A Blueprint for Civil GPS Navigation Message Authentication

Andrew Kerns, Kyle Wesson, and Todd Humphreys

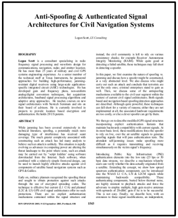
Radionavigation Laboratory
University of Texas at Austin

Applied Research Laboratories
University of Texas at Austin

May 6, 2014

NMA is Gaining Traction

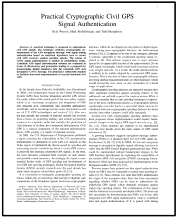
Scott, 2003



Anti-Spoofing & Authenticated Signal Architectures for Civil Navigation Systems

Logan Scott, *LS Consulting*

Wesson et al., 2012



Practical Cryptographic Civil GPS Signal Authentication

Kyle Wesson, Mark Rothlisberger, and Todd Humphreys

...

2014

GPS L2C

Galileo OS

Tradeoff: Overhead vs. Authentication Frequency

- Would you like authentication every 36 seconds?

Tradeoff: Overhead vs. Authentication Frequency

- Would you like authentication every 36 seconds?

uses 100% of available CNAV message slots

Tradeoff: Overhead vs. Authentication Frequency

- Would you like authentication every 36 seconds?

  uses 100% of available CNAV message slots

- What if NMA was restricted to 2% of the CNAV data rate?

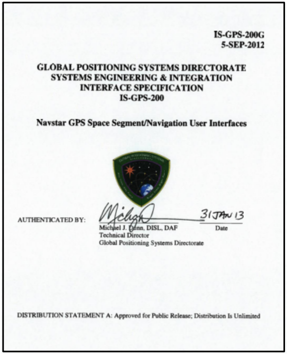
  is it still useful?

case study: 1 message every 9 minutes

Outline

- Introduction to NMA
- Two schools of thought: ECDSA or TESLA?
- Fitting NMA data into CNAV

IS-GPS-200


IS-GPS-200G
5-SEP-2012

GLOBAL POSITIONING SYSTEMS DIRECTORATE
SYSTEMS ENGINEERING & INTEGRATION
INTERFACE SPECIFICATION
IS-GPS-200

Navstar GPS Space Segment/Navigation User Interfaces

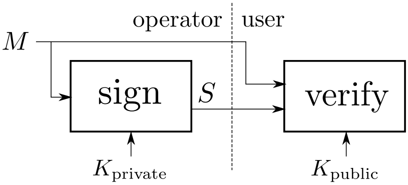
AUTHENTICATED BY: Michael J. Dunn, DISL, DAF, Technical Director, Global Positioning Systems Directorate — 31 JAN 13 (Date)

DISTRIBUTION STATEMENT A: Approved for Public Release; Distribution Is Unlimited

What is GNSS NMA?

Technique to add cryptographic authentication to GNSS navigation data stream [1, 2, 3, 4, 5]

1. GNSS operator signs a section of navigation data M
2. digital signature S is broadcast in navigation data stream
3. users verify (M, S)



Anti-Spoofing with NMA

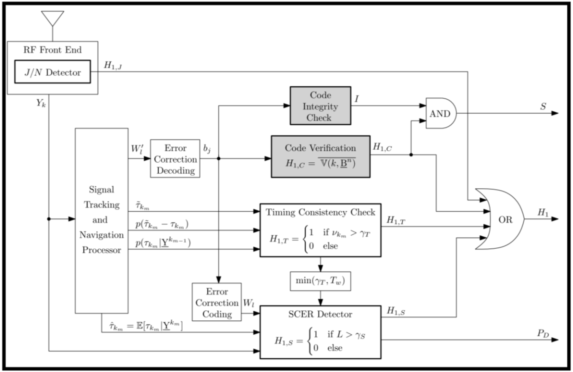
NMA is an attractive anti-spoofing measure:

- minimal burden on a low-cost receiver
- backward compatible
- provides **data authentication**
- enables **signal authentication**

Signal Authentication with NMA

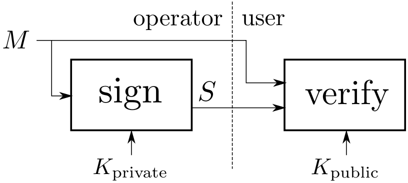
Signal authentication technique developed in [4] and [5]

- ensures underlying GNSS signal is authentic, not just navigation data
- requires μs-level time offset $\delta t_{\text{RX}} < \gamma$



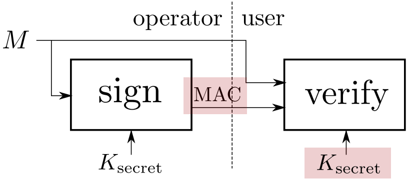
NMA Requires Asymmetric Cryptography

Public-key authentication



- S is a *digital signature*
- users only have public key $\rightarrow$ cannot sign messages

Symmetric-key authentication



- MAC is a *message authentication code*
- users have secret key $\rightarrow$ can sign messages
- length(MAC) < length(S)

What is the required bit strength?

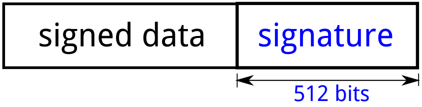
NIST-recommended security level for authentication [6]

b_s	secure until
112	2030
128	> 2030

assume equivalent symmetric-key bit strength b_s = 128 bits

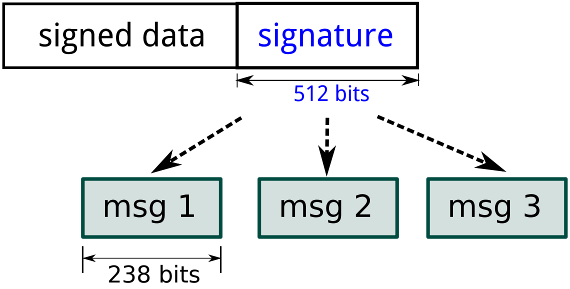
Elliptic Curve Digital Signature Algorithm (ECDSA)

- Standardized public-key authentication scheme
- Assuming P-256 ($b_s = 128$), digital signature is 512 bits



Elliptic Curve Digital Signature Algorithm (ECDSA)

- Standardized public-key authentication scheme
- Assuming P-256 (b_s = 128), digital signature is 512 bits



$T_{\text{ba}} \approx$ 27 minutes

Timed Efficient Stream Loss-Tolerant Authentication

public key — root key

$$K_0 \xleftarrow[h(\cdot)]{} K_1 \xleftarrow[h(\cdot)]{} \cdots \xleftarrow[h(\cdot)]{} K_{N-1} \xleftarrow[h(\cdot)]{} K_N$$

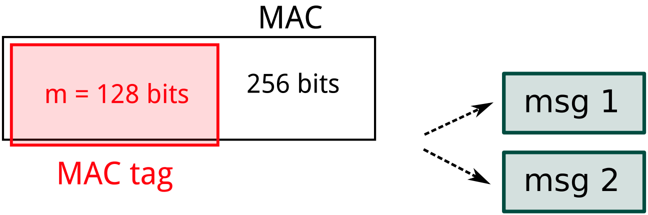
TESLA protocol [7]

- Generate one-way chain of keys
- Broadcast message authentication code MAC (M_i, K_i)
- After delay, broadcast K_i as plaintext
- Receiver checks both MAC and $h^k(K_i) = K_{i-k}$

Note: variant of TESLA where each key is only used for one MAC

TESLA Truncation

- Generate MAC by applying hash function to (M, K_i)
- Truncate MAC to m left-most bits, yielding **MAC tag** [8]



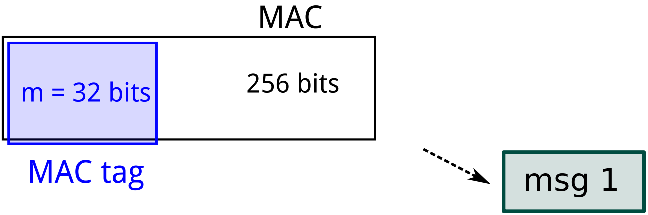
128+m = 256 bits per authentication

$T_{\text{ba}} \approx 18$ minutes

Assume P-256 and 1 NMA-dedicated CNAV message per 9 minutes.

TESLA Truncation

- Generate MAC by applying hash function to (M, K_i)
- Truncate MAC to m left-most bits, yielding **MAC tag** [8]



$128{+}m =$ 160 bits per authentication

$T_{\text{ba}} \approx$ 9 minutes

Assume P-256 and 1 NMA-dedicated CNAV message per 9 minutes.

TESLA Truncation

What is the effect of decreasing m?

Key recovery

- discover a future element of the key chain, or an alternate key that, once the one-way function is applied, matches a previously-disclosed key
- 2^{128} complexity
- decreasing m does not aid attack

MAC tag forgery

TESLA Truncation

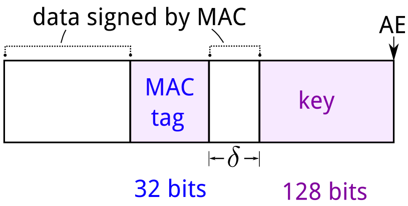
What is the effect of decreasing m?

Key recovery

MAC tag forgery

- forge message or MAC tag without knowing if the MAC tag will pass the victim receiver's verification test
- MAC tags appear random to attacker $\rightarrow$ probability of successfully forging a specific MAC tag is 2^{-m}
- **Ex:** $m = 32$, forgery attempt every 144 seconds for 10 years $\rightarrow$ 1 in 2,000 success rate
- NIST recommends $m \geq 32$ [9]

TESLA Format



- delay δ is critical: key is secret before the delay, but public afterward
- security condition $|\delta t_{\text{RX}}| < \delta$ must hold
- **Ex:** $\delta = 880$ ms

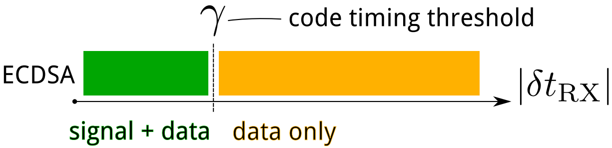
TESLA or ECDSA?

TESLA advantages

- **Lower overhead:** for fixed $b_s = 128$ bits, reduce overhead for one authentication from 512 bits to 160 bits

TESLA disadvantages

- Not standardized
- **Requires approximate time**, $|\delta t_{\text{RX}}| < \delta$



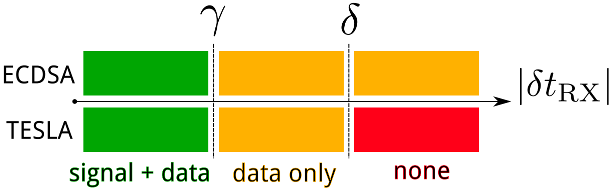
TESLA or ECDSA?

TESLA advantages

- **Lower overhead:** for fixed $b_s = 128$ bits, reduce overhead for one authentication from 512 bits to 160 bits

TESLA disadvantages

- Not standardized
- **Requires approximate time**, $|\delta t_{\mathrm{RX}}| < \delta$



Goal: low overhead without ignoring users in the red box

Hybrid NMA

- auth. spaced equally in time (T_{ba}), but vary in *type*
- k consecutive TESLA type
- followed by one ECDSA type

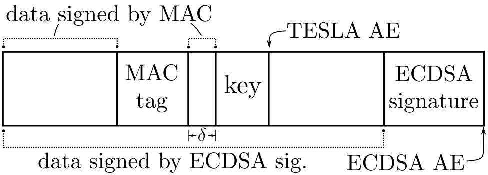


Figure: $k = 1$ hybrid NMA data stream

- only 1 of $(k+1)$ authentications is ECDSA type $\rightarrow$ **low overhead**
- all data signed by ECDSA $\rightarrow$ **cryptographic data authentication** $\forall \delta t_{\text{RX}}$

Three Ways To Transmit NMA Data in CNAV

Data for $(k+1)$ authentications split into

- $238N_{\text{arb}}$ bits in new NMA messages
- $149N_{\text{clk}}$ bits in new clock+NMA messages
- N_e bits exploited from other messages

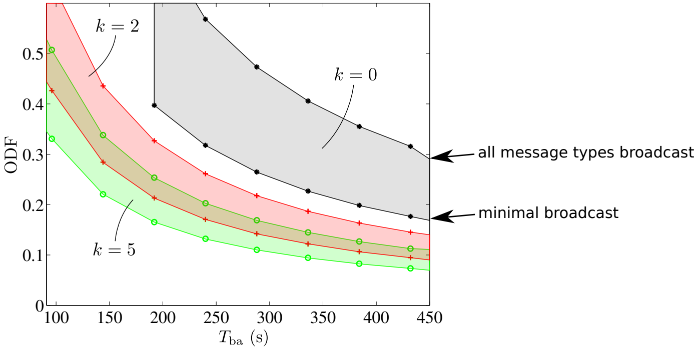
Select $(N_{\text{arb}}, N_{\text{clk}}, N_e)$ to minimize *open data fraction*

$$\text{ODF} = \frac{149N_{\text{clk}} + 238N_{\text{arb}}}{149O_{\text{clk}} + 238O_{\text{arb}}}$$

where $O_{\text{arb}}, O_{\text{clk}}$ are the number of open slots.

Cost Versus Performance

Example result when $N_e = 0$



Example Message Definition

- Choose $k = 5 \quad \rightarrow \quad$ 1 in 6 authentications is ECDSA type
- Choose $N_{\text{clk}} = N_e = 0$

MT	bits	contents
NMA-1	1-32	MAC tag
	38-88	$S_i,\ i \in 1, \ldots, 5$
	89-110	salt
	111-238	TESLA key
NMA-2	1-232	S_6
	233-238	salt

Example Message Definition

- Choose $k = 5 \quad \rightarrow \quad$ 1 in 6 authentications is ECDSA type
- Choose $N_{\text{clk}} = N_e = 0$

MT	bits	contents
NMA-1	1-32	MAC tag
	38-88	$S_i,\ i \in 1, \ldots, 5$
	89-110	salt
	111-238	TESLA key
NMA-2	1-232	S_6
	233-238	salt

$T_{\text{ba}} \approx$ 9 minutes

Conclusions

More efficient NMA without significant security compromises

- TESLA MAC truncation to $m = 32$
- hybrid NMA with all data signed by ECDSA
- optimal $(N_{\text{arb}}, N_{\text{clk}}, N_e)$ w.r.t. ODF cost metric

Conclusions

More efficient NMA without significant security compromises

- TESLA MAC truncation to $m = 32$
- hybrid NMA with all data signed by ECDSA
- optimal $(N_{\text{arb}}, N_{\text{clk}}, N_e)$ w.r.t. ODF cost metric

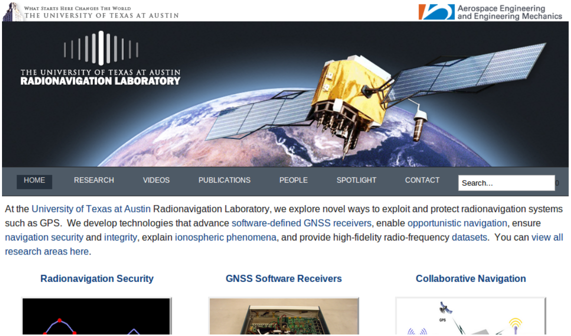
Case study

- 2% of CNAV data rate
- ODF = 6% - 9%

$T_{\text{ba}} \approx$ 9 minutes

Questions?

radionavlab.ae.utexas.edu



References I

[1] L. Scott, “Anti-spoofing and authenticated signal architectures for civil navigation systems,” in *Proceedings of the ION GNSS Meeting*, (Portland, Oregon), pp. 1542–1552, Institute of Navigation, 2003.

[2] G. Becker, S. Lo, D. De Lorenzo, D. Qiu, C. Paar, and P. Enge, “Efficient authentication mechanisms for navigation systems—a radio-navigation case study,” in *Proceedings of the ION GNSS Meeting*, (Savannah, Georgia), Institute of Navigation, September 2009.

[3] S. C. Lo and P. K. Enge, “Authenticating aviation augmentation system broadcasts,” in *Proceedings of the IEEE/ION PLANS Meeting*, (Palm Springs, California), pp. 708–717, Institute of Navigation, 2010.

[4] K. Wesson, M. Rothlisberger, and T. E. Humphreys, “Practical cryptographic civil GPS signal authentication,” *NAVIGATION, Journal of the Institute of Navigation*, vol. 59, no. 3, pp. 177–193, 2012.

[5] T. E. Humphreys, “Detection strategy for cryptographic GNSS anti-spoofing,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 2, pp. 1073–1090, 2013.

[6] NIST, “Digital signature standard,” FIPS PUB 186-4, National Institute of Standards and Technology, July 2013.

References II

[7] A. Perrig, R. Canetti, J. Tygar, and D. Song, "The TESLA broadcast authentication protocol," *RSA CryptoBytes*, vol. 5, no. 2, pp. 2–13, 2002.

[8] NIST, "The keyed-hash message authentication code," FIPS PUB 198-1, National Institute of Standards and Technology, July 2008.

[9] Q. Dang, "Recommendation for applications using approved hash algorithms (revised)," SP 800-107, National Institute of Standards and Technology, Aug. 2007.

[10] Anon., "Systems engineering and integration Interface Specification IS-GPS-200G," tech. rep., Global Positioning System Directorate, 2012. `http://www.gps.gov/technical/icwg/`.

[11] Anon., "ECC brainpool standard curves and curve generation v. 1.0," tech. rep., ECC Brainpool, October 2005.

[12] M. Lochter and J. Merkle, "Elliptic curve cryptography (ECC) brainpool standard curves and curve generation," RFC 5639, Internet Engineering Task Force, March 2010. `http://tools.ietf.org/html/rfc5639`.

[13] NIST, "Recommendation for key management—Part I: General (revision 3)," SP 800-57, National Institute of Standards and Technology, July 2012.

GPS L2 CNAV Specification

CNAV message broadcast intervals [10]

MT	*Contents*	*Minimal*	*Maximal*	*Unallocated*
10	Ephemeris 1	48 sec.	48 sec.	3 bits
11	Ephemeris 2	48 sec.	48 sec.	7 bits
3*	Clock	48 sec.	48 sec.	up to 149 bits
30	Clock, ISC/IONO	288 sec.	288 sec.	12 bits
33	Clock, UTC	288 sec.	288 sec.	51 bits
35	Clock, GGTO	N/A	288 sec.	81 bits
32	Clock, EOP	N/A	30 min.	N/A
37	Clock, Midi Alm.	N/A	32 per 120 min.	N/A
31	Clock, Red. Alm.	N/A	20 min.	N/A
12	Reduced Alm.	N/A	4 per 20 min.	N/A
13	Diff. Corrections	N/A	30 min.	N/A
14	Diff. Corrections	N/A	30 min.	N/A

MT-10	MT-11	clock	arbitrary

ECDSA Curve Selection

ECDSA curves with $b_s = 128$

		curve: *random*	curve: *Koblitz*
field	*binary*	B-283	K-283
field	*prime*	P-256[a]	secp256k1

[a] Also brainpoolP256r1 and brainpoolP256t1 from ECC Brainpool [11][12]

Assume prime field $\rightarrow$ 512-bit signature

Key Distribution

- PKC contains ECDSA and TESLA public keys, period of validity, etc.
- Maximum key period is 1-3 years [13]
- Easily distributed to users with a secure side channel
- Standalone receivers use over-the-air re-keying
    - Initial key inserted by manufacturer
    - Broadcast PKCs are verified via NMA using current key