Design Drivers, Solutions and Robustness Assessment of Navigation Message Authentication for the Galileo Open Service

Ignacio Fernández Hernández, *European Commission*
Vincent Rijmen, *University of Leuven (KU Leuven)*
Gonzalo Seco Granados, *Universitat Autònoma de Barcelona*
Javier Simón, *European GNSS Agency*
Irma Rodríguez, J. David Calle, *GMV*

BIOGRAPHIES

Ignacio Fernández Hernández is currently managing the Galileo Commercial Service at the European Commission. He has previously worked in GNSS in engineering, research and management activities in areas like receivers, standards, EGNOS and ARAIM. He holds a MSC in Electronic Engineering by ICAI, Madrid, and a MBA by LBS, London.

Vincent Rijmen is currently full professor with the Dept. of Electrical Engineering (ESAT) of the University of Leuven (KU Leuven). Previously, he held the Chair of Applied Cryptography at the Graz University of Technology and was Chief Cryptographer of Cryptomathic.

Gonzalo Seco holds PhD degree from Univ. Politecnica de Catalunya and an MBA from IESE. During 2002-2005, he was staff member at the Radionavigation Section in ESTEC/ESA, and involved in the Galileo project and in the development of GNSS receivers and applications. Since 2006, he is associate prof. in the Dept of Telecom. Eng. of Univ. Autonoma de Barcelona (UAB) and head of the Signal Processing for Navigation and Communications (SPCOMNAV) group.

Javier Simon is Service Design Engineer within the European GNSS Agency, currently contributing to the definition and design of the Galileo CS. He holds a MSc. degree in Telecommunications Engineering from the Polytechnic University of Madrid, Spain. Before joining GSA he participated in several projects for the study and design of future GNSS algorithms and systems.

Irma Rodríguez has a MSc in Telecommunication Engineering, from the Universidad Politécnica de Madrid, Spain. She is Head of the GNSS Algorithms and Products Division within the GNSS Business Unit of GMV.

J. David Calle has a MSc in Computer Engineering from the University of Salamanca. He participates in R&D activities related to GNSS algorithms and is the responsible for the Galileo Commercial Service Early Proof Of Concept.

ABSTRACT

As part of its duties concerning Galileo Services & Exploitation, the European Commission is studying services to be offered by Galileo to the GNSS community in the next few years. A service that could be provided without modifying the payload of the current Galileo satellites is the authentication of the navigation information. The Galileo data authentication service would contribute to improving worldwide GNSS security and make Galileo more attractive to user communities.

In order to design and compare different solutions, a conceptual framework including performance indicators is presented. Based on standard navigation performance analysis metrics, the higher-level performance indicators considered in the authentication framework are Availability, Accuracy, Time to First Authenticated Fix and Robustness. Other lower level indicators traced to the above will be defined as well. It is justified why Authentication Error Rate (AER) and Time Between Authentications (TBA) arise as the main indicators. Indicators related to robustness against replay attacks and signal unpredictability, as Maximum Predictable Time (MPT) or Unpredictable Symbol Ratio (USR) will be presented as well.

Secondly, this paper presents contributions to the state-of-the-art of standard symmetric, asymmetric and hybrid (symmetric and asymmetric) authentication approaches for satellite navigation. Implementations of cross-authentication among satellites are discussed, and some schemes based on a Timed Efficient Stream Loss-tolerant Authentication (TESLA) protocol are presented.

The paper then presents and characterizes one Navigation Message Authentication (NMA) solution for the Galileo E1B Open Service signal. Finally, some conclusions and further work are proposed.

MOTIVATION AND CONTEXT

As part of the safety-of-life service, Galileo foresaw to provide a data authentication service for the integrity tables to be transmitted in the I/NAV message. However, as the safety-of-life service has been re-profiled and some requirements of the Galileo system waived, this authentication service is not maintained and, even if it were, it might not satisfy the needs of a broad user base, as its main purpose was to authenticate the Galileo integrity data only and not the navigation data.

In addition, the Galileo Commercial Service foresees to use signals in the E6 band (E6B and E6C) whose spreading codes can be encrypted through a secret key, therefore providing a level of access control and authentication that can be found useful by several user communities.

Due to the high demand to strengthen GNSS open civil signals for consumer or mass market users, the Galileo program is studying how open navigation message authentication could be implemented within the Galileo signal, and what would be the usefulness of it for the mass market users. The analysis of an NMA standalone service is triggered by the following facts:

- In the current and future GNSS context, with around a hundred navigation satellites providing open ranging signals to the users, the public value of Galileo can be enhanced by offering additional services.

- The Galileo signal design and message structure is adequate for introducing authentication, as it allows higher bitrates compared to other GNSS [1][2] and, due to the safety-of-life 're-profiling', a significant amount of bandwidth has been liberated for other uses.

- The impact in terms of risks and cost in the Galileo program of adding NMA is low: The Galileo ground and space architecture can accommodate the transmission of data from an external source into the operational system in real time, opening the possibility to transmit authentication information into the core infrastructure without modifying the operational baseline. Even if modifications to the current Galileo specification might increase performance and robustness of an NMA solution, NMA could be provided in accordance with the current Galileo core system specification.

- Previous literature suggests that, when complemented with additional checks at the receiver, it can provide a reasonable level of protection not only of the satellite data but also against replay attacks [3][4]. In any case, as the Galileo service offering includes spreading-code encrypted signals in the E6B as part of the Commercial Service, an NMA service could be combined with these signals for users that rely on encrypted spreading codes. The proposed NMA scheme could also be upgraded in future Galileo generations available in the next decades.

Due to all these reasons, an additional line of work was started by early 2013 to assess the provision of NMA in the short term through Galileo open signals, with a minimal, if any, disturbance of the Galileo system or operational requirements.

PROTECTION OF NMA AGAINST REPLAY ATTACKS

For the purpose of this work, GNSS authentication has been defined as the guarantee that GNSS information used by the receiver comes from the actual satellites, and not any other source. This implies the authentication of all the information used to compute a position and timing solution, namely:

- The orbital parameters providing satellite positions at a given time.
- A time reference for each satellite signal, which associates the bit transitions to the constellation time and is also used for the satellite position computation. It usually comes from the time of week (TOW).
- Error corrections applied in the navigation solution, including at least satellite clock offsets, and potentially other corrections as ionospheric delays from the ionospheric model provided by the constellation, if ionosphere is not corrected through other means.
- The measurements coming from the signal tracking loops of the receiver, including at least the code phase measurement from the delay lock loop.

Therefore, the elements to be authenticated can be summarized in navigation data and time of arrival. While the authentication of data is straightforward, the authentication of time of arrival is not, and it can only be considered as a protection against certain attacks, under certain conditions and with a certain probability [4].

While an exhaustive analysis of the attacks and conditions is out of scope of this paper, it is considered that signal unpredictability, even if at data level (as opposed to spreading code level) provides enough protection against replay attacks for many applications. The reason is that the first signal samples of each unpredictable symbol cannot be forecast by an attacker because it needs to perform the correlation of the incoming signal with a local replica during some period of time in order to estimate the value of those first samples.

PERFORMANCE FRAMEWORK

In order to design a robust data authentication service for different GNSS user types such as automotive, pedestrian, maritime or even aviation users, several aspects need to be taken into account: Robustness against attacks implies not only that the data authentication is cryptographically secure, but also that the service contributes to the detection of replay attacks. Also important is to maximize the availability of the service: all authentication information needs to be correctly demodulated for a successful authentication, and the environmental conditions of some users, subject to path loss and shadowing effects, may not always allow this. Finally, the solution should be affordable to all user receivers, and combinable with other information sources to add robustness to the user location services.

As a general principle, it is considered that an NMA solution that is optimally designed should provide the same performance as a non-authenticated solution in terms of accuracy, availability and time to first fix, but authenticated with a level of robustness that is considered appropriate for the application.

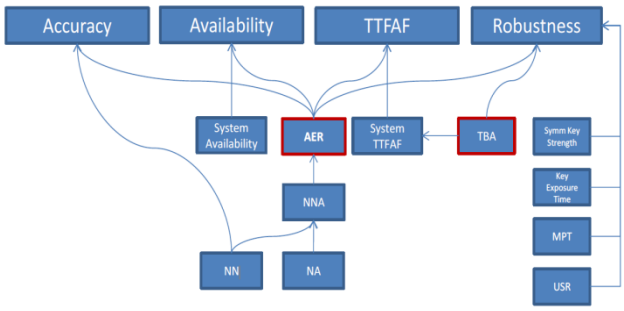

Figure 1 – Authentication Performance Framework

Figure 1 presents the conceptual framework used to design and compare authentication solutions. Further details can be found in [5].

In order to simplify the comparison of performances, four of these indicators will be highlighted and explained. They are the Authentication Error Rate (AER), the Time Between Authentications (TBA), the Maximum Predictable Time (MPT) and the Unpredictable Symbol Ratio (USR). AER and TBA are highlighted in the figure due to their high influence on other indicators.

AER reflects the probability of error of a satellite being authenticated in the absence of attacks but in the presence of disturbances of a real transmission channel.

$$AER = 1 - (1 - BER)^{NNA} \quad (1)$$

Where BER is the bit error rate and *NNA* is the number of bits used in the authentication, which includes the navigation bits to be authenticated (NN), or 'plaintext', and the authentication bits (NA).

TBA reflects the time between two successive authentications of a satellite. It is a design parameter so it need not be computed analytically. For the purpose of this work, no offsetting in the authentication of different satellites has been considered yet (as proposed in [4]). If this were the case, a distinction between 'user' TBA and 'system' TBA should be made.

In real conditions assuming an AER greater than zero, average TBA is obtained from the following formula:

$$\overline{TBA} = \frac{TBA}{1-AER} \quad (2)$$

It should also be noted that TBA and AER drive the Time To First Authenticated Fix (TTFAF). Assuming that the user already has valid (but not yet authenticated) navigation data, or that the receiver receives simultaneously the navigation and authentication data, as is the case in the proposed approaches, the average TTFAF is

$$\overline{TTFAF} = \frac{TBA}{2} + \overline{TBA} \quad (3)$$

TBA/2 is the average of a uniform probability distribution between 0 and TBA, and represents the average time that the receiver will need to wait since data demodulation starts until a new authentication starts (assuming that authentication information is transmitted sequentially, i.e. all the information for the next authentication is transmitted after the last authentication). The average TBA takes into account the fact that, because of data demodulation errors, sometimes the first authentication will be erroneous, increasing the TTFAF. It should also be noted that the signal acquisition time is left out of the computations, as it does not provide valuable information to compare authentication solutions in the context of this work.

MPT reflects the maximum time that, according to the design, the signal will be predictable. It is based on the fact that if any of the inputs of the authentication solution varies, the output authentication information will be fully unpredictable. As the authenticated information will include TOW, the output bits will be unpredictable. The main purpose of reducing MPT is to constrain attacks taking control of the receiver tracking loops. Nevertheless, having unpredictability in the signal very often may also favor snapshot approaches whereby a signal snapshot containing unpredictable features is captured in the receiver, time-tagged with a trusted clock and checked in a server. MPT is a design parameter and therefore can be computed from the NMA data structure.

USR, in the context of this work, reflects the percentage of unpredictable symbols. With this parameter, if a replay detection test statistic based on hypothesis testing is computed from the estimation of unpredictable symbols (or the first samples thereof), USR gives an indicator of how long a receiver has to wait to have a reliable test statistic. USR is a design parameter and therefore can be computed from the NMA data structure.

While the proposed indicators are used to characterize an NMA solution, they can be adapted to authentication solutions protecting the spreading codes as well.

Other requirements and drivers that have been taken into account indirectly in the design but do not appear explicitly in the framework are:

- Receiver limitations:
  - Security related requirements to the receiver shall be minimized, e.g. the storage of secret keys shall be avoided if possible.
  - The solution should not require any additional hardware components than those of a standard low-cost receiver.
  - The solution shall be computationally feasible in a standard low-cost receiver.
- Security: the proposal shall use methods and protocols considered as sufficiently secure by the cryptographic community for the following decades.
- Flexibility: The solution shall be robust over its full lifetime, which can last up to several decades. Therefore, a certain flexibility to cope with potential advances in cryptanalysis is advised.
- Autonomy: The solution shall be as standalone as possible, minimizing public key updates over the lifetime of the receivers.

AUTHENTICATION CONCEPTS

This section presents some authentication concepts and implementation approaches to optimize GNSS authentication:

- "Cross-authentication", implying the case whereby some satellites authenticate other satellites by digitally signing their data.
- A standard Timed Efficient Stream Loss-tolerant Authentication (TESLA) approach [6].
- A TESLA approach using the same chain and keys for all senders at a given time.
- A TESLA approach using the same chain but different keys from different senders at a given time.

Cross-Authentication

In a standard GNSS authentication approach, each satellite will authenticate its own data, as shown in Figure 2, where satellites 1, 2, 3 and 4 transmit standard navigation data P1, P2, P3 and P4 (or 'plaintext' 1, 2, 3 and 4) and a digital signature of it. Ideally, for the provision of authentication, it is preferred that all satellites used by the receiver can transmit authentication information. However, this may not be the case if:

- Satellites from other constellations, over which no control is exercised, are used.
- Satellites from the own constellation cannot provide authentication information over certain periods of time due to limitations in the system architecture or operation.

For this purpose, it is advantageous to develop "cross-authentication" schemes whereby satellites can authenticate others, in a similar way as augmentation or differential systems provide corrections or integrity to other satellites.

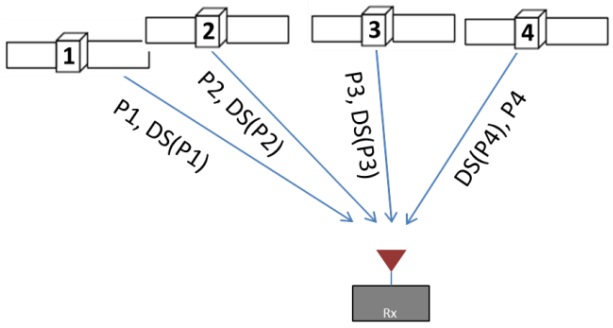

Figure 2 – Standard per-satellite authentication

A NMA solution whereby some satellites authenticate other satellites was presented in [5] and is shown in Figure 3.

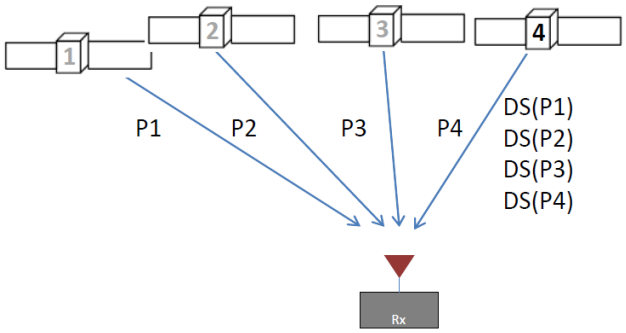

Figure 3 – Cross-authentication

In the above figure, satellites 1, 2 and 3 are transmitting standard navigation data P1, P2, P3 and satellite 4 transmits a digital signature of them, and the signature of its own data DS(P4).

The principal limitation of this approach is that only the satellite transmitting the authentication information contains unpredictable features and is therefore protected against replay attacks.

Cross-authentication can provide data authentication and anti-replay protection for all satellites if the authenticated data is partly unpredictable, as shown in Figure 4.

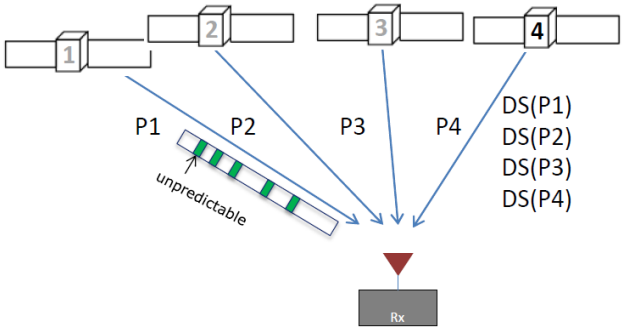

Figure 4 – Cross-authentication scheme with anti-replay protection

TESLA

TESLA is based on the transmission of a message authentication code (MAC) to authenticate the plaintext message and the subsequent transmission of the key used to compute the MAC. This key belongs to a chain of one-way functions. A hash function (e.g. SHA-256) is a one way function, so each element of the chain can be constructed by hashing the previous element. The chain starts with a seed key, which is secret, and ends with a root key that is public and certified through external means so the receiver can consider it as authentic. GNSS authentication through TESLA can be performed in the following way:

- The receiver receives the navigation data and the MAC.
- The receiver later receives the key with which the MAC was generated.
- The receiver re-generates the MAC with the key and the data, which should coincide with the previously received MAC.
- The receiver authenticates the key with a previous key from the chain that is considered authentic, or the root key, by hashing the key the required number of times.

TESLA was initially proposed by [6], where more details about its implementation can be found. It has been also proposed in the domain of GNSS in [5], [7] or [8].

Figure 5 represents a one-way chain, where K_n is the seed key, from which the chain is generated, and K0 represents the root key.

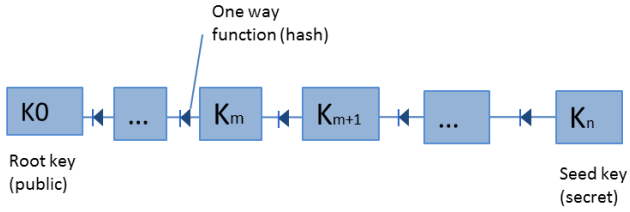

Figure 5 – TESLA one way chain of keys

In a standard TESLA approach, each sender (satellite) uses a different one-way chain, as shown in Figure 6. If a receiver authenticates four satellites, it should receive, in addition to the data to authenticate (P1, P2, P3, P4), four MACs (MAC1, MAC2, MAC3, MAC4) and four keys (K1, K2, K3, K4), one from each satellite, where each key belongs to a different chain.

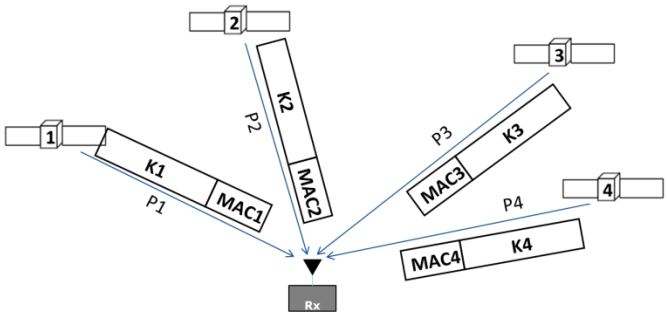

Figure 6 – Standard TESLA approach with a different key and chain per sender

The next section presents an optimization whereby all senders (satellites) use the same key chain, in order to reduce AER.

TESLA with a Single Chain for Several Senders

The proposed concept in this section is also based on a TESLA approach. The main difference to other proposals in the previous work and the literature is the use of a single one-way chain for all senders, as opposed to the use of a single one-way chain for each sender. The main motivation of this choice is to drastically reduce AER: by allowing all the satellites to be authenticated through the same chain, a user needs to receive only a correct key from one satellite to authenticate all satellites. This reduces dramatically the amount of bits required for a PVT computed using data-authenticated satellites.

Not only a single chain will be beneficial for NMA success rate in stationary conditions (i.e. after a certified root key is in possession of the receiver), but also it will help initialization, as only one root key is required for all satellites, as opposed to one root key for each satellite, significantly reducing the time to first authenticated fix. Figure 7 depicts the concept of using the same key from all senders.

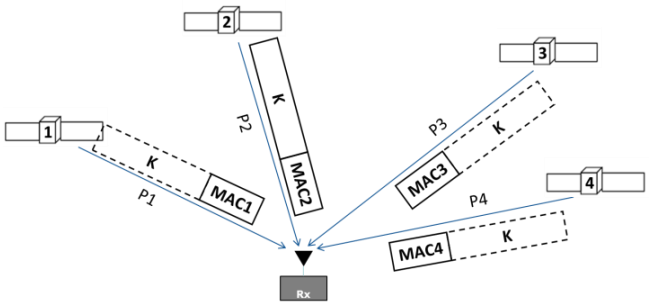

Figure 7 –TESLA approach with the same key for all senders

The proposed solution is especially useful where one or few satellite signals are received in good conditions, surrounded by others at lower elevations or subject to multipath or blockage with worse data demodulation conditions and therefore a much higher bit error rate, which may be the case in urban environments.

It should also be noted that, even if due to shadowing or fading at a certain "authentication frame", which occurs once every TBA seconds, no key is successfully demodulated from any satellite, any key from the next authentication epoch can be used to verify the previous "authentication frame".

One problem that arises when using a single one-way chain for all satellites is that, if the same key is used and transmitted at the same time from all satellites, it will be received at different times by users due to satellite clock offsets and, principally, due to the different time of arrival. For example: The signal from a Galileo satellite at the zenith would take to arrive to the Earth surface approximately h / c seconds, where h is the satellite height and c the speed of light, while the signal from a satellite at around 0 degrees (or at a slightly lower elevation, which may be unrealistic but useful as a boundary in the current example) would be $(\mathrm{sqrt}(r^2 + (h+r)^2)/c$, where r is the earth radius. For a Galileo satellite in a nominal orbit, at 23228 km height [9], and an earth radius of 6371 km, the time difference of arrival is 101 ms - 77.4 ms = 23.6 ms. A spoofer would have therefore some tens of milliseconds to estimate some unpredictable bits from one satellite and replay them with a delay at another one, spoofing the position even if the data used is authentic. Therefore, if all satellites are transmitting the same key at the same time, only the symbols from the satellite closest to the zenith can be considered as unpredictable.

TESLA with a Single Chain and Different Keys from Different Senders

The problem above mentioned can be overcome by transmitting different keys, but still from the same chain, from different satellites. In this way, the key bits would still be unpredictable while any key used for the MAC computations can be recoverable from any later key of the chain.

The main drawback of this approach is that it requires a higher computational power, as the number of one-way operations (hashes) per chain will be higher. For example, if 40 keys (for a nominal GNSS constellation with some margin) are used at every "authentication frame", i.e. every time that a user can authenticate, which occurs with a periodicity defined by TBA, the chain would become 40 times longer. The assessment of the additional CPU needs for this approach is covered in a later section.

Figure 8 and Figure 9 present how the keys of a certain chain would be transmitted and used every time the receiver performs an authentication.

As an example, if TBA is 10 seconds, every 10 seconds each satellite would send at least one MAC authenticating the satellite data with a key that, for the 10-second period of "authentication frame" j, is $K_{j,1}$. This key would be transmitted from SVID1, while SVID 2 would transmit the previous key in the chain $K_{j,2}$, SVID 3 would transmit $K_{j,3}$, and so on. In this way, a spoofer would not be able to predict the key of one satellite from another.

One could argue that SVID2 is closer than SVID1 to the receiver, an attacker could predict $K_{j,1}$ by receiving previously $K_{j,2}$ from the closer satellite. However, as the transmission of the keys is done in parallel during some seconds, only the very few final bits of $K_{j,1}$ could be predicted by having a high amount of bits of $K_{j,2}$, rendering such an attack rather impractical. User implementations could discard the last bits of the key in the anti-replay statistics, as proposed later.

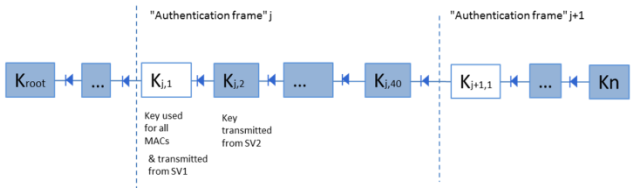

Figure 8 –TESLA one-way chain with different keys from different senders

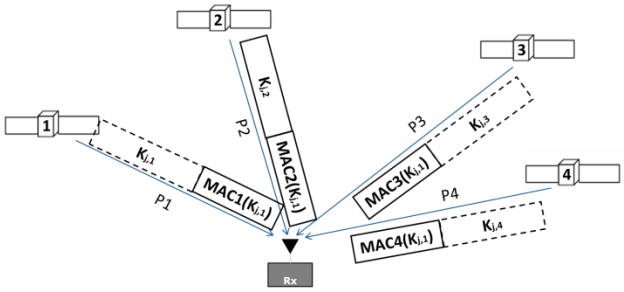

Figure 9 –TESLA single-chain approach with a different key transmitted by each sender

IMPLEMENTATION TRADE-OFFS AND ANALYSES

This section presents some tradeoffs in the design of GNSS authentication services and some preliminary analyses. They are intended to justify some of the design decisions taken in the implementation example presented in the rest of the paper.

Digital Signatures versus Time Delay Asymmetry

To maximize the use of GNSS authentication, cryptographic key management should be simplified as much as possible. This means that asymmetric schemes, whereby the user receivers need only to possess a public key, are preferred to symmetric schemes, whereby the user needs to store a secret key in a security module within the receiver. Asymmetric schemes can be achieved mainly through two ways:

- Digital signatures, as RSA, DSA or elliptic curves [10]. In these schemes, the satellites transmit a digital signature of their navigation data, as described in [4].
- Delayed symmetric key delivery, as TESLA.

The main advantage of authentication through digital signatures is that there are known methods and functions in the cryptographic standards that make them reliable for the cryptographic community [10]. The main disadvantage of authentication through digital signatures, compared to time-delayed symmetric approaches, is the bandwidth required to transmit the authentication information. For example, in order to transmit a digital signature that guarantees a 112-bit to 128-bit level of security, signatures in the order of 500 bits are required (e.g. 512 bits for standard DSA, 128-bit security, or 466 bits for 112-bit security through ECDSA, as per [4]). Other disadvantages may include the computational effort required per authentication, and the fact that some elliptic curves may be subject to patent rights.

The main advantage of schemes based on the delayed delivery of a key used to generate a previously sent authentication code, as TESLA, is the bandwidth reduction and the tolerance to data loss. Their main disadvantages may be that they are not as standardized and accepted by the cryptographic community, and their design must protect the user against more threats, as they rely on a coarse time synchronization of the receiver with a time reference.

In any case, to authenticate the signal-in-space (SIS), the receiver must possess some information that is certified as correct independently from the signal-in-space. This means that even for TESLA approaches, the receiver must possess a public key to authenticate the SIS. However, the frequency with which this public key is used, and the bandwidth associated to this process, can be very low, if a TESLA approach is used.

Preliminary Assessment of CPU Needs of a Single One-Way Chain

To understand the computational power required in a single one-way chain where each satellite transmits a different key, state of the art SHA-2 implementations have been looked at. It is claimed that around 11.5 processor cycles per byte are required [11]. As a rough estimation, a 1GHz processor available in smartphones already would need around 0.4 microseconds for a SHA-256 (i.e. 32 bytes) iteration. For the following assessment, a conservative approach of allocating 1 microsecond per iteration is taken.

The CPU required for a single chain multiple-key TESLA approach, assuming 40 hash iterations per authentication (allowing to cover 30+ satellites) would be therefore 40 microseconds for the whole set of satellites every TBA period, as the operation is required only once for all

satellites, which in absolute terms is very affordable for a standard low-end processor.

As regards the CPU needed to verify a certain key K against an authenticated root key, which is, for example, associated to a time 1 week before the applicability time of K, assuming 40 hash iterations for a 10-second TBA period, it would require 7 days/week * 24 hours/day * 60 minutes/hour * 2 subframes/minute * 3 authentications/subframe * 40 hashes = 2419200 iterations, i.e. around 2.5 seconds, which is highly affordable taking into account that this operation may be required very unfrequently. Therefore, the CPU computing power required seems not a major driver. It should also be noted that, in standard 1-chain-per-sender TESLA approach, the chain verification needs to be computed for each satellite, leading to similar computing power needs.

As a summary, this preliminary assessment shows that CPU power required for a single one-way chain using different keys per satellite is affordable for the CPUs of present and future GNSS terminals.

Security Considerations of a Single One-Way Chain

By using only one chain for all satellites, if the chain is compromised (i.e. the seed key K_n is found), the whole system is compromised. However, if a one-way-per-satellite chain is secure, a one-way-all-satellites chain shall be secure as well, due to the following:

- The 1-way chain security depends on the choice of the hash primitive and hash bit output length. A cryptographically secure design choice for several chains shall be as secure for a one-way all-satellite chain.
- For a certain chain interval, instead of protecting several seed keys K_n the system shall protect only one K_n. The existence of a single vs. multiple parallel chains does not change the system architecture, as the security measures of the system are similar.
- The choice for the hash function and the key length can be done with enough margin to be considered secure for a much higher duration than the validity period of the chain. For example, if a chain is valid for some months, a design choice can be made that is considered cryptographically secure for years. If, due to the higher criticality of compromising K_n, higher security measures are proposed in the system, the validity period of each chain could be shortened, or the key bit length increased, or other system-related measures could be introduced, while maintaining the advantages of the current concept. As shown in the later example, the proposed implementation permits to change the key and MAC lengths and the cryptographic functions in order to cope with future threats over the lifetime of the service.

In summary, using a single chain for all satellites does not seem to reduce the security of the system.

Security Considerations on Key Length, MAC Truncation and One-Way Function Truncation

For applications with bandwidths that are large compared to the output sizes of cryptographic algorithms, it is common to use MAC functions with output sizes of 80 or even 160 bits, e.g. HMAC [12]. It should be noted, however, that meaningful levels of security are achieved already with much shorter output sizes.

Given the low throughput available in GNSS signals, the lengths of the key and the truncated MACs are very sensitive parameters. They drive the number of authentication bits (NA), and therefore affect AER and TBA (the latter assuming a fixed bandwidth), which are reflected in all other indicators. Therefore, their length should be reduced as much as possible, while maintaining security to an acceptable level.

Given that GNSS are one-way systems, an attacker has no control over the message that is authenticated (i.e. it cannot request a satellite to authenticate a given message), or the key used, and the MAC and key transmission occurs with a certain cadence controlled by the system specification. This yields some attacks impractical, permitting the use of very short MAC tags, even shorter than the 32-bit MACs used in many security-critical areas as for example banking.

Assuming the MAC algorithm chosen behaves as a lookup table with a message and a key as entries, and an n-bit random sequence as an output, an attacker could only try to guess the MAC, which is very unlikely even for extremely truncated MACs to very few bits. For example, a MAC as short as 10 bits would be guessed with an average probability of 0.097% (one time out of 1024), rendering the "MAC-guessing" attack very impractical compared to a pure service denial by e.g. jamming the signal. In addition, a receiver can accumulate two or more data authentications before accepting the data as authenticated, in order to reduce the "MAC guessing" attack probability. Given the very short TBAs and the possibility to cross-authenticate several satellites, this permits data authentication with higher probabilities with a delay of few seconds, or even without any delay. Think, for example, of a receiver that only uses a new issue of data (IOD) when is authenticated twice, reducing the "MAC guessing" probability to $1/(1024)^2$ and navigating in the meantime with the previous IOD, which should minimally affect the navigation performance.

As regards the symmetric key length used in the one-way chain, an 80-bit key is considered to be strong for long-enough (e.g. 1 year) chain durations. If the key had to be guaranteed for e.g. 20 or 30 years, longer keys of e.g. 128 bits would be recommended [13]. However, the key has to be robust only for its validity period, which is expected to be less than one year. Secondly, also the lifetime of each message, during which its authenticity must be guaranteed, is very short. This allows the use of shorter keys.

Even so, the proposed implementation allows the use of longer keys in the future, over the lifetime of the system (e.g. up to 128-bit keys), if deemed pertinent, while maximizing bandwidth use when shorter keys are considered secure.

To accommodate keys as short as 80-bits in a standard one-way function as SHA-256 [14] or SHA-3 [15], the one-way function needs to truncate the output of the hash function to the length of the key for every iteration in the chain, as shown below:

$K_m = trunc_{klen}(hash(K_{m+1} \| pattern))$, where

- K_m is the next element in the chain,
- *trunc* is the truncation function to the MSB
- *klen* is the length of the key (e.g. 80 bits)
- *hash* is the hash function used (e.g. SHA-256 or SHA-3)
- K_{m+1} is the previous element in the chain.
- *Pattern* is a bit sequence that is unique for each chain and is disclosed at the beginning of its lifetime, to separate different chains and to prevent brute-force attacks whereby arbitrary chains are pre-computed to predict parts of future chains.

IMPLEMENTATION EXAMPLE: NMA IN GALILEO E1B

This section presents an example of NMA implementation in the Galileo E1B signal. In particular, an NMA solution based on the abovementioned TESLA one-chain-different-keys concept will be inserted into the Galileo I/NAV message structure. The field named as "Reserved 1" in the ICD [1] will be used. This field provides a bandwidth of 40 bits every other second. The decision to use this field, as opposed to other spare fields in the I/NAV message is justified by at least the following reasons:

- The Galileo system allows filling in the information in this field from a source external to the Galileo core infrastructure, therefore minimizing the system impact of adding NMA.
- The scattering of the bits across the navigation message allows reducing MPT.
- The use of the E1 signal has more potential for the target NMA applications, as it shares carrier frequency (1575.45 MHz) with GPS L1 C/A used in mono-frequency receivers and future signals as GPS L1C.

Figure 10 presents the Galileo E1B I/NAV message structure and highlights the position of the "Reserved 1" field. The I/NAV message is convolutionally encoded following a Forward Error Correction scheme and interleaved, as described in the Galileo Open Service SIS ICD [1]. Only the same amount of symbols as unpredictable bits before encoding are considered unpredictable (i.e. 40 symbols in the current case; the demonstration of this is out of scope of this paper). This must be taken into consideration when assessing the MPT and USR parameters.

The proposed solution intends to maximize the performance by looking at the previously mentioned indicators, while keeping a system that is cryptographically secure:

- AER must be reduced to its minimum by reducing the number of authentication bits (NA) and authenticated bits (NN).
- TBA is also reduced to the minimum possible, to allow frequent authentications that increase robustness and reduce TTFAF.
- As regards signal unpredictability, MPT is minimized by insuring that unpredictability is distributed across the navigation message without degrading any of the other parameters. USR is characterized as well.

T_0 (GST_0 sync.) (s)	E5b Sub frame ID	E1-B Content						E1-B Page	E1B Sub frame ID
6	N	Word 6 (2/2)	Res	SAR	Spare	CRC	Res	Odd	N
7	N	Word 7 or 9 (1/2)						Even	N
8	N	Word 7 or 9 (2/2)	Res	SAR	Spare	CRC	Res	Odd	N
9	N	Word 8 or 10 (1/2)						Even	N
10	N	Word 8 or 10 (2/2)	Res	SAR	Spare	CRC	Res	Odd	N
11	N	Reserved (1/2)						Even	N
12	N	Reserved	(2/2)	SAR	Spare	CRC	Res	Odd	N
13	N	Reserved (1/2)						Even	N
14	N	Reserved	(2/2)	SAR	Spare	CRC	Res	Odd	N
15	N	Reserved (1/2)						Even	N
16	N	Reserved	(2/2)	SAR	Spare	CRC	Res	Odd	N
17	N	Reserved (1/2)						Even	N
18	N	Reserved	(2/2)	SAR	Spare	CRC	Res	Odd	N
19	N	Reserved (1/2)						Even	N
20	N	Reserved	(2/2)	SAR	Spare	CRC	Res	Odd	N
21	N	Word 1 (1/2)						Even	N
22	N	Word 1 (2/2)	Res	SAR	Spare	CRC	Res	Odd	N
23	N	Word 3 (1/2)						Even	N
24	N	Word 3 (2/2)	Res	SAR	Spare	CRC	Res	Odd	N
25	N	Word 5 (1/2)						Even	N
26	N	Word 5 (2/2)	Res	SAR	Spare	CRC	Res	Odd	N
27	N	Spare Word (1/2)						Even	N
28	N	Spare Word (2/2)	Res	SAR	Spare	CRC	Res	Odd	N
29	N	Spare Word (1/2)						Even	N
30	N+1	Spare Word (2/2)	Res	SAR	Spare	CRC	Res	Odd	N

Figure 10 –Galileo E1B I/NAV message structure

The 40 bits-per-2-second bandwidth yields 20 bps for a total of 600 bits every I/NAV subframe, after which, in nominal conditions, the I/NAV words are repeated. This 30-second subframe structure has also been taken as a reference for NMA, to facilitate synchronization between the reference time, the navigation data authenticated, and the authentication data. The authentication message structure is therefore synchronized with the Galileo I/NAV navigation message structure.

While a thorough description of each header and field that compose the NMA transmission structure falls out of the scope of this paper, the main data blocks of the proposed implementation are presented in Figure 11 and explained later.

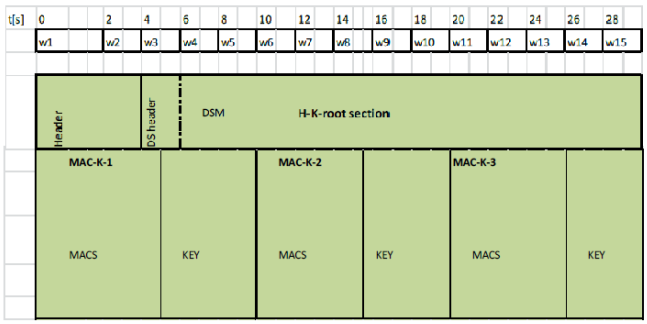

Figure 11 –Galileo NMA message structure within the I/NAV "Reserved 1" field

The top row of Figure 11 shows the subframe time, which goes from 0 to 30 seconds. The second row shows the word order, from 1 to 15 (note that this is different to the Word ID as described in the ICD). Every word, 40 bits will be available for NMA.

As shown in the green area of the figure, the SIS authentication information will be based on two main sections transmitted in parallel:

- "H-K-root" section, with the global header and a digitally signed root key. While the global header is always read, the rest is required only when the user needs a new root key, which should happen very infrequently.
- "MAC-K" section, with the MACs and associated delayed key.

The authentication service will be mainly based on the MAC-K section, which will occupy 32 out of the 40 bits per word, leaving 8 bits per word to the H-K-root section. This implies a total of 120 bits per subframe for the H-K-root section and 480 bits per subframe for the MAC-K sections. The MAC-K authentication implements a TESLA authentication scheme, which is based on the transmission of a MAC, followed by the key used for the MAC (or a related key) with a time delay, as described above. In order to reduce TBA, several MAC-K sections are fitted into a subframe.

To authenticate the keys used in the MAC-K section, a root key (K-root) will be continuously digitally signed and sent in parallel in the H-K-root section. Separating H-K-root and MAC-K sections maintains a constant level of unpredictability and allows more flexibility in the solution design.

Public Key Management

In addition to the SIS information, the system shall provide the public keys through other means than the SIS, allowing the verification of the root key through the DSM. In order to facilitate the use of keys when the receiver is built, several public keys may be required. As public keys will be published just before their validity period (as otherwise the paired private key could be attacked by the knowledge of the public key, even if no information has been yet signed with it), and the requirement to connect the receiver to a network to upload new public keys should be minimised, there must be an overlap in the validity period of several key pairs.

For example, if there were no overlap, a receiver sold just before the end of the validity period of a certain key V_i, would be forced to load the new public key V_{i+1} at the

beginning of its operation. To avoid this, public keys are proposed to be overlapped, so that at a certain time several keys are valid and the system can transmit digital signatures with different keys.

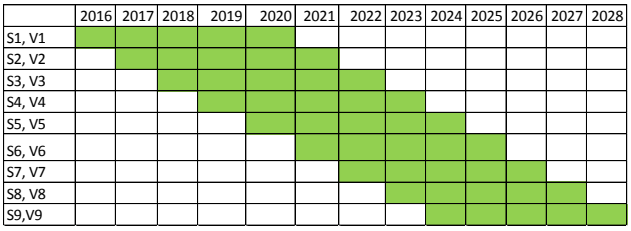

Figure 12 – Example of public-private key pair validity period, assuming each key pair is valid for 5 years

The above figure shows, as a way of example, how several key pairs would be valid for different periods. For example, a receiver manufactured in 2021 could load in its memory the public keys V2, V3, V4, V5 and V6. This would insure a lifetime of 5 years without the need to upgrade the public keys.

A receiver that possesses all keys in force should not have any degradation in the K-root reception performance even if different K-roots are signed with different keys. However, a receiver with a, say, 4 year lifetime without key upload, would receive K-roots for which the public key is not available at the end of its lifetime. It should be noted, however, that the overall H-K-root retrieval is strictly required very rarely in any case.

Public keys can be published in a public site under control of the authentication service provider or through a public key infrastructure. If a private-public key pair has to be revoked, the system will just stop transmitting K-roots with this key, and it will notify through a public internet site (or a public key infrastructure) that the key is revoked. Even receivers without access to the internet will be able to continue functioning, just by using the other public keys.

H-K-Root Section

H-K-root will be transmitted synchronously with the I/NAV subframe. That means that each 30-second subframe a full H-K-root block of 120 bits can be transmitted. The H-K-root section will be composed of a global Header and the K-root digital signature and message (DSM) information. In fact, the global Header and the DSM information could be considered as totally separate sections:

- **Global Header**: it contains parameters like the overall status of the NMA service, the ID of the digital signature that is being transmitted in the DSM section, and a Block ID value that identifies the block within the total number of blocks of the DSM.
- **DS-Header** (Digital Signature Header): header of the root key (K-root) digital signature to be transmitted afterwards, including information like the number of (30-second) blocks that the signature will occupy, or the public key ID (already in possession of the receiver) that will be used to verify the authenticity of the root key K-root. In the current example, it will be transmitted only in the first block of a signature (therefore the dotted line in Figure 11).
- **DSM** (Digital Signature and Message): this field contains the root key (K-root), plus some descriptive fields which shall include the associated time, and the digital signature information. Given that schemes with message recovery may be used, a single section for both the message and the signature has been defined.

The below table shows an example of the transmission of a full H-K-root in the I/NAV subframe. In the proposed example the transmission lasts 5 blocks (i.e. 150 seconds from one satellite), the header is 16 bits, the DS-Header is 8 bits and the DS and message (DMS field) length is up to 512 bits.

field	H	DSH	DSM(1/5)
length	16	8	96
	H	DSM(2/5)	
	16	104	
	H	DSM(3/5)	
	16	104	
	H	DSM(4/5)	
	16	104	
	H	DSM(5/5)	
	16	104	

Table 1 – H-K-root example in 5 subframes/blocks

The type of digital signatures used is still under consideration and is not the main scope of this paper. Schemes as Elliptic-Curve Nyberg-Rueppel, for message recovery, or Schnorr-DSA, as described in [16] and [17] respectively, are plausible options.

In order to speed-up the recovery of a K-root from several satellites, the blocks of a certain signature can be interleaved or offset, as shown in the below example in Table 2. The number of blocks per signature in this example is 5 and satellites are transmitting blocks (B1 to B5) for two different digital signatures (I1 and I2). This would allow the transmission of a 512-bit message and signature (5 * (120 – 16) – 8), where the DS header is 8 bits and the global header is 16 bits.

	t	t+30	t+60	t+90	t+120	t+150
SV1	I1,B1	I1,B2	I1,B3	I1,B4	I1,B5	I2,B1
SV2	I1,B2	I1,B3	I1,B4	I1,B5	I2,B1	I2,B2
SV3	I1,B4	I1,B5	I2,B1	I2,B2	I2,B3	I2,B4
SV4	I1,B1	I1,B2	I1,B3	I1,B4	I1,B5	I2,B1
SV5	I1,B2	I1,B3	I1,B4	I1,B5	I2,B1	I2,B2

Table 2 – H-K-root / header / number of blocks field

In this example, assuming the receiver can receive SV1 to SV5 in good reception conditions, and all satellites are connected to the ground segment, it would complete a K-root in two subframes, or 1 minute. The table shows in green the blocks that are received for the first time, and in orange the blocks that are received for the 2nd, 3rd, etc., time (the selection of green and orange between for example SV1 and SV4 in the first subframe is arbitrary). After 60 seconds, the receiver would have received all DSM blocks from at least one satellite, and after 120 seconds, the receiver should have received all blocks repeated two (block 1), three (block 5), four (blocks 2 and 3) and five (block 4) times, which should provide enough redundancy in degraded conditions.

To add flexibility to the design, a DSM will authenticate not only the K-root and its reference time but also other specific parameters of the chain, as e.g. the key bit length (which can evolve to higher lengths if needed), the MAC size, the one-way function (e.g. SHA-256 or SHA-3), the MAC function (e.g. HMAC-SHA-256, etc.). In this way, the NMA solution gains in flexibility while maintaining its format and specification, which is a highly desirable parameter for a service that can be in operation for several decades.

It should be noted that a receiver does not need to receive a whole block correctly from one satellite. It can just reconstruct a certain block from different satellites, by aggregating the 8-bit parts received correctly from different I/NAV words. This can be verified e.g. by checking the CRC validity of an I/NAV 2-second word, which includes the "Reserved 1" field. If the CRC check is correct, that means that the 8-bit H-K-root portion of that word should be valid as well. It should also be noted that, in principle, there is no reason to change often the DS-ID, other than to use different public-private key pairs, as the same K-root would be transmitted continuously from all satellites for long periods of time.

As a summary, we can say that the H-K-root section will provide the status of the authentication service in the global header and will insure that an authentic [K-root, K-root Time] pair is known to the receiver, allowing it to validate a key of the chain received later. It should be reminded that the receiver will seldom need to decode this section, as it can work with a previously loaded K-root, either from a previous certificate, or from a previous K that is considered authentic (by an older K-root). Note also that different K-roots associated to different times can be sent over the lifetime of a chain, to facilitate the key authentication process.

MAC-K Section

The other section to be transmitted in parallel to the H-K-root will contain the truncated MACs and keys used for authentication. As the H-K-root section consumes 120 bits per subframe, the remaining 480 bits are available for the MAC-K section.

With a key length in the order of 80-100 bits and a truncated MAC length in the order of 10-20 bits, two or three keys can be sent every subframe, one every 15 or 10 seconds, respectively, with its associated MACs. Figure 11 above shows an example with three MAC-K sections per subframe, of 160 bits each.

The MAC-K section is composed by:

- A **MAC section**, which in turn is composed by the **MAC** and a **MAC-Info** section, giving information about the MAC.
- The **key**, which will follow the one-chain multiple-key scheme abovementioned.

The following figure presents the structure of the MAC-K section.

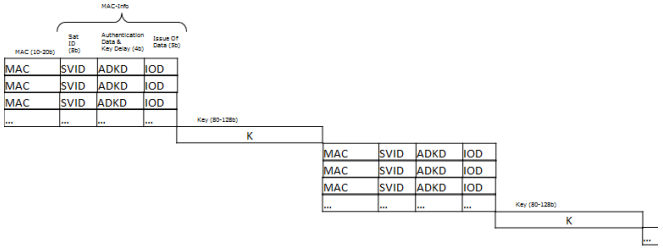

Figure 13 – MAC-K section structure

For a given chain, the output of the MAC function will be truncated (from the MSB) to a length defined in the DSM section. Typically, the permitted values will be between 10-20 bits. In addition to the data, the MACs will authenticate also the system time and the SVID in the following way.

$m = (SVID \parallel GST \parallel navdata)$

$t = trunc_n(MAC(K,m))$

Where

- *m* is the message to be authenticated
- *SVID* is the satellite ID as defined in the MAC-info section.

- *GST* is the Galileo System Time (seconds), as per the OS SIS ICD [1], of the start of the subframe in which the tag (MAC) is transmitted.
- *navdata* is the navigation data authenticated as defined in the MAC-info section.
- *t* is the tag, i.e. the truncated message authentication code transmitted in the signal (depicted as 'MAC' in the figure).
- *trunc(n,p)* is the truncation function whereby the message p is truncated to the n MSB.
- *n* is the truncated MAC length, as defined in the DSM.
- *MAC* is the MAC function used, as defined in the DSM.
- *K* is the key from the one-way chain used for the MAC.

Examples of typical MAC functions used are HMAC-SHA-256, as standardized in [12] and in [23], and CMAC-AES, standardized as Algorithm 5 in [18], and in [19].

The following aspects should be noted about the potential attacks to the MAC authentication process for NMA.

- If the SVID were not added to the MAC, given that all MACs in a MAC-K section are signed with the same key, an attacker could forge the signal and transmit the same navigation data from several satellites, and later replay the first MAC. This attack is prevented by adding the SVID to the authenticated message.

- The GST is added to make the message to be signed always different and increase robustness, even if the navigation data be updated only every hour or so. However, even if it were not added, the MACs would be unpredictable anyway as they use a different key every MAC-K section.

- The signal time is authenticated just by insuring the authenticity of a key with respect to the K-root: if a key of a certain subframe is authenticated using the TOW of that subframe, the TOW must be authentic too as otherwise the hashing process would not lead to the K-root.

The MAC-info section is transmitted contiguously with the MAC and is composed by:

- A Satellite ID field, to allow for cross-authentication of surrounding satellites, either from the same constellation or from others. An 8-bit field has been proposed in the current example allowing for up to 255 satellite IDs.
- An "Authentication Data & Key Delay" field, that can be fit in 4 bits and defined as shown in the below figure.
- A truncated IOD (issue of data) field, which identifies the data to be signed (it may relate to an IOD from the navigation message or another convention may be used, depending on the ADKD case).

field	SVID	ADKD	IOD	*total length*
length	8	4	4	*16*

0	1	2	3	4	5	6	7
Eph & Clk	Iono+	Subframe	Gal F/NAV eph & clk.	GPS L1C eph & clk	SBAS-related TBC	Rsvd	Rsvd
8	9	10	11	12	13	14	15
Rsvd	Rsvd	Rsvd	SLOW-MAC Eph&clk 1 subframe delay	SLOW-MAC Eph&clk 2 subframe delay	SLOW-MAC Eph&clk 3 subframes delay	SLOW-MAC Eph&clk 4 subframes delay	SLOW-MAC Eph&clk 5 subframes delay

Figure 14 –MAC-Info and Authentication Data and Key Delay (ADKD) field

While more ADKD values can be further refined, the field definition is representative of the fact that each transmitted MAC could sign different information not only from different satellites, but also from different signals of the same satellite, and for different sets of information.

For example, in order to reduce AER by authenticating less bits, most of the transmitted MACs could sign only the ephemeris and clock data (ADKD=0), which may be updated at a higher rate than the ionospheric information (ADKD=1).

On the other hand, the overhead due to the dynamic identification of every MAC is significant, as the MAC-info section may be longer than the truncated MAC itself. However, this overhead is compensated with the high flexibility that is obtained by this approach.

Here is a more detailed explanation of the fields for the Galileo system in the ADKD:

- '0', eph&clk: the MAC authenticates the bits of the fields related to the ephemeris, including clock corrections, of the given satellite. It refers to the ephemeris and clock data bits transmitted in the E1 I/NAV message, Words 1 to 4 as per [1]. The SISA field will also be included. The spare bits will not be included.
- '1', iono+: the MAC authenticates the bits of the fields related to the ionospheric corrections, BGDs, satellite health and Galileo system time. It refers to all data bits in E1 I/NAV message Word 5: iono correction, BGDs, HS, DVS and GST.

- '2', subframe: the MAC authenticates the bits of a full subframe. This will include all the Word data bits plus the SAR bits. It will not include other reserved fields or the CRC.
- '11' to '15': In order to prevent attacks whereby the attacker would rebroadcast a right key with forged navigation and MACs, which could happen if the receiver has a time uncertainty in the order of the seconds between the reception of the MACs and the key, the concept of "slow MACs" is added. A "slow MAC" is a MAC generated with a key that will be broadcast some subframes later. For example, if ADKD field is 12, it means that the receiver will get the key associated to that MAC exactly with 2 subframes (60 seconds) delay with respect to the time it would have received it in normal conditions.

The number of navigation bits (NN) to be authenticated would be 544 bits in cases '0' and '11' to '15', 99 bits in case '1', and 2250 bits in case '2'. The latter can be optimized to discard some unused bits.

Bandwidth Allocation Analysis

The following figure presents a bandwidth allocation analysis between MAC lengths and key lengths. Based on this analysis, preferred implementations can be selected to avoid having unused bits in the MAC-K sections.

In the current example, the following input parameters are considered:

- MAC-K sections total bandwidth: 480 bits.
- Number of MAC-K sections per subframe: 3
- Total length per MAC-K section: 160 bits.
- MAC header length (including IOD, SVID, ADKD): 16 bits.

Table 3 shows the number of MACs that can be transmitted in a MAC-K section, for given key lengths and truncated MAC lengths. The table shows in yellow the combinations that yield no spare bits (i.e. all of the 480 bits are used by the MAC-K sections).

	NUMBER OF MACS										MAC length
Key length	10	11	12	13	14	15	16	17	18	19	20
80	3	2	2	2	2	2	2	2	2	2	2
81	3	2	2	2	2	2	2	2	2	2	2
82	3	2	2	2	2	2	2	2	2	2	2
83	2	2	2	2	2	2	2	2	2	2	2
84	2	2	2	2	2	2	2	2	2	2	2
85	2	2	2	2	2	2	2	2	2	2	2
86	2	2	2	2	2	2	2	2	2	2	2
87	2	2	2	2	2	2	2	2	2	2	2
88	2	2	2	2	2	2	2	2	2	2	2
89	2	2	2	2	2	2	2	2	2	2	1
90	2	2	2	2	2	2	2	2	2	2	1
91	2	2	2	2	2	2	2	2	2	1	1
92	2	2	2	2	2	2	2	2	2	1	1
93	2	2	2	2	2	2	2	2	1	1	1
94	2	2	2	2	2	2	2	2	1	1	1
95	2	2	2	2	2	2	2	1	1	1	1
96	2	2	2	2	2	2	2	1	1	1	1
97	2	2	2	2	2	2	1	1	1	1	1
98	2	2	2	2	2	2	1	1	1	1	1
99	2	2	2	2	2	1	1	1	1	1	1
100	2	2	2	2	2	1	1	1	1	1	1
101	2	2	2	2	1	1	1	1	1	1	1
102	2	2	2	2	1	1	1	1	1	1	1
103	2	2	2	1	1	1	1	1	1	1	1
104	2	2	2	1	1	1	1	1	1	1	1
105	2	2	1	1	1	1	1	1	1	1	1
106	2	2	1	1	1	1	1	1	1	1	1
107	2	1	1	1	1	1	1	1	1	1	1
108	2	1	1	1	1	1	1	1	1	1	1
109	1	1	1	1	1	1	1	1	1	1	1
110	1	1	1	1	1	1	1	1	1	1	1
111	1	1	1	1	1	1	1	1	1	1	1
112	1	1	1	1	1	1	1	1	1	1	1

Table 3 – Number of possible MACs for a given MAC length and Key length

It can be shown that by keeping a truncated MAC length of 10 bits, and a key length of 82 bits or less, 3 MACs per MAC-K section can be transmitted, for a total of 9 MACs, i.e. 9 data-authenticated satellites, every 30 seconds.

It is worth mentioning that thanks to the proposed improvements, the high bitrate of Galileo signal data components with respect to other GNSS, the relatively high bandwidth available for authentication from the SoL re-scoping, and the availability of the "reserved 1" field, the Galileo programme has the opportunity to provide a highly robust authentication service more than an order of magnitude faster of what has been proposed in the state of the art for implementation in other systems.

PERFORMANCE COMPARISON

This section characterizes the proposed solution in terms of TBA, AER, MPT and USR. Introducing the cross-authentication approach makes the characterization not as straightforward as if each satellite were self-authenticating. This leads to some extra explanations and indicators described below.

Summarizing, the implementation to be characterized implies:

- Key length of 82 bits
- MAC length of 10 bits
- MAC-Info length of 16 bits
- One MAC-K section every 10 seconds

The characterization below excludes the processing of the H-K-root. It is assumed that the receiver has an authentic K-root.

Time Between Authentications
As it has been mentioned above, the TBA obtained is 10 seconds, i.e. every 10 seconds a key and 3 MACs (or tags) are transmitted. A satellite can self-authenticate one, two or three times per 30-second subframe, leaving the remaining space for authenticating other satellites or other data. Therefore, for a given satellite, the data from that satellite, plus two other satellites, can be authenticated.

Maximum Predictable Time
This parameter depends on how the encoding and interleaving of the Galileo I/NAV message encodes the unpredictable bits in symbols (some of which will be predictable and some of which not) and spreads them across the transmitted message. It comes out that all of the unpredictable symbols of every 2-second word are transmitted in a period of 0.4 to 0.5 seconds, leaving the remaining 1.6 to 1.5 seconds fully predictable. As a reference, a MPT of 1.6 seconds will be taken.

This is based on the assumption that every 40-bit "Reserved 1" field contains unpredictable information. As every "Reserved 1" field contains 32 bits of a MAC-K section, and the longest predictable interval of a MAC-K section is the 16 bits of the MAC-info field, all the "Reserved 1" fields will contain some unpredictability.

Unpredictable Symbol Ratio
This parameter is calculated under the assumption that all symbols are predictable except the MAC and the key bits, excluding the last 10 bits of the key, to avoid attacks whereby the last key bits are deduced and rebroadcast. That gives a total of 3*(3*10+72) = 306 bits, or 306 symbols per subframe. The total of 7500 symbols per subframe (250 sps * 30 seconds) yields a USR of 4.08%.

That means that, in average, there are 0.0408*250 = 10.2 symbols per second that are unpredictable, from which an anti-replay test statistic could be derived.

Authentication Error Rate
As shown in equation (1), AER depends on bit error rate (BER) and number of navigation and authentication bits (NNA) to be demodulated. BER, in turn, can be bounded by the following formula (3) [20], assuming a soft decision Viterbi decoder and a static receiver under an AWGN channel and stable PLL tracking:

$$BER \leq \frac{1}{2} \cdot (36D^{10} + 21D^{12} + 1404D^{14} + 11633D^{16}) \quad (3)$$

Where

$$D = \mathrm{e}^{-\frac{C/N_0}{2Rb}} \quad (4)$$

And *Rb* is the number of bits per second. AER vs C/N_0 is shown in Figure 15 for three cases, where ADKD is '0', '1' and '2'.

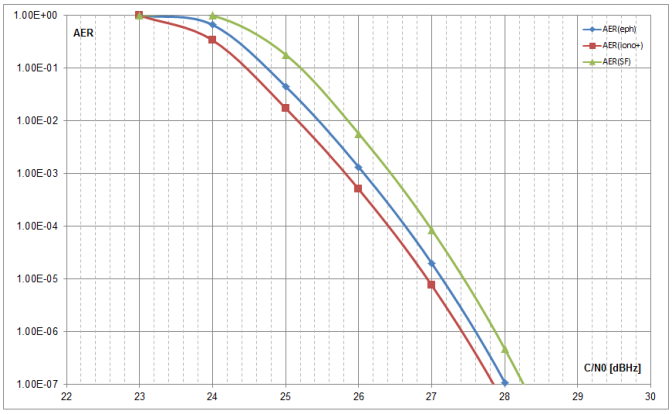

Figure 15 –AER vs C/N_0 for I/NAV ephemeris and clock authentication (ADKD=0 –blue-, 1 –red-, 2 –green-)

The figure shows that, under these assumptions, very low AER are obtained even at low C/N_0 values. For example, an AER of 1% is obtained with a C/N_0 between 25dBHz and 26 dBHz for all cases. It also shows that below 24 dBHz NMA is barely usable. The authentication performances are as expected and in line with the I/NAV demodulation performances. A receiver able to successfully demodulate the navigation data should also authenticate with a reasonably high availability.

The following table summarises the performances of the proposed authentication solution example.

Indicator	**Value**	**Comments**
NN	436 bits 99 bits 2250 bits	For ADKD=0 (top), 1 (middle) and 2 (bottom).
NA	108 bits	MAC-info: 16 bits MAC: 10 bits Key: 82 bits
4SAT NA	186 bits	MAC-info: 4*16 MAC: 4*10 Key: 82
NNA	544 bits 207 bits 2358 bits	For ADKD=0 (top), 1 (middle) and 2 (bottom).
TBA	**10 sec**	Including up to 3

		authenticated satellites.
MPT	**~1.6 sec**	
USR	**4.08%**	306 unpredictable symbols every 30 seconds.
AER	**See** Figure 15	

Table 4 – Performance characterization of the proposed authentication solution

It should be noted that other indicators as average TBA and TTFAF can be computed with the above indicators, as shown in previous sections.

COMPARISON OF DIFFERENT AUTHENTICATION SOLUTIONS

As the AER result does not show in an evident way the advantages of the proposed solution compared to other authentication solutions, the following figure presents the "four satellite AER" for a given BER. It represents probability that 4 satellites are correctly authenticated under a noisy channel (AWGN in this case), versus a given BER. For clarity, only NA (as opposed to NNA) has been considered. This may represent the real case whereby a receiver has already received the slow time varying navigation information (ephemeris, ionospheric model, etc.), while it still requires successful authentications to verify it is not under a replay attack. The first 3 cases are already proposed in [5]:

- AER-NA-DS: 466-bit digital signature, one per satellite.
- AE-NA-STD-TESLA: Standard TESLA approach, with 224-bit keys and 15-bit truncated MACs.
- AER-NA-1C-TESLA: 1-chain TESLA approach, with 224-bit keys and 15-bit truncated MACs.
- AER-NA-1C-TESLA-F: 1-chain TESLA "fast" approach as previously described, with 10-bit MACs and 82-bit keys.

As expected, a lower NA leads to a higher availability of the service in equal conditions, which makes the service more robust. For example, to achieve a 4-satellite NA-only AER of 10% a BER of $5*10^{-5}$ in case of a digital signature is required, but only a BER of around 10^{-3} is required in the "fast TESLA" case described in this note.

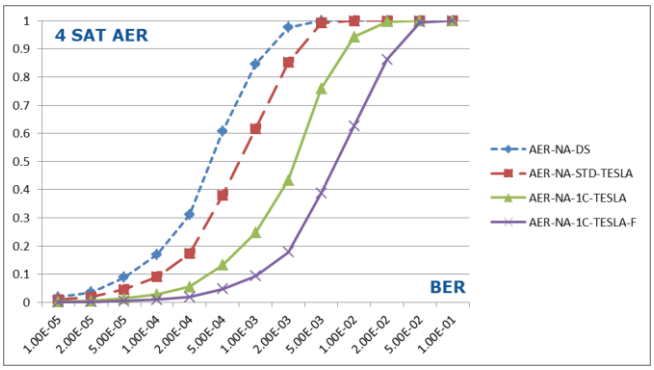

Figure 16 – Four-satellite AER vs BER

CONCLUSIONS

This paper has presented the main motivations for the Galileo programme to study the provision of an open navigation message authentication (NMA) service. Thanks to the current system and signal design, which includes a high bitrate relative to other GNSS, bandwidth available from the former safety-of-life service, and an external transmission channel to transmit 20bps continuously, a highly performant NMA service can be provided at a low effort.

The paper also discusses the value of NMA against replay attacks, justifying why signal unpredictability is maximised, and develops a framework to analyse NMA solutions from a GNSS service provider point of view. Time Between Authentications (TBA) and Authentication Error Rate (AER) arise as the main performance indicators.

Some novel concepts to provide authentication are presented, including the translation of the cross-authentication concept to a TESLA implementation, whereby all satellites transmit keys from the same chain and MACs for contiguous satellites, allowing receivers to use authentication data from some satellites to authenticate others. The paper then discusses some security considerations about the use of a single chain at a time, and the reduction of key and MAC sizes in order to fit with the constraints of a very low bandwidth channel as is the case for GNSS signals.

A concrete implementation of the concept is proposed in the Galileo E1B I/NAV message structure, using the "Reserved 1" field that provides 40 bits every other second. The data fields required for the general understanding of the solution are presented. They are divided in a "H-K-root" section to transmit a header and a signed root key, and a "MAC-K" section to transmit the MACs and keys used regularly for authentication.

A flexible approach is presented whereby the SIS data can inform the receiver about the length of the MACs and keys, and other parameters, in a way that the system robustness can evolve over its lifetime if new threats appear.

The proposed solution is then characterised, mainly in terms of TBA, AER and signal unpredictability through MPT and USR. The indicators show that a highly performant solution can be implemented in Galileo. Particularly, a TBA as low as 10 seconds provides a very low time to first authenticated fix and a high robustness, while maintaining a sufficient cryptographic security level. AER can also be highly reduced, thanks to the low number of bits required for authentication and the possibility to get most of them from the best visible satellite(s), maximizing the availability of the authentication service even in degraded conditions.

FURTHER WORK

The work presented in this paper relates to the implementation of NMA. However, the framework proposed can also be used for spreading code-based authentication solutions. The currently proposed solution might evolve in future Galileo (or other GNSS) generations towards the interleaving of unpredictable chips in the spreading codes generated with a key transmitted every 10 seconds, in which case the current NMA structure could be maintained and complemented in future satellite payloads in the following decades, for example with signal watermarking solutions at spreading code level [22].

Another potentially relevant work to be done to maximize benefits from this solution is the definition of the level of protection provided that some satellites in the position solution may be:

- not authenticated at all.
- only data-authenticated.
- data and TOA-authenticated, where TOA authentication has an associated confidence level based on a test statistic.
- the above, combined with other authentication measures at other stages of the receiver, as AGC, J/N detectors, trusted clocks, inertial sensors, antenna arrays, etc.

Also, for fully proofed location applications, GNSS authentication shall be combined with other cyber-security measures at the receiver, user terminal, communication channel and server [21].

In the short to mid-term, the proposed authentication solution, or an evolution of it, will be transmitted in the Galileo signal-in-space as part of the AALECS (Authentic and Accurate Location Experimentation with the Commercial Service) project [24], in either the E1B or the E6B Galileo signals, for testing purposes. A proof-of-concept platform has already been developed and tested with real SIS, showing the feasibility of Galileo authentication [25]. The characterization of the proposed approach in degraded conditions, with both simulated and real signals, is foreseen to be performed as part of this project.

Eventually, the main area of the future work will hopefully be to develop, test, qualify, operate and maintain an open and civil GNSS authentication service for the benefit of the worldwide GNSS civil community in the years to come.

ACKNOWLEDGMENTS

The authors thank the AALECS team members involved in authentication (E. Carbonell, P. Walker, O. Pozzobon, S. Fantinato, M. Canale) and the GNSS team at the Joint Research Centre's IPSC, A6 Unit, Ispra.

DISCLAIMER

The material in this paper does not represent any official view of the EU or its Member States. The solutions proposed will not necessarily be included in future Galileo operational services.

REFERENCES

[1] European Union (2010). European GNSS (Galileo) Open Service Signal In Space Interface Control Document. OD SIS ICD, Issue 1.1, September 2010

[2] GPS Interface Specification IS-GPS-200 21, Mar 2014.

[3] Humphreys, T. (2013). Detection Strategy for Cryptographic GNSS Anti-Spoofing. IEEE Transactions on Aerospace and Electronic Systems (Volume:49, Issue: 2)

[4] Wesson, K., Rothlisberger, M., Humphreys, T. (2012). Practical Cryptographic Civil GPS Signal Authentication. The Journal of the Institute of Navigation, February 2012

[5] Fernández-Hernández, I. (2014), GNSS Authentication: Design Parameters and Service Concepts, Proceedings of the European Navigation Conference 2014, Rotterdam, Netherlands, n.150.

[6] Perrig, A., Canetti, R., Tygar, J. D., Song, D. (2002), The TESLA Broadcast Authentication Protocol. CryptoBytes, 5:2, Summer/Fall 2002, pp. 2-13

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

[8] Lo, S., Enge, P., Aviation Augmentation System Broadcasts, IEEE/ION Position Location and Navigation Symposium (PLANS), 2010

[9] Perez-Bartolomé, J., Maufroid, X., Fernández-Hernández, I., López-Salcedo, J.A., Seco-Granados, G., Introduction to Galileo System, Galileo Positioning Technology, Springer, 2014.

[10] Menezes, A.J., Van Oorschot , P., Vanstone, S.A. (1997). Handbook of Applied Cryptography, CRCPress, Inc.

[11] Guilford, J., Yap, K., Gopal, V., IA Architects - Fast SHA-256 Implementations on Intel Architecture Processors - White Paper - Intel Corporation – May 2012

[12] Krawczyk, H., Bellare, M., Canetti R. - RFC 2104 (1997), Keyed-Hashing for Message Authentication. Network Working Group

[13] Algorithms, Key Sizes and Parameters Report, 2013 recommendations. ENISA, October 2013.

[14] FIPS PUB 180-4 - Federal Information Processing Standards Publication - Secure Hash Standard (SHS) - National Institute of Standards and Technology Gaithersburg, MD 20899-8900 - March 2012

[15] DRAFT FIPS PUB 202 - Federal Information Processing Standards Publication - SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions - National Institute of Standards and Technology Gaithersburg, MD 20899-8900 - May 2014

[16] ISO/IEC 9797-3. Information technology – Security techniques – Digital signatures giving message recovery – Part 3: Discrete logarithm based mechanisms. International Organization for Standardization

[17] ISO/IEC 14888-3. Information technology – Security techniques – Digital signatures with appendix – Part 3: Discrete logarithm based mechanisms – Amendment 1. International Organization for Standardization, 2009.

[18] ISO/IEC 9797-1:2011. Information technology – Security techniques – Digital signatures giving message recovery – Part 1: Mechanisms using a block cipher. International Organization for Standardization, 2011.

[19] NIST SP800-38B. Recommendation for block cipher modes of operation: the CMAC mode for authentication.

[20] Kaplan, E.D., Hegarty, C.J. Understanding GPS Principles and Applications (2nd Edition). Artech House Inc., 2006.

[21] Pujante, A., Location Authentication - Enabling New Smartphone Apps, Inside GNSS, Issue May/June 2014, p 48-52.

[22] Scott, L. (2003). Anti-Spoofing & Authenticated Signal Architectures for Civil Navigation Systems. ION GPS 2003, Portland, Oregon USA

[23] FIPS PUB 198-1 - Federal Information Processing Standards Publication – The Keyed-Hash Message Authentication Code (HMAC) - National Institute of Standards and Technology Gaithersburg, MD 20899-8900 – July 2008

[24] Rodríguez, I., Tobías, G., Calle, D., Martín, J.M., Pozzobon, O., Canale, M., Maharaj, D., Walker, P., Göhler, E., Toor, P., Fernández, I., Preparing for the Galileo Commercial Service – Proof of Concept and Demonstrator Development, Proceedings of the ION GNSS+ 2014, Tampa, U.S.

[25] Calle, J. D., Carbonell, E., Rodríguez, I., Tobías, G., Göhler, E., Pozzobon, O., Cannale, M., Fernández, I., Galileo Commercial Service from the Early Definition to the Early Proof-Of-Concept, Proceedings of the ION GNSS+ 2014, Tampa, U.S.