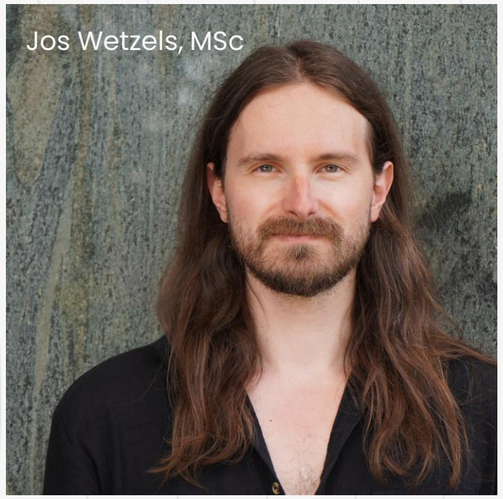
August 2025

2 COPS 2 BROADCASTING

TETRA End-To-End Under Scrutiny

By Midnight Blue



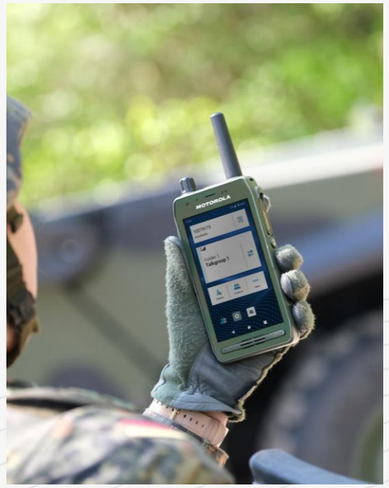
Midnight Blue

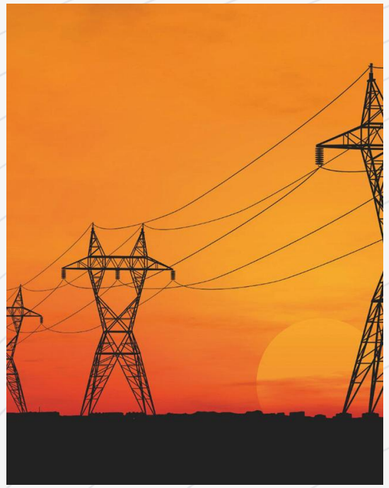
Background

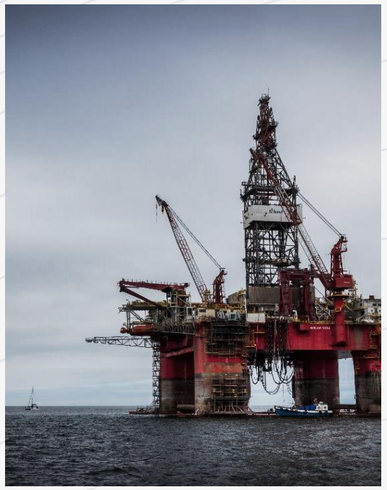
(what you absolutely need to know)

What is TETRA?

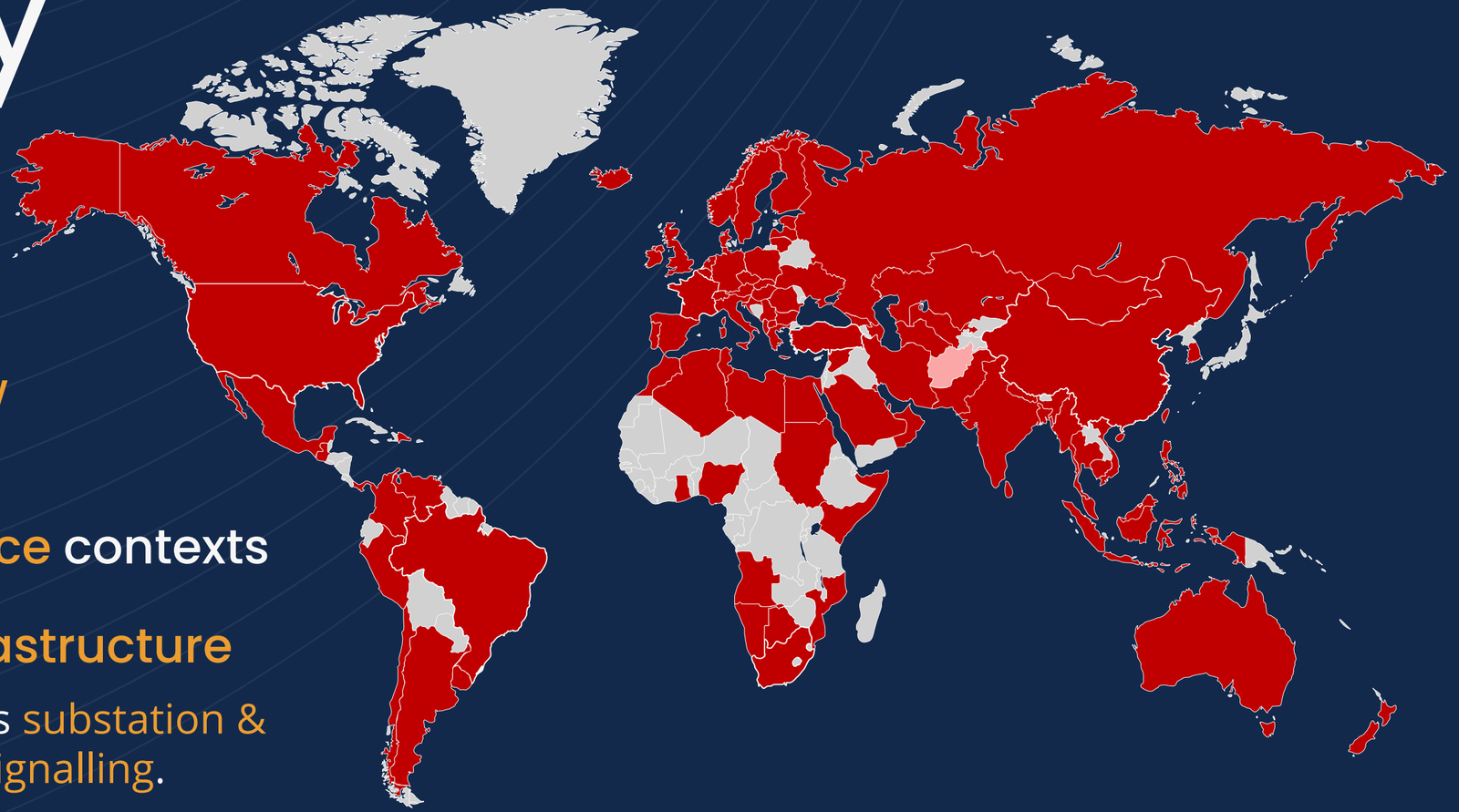
- Globally used radio technology
  - Competes with P25, DMR, TETRAPOL
- Standardized in 1995 by ETSI
  - Known for GSM, 3G/4G/5G, GMR, etc.
- Used for voice & data communications incl. machine-to-machine
- Historically secretive approach to security







Globally used



- Leading standard for law enforcement
- Use in military, intelligence contexts
- Popular with Critical infrastructure
  - Also for SCADA WAN, such as substation & pipeline control, or railway signalling.

*Based on OSINT

Scrutinizing TETRA

- **TETRA cryptography was closely held secret since 1995**
  - Safe to assume most major states had specs
- **In 2023, we reverse-engineered all crypto in the ETSI TETRA standard – which excludes E2EE**
- **We identified significant vulnerabilities in the standard**
  - And 12 more CVEs in base stations, mobile devices and microcontroller ROM code

TETRA BURST Summary

- Backdoored TEA1 cipher offering 32 instead of 80 bits of security
- Keystream recovery attack, regardless of cipher
  - Full breach of confidentiality and integrity
  - Harder to carry out in practice
- Other vulns not relevant for today

Mitigative efforts

following our disclosures

- **Standards revision**
  - ETSI TS 100 392-7 V4.1.1 (2022-10)

    “if the [network time] deviates from the expected value, the MS should [take action]”
- **Large scale patching efforts**
- **Migration away from TEA1**
  - For instance, through dual-cipher networks
- **End-to-end as a mitigation**
  - Expensive, proprietary

TETRA security

Public standard, proprietary crypto

- **TAA suite**
  - Authentication, OTAR
  - Identity encryption
  - Remote disable
- **TEA Air Interface Encryption algorithms**
  - Voice and data (Air Interface Encryption (AIE))
    - **TEA1**, **TEA4**, **TEA7**: Readily exportable
    - **TEA2**, **TEA5**: European public safety
    - **TEA3**, **TEA6**: Extra-European public safety
    - *All vulnerable to non-crypto keystream recovery attack*
- **End-to-End**
  - Super secretive topic

Today's motivation

- **TETRA End-To-End encryption has never been more essential**
  - Very sensitive use cases
  - Defense against suboptimal AIE security
  - Nordic countries will require next-gen E2EE interoperability with TETRA E2EE*

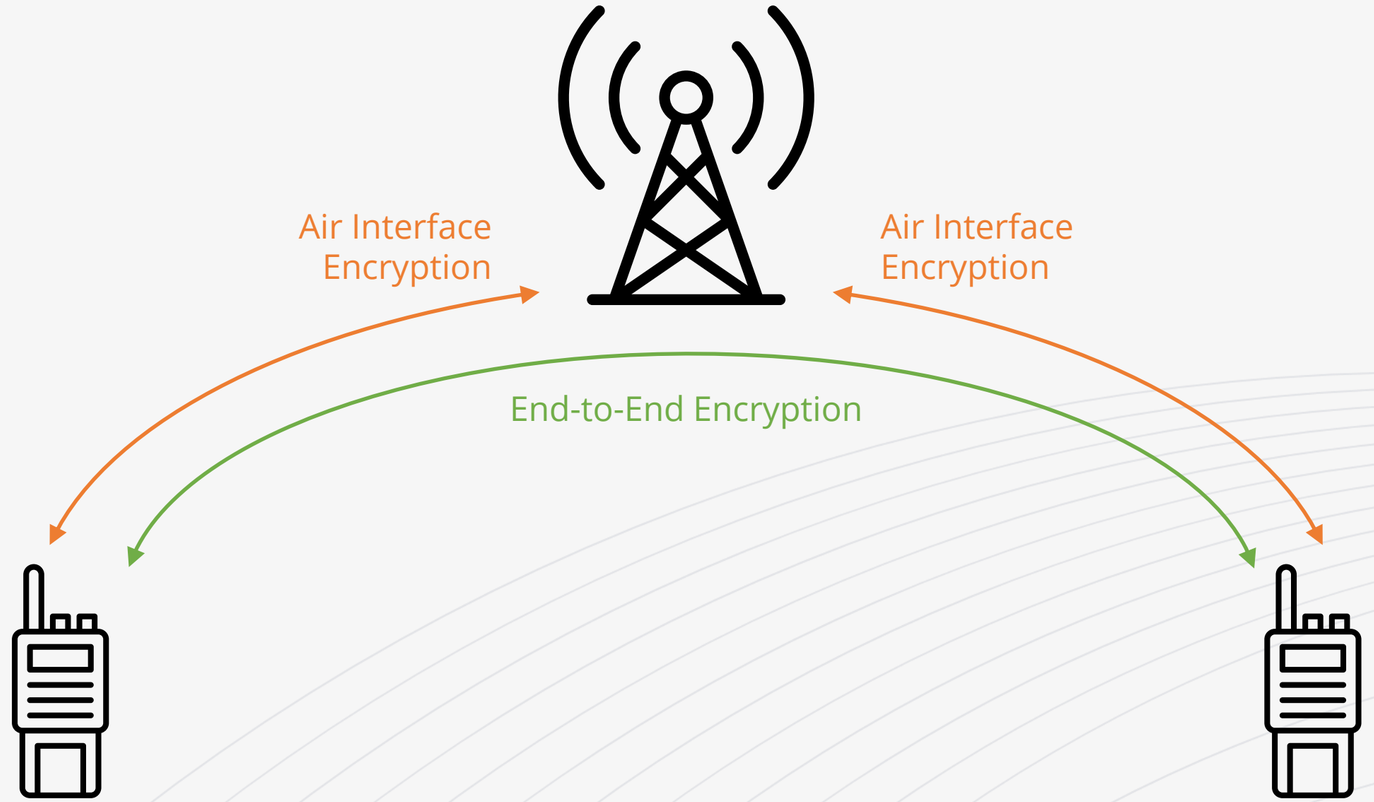
nlnet FOUNDATION

"The device should have the capability to support end-to-end encryption solutions that are interoperable with currently deployed end-to-end solutions in TETRA devices."

* https://www.dsb.no/siteassets/nodnett/nytt-nodnett/ppdr-rugged-handheld-device-for-heavy-use-nccom-whitepaper.pdf

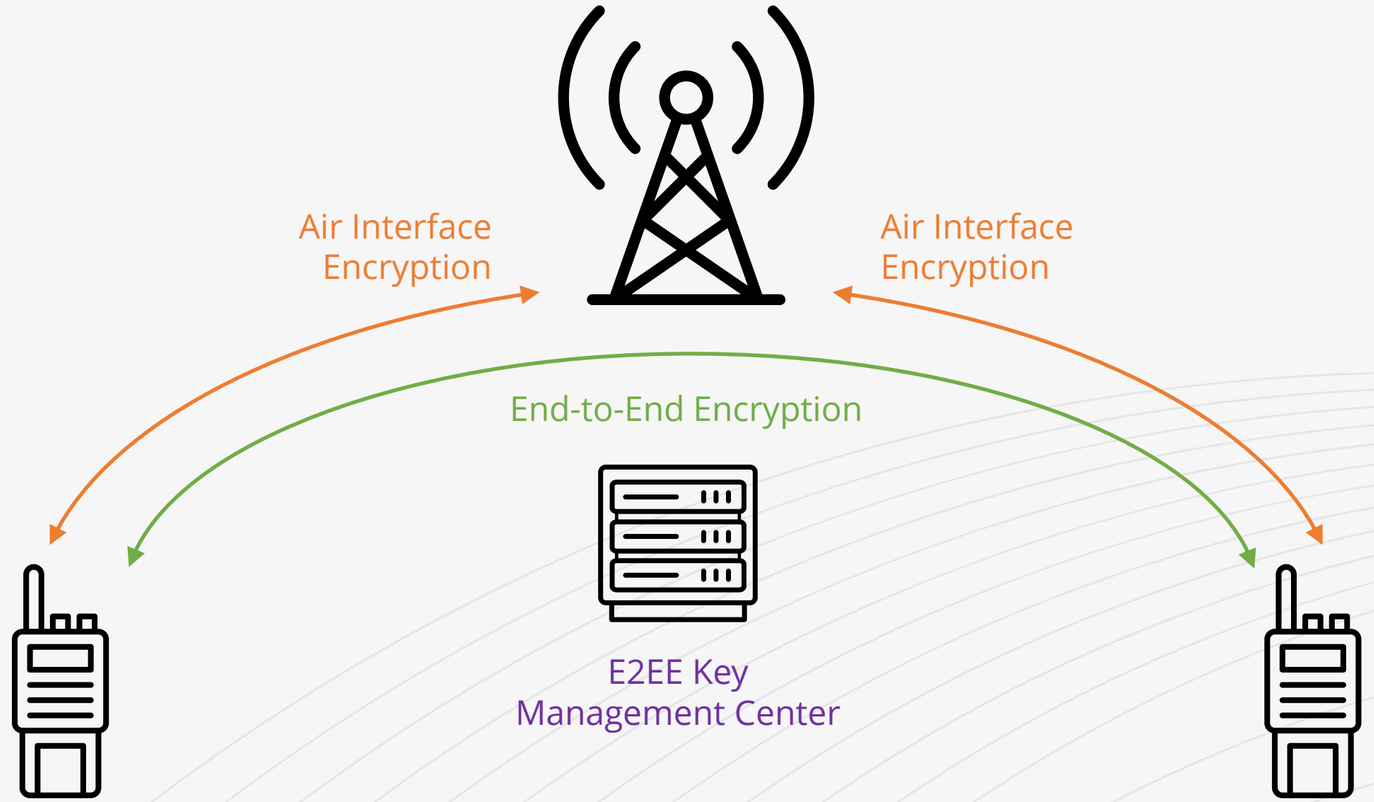
TETRA E2EE

E2EE versus AIE



AIE	Air Interface Encryption	Infra ↔ radio
E2EE	End-to-End Encryption	Between radios

E2EE versus AIE



AIE	Air Interface Encryption	Infra ↔ radio
E2EE	End-to-End Encryption	Between radios

TETRA E2EE

- **Most sensitive use-cases**
  - Special forces, intelligence agencies
- **Incredibly hard to determine users**
  - But OSINT & industry sources indicate widespread usage in:
    Europe, LATAM, Middle East, South Asia

Hytera AIRBUS sepura LEONARDO SECTRA

Motorola E2EE options

- **Smartcard (“SIM”)**
  Support for 3rd party SIMs (eg Sectra)

  Won’t ever get one ☹
  Expect serious hardening
- **UCM**
  Universal Crypto Module
  Prev gen devices

  TETRA/P25/LTE
- **CRYPTR micro**
  SD card form factor, SDIO interface
  Current generation devices

  Based on MACE engine (like UCM)

Great

- Got CRYPTR Micro off Ebay
- Looked the part
- Let’s dig into SDIO

Great scam-per weight-ratio ☹

.. It didn’t work

- Got CRYPTR Micro off Ebay
- Looked the part
- Let’s dig into SDIO

Sepura E2EE

SECURITY SERVICES

TETRA:
- Authentication
- Class 1, 2,3 & 3G TETRA security
- Air Interface encryption TEA1/2/3/4[8]
- Smart card E2E encryption[8]
- Embedded E2E encryption[8]
- Enhanced security module (ESM)
- In-country E2E Encryption algorithm[8]
- Multiple E2E algorithm[8]

- Many different E2EE options
- Embedded? Is that software?
- OMAP-L138-based model
  - Same SoC as MTM5400

Source: https://www.youtube.com/watch?v=NP5pKX24x_I

Sepura

Slapping more money on the counter

- **Purchased a few civilian, TEA1-enabled radios**
- **We previously found a code execution vulnerability**
  - Weak filesystem access control checks (CVE-2025-52945)
  - Head start!
- **Built tooling, implemented attacks, extracted crypto**
  - That’s a talk on its own
  - Skipping details in the interest of time

TETRA E2EE

The nitty gritty

E2EE

Variants

- TETRA E2EE implementation is proprietary
- TCCA has ‘recommendations’
  - SFPG Recommendation 02, 07, 08
- We believe Sepura implementation closely adheres to TCCA reccs

E2EE

NDAs, again…

- TCCA SFPG recommendations are under NDA ☹
  - End users we have spoken have no idea how E2EE works

The access to these documents is for TCCA members
Non TCCA members can have access to the SFPG Recommendations, if their request is supported by an TCCA member

NDAs for SFPG Recommendations from SFPG@TCCA.INFO

https://tcca.info/documents/Cyber_security_workshop-presentation.pdf/

E2EE

NDAs, again…

- TCCA SFPG recommendations are under NDA ☹
  - End users we have spoken have no idea how E2EE works
- However, a few days ago we found this online…

End to end encryption in Public Safety TETRA networks

By B.W. Murgatroyd

ICTU UK Home Office

TETRA was designed from the outset with security as one of its principal features. Standard security features include strong mutual authentication, dynamic cipher key encryption and secure terminal disabling.

Some users require an additional degree of security and a methodology has been developed by the TETRA MoU Security and Fraud Prevention Group (SFPG) to implement end to end encryption for voice and short data services.

This paper examines the reasons for choosing end to end encryption as a counter to perceived threats, shows the detail of the end to end encryption recommendation and examines the limitations of end to end encryption and the benefits of using it in conjunction with the standard TETRA security mechanisms.

Figure 5: Synchronization Frame

Information element	Length	Remark
STD	1 bit	Set to 0 if conforming to rec. 02.
ALGORITHMID	10 bits	Identifies the Voice traffic cryptographic system.
SV	64 bits	The synchronisation vector
PTS	2 bits	2-bits of a 16-bit timestamp
KEYID	20 bits	The Key identifier
CCSUM	22 bits	Cryptographic checksum

Table 2: Synchronization Frame Structure

Key Management Messages (Algorithm E4)

The SDS messages associated with Key Management require protection. Algorithm E4 is used for this in conjunction with the Signalling Encryption Key which may be a key in its own right or be a GEK or UEK. It is assumed that algorithm E4 is a block cipher that is used in a Cipher-Block-Chaining mode with a random IV. The block size of E4 is not specified.

End to end Key Management

End-to-end key management is totally separate from standard TETRA Air Interface key management and is not part of the SwMI. The short data service is used to send Over the Air Keying (OTAK):

https://digital-library.theiet.org/doi/10.1049/ic%3A20030015

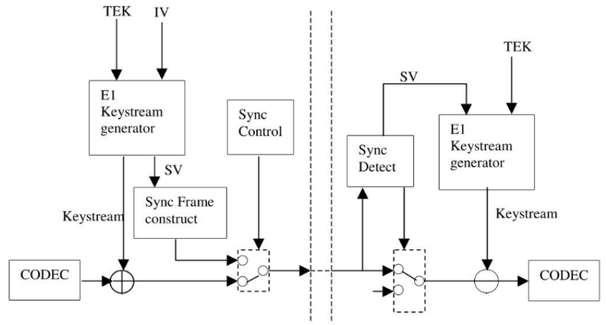
E2EE

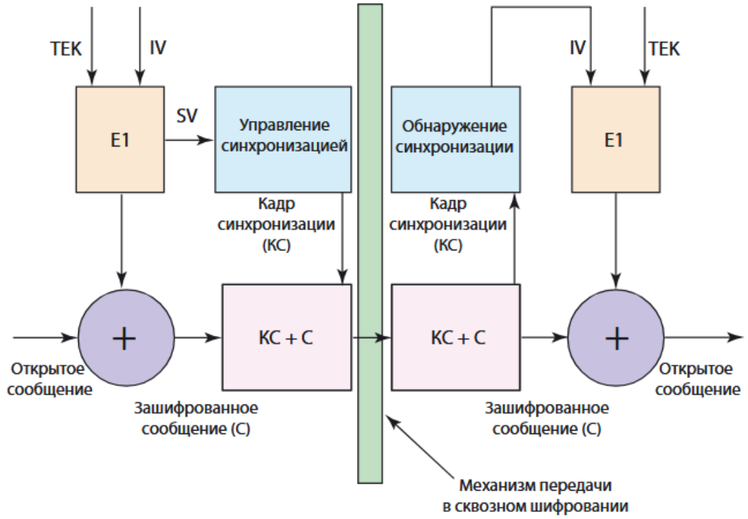
NDAs, again…

- **We also found 7+ Chinese and 1 Russian publications with full TETRA E2EE details**
  - Clearly based directly on SFPG recs or Murgatroyd paper
  - Both CN+RU universities are sanctioned entities
  - No indication of offensive intent
- **Clearly these NDAs don't aid security at all**
  - Only ones who don't know how E2EE works are end users 🙃

加密和同步

- 通过在U-面上窃取语音帧（半个时隙）实现同步数据的传输；
- 同步帧被当作单独的半个时隙通过STCH, 1…4 SV/sec传输；
- 在每个传输开始时，由发送方MS选择一个随机的初始向量；
- 在传输期间，密钥生成器要不间断的生成密钥流
- 同步帧包括初始向量，算法标识和密钥标识





TETRA标准中的端到端加密

- **ETS 300 392-7,** 描述了采用同步流加密实现的对称加密系统的标准机制
- **TETRA MoU SFPG (Security and Fraud Prevention Group)：** 一个缺省的端到端加密框架。用户使用这个框架确定他们自己的端到端加密系统。
  - **TETRA Mou SFPG, Recommendation 02 –** 端到端加密
    - 一个公开的算法 **(IDEA)**
    - 或用户特定的算法
    - -推荐只供**TETRA MoU**用户使用
  - **TETRA MoU SFPG, Recommendation 07 -**端到端加密的**SDS**
    - 基于**Rec 02** 的 **SDS**
    - 采用通用密钥管理的加密数据
    - -推荐只供**TETRA MoU**用户使用
  - **TETRA MoU SFPG, Recommendation 08 –** 采用 **SIM**来进行端到端加密
    - 正在制定中, 现有一个草案

Core E2EE functions

E-functions define E2EE crypto functions, regardless of underlying cryptographic algorithm

- **E2**, **E4**: Key management
- **E1**, **E3**: Traffic encryption
- **E5**, **E6**: SDS encryption
- *Packet Data encryption*

Underpinning cryptography

Support for several underpinning cryptographic primitives

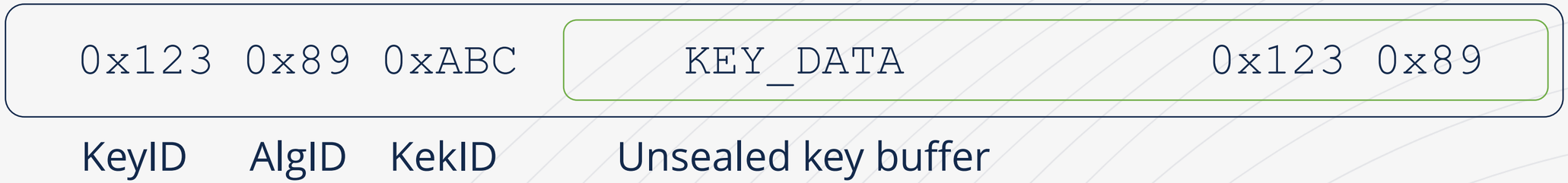
- **AES-128, AES-256**
  - Both excellent, well-understood ciphers
- **IDEA**
  - International Data Encryption Algorithm
  - Deprecated but not bad per se
- **Customer algorithms**

Receiving a TEK key

- MS (portophone) receives OTAK msg with metadata and encrypted data

Receiving a TEK key

- MS (portophone) receives OTAK msg with metadata and encrypted data
- Straightforward decryption (E2 using Key Encryption Key (KEK) in CBC mode)



Receiving a TEK key

- MS (portophone) receives OTAK msg with metadata and encrypted data
- Straightforward decryption (E2 using Key Encryption Key (KEK) in CBC mode)
- Integrity check based on expected KeyID / AlgID

Weakened algorithm

```
static void ALG87_PROCESS_KEY(uint8_t *lpKey_InOut) {

    lpKey_InOut[0] = lpKey_InOut[0xE];
    lpKey_InOut[1] = lpKey_InOut[0xF];
    for (int i = 0; i < 7; i++) {
        lpKey_InOut[i + 2] = lpKey_InOut[i + 9];
    }
}
```

- One algorithm (0x87) invokes an additional key-processing function
- Reduces AES128 key to 56 bits of effective entropy
- Clearly for exportability reasons
  - But are asset owners informed?

Covert

or overt?

Some vendors mention export controlled algos

Others suggest no reduction (except in leaks)

Public tenders/RFP across the world *never* mention AES56/64

AIE algorithms:	TEA1, TEA2, TEA3
E2EE algorithms:	AES export/128, custom algorithm (IDEA, AES256, Customer developed up to 256)

Note: Export controls apply when ordering encryption.

Quick Start Guide

SUPPORTED ENCRYPTION ALGORITHMS

Algorithms
- 128-bit AES
- 256-bit AES

MXP600 350-470 ROM AES 128 CLR

MXP600 350-470 ROM AES 128 TEA1

MXP600 350-470 ROM AES 128 TEA3

MXP600 350-470 ROM AES 256 CLR

MXP600 350-470 ROM AES 256 TEA1

The length of the Traffic Key (stated in Bits) is subject to export control regulations and hence the CMC will be factory configured to support 128, 64 or 56 bit key lengths.

Initially the CMC will support the AES-128 algorithm however it is expected that other algorithms will become available in future software releases.

Export control regulations will determine which algorithms may be supplied and also the permitted length of the Traffic Keys (stated in Bits). For UK and Western European operations, 128 bit keys will typically be used however 56 and 64 bit keys are also supported.

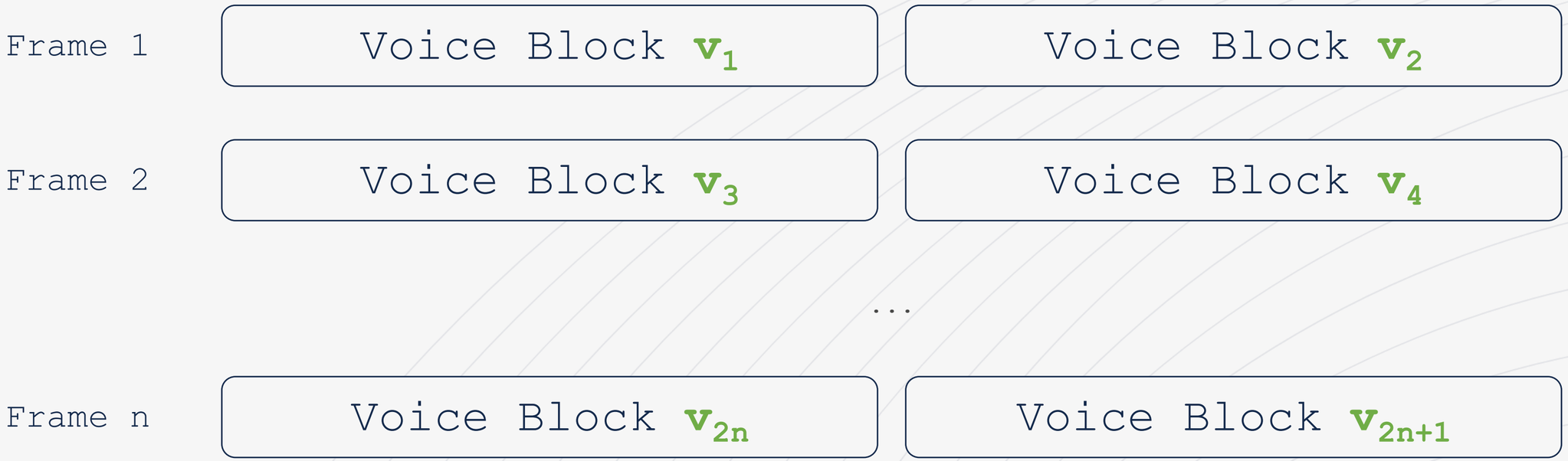
Materials on: Sepura MOD-05-166, Leonardo Puma T3 Plus, Motorola KVL6K / MTP830S / MXP600

TETRA E2EE

Call encryption

Non-E2EE calling

- **Voice data is split in ~25ms blocks**
  - Denoted v_1, v_2, etc
- **Encoded and sent as traffic stream**
  - Two blocks per timeslot
  - Some traffic loss is acceptable



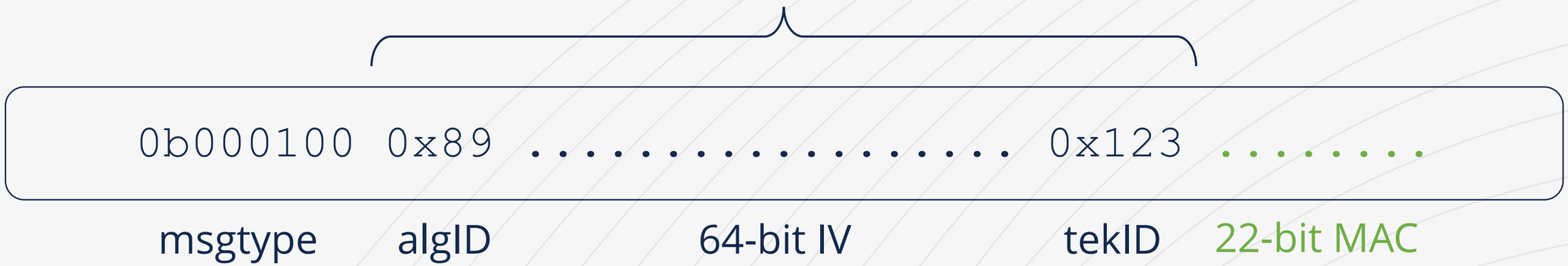
Encrypting traffic

- **Some traffic blocks now used for signaling cryptographic parameters**
  - “Slot stealing”: repurpose a voice block for signaling purposes
- **Parameters chosen by traffic initiator**
  - So, not by the infrastructure
- **Sent in E2EE SyncFrame**

SyncFrame dissected

- Contains encryption parameters including Initialization Vector (IV)
- Also contains Message Authentication Code (MAC) to prevent tampering

Accept SyncFrame if received MAC equals `MAC(algID, IV, tekID)`



Simplified

E2EE calling

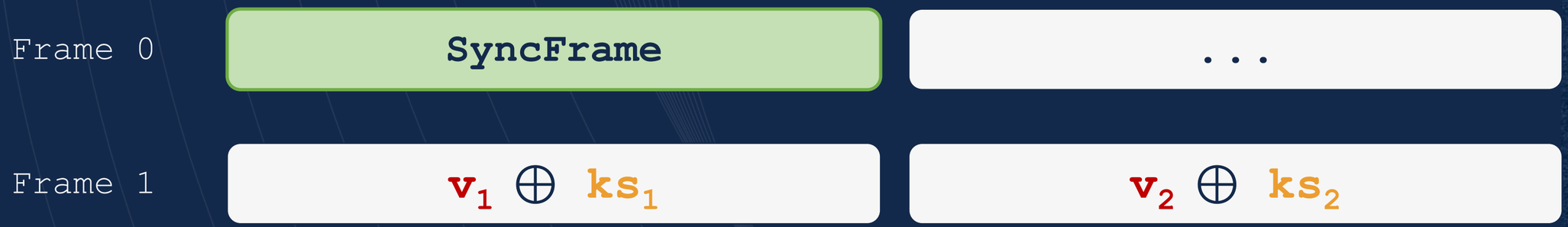
- **Omitting some details given stakeholder concerns; rough outline:**
  - Syncframe instructs which keystreams to generate
  - Voice XOR'ed with keystream for en- / decryption
  - Periodic syncframe resolves any desync issues

Frame 0	**SyncFrame**	. . .
Frame 1	$v_1 \oplus ks_1$	$v_2 \oplus ks_2$
Frame 2	$v_3 \oplus ks_3$	$v_4 \oplus ks_4$
Frame n	**SyncFrame'**	$v_6 \oplus ks_6$

Weak design

Assume we have recovered a few blocks of keystream ks_1 .. ks_n

- We open a call and re-play the original syncframe
- Then, we inject malicious voice traffic v, encrypted with our recovered ks
- Indistinguishable from valid!



Unpredictability

- How to get known plaintexts?
- Here's the first voice frames from three call starts
  - Notice anything?

Call 1	Frame 1	01011000 ... 11010100	11010101 ... 00100100
	Frame 2	01100101 ... 11010001	11000101 ... 00000010
	Frame 3	01011101 ... 00000101	01101000 ... 01100001
	Frame 4	00010001 ... 10101110	11101110 ... 01001010
Call 2	Frame 1	01011000 ... 11010100	11010101 ... 00100100
	Frame 2	01100101 ... 11010001	11000101 ... 00000010
	Frame 3	01011101 ... 00000101	01101000 ... 01100001
	Frame 4	00010001 ... 10101110	11101110 ... 01001010
Call 3	Frame 1	01011000 ... 11010100	11010101 ... 00100100
	Frame 2	01100101 ... 11010001	11000101 ... 00000010
	Frame 3	01011101 ... 00000101	01101000 ... 01100001
	Frame 4	00010001 ... 10101110	11101110 ... 01001010

Example denotes two 137-bit blocks per frame. Sequences edited for educational reasons.

~~Un~~predictability

- Here’s the first voice frames from three call starts
  - Not all but multiple radios
  - *Won’t disclose which for now*
- Predictable plaintext achieved 💪

Example denotes two 137-bit blocks per frame. Sequences edited for educational reasons.

Arbitrary length voice injection

- We can re-inject our captured syncframe
  - Recipient re-synchronizes, re-uses same ks
- Arbitrary length voice injection!

Frame 0	SyncFrame(tekId=K,IV=X_1)	...
Frame 1	$v_1 \oplus ks_1$	$v_2 \oplus ks_2$
Frame 2	SyncFrame(tekId=K,IV=X_1)	$v_4 \oplus ks_4$
Frame 3	$v_1 \oplus ks_1$	$v_2 \oplus ks_2$

E2EE SDS

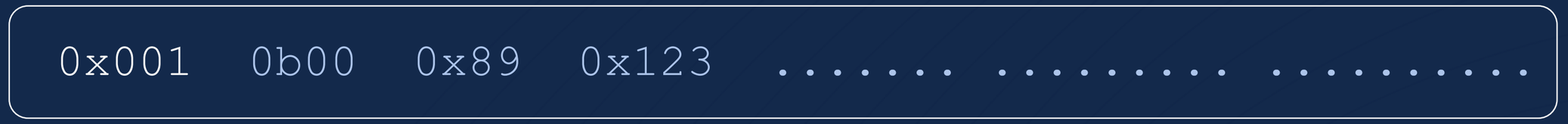
text messages

Very briefly:

- E2EE SDS has IV and MAC protecting msg
- However, SDS counter protected
- Hence, SDS are fully replayable

Accept SDS if received MAC equals:

`MAC(svType, algId, tekId, IV, plaintext)`



Counter svType AlgID TekID IV MAC ~~ciphertext~~ plaintext

Hold on..

Some disclaimers apply

- **We've investigated Sepura's Embedded E2EE**
  - Implementation of TCCA SFPG recommendations
  - Other implementations may or may not share these issues

- **We've left the original TETRA Air Interface Encryption out of the equation**

Regarding Air Interface Encryption

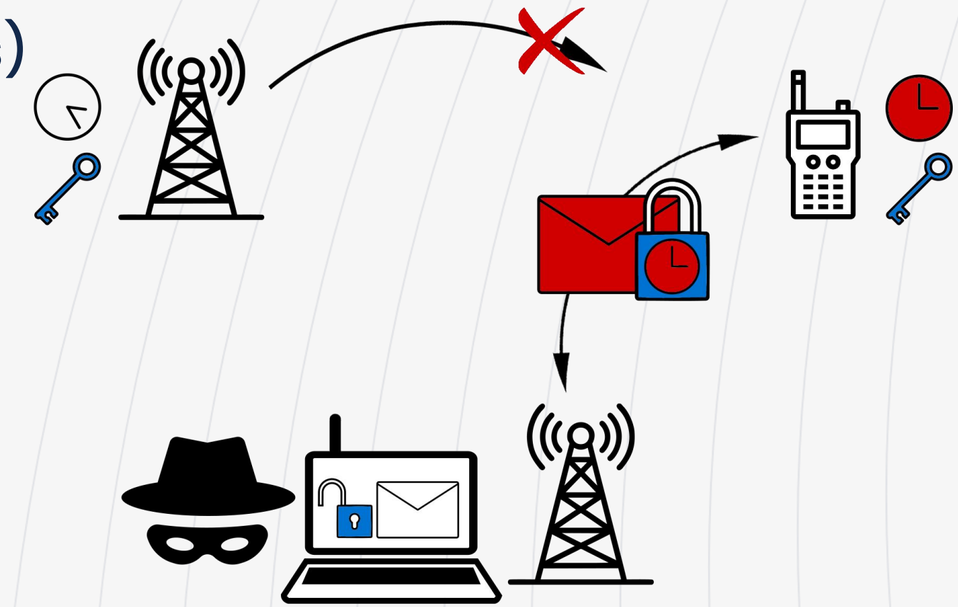
- E2EE is a layer on top of the AIE
- TETRA:BURST found several issues
  - One fixed with ETSI standards revision

Let’s discuss AIE resilience

AIE weakness: ETSI's patch for CVE-2022-24401 (keystream recovery)

Refresher: CVE-2022-24401 Keystream oracle attack

- Capture interesting encrypted message at time T
- Target MS (portophone) (any, with same keys)
- Overpower legitimate signal
- Set MS time to time T
- *Recover keystream for that time*
- …
- Profit

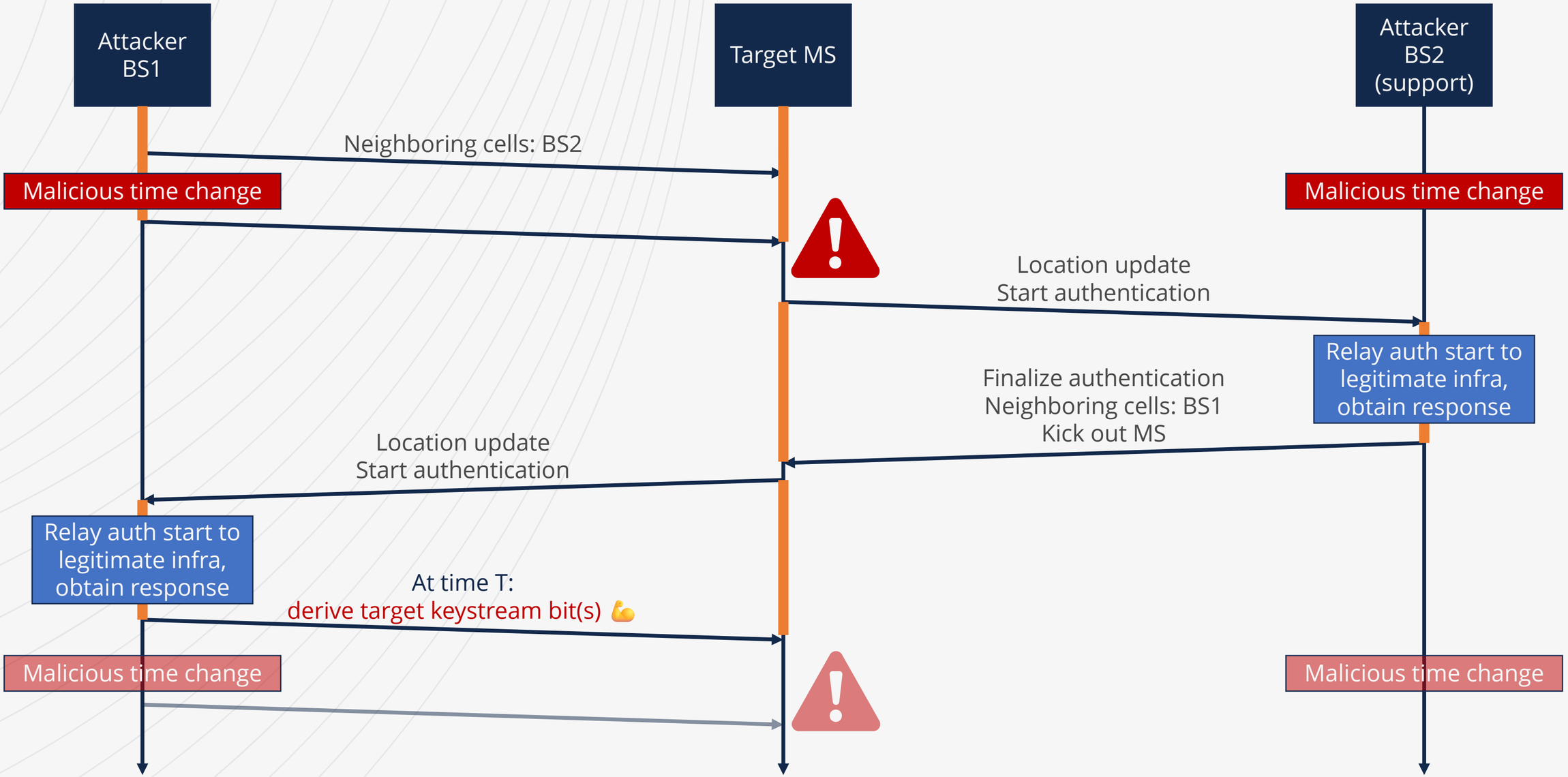


ETSI's fix

- **When MS encounters an unexpected IV (time) change, the MS *should consider*:**
  - ❌ Switch to a neighboring cell that meets the expected IV
  - ❌ Initiate encrypted re-authentication of the current cell (often unsupported) using the attacker-provided IV
  - ✅ If both of the above fail: switch to neighboring cell that meets the attacker-provided IV

- **Root cause not addressed**

- **Patch workaround:**
  - Secondary attack frequency


Attacker BS1
Target MS
Attacker BS2 (support)
Neighboring cells: BS2
Malicious time change
Malicious time change
Location update
Start authentication
Relay auth start to legitimate infra, obtain response
Finalize authentication
Neighboring cells: BS1
Kick out MS
Location update
Start authentication
Relay auth start to legitimate infra, obtain response
At time T:
derive target keystream bit(s) 💪
Malicious time change
Malicious time change

Multi Cipher Networks

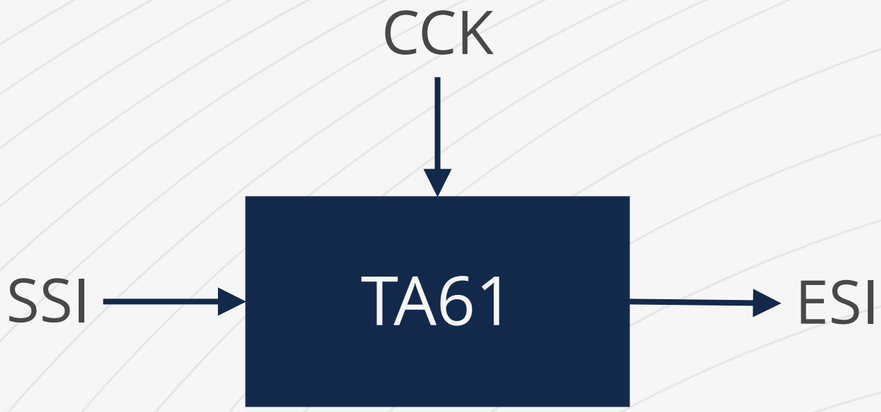
Multi-cipher networks

- TEA1 is obviously broken as an AIE algorithm
- Many parties started moving away from TEA1 following TETRA:BURST
- One solution: multi-cipher networks
  - Combine TEA1 and TEA2/TEA3 on the same network
  - Migrate most important radios first

Identity encryption
(this is relevant)

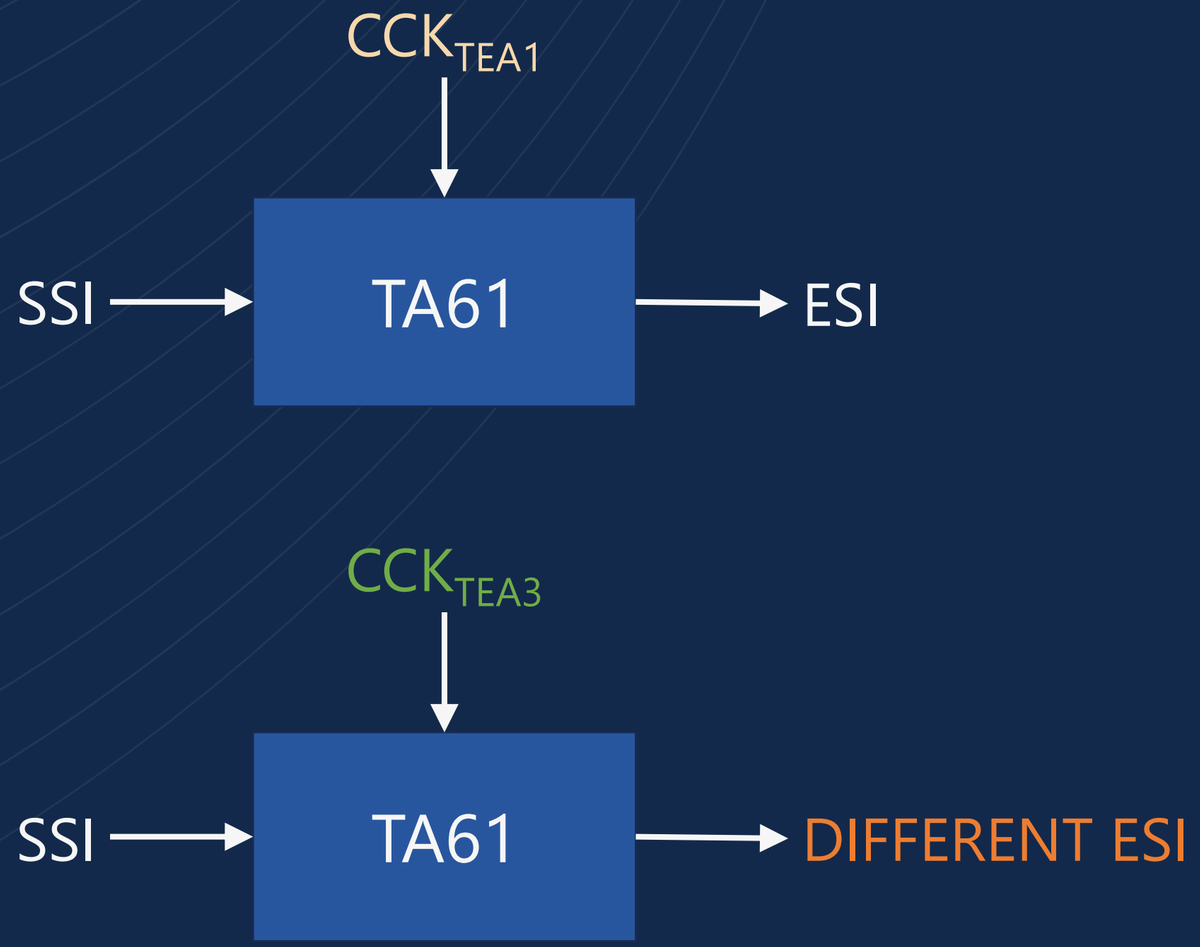
- **TETRA uses the CCK network key identity encryption**
- **Prevents trivial traffic analysis**

  (we broke it, but that's irrelevant here)



Distinct keys

If TEA1 and TEA3 are to operate with distinct CCKs, identity encryption would break



~~Distinct keys~~

no actually there's just one shhhhht 🤫

- The CCK is the same for all TEAs on a network
- A single TEA1 radio compromises traffic for the entire network
- If keys are not rotated, the network remains compromised indefinitely

Turns out, no one talks about this

One weird trick

We thought of a solution

- Keep an eye on our blog

OEMs, this one's for you 🎁

- Be the first to support multi-cipher with distinct CCKs, drop us an email

More caveats, pitfall, misery

Plaintext uplink injection

A downlink keystream oracle on mixed plain/encrypted networks

Plaintext downlink injection

Accepted by MS (!), uplink keystream oracle or standalone attack vector

Key rotation

- Many networks don’t rotate
  - Efficient keystream recovery options exist
- Worse: reports of use of default TETRA encryption keys

AIE: the status quo

All TEA1 networks:	compromised
Mixed cipher networks:	compromised
No/slow key rotation networks:	compromised*
For the rest: “improved” keystream recovery attack	compromised*
Beyond-current patch level, TEA2, key rotation, maybe GCKs or E2EE	okay**

* All class 2, class 3 with caveats

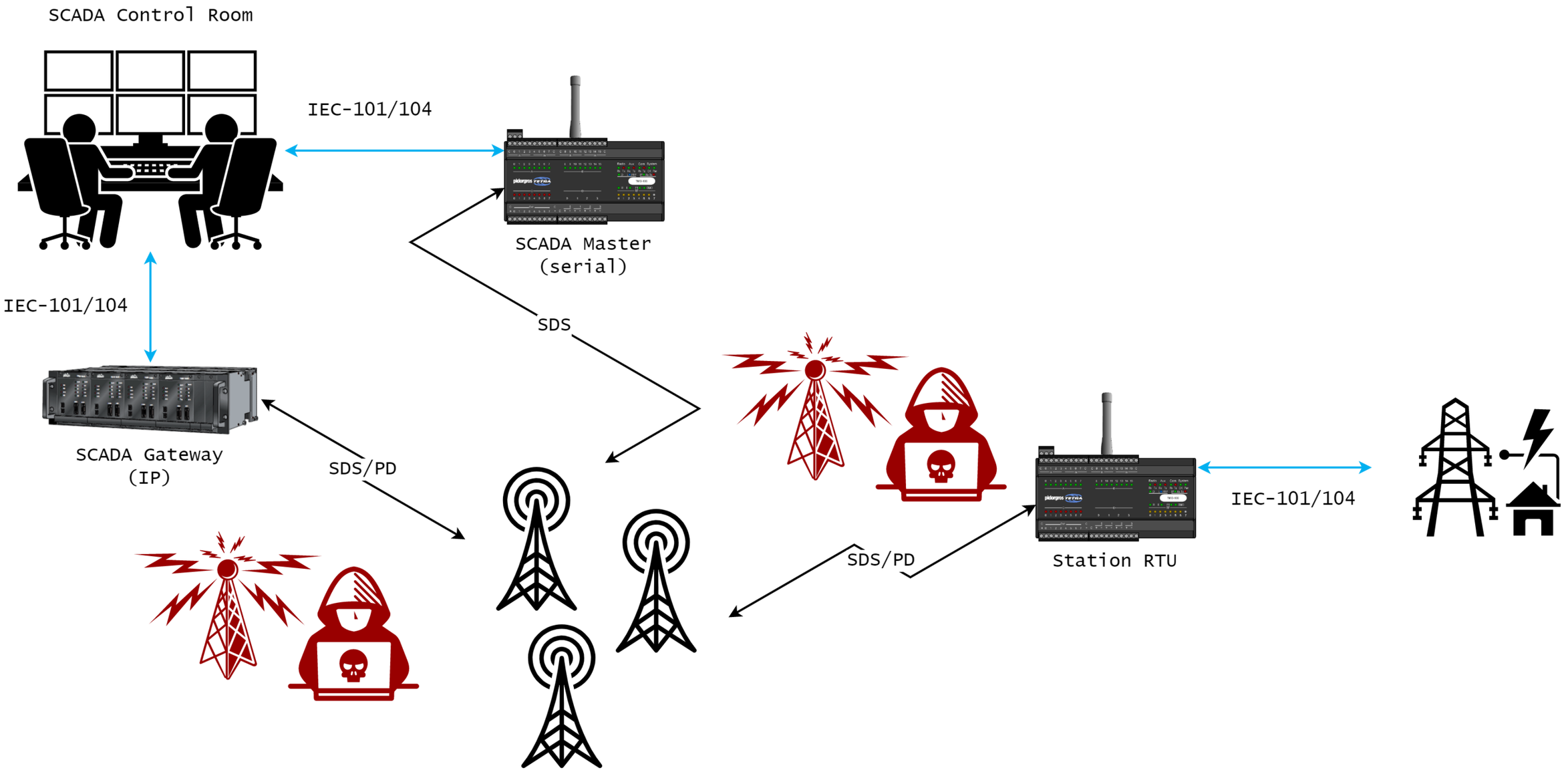
** Still caveats apply, such as unencrypted downlink injection

Traffic injection

“Traffic injection can’t be done”

– Many TETRA experts

- **Need to handle:**
  - Synchronization
  - Interference from other devices
  - Keystream recovery? Often not required
  - Registration/authentication


SCADA Control Room
IEC-101/104
SCADA Master
(serial)
IEC-101/104
SDS
SCADA Gateway
(IP)
SDS/PD
SDS/PD
Station RTU
IEC-101/104

Conclusion

"*Transparency is at the root of ETSI, in our governance and technical work.*"

- Luis Jorge Romero,
  ETSI Director-General

- **First public analysis of a TETRA E2EE solution**
  - Uncovered weakened variant
  - Confidentiality is OK, integrity/authenticity not so much
- **Revealed further serious ecosystem issues**
- **As always:**
  - Don't trust black box solutions
  - Be skeptical of vendor claims & recommendations
  - Perform in-depth technical assessment before procurement, not checkbox compliance
  - Pressure vendors for transparency

Questions?

Social

-

Web

- midnightblue.nl
- tetraburst.com

Contact

- j.wetzels@midnightblue.nl
- c.meijer@midnightblue.nl
- w.bokslag@midnightblue.nl