Keeping the Spoofs Out

Signal Authentication Services for Future GNSS

GNSS signal authentication is a requirement for a number of applications. The article reviews the navigation message authentication concept and its limitations and proposes a new authentication scheme based on signal authentication sequences that can be integrated in GNSS. The method works on systems that provide an open and encrypted service on the same frequency and would require minimum changes to the system.

OSCAR POZZOBON
QASCOM S.r.l

The demand for techniques capable of authenticating the GNSS signals and detecting simulation attacks (spoofing) has increased exponentially in the last years, mainly targeted to financial and safety critical applications.

Associated proposals and developments addressing these issues focused on two different approaches: user segment authentication services that leveraged existing services in order to detect signal spoofing and that integrated signal authentication services into the GNSS system itself.

Although the first approach can work with existing GNSS systems and those under development that do not provide a signal authentication services — such as the GPS C/A-code and Galileo E1B signals — the latter approach requires a new system design and/or modification of existing system architecture.

This article focuses on the integration of authentication services into future GNSSes, first explaining the architecture design for the various components of ground, space, and user segments. The article concludes with a discussion of the anticipated performance of the proposed authentication scheme and a comparison of different deployment architectures.

Authentication: Signals or Messages?

Our discussion begins with the previously proposed navigation message authentication (NMA) option described in an article by C. Wullems *et alia* and the new signal authentication sequences (SAS) scheme proposed in the paper by O. Pozzobon *et alia* (2), both of which are listed in the Additional Resources section near the end of this article. NMA refers to the cryptographic authentication of the messages only, while SAS refers to the authentication of the encrypted signal through the release of short encrypted sequences embedded in the navigation data. SAS security measures assume that the encrypted signal cannot be predicted.

Spreading code encryption (SCE) is the preferred option to limit access to a GNSS signal and, therefore, to the system's positioning and time functions. However, if the only objective of a service is to provide signal authentication (robustness against signal spoofing), NMA and SAS are preferable as they can reduce the cost of the receiver, providing full navigation access to users who have no access to the authentication infrastructure.

The first attempt to integrate an authentication mechanism for open signals in GNSS was introduced by L. Scott in a 2003 paper (see Additional Resources for full citation). Scott based his concept on secret spreading sequences, called spread spectrum security codes (SSSCs), that were modulated in the signal for 10 milliseconds every 1 second of modulation with a known spreading sequence. SSSCs are transmitted in the navigation messages and used for correlation with the received signal in order to verify the authenticity. In his proposal Scott also outlined a scheme for authenticating the data.

One limitation of such an approach is the need to modify an existing modulation scheme, which has significant consequences involving alterations in

the system infrastructure. Furthermore, introducing noise (the receiver can't track the code during the 10 milliseconds of SSSC modulation could create implications in some delay locked loop (DLL) and phase locked loop (PLL) receiver designs. A similar SSSC concept was described the following year in a paper by M.G. Kuhn.

An authentication scheme based on navigation messages only was proposed later in the paper by C. Wullems et alia and further explained in articles by G. Hein *et alia* (see Additional Resources).

NMA Schemes

Navigation data can be authenticated with cryptographic schemes such as digital signatures or message authentication codes (MACs). Previous work proposed the use of modified timed efficient stream loss-tolerant authentication (TESLA) protocols in order to reduce the message overhead and computation on the GNSS receiver.

A fundamental parameter in the design of NMA schemes is to include in the cryptographic integrity scheme at least the message transmission time reference — time of week (TOW) and week number (WN) — and the satellite ephemerides, as they are used to help determine the pseudorange. However, because the time is repeated over the weeks, leaving the ephemeris the only unpredictable information, the introduction of unpredictable information such as a secure random number is required in order to avoid so-called "replay attacks" in which a valid data transmission is maliciously or fraudulently repeated or delayed.

Figure 1 shows a hypothetical NMA scheme in which a "nonce" is introduced through a secure random number–generation function in order to increase the stochastic property of the data. In cryptography, a nonce is a value that is used only once within a specified context. For example, as described in the Galileo Open Service Signal in Space Interface Control Document (OS-SIS-ICD), Galileo F/NAV messages Page Type 1 has 26 spare bits that could be used for inserting a nonce. The nonce entropy and size are typically designed with respect to the probability that a system will experience a "brute force attack," which quantifies the likelihood that an attacker will have to guess the entire message and reuse the authentication message.

FIGURE 1 Navigation message authentication scheme

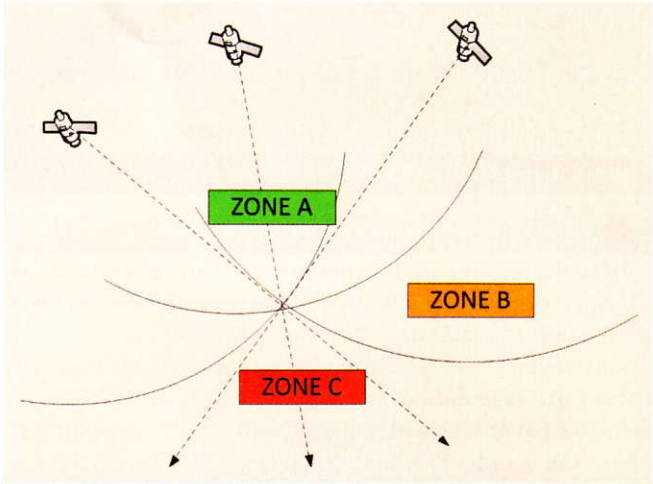

FIGURE 2 "Spoofable" areas in NMA scheme with support of a trusted clock (trusted receiver)

In such an authentication scheme, the NMA messages could be generated on the ground (where the ground control center knows the ephemerides and the keys, and can generate all the NMA for the various time slots). In this case, the satellite only needs to synchronize the insertion of the nonce in the correct subframe or page (i.e., no encryption operation on the satellites). NMA data could also be advanced in time in order to allow a faster "time-to-authentication."

An NMA scheme is vulnerable to three types of attacks:

- **All SV data replay attack.** The messages are acquired and later replayed to the receiver. This would cause a time jump easily verifiable with a trusted clock (trusted receiver) as demonstrated in the paper by O. Pozzobon *et alia* (1).
- **Selective SV delay attack.** In order avoid time jumps, an attacker could delay only the desired SVs in order to manipulate the position solution function by increasing a pseudorange.

  Figure 2 shows the "spoofable" areas in this kind of attack: Zone A is the unpredictable area, in which a spoofer could not predict the authenticated navigation messages in the signals traveling through space.

  Zone B is an area where some satellites could be spoofed, but this can be detected by anti-spoofing receiver autonomous integrity monitoring (AS-RAIM) algorithms, as the position solution obtained from different satellites would contain inconsistencies due to the positive delay contribution in the pseudoranges.

  Zone C is the only practical area of spoofing in this type of attack, as the position solution could still be projected in order to appear consistent after AS-RAIM verification.

  However, in most of the scenarios with six to eight satellites in view at low elevation, this area would be practically reduced to a vertical range, and the security function might be designed to support only some particular geometries obtained only by SVs above some degree of elevation, because high elevation SVs could deceive the system while low elevation would be detected by the AS-RAIM.

  For road applications a receiver could verify its position with a digital elevation model (DEM) in order to verify the consistency. Further research is needed in this domain to study which geometries could provide sufficient security for NMA.
- **Early bit detection attack.** NMA authentication limits the possibility of predicting authenticated naviga-

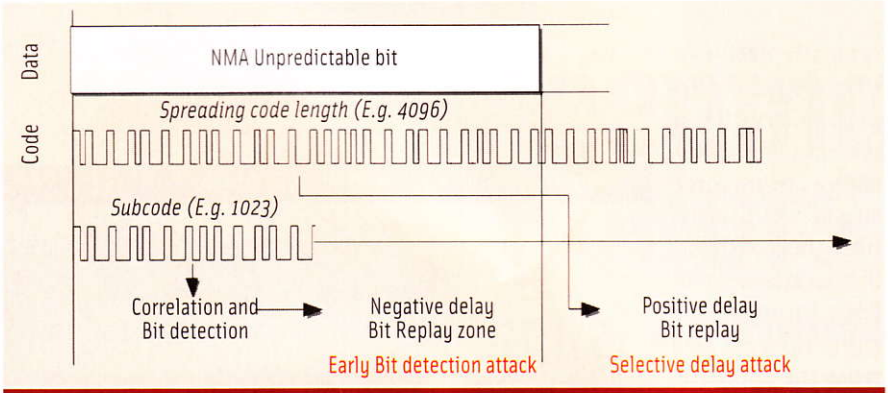

FIGURE 3 Early bit detection attack with sub-code

tion data before the NMA message is received. This means that the receiver needs to perform code and phase tracking, extract the bits, and decode the message in order to know the unpredictable data. (See **Figure 3**.)

However, GNSS satellite navigation payloads typically spread message data over long codes (e.g., 4,092 chips for Galileo E1B) or repeated sequences (20 x 1,023 chips for GPS C/A-code) in order to reduce the bit error rate. An attacker can attempt to integrate a shorter time or correlate a subset of the code in order to detect the bits and replay the data with the intended delay in order to create and transmit a negative delay in the pseudorange.

The probability of success for such attacks is dependent on the carrier-to-noise (C/N_0) ratio. The paper by D. H. Arze Pando listed in Additional Resources discusses this method of attack and provides some statistical results.

Figure 4 shows the normalized auto-correlation function (ACF) of a Galileo E1B code of 4,092 chips (green line) assuming a four megahertz filter. The figure shows graphically that a subset of the code (1,023 chips, blue line) could still be detected by a discriminator function, while a shorter code (102 chips, red line) would not be sufficient. A 1,023-chip code could be used to introduce negative delays in the pseudorange of up to three milliseconds (resulting in a 900-kilometer ranging error).

Architecture of Signal Authentication Sequences

Access control to satellite navigation signals can be implemented at the data level or at the signal modulation level. Data-level access control foresees the encryption of the messages (in whole or part). With this approach a receiver can perform the code search process and track the code delay and phase, but it cannot decode the encrypted message content. Therefore, if parameters such as transmitted time, ephemeris, and clock errors are encrypted, a position, velocity and time (PVT) solution cannot be computed.

Signal-level access control requires the encryption of the ranging codes. A direct block cipher encryption of the code is typically not a preferred design option because a time-limited code can be captured with a high-gain directional antenna.

A more robust approach is the use of a stream cipher, as the code never repeats. A stream cipher is a symmetric key cipher where plain bits are combined with a pseudorandom cipher bit stream (keystream), typically by an exclusive-or (xor) operation. In order to encrypt a pseudorandom noise (PRN) sequence, the plain sequence is modulo 2–summed with the stream cipher, resulting in an encrypted PRN sequence.

For the purpose of the concept demonstration, we assumed a BPSK signal with an open code modulated in-phase and an encrypted code modulated in quadrature. The transmitted signal (neglecting signal amplitude) will be generated as follows:

$$s(t) = \sum_{k=1}^{N}\left[O_a^{\,k}(t)\,D^k(t)\cos\left(2\pi f t_{L1}\right)\right] + \sum_{k=1}^{N}\left[O_b^{\,k}(t)\,SC^k D^k(t)\sin\left(2\pi f t_{L1}\right)\right] \quad (1)$$

where N is the number of visible satellites, $O_a^{\,k}$ and $O_b^{\,k}$ are the publicly known spreading codes for every K satellite, SC^k is the stream cipher, and D^k is the transmitted data. The same concept can be applied to a coherent adaptive subcarrier modulation (CASM) in which multiple channels are multiplied together.

One design factor of interest is the frequency of the stream cipher versus the chipping frequency of the PRN sequence. For our analysis we define this variable as a binary stream-cipher carrier (BSC):

$$BSC(m,n) \text{ or } BSC(F_{sc},F_c) \quad (2)$$

where F_{SC} is the stream cipher SC^k frequency, F_C is the non-encrypted code $O_b^{\,k}$ chipping frequency, and $m = F_C / F_{SC}$, $n = F_C / F_{ref}$, and F_{ref} = 1,023 Mcps comprise a set of terms describing the GPS C/A reference code. For example, a signal with a $O_b^{\,k}$ chipping rate of 10.23 megahertz and a stream cipher SC^k frequency of 1 megahertz will be encrypted with a *BSC(10,10)*.

The objective of our proposed system is to authenticate the open GNSS signal. The proposed security architecture can be integrated into GNSSes that use direct-sequence spread spectrum (DSSS) as their modulation technique. This approach can provide on the same frequency an open signal service, where the spreading code is publicly released, and an encrypted service where the spreading code is ciphered.

The security concept is based on the unpredictability of the encrypted PN sequence, which is assumed to be generated *a priori* by a secure function. The concept is as follows: the stream cipher SC^k is observed in a predetermined period. As shown in **Figure 5**, a portion of the binary sequence is extracted with the SAS epoch time reference for a specific time frame, e.g., SC^k [0:5000, n_0] if 5,000 chips are observed at the discrete time n_0.

The sequence is processed and transmitted in the open-service navigation messages together with an authentication and/or encryption scheme. This message is defined as the signal authentication sequence (SAS) defined as follows:

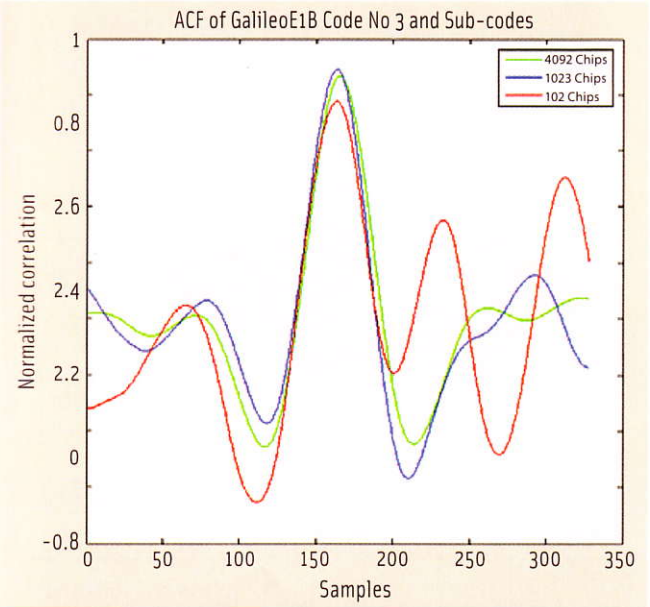

FIGURE 4 ACF of Galileo E1B code No. 3 and sub-codes used for early bit detection attack

$$SAS^k = SC^k[n_0 : n_0 + l] \quad (3)$$

where l is the length of the SAS code and n_0 is the first chip of the SC^k observation time.

During the open-service data decoding process and after authentication verification or decryption, the receiver obtains the SAS, generates the PN sequence for that specific epoch, and correlates it with samples of the encrypted code. The correlation result is fed to a security algorithm that determines the signal security state based on an estimated threshold.

Architecture overview

Figure 6 describes a high-level architecture of the SAS authentication architecture. The SAS messages are generated at the ground segment and uploaded to the satellites together with the navigation messages.

The SAS messages are transmitted in the open signal and received by the user receiver, which verifies the integrity and/or decrypts the data content. The user receiver also acquires the encrypted message at the predefined epoch and verifies the signal's authenticity with an algorithm, which we will describe later.

Ground segment

SAS messages are generated by the ground segment. The idea is that a proper dedicated service uses a key management facility (KMF) and an encrypted code generation facility (ECGF) to obtain the cipher stream in a particular epoch as defined by equation (3). The binary sequence is than formatted and packed into the messages by a message generation facility (MGF) **Figure 7** presents a block diagram of the ground segment design.

Space Segment

Signal authentication sequences are transmitted in the open-signal data messages. Considering the evolution of GNSS and

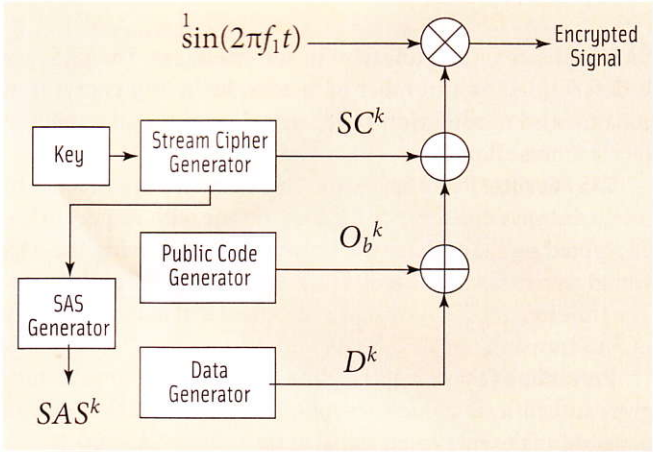

FIGURE 5 Example of signal authentication sequence (SAS) generator

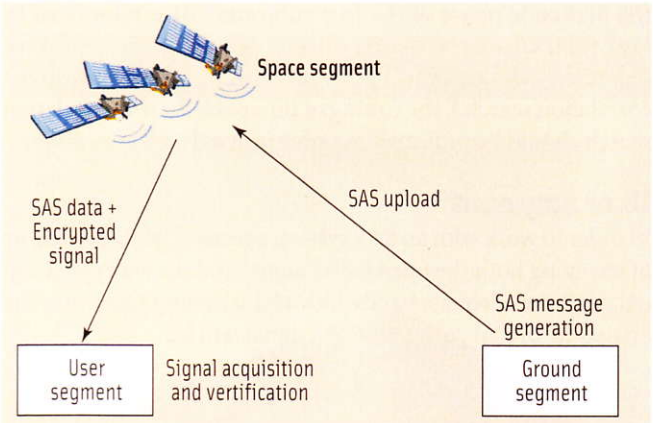

FIGURE 6 SAS architecture overview

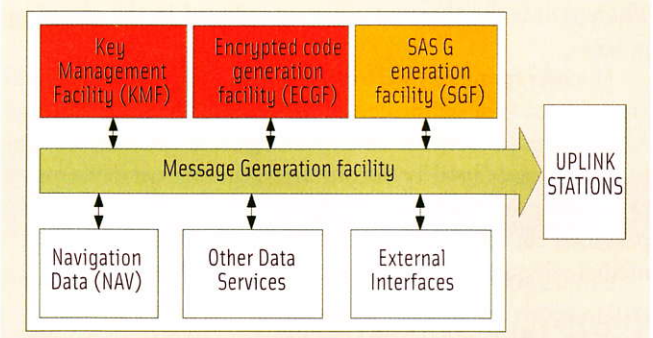

FIGURE 7 Ground segment SAS block diagram

the number of different existing GNSS services, the objective of this article is not to define a protocol for the SAS. Instead, it considers various considerations issues for design parameters and specifications are provided:

Message Overhead and SAS Data Truncation. A number of navigation data must be received by the receiver with a certain priority, such as time of week (TOW), clock corrections, and ephemeris data. Therefore, SAS messages shall be transmitted in order to not interfere with such data.

Depending on the channel bit rate and spare available data, the designer of the system can decide to transmit the entire

SAS sequence or to truncate it in sub-messages. The SAS size is determined by a number of factors, including encryption scheme and modulation type, signal power, and expected receiver noise floor.

SAS Advance/Delay Approach. There are two approaches to SAS transmission: advance the SAS in time with respect to the encrypted signal or delay the transmission. Delaying the SAS would reduce the risk that an attacker regenerates the sequence. The time to alert (TTA) would be proportional to the frequency of SAS transmissions.

Preventing Fake Encryption. The SAS message should integrate authentication and encryption so that an attacker cannot generate a fake encrypted signal at the precise SAS epoch.

Timing. The SAS time reference could be set with two approaches. A predetermined recurring time slot (for example, the first code phase of the first subframe) or randomized in order to increase the security. (In the latter case the SAS message should indicate the precise epoch in which to perform the correlation search.) The timing of the epoch for the correlation search should be projected in order to avoid a bit transition.

User segment

In order to work with an SAS system, a receiver must be capable of receiving both the open GNSS signal and the encrypted signal, with the adequate bandwidth and sampling rate. After the analog-to-digital conversion the signal will be:

$$s(n)=\sum_{k=1}^{N}\left[O_a^{\,k}(n)D^k(n)\cos(\omega_{IF}n)\right]+ \quad (4)$$
$$\sum_{k=1}^{N}\left[O_b^{\,k}(n)SC^k(n)D^k(n)\sin(\omega_{IF}n)\right]+e(n)$$

where $e(n)$ is the thermal noise introduced in the sampling process.

Doppler frequency wipe off as well as code and phase locks are assumed to be performed on the open code $O_a^{\,k}$. The receiver will attempt to store the encrypted signal at the discrete time $[n_0: n_0+l]$ as defined by the protocol. After the carrier removal by multiplication with $\sin(\omega_{IF}n)$ and after application of a low-pass filter cut downconvert the received $2\omega_{IF}$ frequency to intermediate frequency (IF), the remaining signal is:

$$\sum_{k=1}^{N}\left[\frac{1}{2}O_b^{\,k}(n)SC^k(n)D^k(n)\right]+e(n) \quad (5)$$

The receiver can generate the spreading sequence as the modulo 2 sum of the SAS code defined in equation (3) and the public spreading code, resulting in a short, local security code replica (SCR):

$$SCR^j(n)=SAS^k(n)O_b^{\,k}(n) \quad (6)$$

where j is the specific satellite code.

A security processing function will evaluate the correlation value C^j defined in equation (7) of the encrypted signal and the local replica based on a threshold for every k satellite.

$$C^j=\sum_{n=n_0}^{n_0+l}SCR^j(n)\left\{\sum_{k=1}^{N}\left[O_b^{\,k}(n)SC^k(n)D^k(n)\right]+e(n)\right\} \quad (7)$$

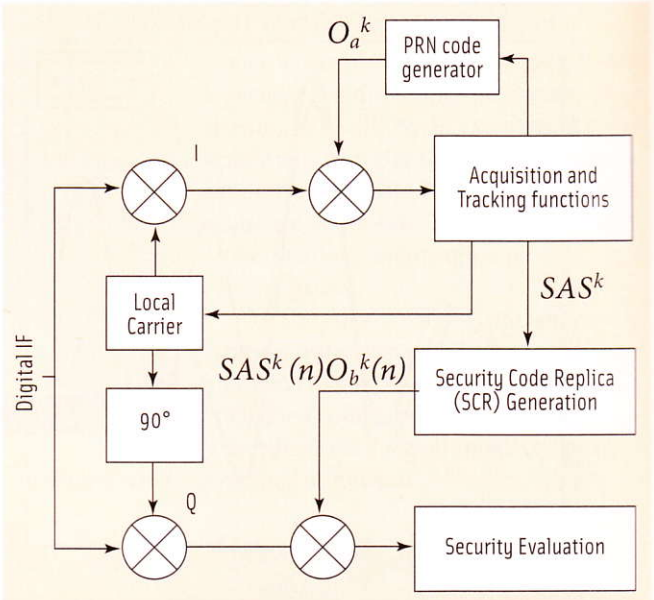

FIGURE 8 SAS receiver example

This security processing function will determine the signal authentication state based on a threshold that can be set as parameter in the receiver. **Figure 8** shows a hypothetical example of SAS receiver.

Test Simulations

The proposed method has been tested with a Matlab simulation in order to prove the feasibility of the authentication scheme. Furthermore, the receiver operating characteristic (ROC) can be analyzed based on the probability of false positives (spoofing detected erroneously) and false negatives (spoofing not detected).

The SAS transmission block simulates a BSC(m,n) signal, where the spreading code has a 10.23 megahertz chipping rate, a BOC modulation, and the stream cipher frequency m and number of satellites n can be set in the software. A subset of the stream cipher is used to generate the SAS, that is stored simulating a transmission in the open signal. The spreading code and stream cipher are modulo 2 summed, obtaining the final modulation code. The software correlates the SAS with the encrypted code in the predetermined period and performs analysis on the correlation results.

The idea is that a correlation peak indicates a correspondence between the SAS and the unpredictable encrypted code, resulting in a high confidence that the signal is authentic. (As mentioned previously, the security is based on the fact that an attacker could not generate the encrypted signal.) A low correlation value means that the SAS is different from the encrypted code, indicating a possible spoofing attack.

The simulation incorporated the following parameters:

- **SAS length.** Different values of this parameter have been tested during the study.
- **Correlation threshold.** Defined as the ratio between the highest correlation value and lowest one.
- **Visible satellites.** The sum of different PN sequences results

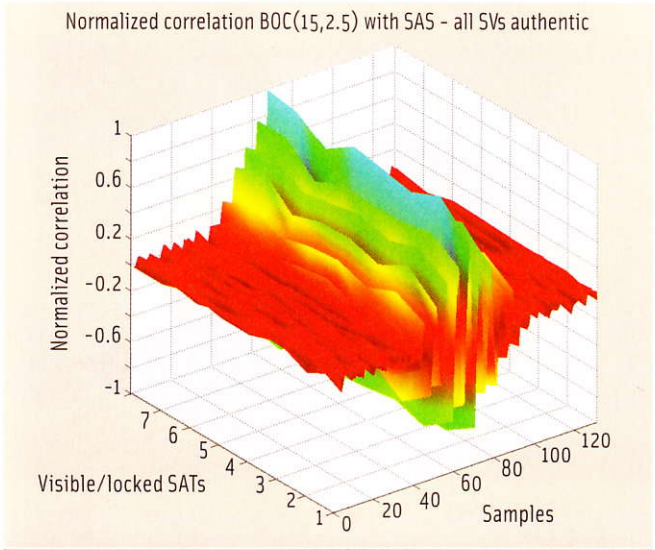

FIGURE 9 Simulation of SAS implementation in which encrypted code samples in the receiver are correlated with satellite signals. In this case, the normalized correlation peak is greater than 0.9, with green colors indicating that all satellites have been authenticated

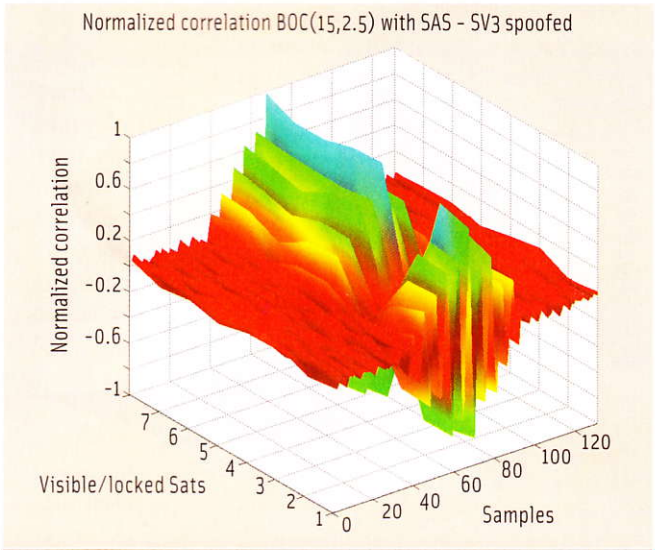

FIGURE 10 Simulation of SAS implementation in which one satellite is spoofed: SV3, spoofing indicated by red color

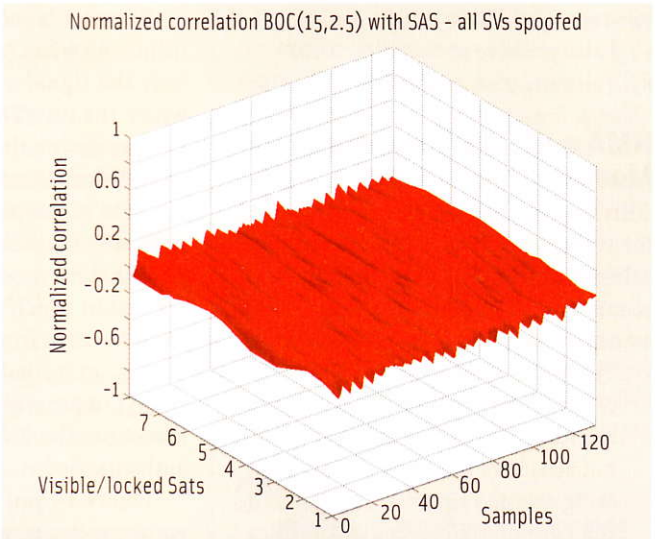

FIGURE 11 Simulation of SAS implementation in which all satellites are spoofed

in multiple access interference (MAI), reducing the system performance. The value for the simulation has been set to 8.

- **Binary stream cipher carrier frequency** — BSC(20,10) for the simulation
- **BOC modulation parameter** — (15,2.5)
- **Length of correlation windows** — 20 chips
- **Received power** — -157dBW
- **Bandwidth** —25 megahertz

Detecting Spoofed Signals

Figure 9 shows the results with a SAS length of 5,000 chips. Combined with a BSC(20,10) carrier, this results in a local code SCR length on the order of 10^5. (For visualization purposes the codes has been shifted in order to align the SAS position of every space vehicle — SV. Also, in Figures 8–10, a green color indicates authentication and red indicates spoofing.) The figure graphic shows a BOC(15,2.5) signal with a correlation peak corresponding to the SAS codes (>0.9 normalized correlation), indicating a condition in which all the signals are authentic.

In Figure 10 we have spoofed SV 3. The software generated a random PRN code instead of the original encrypted sequence, simulating a single-SV signal spoofing (such as might be performed with a receiver-spoofer) or by buffering and retransmission (with delay) of the original signal. The results show that all the satellites are authenticated except SV 3 where the code replica SCR^3 had a noticeably low correlation with the spoofed sentence.

Figure 11 shows the case where all satellites are spoofed. The correlation at the SAS epoch is lower (<0.1 norm.).

False Positives & Negatives

Tests have been performed using a large number of simulations in order to determine the probabilities of generating false negatives (not detecting a spoofed signal) and false positives (erroneously identifying a satellite signal as being spoofed). The correlation threshold is defined as the ratio between the highest correlation value and the lowest one.

Varying the SAS length and the threshold, two plots are shown. Figure 12 shows the false positive variation. It can be seen that increasing the SAS length decreases the probability of false positives. False negatives, however, do not seem to be significantly affected by the length of the signal authentication sequence within the plot range of 500 to 5,000 chips (See Figure 13). (Note that the labeling of the graph axes is reversed in the two figures.)

The correlation threshold also affects the two plots differently: increasing the threshold produces more false positives while false negatives decrease. This is because the correlation value must be higher in order to pass the security threshold, and a threshold too high might exclude even satellites that are authentic.

From this analysis, we can conclude that a good compromise would be an SAS length of 5,000 chips and a correlation threshold

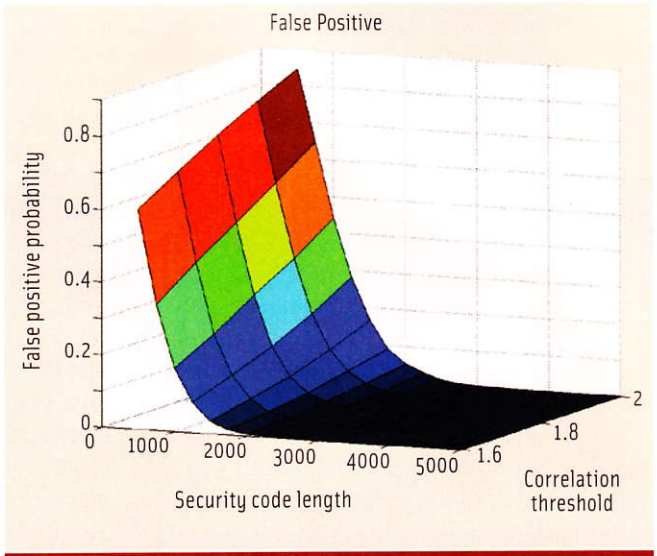

FIGURE 12 False positives

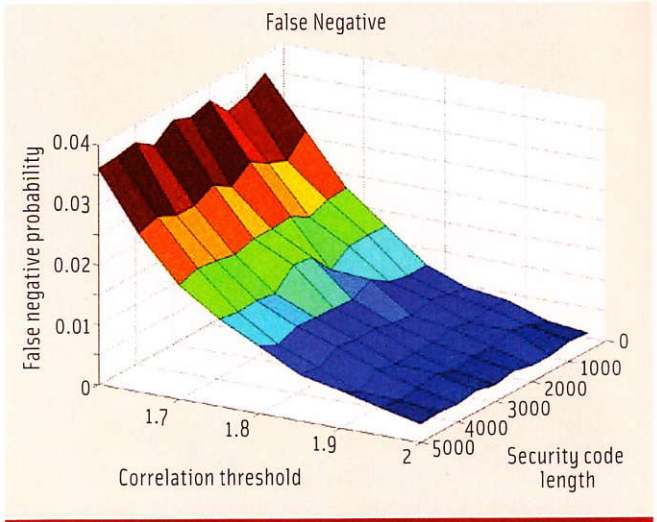

FIGURE 13 False negatives

of 2. With these values we obtain the following probabilities:

- False positive probability: 4×10^{-5}
- False negative probability: 3.31×10^{-3}

NMA and SAS Operating Modes

Three operating modes are proposed for the deployment of NMA and SAS schemes, depending on the user and security requirements (see **Table 1**). We assume two different security contexts:

- *Trust of the antenna proximity.* The receiver user is trusted (that is, has no interests to tamper with the receiver), but attackers might be nearby (radiating spoofed signals). This could be the case for safety-critical applications such as rails and aviation.
- *Trust of the receiver.* Both the antenna proximity context and the user receiver are not trusted (are willing to either radiate spoofed signals or tamper with the receiver). This could be the case for financial critical applications such as road tolls for example.

Mode	Trust of the antenna proximity	Trust of the receiver
(A) User based signal authentication	No	Yes
(B) Remote authentication service	No	Yes
(C) User based signal authentication and PVT integrity	No	No

TABLE 1. NMA and SAS Operating Modes

The user-based signal authentication (Mode A) refers to the authentication of only the signal and assumes a context where the user derives no benefit from compromising the receiver. The second mode (Mode B) refers to the same context, but data is sent to a remote service that will verify the authenticity of the signal.

The third mode (Mode C) refers to a context in which the user is not trusted (could benefit from spoofing the receiver, e.g., in a road tolling scheme) and could tamper with the data and receiver; therefore, the PVT output needs to be authenticated.

Figure 14 portrays Mode A. Navigation message authentication or signal authentication sequence data are received either via space or via ground communication (authentication service provider). In an NMA scheme, the receiver passes the navigation data (or a hash of it) and the NMA sentence to an authentication security module (ASM) for verification.

In the SAS scheme, the receiver passes an encrypted signal sample and the encrypted SAS message to the ASM. The authentication security module will decrypt the SAS message, generate the security code replica, attempt to correlate the codes and satellite signals, and return the authentication state. The ASM can support both symmetric and public key cryptography.

The remote authentication mode (**Figure 15**) foresees a receiver that sends data to an authentication service provider for a post-processing verification. In the NMA scheme, the receiver will send the navigation data (or a hash of it) and pseudorange to the authentication provider, which will verify the consistency of the message and the position solution. In the SAS scheme the receiver will transmit a sample of the encrypted signal (in a precise epoch) to the authentication pro-

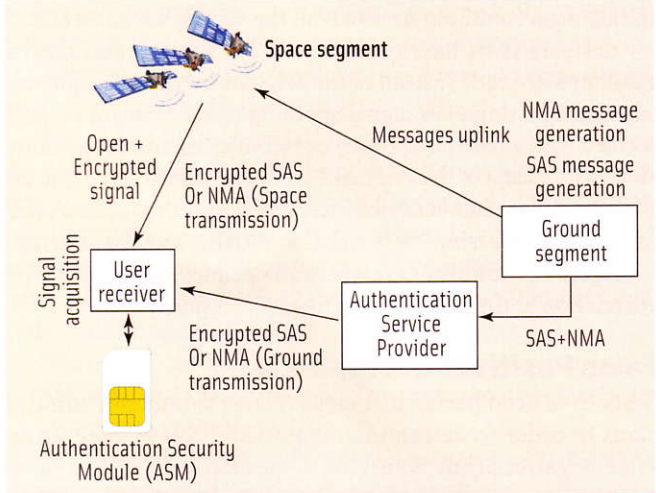

FIGURE 14 User-based signal authentication mode

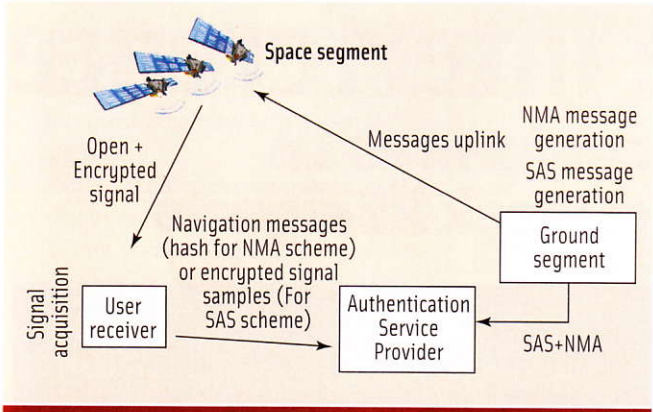

FIGURE 15 Remote authentication mode

vider, which will verify the authenticity of the signal with the SAS messages.

The user-based signal authentication and PVT integrity mode (Mode C) has the same architecture as mode A but foresees a context where the attacker will attempt to tamper with the receiver. Therefore, the ASM must be integrated in a tamper-resistant portion of the user equipment, otherwise an attacker will attempt to tamper with the communication between the receiver and the ASM. The article by O. Pozzobon *et alia* (1) discusses aspects of tamper-resistant GNSS receivers.

Conclusions

This article presented a new concept for an authentication mechanism and architectures that could be integrated into a GNSS system. The proposed method imposes a minimum effect on the system design, as only the data subsystem would be involved in a hypothetical update. No signal modifications are required for the implementation of the NMA and SAS schemes presented in this article.

Test results indicate that the signal authentication sequences concept can be used for authentication in systems that provide both open and encrypted signals, achieving a higher security level compared to navigation message authentication schemes (see **Table 2**), and that the security achievable with SAS with respect to spoofing attacks is comparable to that achievable with spreading code encryption, both being based on the security of the encrypted signal.

Spoofing Attack	Prevention with NMA	Prevention with SAS
Full signal simulation	Yes	Yes
Single channel attack	Yes	Yes
SV data replay attack	No[1]	Yes
Selective delay attack	No[2]	Yes
Early bit detection attack	No	Yes
Signal record and replay attack	No	Needs special HW (C/N_0 of repeated signal might not be sufficient to trigger the security evaluation threshold)

[1] Can be detected with the use of a trusted clock on the receiver
[2] Can be detected with AS-RAIM (Anti-spoofing receiver autonomous integrity monitoring)

TABLE 2. Comparison of NMA and SAS Performance against Various Types of Spoofing

Acknowledgments

The author would like to thank Andrea Dalla Chiara, Luca Canzian, and Matteo Danieletto from the University of Padova for their support in the simulation of the SAS scheme.

References

[1] Arze Pando, D. H., "Distance-Decreasing Attack in Global Navigation Satellite System," 2009 project, School of Computer and Communication Sciences (I&C), Swiss Federal Institute of Technology (EPFL)

[2] Galileo Interface Control Document, OD SIS ICD, Issue 1, February 2010

[3] Hein, G., and F. Kneissl, J.-A. Ávila-Rodriguez, and S. Wallner, "Authenticating GNSS: Proofs against spoofs, Part 1," *Inside GNSS*, July/August 2007, pp. 58-63

[4] Hein, G., and F. Kneissl, J.-A. Avila-Rodriguez, and S. Wallner, "Authenticating GNSS: Proofs against spoofs, Part 2," *Inside GNSS*, September/October 2007, pp. 71-78

[5] Kuhn, M. G., "An Asymmetric Security Mechanism for Navigation Signals," Proceedings of the 6th Information Hiding Workshop, pp. 239–252, 2004

[6] Lo, S., and D. De Lorenzo, P. Enge, D. Akos, and P. Bradley, "Signal Authentication, A Secure Civil GNSS for Today," *Inside GNSS*, Vol. 4, No. 5, September/October 2009

[7] Pozzobon, O. (1), and C. Wullems, and M. Detratti "Security Considerations in the Design of Tamper-Resistant GNSS Receivers," IEEE/NAVITEC 2010, Noordwijk, The Netherlands, December 10, 2010

[8] Pozzobon, O. (2), and L. Canzian, A. Dalla Chiara, and M. Danieletto "Anti-Spoofing and Open GNSS Signal Authentication with Signal Authentication Sequences," IEEE/NAVITEC 2010, Noordwijk, The Netherlands, 10 December 2010

[9] Scott, L., "Anti-Spoofing & Authenticated Signal Architectures for Civil Navigation Systems," *Proceedings of the Institute of Navigation GPS/GNSS 2003 conference*, pp. 1543-1552, 2003

[10] Trusted Innovative GNSS receiver (TIGER) project, Galileo Supervisory Authority grant agreement n° 228443, <www.tiger-project.eu>

[11] Wullems, C., and O. Pozzobon and K. Kubik, "Signal Authentication and Integrity Schemes for Next Generation Global Navigation Satellite Systems," *Proceedings of the European Navigation Conference GNSS, 2005*

Author

Oscar Pozzobon is the director and co-founder of Qascom S.r.l., Italy. He received a degree in Information Technology Engineering from University of Padova and a master's degree in telecommunication engineering from the School of Information Technology and Electrical Engineering, University of Queensland, Australia. He is currently involved in different industrial and government projects regarding the design and development of secure GNSS receiver, GNSS signal authentication, and anti-spoofing schemes. He has been involved in engineering activities with the European GNSS Agency, including the design, development, and testing of the 'Trusted Innovative GNSS receiver' (TIGER) project and the 'Precise and secUre autoMotive trAcking' (PUMA) project. His main interest are satellite navigation, cryptography and hardware security, where he holds three patents and more than 20 publications. IG