GPS Radio Hacking – What the Hell Time Is It?
Radiohacking GPS – víme, která bije?

Tomáš Rosa

Tomáš Rosa
tomas.rosa@rb.cz

Dr. Tomáš Rosa holds Ph.D. in cryptology with the Best Doctoral Work Award of the Rector of Czech Technical University for 2004. He is focused on mathematical and physical methods of computer security, especially on embedded and radio applications. He helped to improve several worldwide standards, namely TLS protocol, EMV payment scheme, and Bluetooth.

Tomáš Rosa je doktorem v oboru kryptologie s Cenou rektora ČVUT za rok 2004. Věnuje se matematicko-fyzikálním metodám počítačové bezpečnosti, speciálně pak vestavěným a rádiovým aplikacím. Pomohl zlepšit několik celosvětových standardů, konkrétně protokol TLS, platební schéma EMV a bezdrátový standard Bluetooth.

GPS Radio Hacking – What the Hell Time Is It?

Possibility of position, velocity, and also time spoofing of civil GPS is a well known implication of the deliberately missing cryptographic protection of the L1 C/A satellite signal. Practical feasibility was already demonstrated several times, so is there anything new? It is in the massive rise of software-defined radio (SDR) phenomenon, that will soon allow even script kiddies to download the exploit code from the internet and let it run. Are you still sure where are you and what time is it? Does it hold for your infrastructure, too? Countermeasures are uneasy, proper understanding is the key.

Radiohacking GPS – víme, která bije?

Možnost podvržení polohy, rychlosti a také času civilní GPS je známým důsledkem záměrné absence kryptografické ochrany družicového signálu L1 C/A. Praktická schůdnost byla také už několikrát prokázána, ergo co je zde nového? Jde o masivní nástup fenoménu softwarového rádia (SDR), které brzy dovolí i hackerským zelenáčům stáhnout si útočný program z internetu a spustit ho. Stále jste si jisti, kolik je hodin a kde jste? Platí to i o vaší infrastruktuře? Obrana je nesnadná, klíčem je pochopení problému.

1 Introduction

GPS receivers for position, velocity, and time (PVT) measurements based on the civil service known as C/A (Coarse Acquisition) or L1 C/A [1], [8], [24], [37], [39], where L1 stands for the satellite downlink frequency of 1575.42 MHz, are almost ubiquitous. They have also successfully found their way into many IT systems where they help to establish the spacetime topology and synchronization [8]. Many of these systems are also parts of many critical infrastructures [10], [40]. This is not surprising, as this service seems to be a really marvellous gift coming to us from the sky in almost any place on Earth. There is, however, a considerable pitfall in that this service is by no means as robust and bullet-proof as we would like it to be [2], [10], [16], [17], [26], [31], [34], [35], [40], [41].

Before starting the main topic, three clarifications are apposite. First, we will often use the GNSS (Global Navigation Satellite System[s]) shorthand when referring to general properties that are common to most of the contemporary satellite navigation systems. Today, GNSS set includes mainly the following instances [1]:

- BeiDou-2 (former COMPASS) developed and maintained by China.
- Galileo of the Europen Union being built by the European GNSS Agency (GSA) residing in Prague.
- GLONASS (Globalnaia Navigationnaia Sputnikovaia Sistema) designed and operated by Russia.
- NAVSTAR Global Positioning System (GPS) of the United States.

The second note is on the service(s) we are going to talk about. Basically any GNSS offers two kinds of signal: open and reserved ones. In case of GPS, this division is classically made into the civil and military signals. Despite not serving a united army force, the European Galileo [28] also has its security enhanced flagship called Public Regulated Service (PRS) [1], [32]. However, it seems to be nearly impossible to get any plausible public review of these services, their security goals, protections that are effectively applied, and last but not least the end-user license policy. Their usage in civil applications seems to be therefore very limited. This is, unfortunately, also true for the PRS of Galileo, although ideas exist on how to theoretically overcome its restrictions [32]. Anyway, we will focus solely on the open or civil GNSS services here.

Finally, please note this is meant to be a relatively quick overview of the GNSS/GPS hacking state of the art to support the invited talk at IS2 2016. It has to be accessible to a broad audience, including technical experts as well as chief security officers, so the style has to be concise and clear. The result is a kind of author's essay, rather than a comprehensive study. The interested reader is kindly referred to the references, namely [1], [8], [24], [27], [37], [39] for GNSS theory and practice, and [10] for a monography devoted to GNSS vulnerabilities and attacks. The practical experiments touched here are detailed in author's technical presentation [31] that can be also seen as a kind of scholastic supplement to the recent practical hacking demonstrations [16] and [41].

The rest of this overview is organised as follows: In part two, we review the basic positioning principles of GNSS/GPS. We then continue with the vulnerabilities and possible attacks on the L1 C/A civil service in part three. Practical experiments [31] are then briefly noted in part four. In part five, we discuss on how much of help we can expect from the Satellite-Based Augmentation System(s) (SBAS) that are used, for instance, to elevate the L1 C/A service quality to conform with the increased aviation safety demands. In part six, we review the basic countermeasure ideas, and we finally conclude in part seven. In that part, the very recent GSA announcements on the emerging Galileo Open Service security enhacements [53] is also discussed.

2 GNSS/GPS Positioning and Timing Reviewed

Generally speaking, any GNSS consists of three major parts: the space, ground, and user segments. The space segment consists of several (in case of GPS at least 24) operational satellites that continuously broadcast their signal in space (SIS) towards the Earth [1], [24], [28]. The ground segment is responsible for satellite maintenance, including periodical checking of the SIS quality and uploading possible corrections to the respective space vehicles (SVs). The user segment encapsulates all those particular client receivers that consume SIS broadcasted from the selected satellites to compute user position, velocity, and time (PVT). In this paper, we are mainly interested in the SIS properties [1], [12], [15], [24], [27] and processing [37].

The method behind GNSS positioning is called a triangulation. Having given the position (x_1, y_1, z_1) of one SV and the actual distance r_1 in between this particular SV and the user, the receiver can deduce that its spatial coordinates (x_u, y_u, z_u) must be on a sphere satisfying

$$r_1 = [(x_1 - x_\mathrm{u})^2 + (y_1 - y_\mathrm{u})^2 + (z_1 - z_\mathrm{u})^2]^{1/2}.$$

Having given one more SV distance r_2 and position (x_2, y_2, z_2), the receiver knows it must be on the intersection of two such spheres which is a circle in the plane of the intersection satisfying

$$r_1 = [(x_1 - x_\mathrm{u})^2 + (y_1 - y_\mathrm{u})^2 + (z_1 - z_\mathrm{u})^2]^{1/2},$$

$$r_2 = [(x_2 - x_\mathrm{u})^2 + (y_2 - y_\mathrm{u})^2 + (z_2 - z_\mathrm{u})^2]^{1/2}.$$

Third SV at distance r_3 and position (x_3, y_3, z_3) further reduces the possible user coordinates to at most two points in the space satisfying the equations

$$r_1 = [(x_1 - x_\mathrm{u})^2 + (y_1 - y_\mathrm{u})^2 + (z_1 - z_\mathrm{u})^2]^{1/2},$$

$$r_2 = [(x_2 - x_\mathrm{u})^2 + (y_2 - y_\mathrm{u})^2 + (z_2 - z_\mathrm{u})^2]^{1/2},$$

$$r_3 = [(x_3 - x_\mathrm{u})^2 + (y_3 - y_\mathrm{u})^2 + (z_3 - z_\mathrm{u})^2]^{1/2}.$$

This remaining uncertainty is of no problem most of the time, since one of these points can be usually further eliminated basing on a contextual information, such as the receiver is supposed to be on the Earth surface and the second point is then too high to be true.

In GNSS, however, we cannot measure the distances directly. Instead, we are looking at the delay in between the SIS transmission from the respective SV (t_send) and its reception by the receiver (t_recv). Since the speed of light denoted c is finite and invariant, there is a known relation holding in free space

$$r_i = (t_{i\text{,recv}} - t_{i\text{,send}})c.$$

The receiver's clock is, however, not perfectly synchronized with the GNSS master time. Therefore, we are not getting the true ranges r_i this way, but a sort of pseudoranges denoted ρ_i instead. The aforementioned equations are then in the form

$$\rho_i = (t_{i\text{,recv}} + \delta - t_{i\text{,send}})c = [(x_i - x_\mathrm{u})^2 + (y_i - y_\mathrm{u})^2 + (z_i - z_\mathrm{u})^2]^{1/2} + \Delta_\mathrm{u},\ 1 \le i \le N,$$

where Δ_u is the value of the distance error, introduced by the fact we are observing pseudoranges instead of the direct true ranges, and N is the number of equations.

Seeing the positioning problem [37] this way, we can understand that instead of the three unknowns, we have to recover also the fourth variable denoted Δ_u. Assuming there is no generally usable dependence among these variables, it follows that instead of three equations we need at least four of them. That means $N \ge 4$, which in turn means we have to observe SIS for at least four satellites.

The good point is, however, that this way we are also implicitly getting the δ correction of the local receiver clock with respect to the GNSS master time with a precision that is approaching the atomic time [25] of the space segment (further supervised by the ground segment). Despite seeming as a by-product, this is so valuable output that in many applications the particular GNSS service is solely used for the precise timing purpose. In particular, the C/A service of GPS is the most widely used one here [8].

According to the way we observe $t_{i,\mathrm{send}}$ it is important to emphasize GNSS is generally not a pulsed system. The time-sent information is continuously embedded into the SIS being generated by the respective SV [1], [8], [15], [24], [27] and we are recovering the $t_{i,\mathrm{send}}$ values by continuously observing the respective SIS and maintaining a delayed replica clock for each SV. This procedure is known as a satellite tracking and it is directly linked to the number of receiver channels available. The receiver can track in parallel as many satellites as many channels it has.

Note also that we can in theory lower the number of equations, and so the satellites needed to be tracked, by knowing some of those PVT variables a priori. For instance, if we know the receiver position from another geoinformatical source, we can recover the precise timing signal by just a single channel receiver and at least one strong SV in the sky view [8].

A simple illustration of the whole tracking process leading to position, velocity, and time estimation is given in Figure 1. Please bear on mind the process of PVT computation [37] has been simplified here considerably to exemplify the main principle. In practice, there are further SIS distortions the receiver has to cope with, namely gravitational field perturbations affecting satellite orbits [25], ionospheric and tropospheric phase shifts, local neighbourhood reflections etc. [24], [28]. Lot of the atmospheric effects, for instance, are then much easier to correct provided we can observe their impact on signals being transmitted on at least two different frequencies (not just L1 only). This is the reason why the modernised GNSS services usually seek for an allocation of multiple frequencies for their SIS and declare this as one of their key benefits [1], [12], [28].

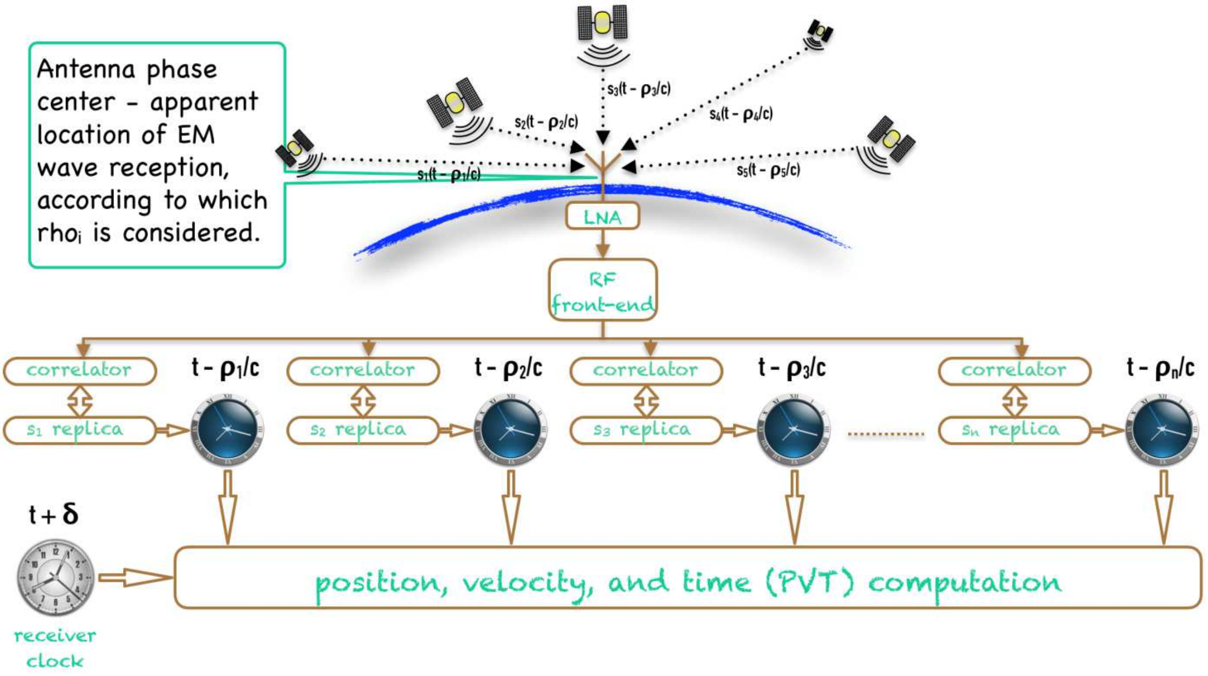


Figure 1: GNSS tracking loop showing the role of replica clocks derived from the satellite signals.

In Figure 1, there also is the antenna phase center indicated [1], [8], [20], [23]. For those who ever wondered of what place is the position, velocity, and time relevant to, this is the answer. It is an apparent spot at or close to the antenna where the incident electromagnetic wave seems to get converted to the electrical antenna current. So, for example, if we would have an antenna at the roof top connected with a processing device several floors below the roof, the PVT computation would be still valid with respect to a spot at the antenna on the roof. Antennas designed for precise PVT measurements have their phase center(s) (as it may vary with the angle of the wave arrival) marked in their data sheets.

3 GPS L1 C/A Vulnerabilities and Attacks

On one hand, we can hardly call the GPS L1 C/A service vulnerabilities a "discovery", since there was never any kind of a robust protection seriously considered [1], [24]. Perhaps, there was some hope that the attacks would not be easy to carry out [40], but that was apparently all. For those who required a better protection (and were also among the selected ones), there was the military-grade service developed. Unfortunately, practically all GNSS, including the modernised GPS services [1], continue with this "cold war" viewpoint, which is an obvious misconception nowadays [10].

The main reason why we shall also consider the GNSS civil services protection seriously, now can be called a software-defined radio (SDR) [13], [19], [36]. Generally speaking, this is a universal radioelectronic device that allows almost complete redesign of its signal processing parts by just loading the appropriate firmware. When it comes to hacking, we can tell that what used to be a question of deep radio understanding [30] together with a practical HW skill [33], is now becoming a question of a few off-the-shelf components [31], [36], [43], a basic course in DSP (Digital Signal Processing) [21], [22], and widespread SW frameworks [13], [36].

It is important to understand that the core of a contemporary radio attack is not the universal HW itself [49], but the software realizing the signal processing flow-graph [17], [50]. Being written just once, this piece of software can be then shared, downloaded, and executed all around the world very easily. Just like any other so-called exploit code, as we know them from computer security everyday practice very well. The practical experiments noted below fully support this argument [31].

So, what can the attacker do, having been equipped with the right SDR and having loaded the right exploit SW into it? The basic classification of GNSS attacks is as follows [10]:

- jamming,
- meaconing (record & replay),
- spoofing.

Jamming attack is basically a DoS (Denial of Service) that is illustrated in Figure 2. It seems like a trivial attack, but it can be really demanding, provided the attacker aims for a broad and reliable effect without being easily identified due to a big power transmission. Not surprisingly, several involved jamming strategies have been developed [10], [15].

Due to the ubiquitous implementation of GNSS receivers in our modern IT infrastructure, the jamming attack can have rather devastating consequences, nowadays. It seems like the developers had the assumption that unless there is the end of the world, the GPS L1 C/A service in particular must be always available. Several interesting real-life incidents are assembled in [10] illustrating the general public services practically collapsed after the GPS signal was jammed. This shall be fully reflected in our future risk analyses.

The jamming threat is even more emphasized by the fact it seems to be a radio-based attack on GNSS that can hardly be prevented by a cryptographic signal processing only [10] which otherwise is clearly a preferred approach (cf. the discussion below). Therefore, even in the case of military-grade signals, there is always a place for purely radio-oriented countermeasures, including the use of smart antennas [20], [23].

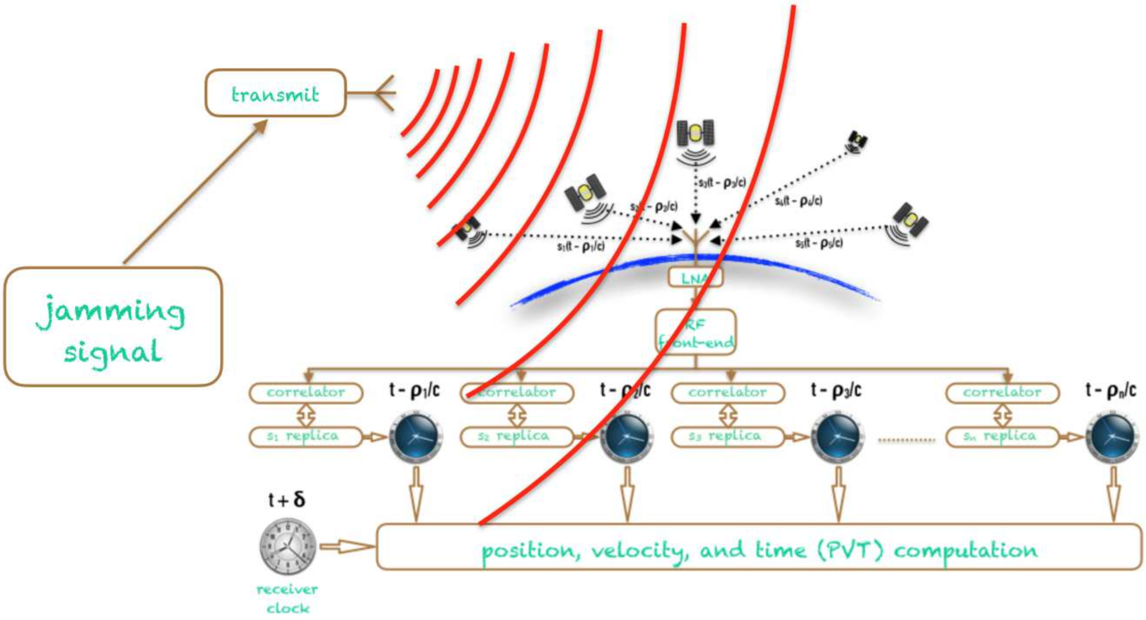


Figure 2: Denial of Service by the original satellite signal jamming.

Another attack on GNSS is the meaconing illustrated in Figure 3, which is basically another name for the record & replay attack. The record phase starts similarly to an ordinary receiver signal processing flow-graph. Instead of processing the RF signal samples towards getting PVT, however, the attacker just only records these samples to a suitable medium. During the replay phase, they use the captured samples as a source for the quadrature modulation [22] of a regenerated L1 carrier that is in turn transmitted towards the victim's receiver. That receiver then can see the fake signal like it was coming from the original satellites.

A simple variant of the meaconing attack can, instead of offline storing, transfer the original samples over e.g. internet to some other place and replay them in almost real time there. This way, we can instantly "move" the remote sensor to the place of our receiving antenna.

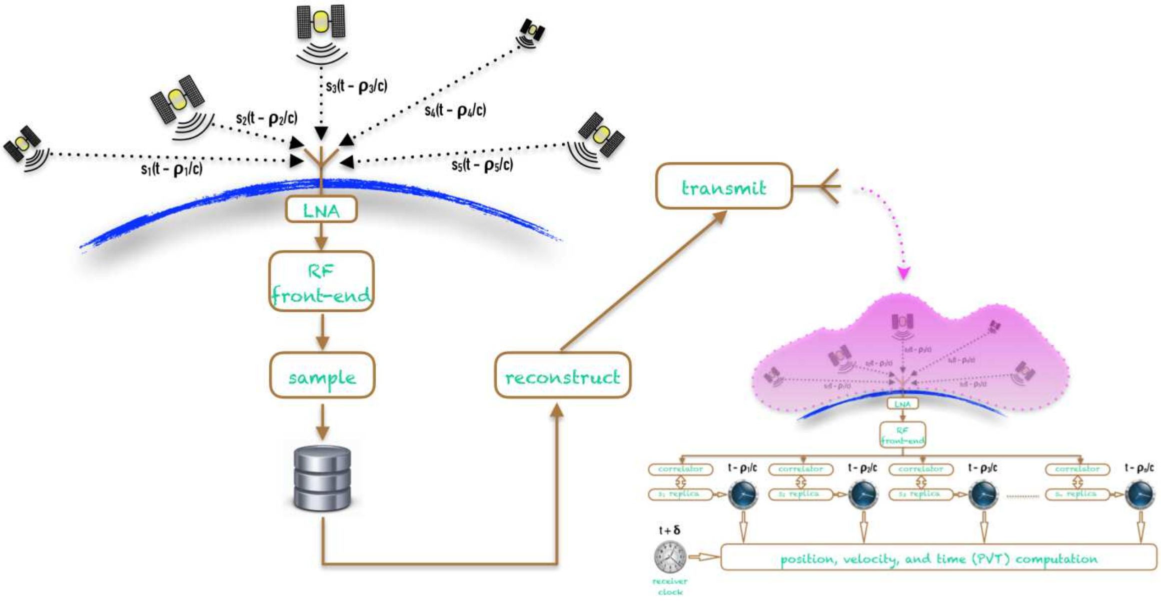


Figure 3: Meaconing (record & replay) attack on GNSS.

The third kind of attacks on GNSS, called spoofing, is illustrated in Figure 4. This time the fake signal is not based on a simple recording, but it is created synthetically instead. In particular, Figure 4 shows a robust way on how to generate the spoofing signal from scratch that is able to naturally mimic the physical properties of the original signal, namely a smooth Doppler shift [24], [28]. We use the PVT values as an input and according to the equations mentioned in part two we derive the expected values of the replica clocks dials together with their expected tick speed. Then we use the dial values as phase variables for the fake signal synthesis that is then modulated on the L1 carrier and transmitted.

It shall be understood that both meaconing and spoofing attacks can be in fact carried out by practically any GNSS radio debugging tool. Actually, it took a surprisingly long time till people recognized that these development simulators and testers, being equipped with a simple output power amplifier and antenna, can be also used for a real attack [40]. On the other hand, there are, however, reasons on why to develop a new line of these tools in parallel. The first is these "superprofessional" tools seem to be often somehow overpriced for the purpose we need them, while at the same time lacking security relevant functions such as, for instance, navigation data fuzzing modules to search for receiver firmware vulnerabilities and exploits [26]. It is questionable whether these functions would otherwise be ever implemented into those professional, development-style black boxes. According to author's own experience, their manufacturers do not seem to appreciate the potential of GNSS security research too much, as, for instance, a security conference review was unfortunately (as the device was really interesting) not enough to qualify for a "free" 14-day trial test period of one of them.

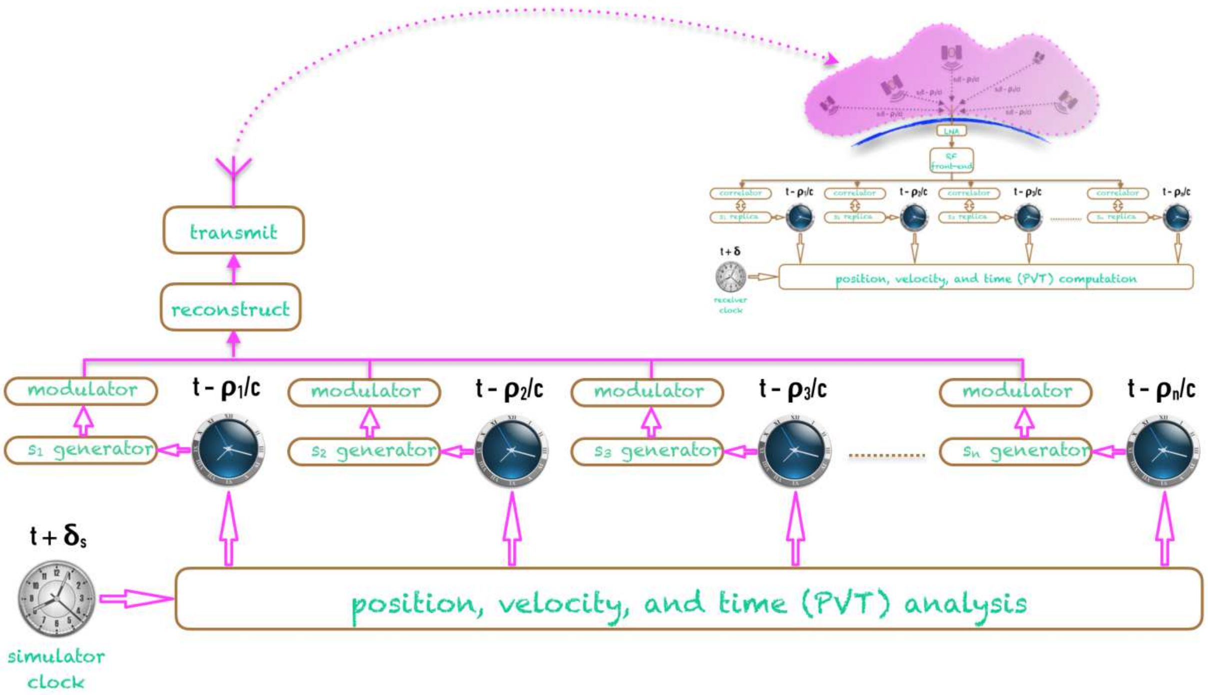


Figure 4: GNSS spoofing attack based on a synthetic signal generation by the tracking reversal process.

The simple attack classification noted here can be further subdivided based on, for instance, what are the requirements on the targeted receiver. Some of them may require a cold start, while sometimes a warm start (by remembering the last PVT and satellite positions) is enough. There are even attacks that can hijack the receiver while it is actively tracking the original SIS [17]. This usually requires the spoofing devices to also track and estimate the original SIS and carefully blend its own fake signal with it. We can take the SCER (Security Code Estimation and Replay Attack) as an example [10], [51]. Another subdivision can be based on the amount of independent transmitters the attacker can use to improve the spatial diversity of the fake signal and so to bypass spatial diversity countermeasures [38].

4 Practical Experiments

The low-cost, simplistic meaconing and spoofing attacks were successfully verified on GPS L1 C/A in a rather simplistic setup. Details allowing objective replication of these experiments are given in [31]. As the SDR platform, we have used USRP N210 [49] with the UBX-40 daughterboard [48]. For the meaconing experiment, we also used a custom based front-end in between the active GPS antenna and the UBX-40 daughterboard that was, however, based solely on off-the-shelf components [43]. Its purpose was to carefully amplify the very weak GNSS signals without introducing too much additive noise [11] over the level inevitably captured by the antenna [20].

To verify the fake signal, we used a state-of-the-art GNSS receiver uBlox NEO-M8N installed as a USB component peripheral on a development breakout board, which is still an off-the-shelf product [44], [45]. After a proper attenuation and DC current blocking in between UBX-40 output port and the GNSS receiver input, we used a direct coaxial cable connection in between these two components. This way we could avoid any direct fake signal emanation, which is clearly more than desirable. The detailed report [31] also shows how to really transmit the forged signal, but this is just for the sake of completeness and we can by no means recommend this.

Data from the uBlox receiver were processed and presented via uCenter [46], [47], which is a free of charge development SW running on practically any reasonable platform equipped with MS Windows Vista or higher. uCenter dashboard example is shown in Figure 5 below.

Despite there being professional (and still somehow overpriced) SW tools for GNSS/GPS signal manipulation, we stayed with a very modest setup here to exemplify our argument of the massive attacks threat accelerated by the SDR phenomenon. In particular, for the meaconing (record & replay) attack experiment, we employed the `rx_samples_to_file` and `tx_samples_from_file` example codes that are installed together with the N210 device driver (USRP Hardware Driver – UHD) source tree. Details of their invocation are given in [31].

For the preparation of the fake signal used in the spoofing attack, we used a simple but very handy open source project GPS-SDR-SIM by Takuji Ebinuma et alia [50]. The core is a C module that is easy to compile almost everywhere, where we have the OpenMP framework for shared-memory parallel programming (open source code, not specific to GNSS). This utility prepares the simulated GPS L1 C/A signal as an offline file with its quadrature (complex) envelope samples [22]. The final transmission step is then, therefore, de facto the same as in the meaconing attack experiment. Since it uses a different data precision, however, `tx_samples_from_file` should be edited slightly. This would be really easy, but since GPS-SDR-SIM comes with its own "player" code [50], we have stayed with that. In particular, this player is a Python code that creates a simple flow-graph based on the GNU Radio framework [13]. This is again an open source project serving as the platform of first choice for many academic projects. All the details necessary for an independent verification of these experiments are given in [31].

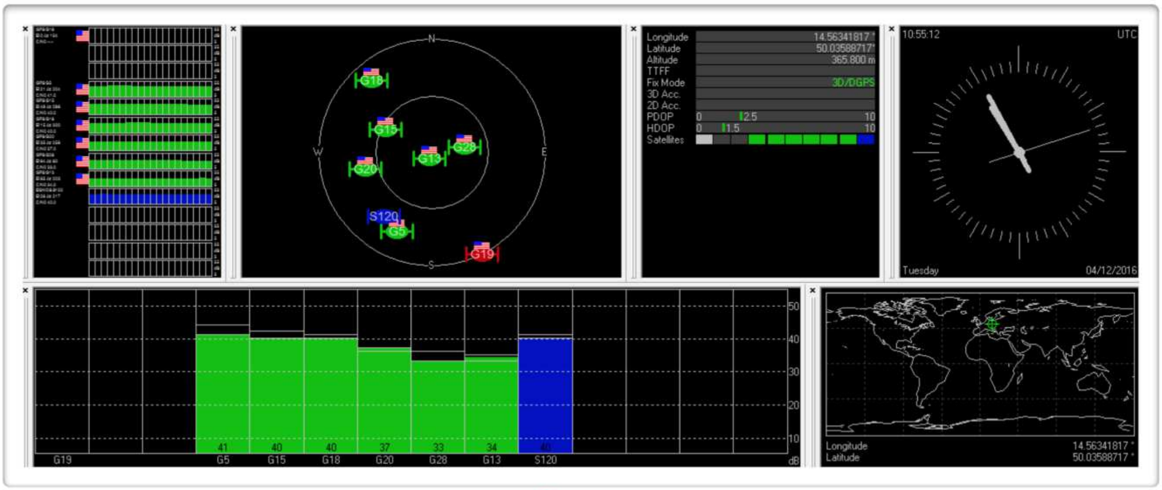
5 SBAS to the Rescue?

Even without intentional attacks, the civil GPS C/A service is not as reliable as necessary to be directly and solely usable in safety-critical applications such as flight security. Services known as Satellite-Based Augmentation System [1], [28] have been devised to constantly monitor the integrity of C/A and report its safety status together with possible differential GPS (DGPS) correction parameters via geostationary satellites. To be easily accessible for simple receivers, the signals coming from these auxiliary satellites are also transmitted on L1 with almost the same CDMA [15] scheme. They differ in the data modulation speed and payload [1], but this is relatively easy to handle in the receiver.

In particular, the Europe is practically covered with SBAS named EGNOS (European Geostationary Navigation Overlay System) that is also operated by GSA [28]. Contrary to Galileo, however, this system is in the production grade version, now, being enjoyed by many civil pilots landing on the old continent.

Provided SBAS is such a wonderful safety system built around the C/A service, it is natural to ask whether it is also a countermeasure against the attacks discussed above. To understand the answer, it is important to see that SBAS provides a safety assurance based on SIS observed by its reference monitoring stations. It cannot check the local reception of the (possibly) fake signal being received by the individual users. It is oriented to measure and evaluate the quality of the original SIS as it is being transmitted by the original satellites globally. Therefore, a reception of local invalid signals transmitted by anybody else is not addressed by today’s SBAS [1]. So, the answer is unfortunately no, it cannot prevent the attacks discussed here.

To practically demonstrate this, we present the Figure 5 that shows a dashboard of the reference GNSS receiver used in our meaconing attack experiment. Since there was also an EGNOS satellite in the sky view during the original SIS recording phase and since the EGNOS (as any other SBAS) produces RF signal that is highly compatible with L1 C/A, that augmentation signal had been recorded as well. In Figure 5, we can see the EGNOS channel was successfully recognized and employed by the receiver during the replay phase of our demo attack. So, as a kind of by-product, we also have a working example of the EGNOS replay attack here. Actually, we see a classical example of false sense of security as the receiver dashboard shows the EGNOS signal has been applied to get a DGPS (Differential GPS) precision grade [1], [24], [37] thereby suggesting everything should be more than fine.



Longitude 14.56341817 °
Latitude 50.03588717 °
Altitude 365.800 m
TTFF
Fix Mode 3D/DGPS
3D Acc.
2D Acc.
PDOP 0 2.5 10
HDOP 0 1.5 10
Satellites

10:55:12 UTC
Tuesday 04/12/2016

Longitude 14.56341817 °
Latitude 50.03588717 °

Note we have also successfully recorded the SBAS/EGNOS signal channel PRN120 coming from Inmarsat 3F2 AOR-E. The DGPS indicator above shows this signal has already been used for a fix assurance.

Figure 5: Successful meaconing attack on GPS L1 C/A that also covers the EGNOS signal.

6 Countermeasures

Actual and comprehensive review of countermeasures can be found in monography [10] as well the references noted there, cf. also [18], [23], [38]. Despite exposing the label “interference” in its title, it also covers jamming, meaconing, and spoofing as a special kind of man-made intentional interferences. Many classifications are also noted there, however, we will use a very simple three-category scheme for our purpose here:

- countermeasures that also improve SIS definition,
- methods based on receiver radio signal processing only,
- PVT postprocessing verifications.

Note we are still talking about the civil GNSS services, as these are the only ones that are granted for general IT companies. If we understand cryptography as the science bringing information protection constructions based on rigorous mathematical or physical arguments, then we have to clearly admit that cryptography shall be applied to SIS processing if we really want to get a robust GNSS security [10], [14], [51], [52]. That in turns means the first countermeasures category is the one that shall be preferred, so we pay a higher attention to it here. Truly, this is the case of e.g. military GPS signals. Unfortunately, the only “protection” explicitly noted, for instance, in the actual definition of the emerging Galileo Open Service SIS [12], besides a convolutional forward error correction (FEC) code, is a simple CRC. Interestingly, the sole inclusion of FEC with a binary interleaving [12] can actually work as a “spoofing obstacle”, but this is more or less a by-product of the receiver hijacking problem [17] and it is clearly no protection against direct meaconing or synthetic signal spoofing. On the other hand, the interleaved FEC can also potentially even help the attacker with the SCER attack [10], provided the detector [52] was not tailored properly to its presence by reflecting its impact on the processing flow-graph together with the residual attacker’s security code estimator advantage [51]. That said FEC incorporation is a tricky part deserving certain attention.

As we have the open signals without any cryptographic protection, now (hopefully, this will change), we have to focus on the remaining two categories. Anyway, it should be emphasized that even with a cryptographic protection in place, its verification needs to be understood as a probabilistic problem [51] to be able to defeat SCER-like attacks [10]. Instead of a simple yes-no algebraic algorithm, we need a statistical signal detector that allows us to distinguish between the original and fake signals [52]. It follows we still need some radio signal processing techniques to help us. This is especially true when coping with jamming. A question may arise on how far the cryptography is useful at all, provided it is hardly enough in itself. In this light, we shall see the cryptographic protection of SIS as a significant enabling factor that allows us to design and use much more powerful attack detection techniques than what remains when no cryptography was employed.

To that end, the cryptographic protection goals shall be precisely stated and the cryptosystem shall be then carefully tailored to the RF signal processing needs. We emphasize again, this is not only to achieve certain efficiency, but mainly to fulfil those security goals at all. As an example, we may consider the well-known method of “codeless” tracking that has been developed to gain some information from the military GPS signal that is in turn used to get rid of the effects like the ionospheric distortion [42]. At least, this is how it is known in the satellite navigation community. Practically the same article could had been, however, issued in proceedings on cryptology as a successful partial cryptanalysis of the military GPS scheme! This illustrates nicely that designing a broadcast radio signal protection is not as easy as solving a simple general data encryption exercise. We need to work with the low-level signal properties to fully understand on how to really achieve our security goals [51].

Anyway, the radio signal processing tries to identify both intentional as well as unintentional interferences basing on statistical properties of the RF samples being processed [10], [15], [52]. As the computational power of receiver controllers grows up, more and more involved methods can be used. Unfortunately, regarding the intentional interference (jamming, meaconing, and spoofing), no provable or at least universal purely radioelectronical countermeasure has been found and it does not seem there is something like that on the horizon [10].

PVT postprocessing addresses these problems from a different perspective. Instead of trying to get error-free receiver output, it combines the respective contributions of several independent PVT measurement technologies and devices. This way we can, for instance, combine the GPS service with GLONASS (or Galileo in the near future) or with inertial sensors like accelerometers, gyroscopes, barometers, etc., to be able to mutually cross-check the individual results. Mathematically speaking, this combination is typically based on the statistical Kalman filtering theory [27], [37].

In summary, the only rigorous approach to meaconing and spoofing, however, remains via cryptography. Anything else can be seen like a cat-and-mouse game or the virus-antivirus fight. Once we take the current best spoofer, for instance, we can look at its signal artefacts and design our detector to go especially after them [10]. The spoofing device author, however, can learn this, tweak their device a little bit and our detector falls short. And the whole story repeats again, and again…

7 Conclusion

The possibility of easy jamming, meaconing, and spoofing attacks, thereby of the free manipulation with the receiver apparent position, velocity, and time, is a long overlooked problem of practically any open GNSS service. The most popular target seems to be the widespread GPS L1 C/A signal. Any other unprotected GNSS service can, however, follow soon, including the emerging Galileo Open Service [12]. Furthermore, this is all getting accelerated by the software-defined radio paradigm that has successfully established its place in many hacker arsenal toolboxes [16], [41]. Since SDR allows a variant of the “write once, run anywhere” approach, we can expect even so-called “script kiddies” will be soon able to play with GNSS attacks all around the world.

One may argue this is not a discovery, since a robust protection of the civil services was never seriously considered. However, this is right the point that needs to be radically revised, now as the threat model has clearly changed. We can see the public GNSS hacking and security research papers mainly as an appeal on designers of these services, as well as a clear warning to its application engineers to be careful with what they trust to.

As the cold war era is over and the civil sector plays important role in the critical infrastructure today, it is right the civil service that deserves a robust security protection, now. Designing a new service such as the Galileo Open Service without any solid cryptographic protection would be actually bringing up a new system that is by-design broken. We can hardly agree with the position like that civil applications do not deserve an accessible and robust protection of their GNSS signals. On one hand, it is understood that services such as Galileo Public Regulated Service (PRS) should exist. Their role shall be, however, mainly seen in the possibility to temporarily (!) restrict the access to the GNSS service in time of war, severe terrorist attacks, dramatic regional destabilization, etc. They can also broadcast some additional data that can be used for investigation, international police coordination, natural disaster recovery, and so on. Their role is, however, by no means in distinguishing the usable (i.e. protected and safe) GNSS services from the unusable (i.e. unprotected and unsafe) ones.

Interestingly, in the time of finishing this manuscript, GSA announced it will provide the Navigation Message Authentication (NMA) protection [10], [52] right with the Galileo Open Service signal as of 2018 [53]. Cryptographically speaking, NMA allows the navigation data origin authentication by a form of a digital signature or message authentication code (MAC). As any such protection embedded into a broadcast data, in itself, it can only protect against a very harsh meaconing or a totally synthetic signal spoofing, but it usually fails against a bit more clever variants of these attacks. To be robust and useful, it needs to be made in a form of a statistical signal detector allowing to distinguish also the relevant SCER variants [10], [51], [52], as we have discussed it in part six above. This all needs to be detailed in the new SIS definition to fairly get all the receiver manufacturers to the same ground, as we clearly cannot assume all users will check the particular implementations by themselves to see whether they are vulnerable or not. Since it is actually less than two years to the planned start of this new Open Service version and the SIS definition remains practically intact [12], this all brings more questions than answers. This is somehow unfortunate, as such an unprecedented (with respect to the rest of the GNSS world) step would otherwise deserve clear ovations.

Acknowledgement

The author is grateful to Jiří Buček (FIT, Czech Technical University in Prague), Tomáš Jabůrek and Radek Komanický (Raiffeisen Bank), and Martin Opava (truconneXion). They all have contributed with a non-trivial effort to make the GNSS security research possible and pleasant.

References

[1] Betz, J.-W.: *Engineering Satellite-Based Navigation and Timing: Global Navigation Satellite Systems, Signals, and Receivers*, IEEE Press, John Wiley & Sons, 2016.

[2] Bonebrake C. and O'Neil, L.-R.: Attacks on GPS Time Reliability, *IEEE Security & Privacy*, May/June 2014, pp. 82-84, 2014.

[3] Borre, K., Akos, D.-M., Bertelsen, N., Rinder, P., Jensen, S.-H.: *A Software-Defined GPS and Galileo Receiver A Single-Frequency Approach (Applied and Numerical Harmonic Analysis Series)*, Birkhauser Boston, 2007.

[4] Chen, J., Zhang, S., Wang, H., and Zhang, X.: Practicing a record-and-replay system on USRP, *In Proc. of the second workshop on Software radio implementation forum*, pp. 61-64. ACM, 2013.

[5] Di, R.: *A USRP-Based Flexible GNSS Signal Recording and Playback System: Performance Evaluation and Study*, M.Sc. Thesis, Miami University, Oxford, Ohio, 2013.

[6] Di, R., Peng, S., Taylor, S., and Morton, Y.: A USRP-Based GNSS and Interference Signal Generator and Playback System, *In Position Location and Navigation Symposium (PLANS) 2012*, pp. 470-478, IEEE, 2012.

[7] Diggelen, van F.: A-GPS: *Assisted GPS, GNSS, and SBAS (GNSS Technology and Applications Series)*, First Edition, Artech House, 2009.

[8] Doberstein, D.: *Fundamentals of GPS Receivers – A Hardware Approach*, Springer, 2011.

[9] Dong, L.: *IF GPS Signal Simulator Development and Verification*, M.Sc. Thesis, University of Calgary, Alberta, 2003.

[10] Dovis, F. (Ed.): *Gnss Interference, Threats, and Countermeasures (Gnss Technology and Applications Series)*, Artech House Publishers, 2015.

[11] Etten, van W.-C.: *Introduction to Random Signals and Noise*, First Edition, Wiley, 2005.

[12] *European GNSS (Galileo) Open Service (OS), Signal In Space (SIS) Interface Control Document (ICD)*, Europen Union, November, 2015.

[13] Grayver, E.: *Implementing Software Defined Radio*, Springer, 2012.

[14] Hernandez, I.-G., Rodriguez, I., Tobias, G., Calle, J.-D., Carbonell, E., Seco-Granados, G., Simon, J., and Blasi, R.: Galileo Commercial Service – Testing GNSS High Accuracy and Authentication, *Inside GNSS*, January/February 2015, pp. 38-48, 2015.

[15] Holmes, J.-K.: *Spread Spectrum Systems for GNSS and Wireless Communications (GNSS Technology and Applications Series)*, Artech House, 2007.

[16] Huang, L. and Yang, Q.: GPS Spoofing – Low-cost GPS Simulator, *DEF CON 23*, Las Vegas, August 6th – 9th, 2015.

[17] Humphreys, T.-E., Ledvina, B.-M., Psiaki, M.-L., O'Hanlon, W.-O., and Kintner, P.-M., Jr.: Assessing the Spoofing Threat: Development of a Portable GPS Civilian Spoofer, *In Proc. of the ION GNSS international technical meeting of the satellite division*, vol. 55, p. 56, 2008.

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

[19] Johnson, C.-R., Jr., Sethares, W.-A., and Klein, A.-G.: *Software Receiver Design – Build Your Own Digital Communications System in Five Easy Steps*, Cambridge University Press, 2011.

[20] Kraus, J.-D. and Marhefka, R.-J.: *Antennas For All Applications*, Third Edition, McGraw-Hill, 2003.

[21] Lathi, B.-P. and Green, R.-A.: *Essentials of Digital Signal Processing*, Cambridge University Press, 2014.

[22] Lyons, R.-G.: *Understanding Digital Signal Processing*, Third Edition, Prentice Hall, 2011.

[23] McMilin, E.-B., Chen, Y.-H., De Lorenzo, D.-S., Akos, D.-M., Walter, T.-F., Lee, T.-H., Enge, P.-K.: Single Antenna, Dual Use: Theory and Field Trial Results for Aerial Applications of Anti-Jam and Spoof Detection, *Inside GNSS*, September/October 2015, pp. 40-53, 2015.

[24] Misra, P. and Enge, P.: *Global Positioning System – Signals, Measurements, and Performance*, Revised Second Edition, Ganga-Jamuna Press, 2012.

[25] Montenbruck, O. and Gill, E.: *Satellite Orbits: Models, Methods and Applications*, HAR/CDR edition, Springer, 2011.

[26] Nighswander, T., Ledvina, B., Diamond, J., Brumley, R., and Brumley, D.: GPS Software Attacks, *In Proc. of the 2012 ACM conference on Computer and communications security*, pp. 450-461, ACM, 2012.

[27] Noureldin, A., Karamat, T.-B., Georgy, J.: *Fundamentals of Inertial Navigation, Satellite-based Positioning and their Integration*, Springer, 2013.

[28] Nurmi, J., Lohan, E.-S., Sand, S., and Hurskainen, H. (Eds): *GALILEO Positioning Technology*, Springer, 2015.

[29] Perring, A., Canetti, R., Tygar, J.-D., and Song, D.: The TESLA Broadcast Authentication Protocol, *In CryptoBytes*, 5:2, Summer/Fall 2002, pp. 2-13, 2002.

[30] Razavi, B.: *RF Microelectronics*, Second Edition, Prentice Hall, 2011.

[31] Rosa, T.: *GNSS/GPS Radio Hacking – From Beautiful Equations to Serious Threats*, In QuBit Conference 2016, Prague, April 12th – 14th, 2016, [http://crypto.hyperlink.cz/files/rosa-qubit-2016.pdf].

[32] Rugamer, A., Stahl, M., Lukcin, I., Rohmer, G.: Privacy Protected Localization and Authentication of Georeferenced Measurements using Galileo PRS, *In Position, Location and Navigation Symposium-PLANS 2014*, 2014 IEEE/ION, pp. 478-486, IEEE, 2014.

[33] Rutledge, D.: *The Electronics of Radio*, Cambridge University Press, 1999.

[34] Shepard, D.-P. and Humphreys, T.-E.: Characterization of Receiver Response to Spoofing Attack, *In Proc. of the 24th International Technical Meeting of The Satellite Division of the Institute of Navigation*, p. 2608, 2011.

[35] Shepard, D.-P., Humphreys, T.-E., and Fansler, A.-A.: Evaluation of the Vulnerability of Phasor Measurement Units to GPS Spoofing Attacks, *International Journal of Critical Infrastructure Protection 5*, no. 3, pp. 146-153, 2012.

[36] Stewart, R.-W., Barlee, K.-W., and, Atkinson, D.-S.-W.: *Software Defined Radio using MATLAB & Simulink and the RTL-SDR*, Strathclyde Academic Media, 2015.

[37] Strang, G. and Borre, K.: *Algorithms for Global Positioning*, Wellesley-Cambridge Press, 2012.

[38] Tippenhauer, N.-O., Poepper, C., Rasmussen, K.-B., and Capkun, S.: On the requirements for successful GPS spoofing attacks, *In Proc. of the 18th ACM conference on Computer and communications security*, pp. 75-86. ACM, 2011.

[39] Tsui, J.-B.-Y.: *Fundamentals of Global Positioning System Receivers: A Software Approach, Second Edition*, Wiley-Interscience, 2005.

[40] John A. Volpe National Transportation Systems Center: *Vulnerability Assessment of the Transportation Infrastructure Relying on the Global Positioning System*, Final Report for the Office of the Assistant Secretary for Transportation Policy, U.S. Department of Transportation, August 29, 2001.

[41] Wang, K., Chen, S., and Pan, A.: Time and Position Spoofing with Open Source Projects, *BlackHat EU 2015*, November 12th – 13th, 2015.

[42] Woo, K.-T.: Optimum Semicodeless Carrier-Phase Tracking of L2, *In Proc. of the 1999 International Technical Meeting of the Satellite Division of the Institute of Navigation*, also appeared in *Navigation 47*, no. 2, pp. 82-99, 2000.

[43] Mini-Circuits, RF/IF Microwave Components DC to 40 GHz, http://www.minicircuits.com.

[44] uBlox GNSS module overview, UBX-14000426-R06, uBlox, http://www.u-blox.com.

[45] NEO-M8, u-blox M8 concurrent GNSS modules, Data Sheet, UBX-13003366-R10, uBlox, http://www.u-blox.com.

[46] u-center, GNSS evaluation software, UBX-13003929-R06, uBlox, http://www.u-blox.com.

[47] u-center, GNSS evaluation software for Windows, User Guide, UBX-13005250-R10, uBlox, http://www.u-blox.com.

[48] UBX Daughterboard, Data Sheet, Ettus Research, http://www.ettus.com

[49] USRP N210, Data Sheet, Ettus Research, http://www.ettus.com

[50] GPS-SDR-SIM: Software-Defined GPS Signal Simulator, https://github.com/osqzss/gps-sdr-sim

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

[52] Wesson, K., Rothlisberger, M., and Humphreys, T.-E.: Practical cryptographic civil GPS signal authentication, *Navigation*, 59(3), pp. 177-193, 2012.

[53] GSA: Assuring authentication for all, April 18th, 2016, http://www.gsa.europa.eu/news/assuring-authentication-all