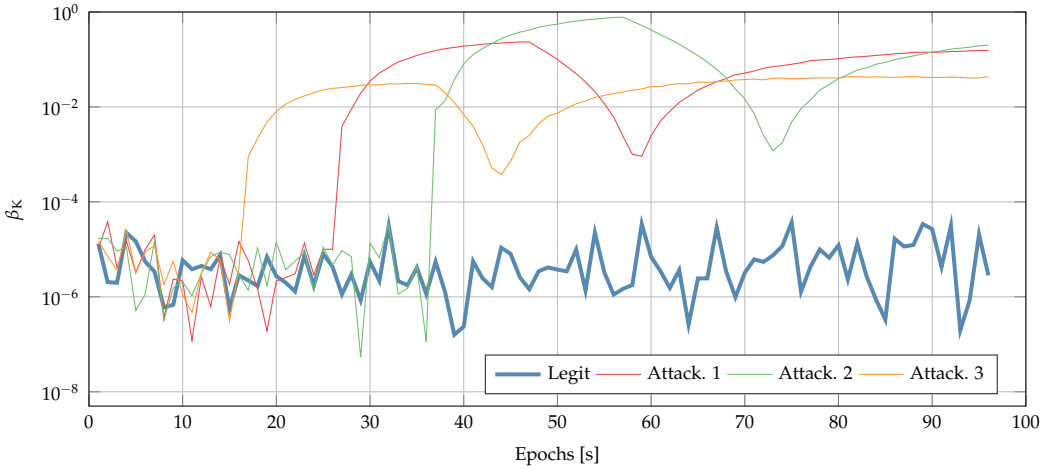


Figure 9. Test statistic β_K used by the innovation testing: comparison between legitimate (blue) and under-attack scenarios.

7. Conclusions

In this work, we presented a secure timing protocol that may be used, for instance, by Industry 4.0 applications to synchronize multiple IoT devices within a facility. We considered a scenario where the master clock was securely connected to a GNSS receiver, and all the devices or sensors aimed to be synchronized. The protocol was based upon the new Galileo ACAS protocol and relied only on authenticated measurements to obtain the clock correction.

The procedure was composed by three blocks: first, exploiting the fact that the facility position is known, the receiver processes the E6C measurements to obtain an estimation of the receiver clock bias and drift; second, the receiver merges the previously obtained measurements to compute the current clock bias estimation by fitting either a linear or a quadratic least-squares model, or by using a Kalman filter. Lastly, we also considered the employment of a security evaluation phase where we tested the consistency of each new measurement with the previously estimated model. For this task, we considered two methods: clock monitoring and innovation test. We validated the proposed procedure using an experimental dataset collected with a Septentrio PolaRx5 receiver, and simulated data considering both legitimate and under-attack conditions. The obtained numerical and experimental results show that our protocol was both able to compute a reliable timing correction and to reject time-push attacks.

Author Contributions: Conceptualization, F.A., L.C., N.L., S.T. and N.M.; methodology, F.A. and L.C.; software, F.A. and L.C.; validation, F.A. and L.C.; formal analysis, F.A. and L.C.; investigation, F.A. and L.C.; writing—original draft preparation, F.A. and L.C.; writing—review and editing, F.A., L.C., N.L., S.T. and N.M.; visualization, F.A. and L.C.; supervision, N.L. and S.T. All authors have read and agreed to the published version of the manuscript.

Funding: this research was funded by Regione Veneto under the project VIrtualization and Remotization for Resilient and Efficient Manufacturing (VIR2EM), POR FESR 2014-2020 DGR n. 822/2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

ACAS	Assisted commercial authentication service
BGD	Broadcast group delay
CAS	Commercial authentication service
CHIMERA	Chips-message robust authentication
CS	Commercial service
ECS	Encrypted code sequence
FA	False alarm
GLRT	Generalized likelihood ratio test
GNSS	Global navigation satellite system
GSC	GNSS service center
GST	Galileo system time
IoT	Internet of Things
LAN	Local area network
LOS	Line of sight
LS	Least squares
MAC	Message authentication code
MD	Missed detection
MSE	Mean square error
NMA	Navigation message authentication
NTP	Network time protocol
OS-NMA	Open service navigation message authentication
ppm	Parts per million
PRN	Pseudo-random noise
PTP	Precision time protocol
PVT	Position, velocity, and time
RECS	Re-encrypted code sequence
ROC	Receiver operating characteristic
SBF	Septentrio binary format
SCA	Spreading code authentication
SCE	Spreading code encryption
SSSCs	Spread spectrum security codes
TESLA	Timed-efficient stream loss-tolerant authentication
UTC	Coordinated universal time
VCTCXO	Voltage-controlled and temperature-controlled crystal oscillator

References

1. Fernandez-Hernandez, I.; Walter, T.; Neish, A.; O'Driscoll, C. Independent Time Synchronization for Resilient GNSS Receivers. In Proceedings of the 2020 International Technical Meeting of The Institute of Navigation (ION), San Diego, CA, USA, 21–24 January 2020; pp. 964–978.
2. Mills, D.L. *Computer Network Time Synchronization: The Network Time Protocol on Earth and in Space, Second Edition*; CRC Press: Boca Raton, FL, USA, 2016.
3. Watt, S.T.; Achanta, S.; Abubakari, H.; Sagen, E.; Korkmaz, Z.; Ahmed, H. Understanding and applying precision time protocol. In Proceedings of the 2015 Saudi Arabia Smart Grid (SASG), Jeddah, Saudi Arabia, 7–9 December 2015; pp. 1–7. [CrossRef]
4. Hernández, I.F.; Ashur, T.; Rijmen, V.; Sarto, C.; Cancela, S.; Calle, D. Toward an Operational Navigation Message Authentication Service: Proposal and Justification of Additional OSNMA Protocol Features. In Proceedings of the 2019 European Navigation Conference (ENC), Warsaw, Poland, 9–12 April 2019; pp. 1–6. [CrossRef]
5. Perrig, A.; Tygar, J.D. TESLA Broadcast Authentication. In *Secure Broadcast Communication: In Wired and Wireless Networks*; Springer: Boston, MA, USA, 2003; pp. 29–53. [CrossRef]
6. Fernández-Hernández, I.; Rijmen, V.; Seco-Granados, G.; Simon, J.; Rodríguez, I.; Calle, J.D. A Navigation Message Authentication Proposal for the Galileo Open Service. *NAVIGATION J. Inst. Navig.* **2016**, *63*, 85–102. [CrossRef]
7. Terris-Gallego, R.; Fernandez-Hernandez, I.; López-Salcedo, J.A.; Seco-Granados, G. Guidelines for Galileo Assisted Commercial Authentication Service Implementation. In Proceedings of the 2022 International Conference on Localization and GNSS (ICL-GNSS), Tampere, Finland, 7–9 June 2022; pp. 1–7. [CrossRef]
8. Fernandez-Hernandez, I.; Cancela, S.; Terris-Gallego, R.; Seco-Granados, G.; López-Salcedo, J.A.; O'Driscoll, C.; Winkel, J.; Chiara, A.d.; Sarto, C.; Rijmen, V.; et al. Semi-Assisted Signal Authentication based on Galileo ACAS. *arXiv* **2022**, arXiv:2204.14026.

9. Kuhn, M.G. An Asymmetric Security Mechanism for Navigation Signals. In *Information Hiding*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 239–252.
10. Scott, L. Anti-Spoofing & Authenticated Signal Architectures for Civil Navigation Systems. In Proceedings of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003), Portland, OR, USA, 9–12 September 2003; pp. 1543–1552.
11. Scott, L. Proving Location Using GPS Location Signatures: Why it is Needed and a Way to Do It. In Proceedings of the 26th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2013), Nashville, TN, USA, 16–20 September 2013; pp. 2880–2892.
12. Anderson, J.M.; Carroll, K.L.; DeVilbiss, N.P.; Gillis, J.T.; Hinks, J.C.; O'Hanlon, B.W.; Rushanan, J.J.; Scott, L.; Yazdi, R.A. Chips-Message Robust Authentication (Chimera) for GPS Civilian Signals. In Proceedings of the 30th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2017), Portland, OR, USA, 25–29 September 2017; pp. 2388–2416. [CrossRef]
13. Laurenti, N.; Poltronieri, A. Optimal Compromise among Security, Availability and Resources in the Design of Sequences for GNSS Spreading Code Authentication. In Proceedings of the 2020 International Conference on Localization and GNSS (ICL-GNSS), Tampere, Finland, 2–4 June 2020; pp. 1–6. [CrossRef]
14. Yang, Q.; Zhang, Y.; Tang, C.; Lian, J. A Combined Antijamming and Antispoofing Algorithm for GPS Arrays. *Int. J. Antennas Propag.* **2019**, *2019*, 8012569. [CrossRef]
15. Meurer, M.; Konovaltsev, A.; Appel, M.; Cuntz, M. Direction-of-Arrival Assisted Sequential Spoofing Detection and Mitigation. In Proceedings of the 2016 International Technical Meeting of the Institute of Navigation (ION), Monterey, CA, USA, 25–28 January 2016. [CrossRef]
16. Van Diggelen, F. *A-GPS: Assisted GPS, GNSS, and SBAS*; Artech House: Boston, MA, USA, 2009.
17. Kaplan, E.D.; Hegarty, C.J. *Understanding GPS, Principles and Applications*, 2nd ed.; Artech House: Boston, MA, USA, 2005.
18. Klobuchar, J.A. Ionospheric Time-Delay Algorithm for Single-Frequency GPS Users. *IEEE Trans. Aerosp. Electron. Syst.* **1987**, *AES-23*, 325–331. [CrossRef]
19. EUSPA. Ionospheric Correction Algorithm for Galileo Single Frequency Users. Available online: https://www.gsc-europa.eu/sites/default/files/sites/all/files/Galileo_Ionospheric_Model.pdf (accessed on 30 July 2022).
20. Sezen, U.; Gulyaeva, T.; Arikan, F. Online computation of International Reference Ionosphere Extended to Plasmasphere (IRI-Plas) model for space weather. *Geod. Geodyn.* **2018**, *9*, 347–357. [CrossRef]
21. Ardizzon, F.; Caparra, G.; Fernandez-Hernandez, I.; O'Driscoll, C. *A Blueprint for Multi-Frequency and Multi-Constellation PVT Assurance*; NAVITEC: Noordwijk, NL, USA, 2022.
22. Walter, T.; Blanch, J.; DeGroot, L.; Norman, L.; Joerger, M. Ionospheric Rates of Change. In Proceedings of the 31st International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2018), Miami, FL, USA, 24–28 September 2018; pp. 4158–4170. [CrossRef]
23. Galileo Signal-in-Space Interface Control Document. Available online: https://www.gsc-europa.eu/sites/default/files/sites/all/files/Galileo_OS_SIS_ICD_v2.0.pdf (accessed on 30 July 2022).
24. Kay, S.M. *Fundamentals of Statistical Signal Processing: Estimation Theory*; Prentice Hall: Upper Saddle River, NJ, USA, 1997.
25. Broumandan, A.; Lachapelle, G. Spoofing Detection Using GNSS/INS/Odometer Coupling for Vehicular Navigation. *Sensors* **2018**, *18*, 1305. [CrossRef] [PubMed]
26. Liu, Y.; Li, S.; Qiangwen, F.; Liu, Z. Impact assessment of GNSS spoofing attacks on INS/GNSS integrated navigation system. *Sensors* **2018**, *18*, 1433. [CrossRef] [PubMed]
27. Ardizzon, F.; Laurenti, N.; Sarto, C.; Gamba, G. It's Galileo time: Options for crystal oscillators in OSNMA-enabled receivers. *GPS World* **2022**, *33*, 16–19.
28. Ceccato, S.; Formaggio, F.; Caparra, G.; Laurenti, N.; Tomasin, S. Exploiting Side-Information For Resilient GNSS Positioning in Mobile Phones. In Proceedings of the IEEE/ION Position Location and Navigation Symposium (PLANS), Monterey, CA, USA, 23–26 April 2018. [CrossRef]