# CRYPTO AGILITY DEFINITIONS FOR SPACE SYSTEMS

**JANNIK MÄHN,**
**MATTHIAS MÜLLER & KARIN ZIELINSKI**

**05.11.2025**

01

# OHB – COMPANY INTRODUCTION

# OHB SE & OHB SYSTEM AG

WE.CREATE.SPACE

**1**

- Both: Main Quarters in Bremen, Germany
- Pioneering role in European space missions
- Ca. 3000 Employees

---

**2**

- 3 Core Segments:
  - Space Systems
  - Aerospace
  - Digital

---

**3**

- Earth Observation
- Navigation
- Telecommunication
- Science and Exploration
- Reconnaissance

# DEPARTMENT: CRYPTOGRAPHIC SYSTEMS

## WHAT DO WE DO?

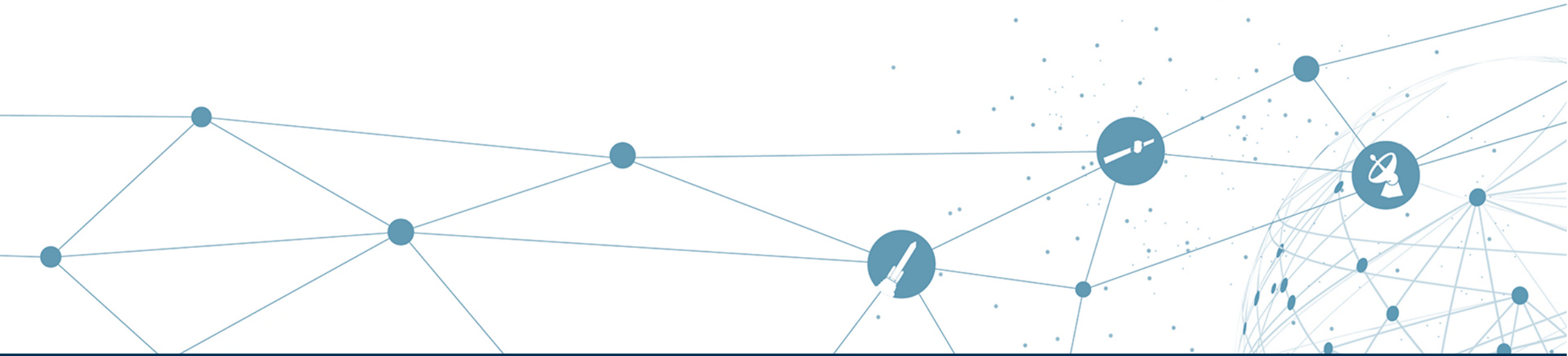
- We develop **security units** that protect communication cryptographically
  - complete design
  - different classification levels
  - between satellite and ground
  - with endorsed cryptographic methods
  - with additional security components

OHB Satellite
Security Unit

- Adhering to **constraints** from satellite **environment and** from **security agencies**
  - tested and proven functionality
  - space qualified
  - approved/certified

02

# CRYPTO AGILITY FOR SPACE SYSTEMS



# THE NEED FOR CRYPTO AGILITY

## QUANTUM COMPUTER THREAT

- If, when, what?
  - Verifiable Quantum Advantage
- Store-now-Decrypt-later-Attacks
  - Need to act today
- NIST PQC-Competition
  - First standardized algorithms

Source:
www.academy.fraunhofer.de/weiterbildung

# STANDARDIZED POST-QUANTUM CRYPTOGRAPHY

## NIST COMPETITION

- FIPS 203:
  - **ML-KEM**
  - Derived from: Crystals-Kyber
- FIPS 204:
  - **ML-DSA**
  - Derived from: Crystals-Dilithium
- FIPS 205:
  - **SLH-DSA**
  - Based on: SPHINCS$^+$

- More to come …
- Other algorithms endorsed

- **Threats to PQC Cryptosystems:**
  - Novel **mathematical attacks**
  - Novel **side-channel attacks**
  - New regulations from **security agencies**
  - Etc.

# WHAT IS CRYPTO-AGILITY

## LITERATURE REVIEW

- Algorithms with *Agility-in-mind*
  - TLS
  - SSH
  - IKEv2

    → Negotiation protocols

  - We consider:

    → Opinionated protocols

- Possibly Agile Components:
  - Software
  - Parameter Set
  - Algorithm
  - Cryptographic Functions
  - Hardware
  - Etc.

# CRYPTO-AGILITY DEFINITION

LITERATURE REVIEW

**Definition:**

Crypto Agility *is a* ***theoretical or practical approach, objective, or property*** *which provides capabilities for* ***setting up, and modifying*** *encryption methods and keying material in a* ***flexible and efficient*** *way while preserving business continuity.*

Source:

Näther et. al., “Toward a Common Understanding of Cryptographic Agility – A Systematic Review,” 2025

# SPACE SYSTEMS DEFINITION

## RELEVANT ASPECTS

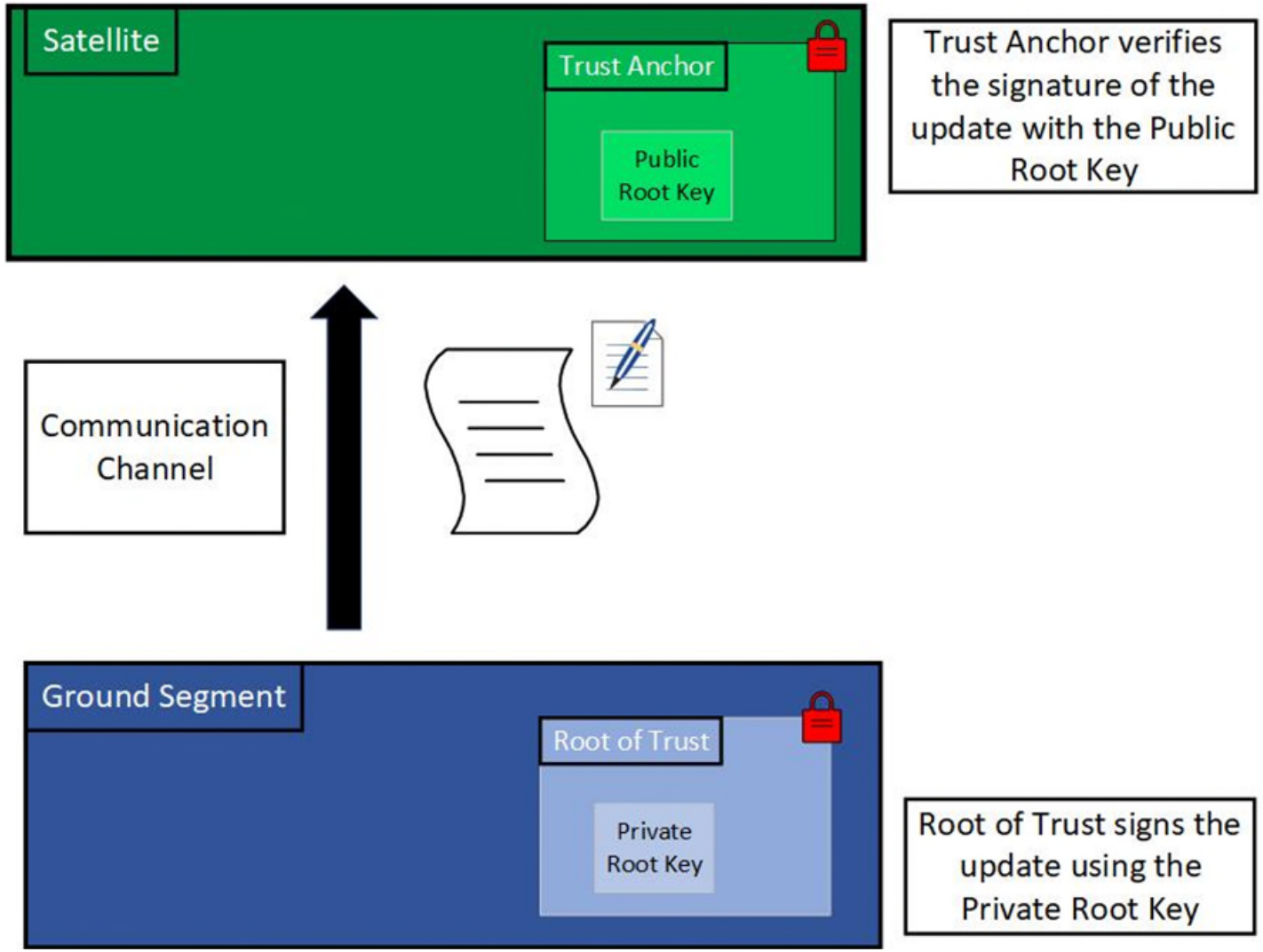
| Crypto Agility – Fundamental Definitions | | | | |
|---|---|---|---|---|
| *Crypto agility is a theoretical or practical approach, objective, or property which provides capabilities for setting up, identifying, and modifying encryption methods and keying material in a flexible and efficient way while preserving business continuity.* | | | | |
| This is the general definition used throughout this document. Where applicable, the notion can be specified as follows: | | | | |
| **Adaption Agility** | **Migration Agility** | **Hardware Agility** | **Software Agility** | **Design Agility** |
| *The ability to change small parts of the crypto unit, e.g., the parameter set of a certain algorithm to change their security level.* | *The ability to exchange entire cryptographic implementations, or the selection between different algorithms.* | *The ability to reprogram the hardware of the security module.* | *The ability to update software components of the security unit.* | *The ability to exchange single components of the security unit without changing the internal interface, as well as the ability to be algorithm independent. Also, general design rules that allow for platform agility.* |

02

# REALIZING CRYPTO AGILITY IN SPACE SYSTEMS

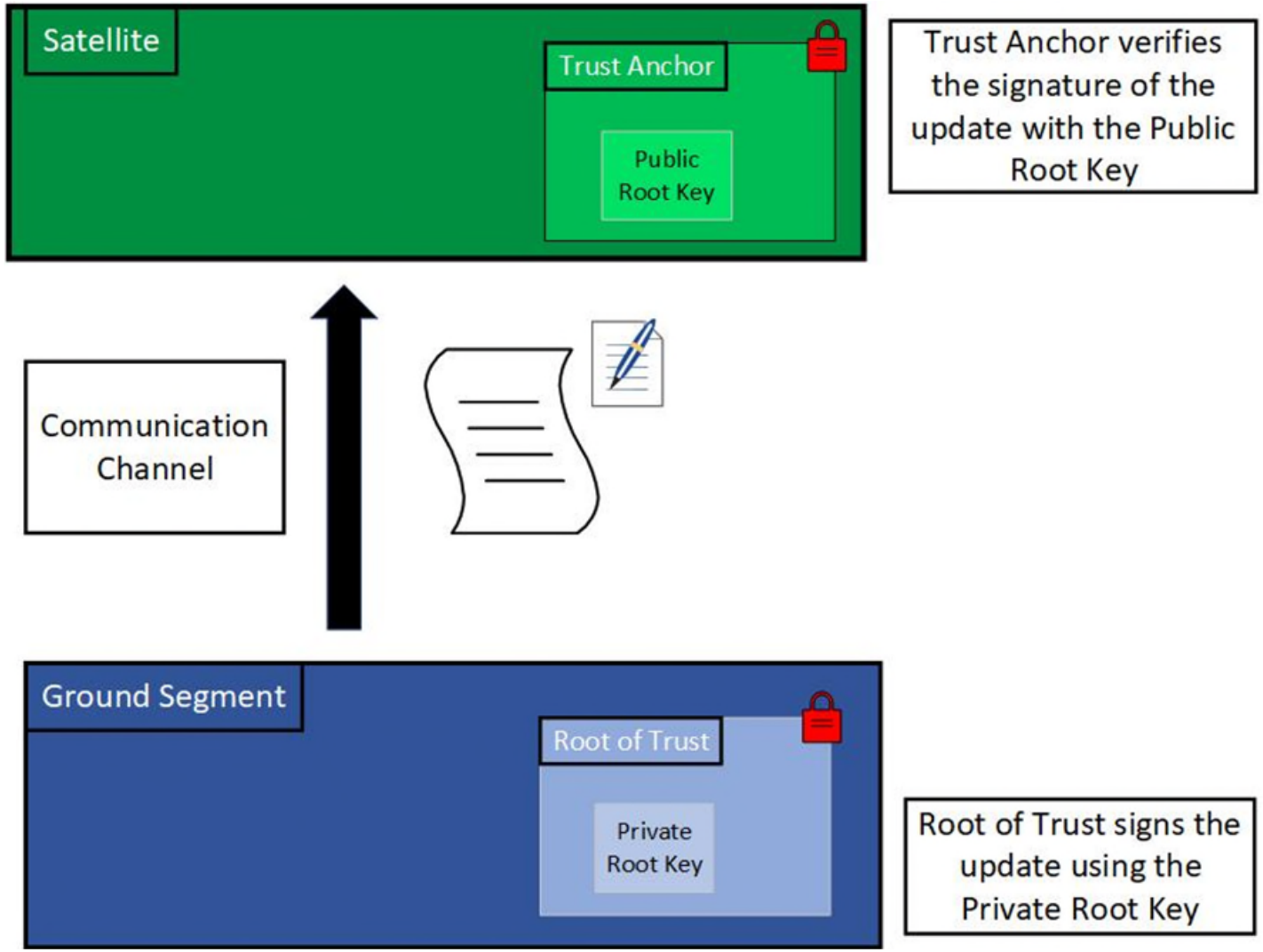
# AUTHENTICATED INITIALIZATION PROCESS

## TRUST ANCHOR & ROOT OF TRUST



# AUTHENTICATED INITIALIZATION PROCESS

## TRUST ANCHOR & ROOT OF TRUST




**Entirely logical separation from normal authentication channel!**

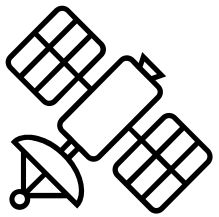
# AUTHENTICATED INITIALIZATION PROCESS

## DEFINITIONS

- **Certification Authority:**
  - the **ground unit**
  - that is **authorized** to initiate updates
- **Root Key:**
  - **public-private key pair** of the signature scheme
  - **private-key** belongs to ground unit
  - **public-key** is stored in the satellite
  - *Stateful **hash-based** signature endorsed*
- **Root of Trust:**
  - a **security module** in the **ground**, that:
    - stores the **private** part of root **key**
    - **signs** the update-files
    - **provides** an **anti-replay** protection
- **Trust Anchor:**
  - Security module in the satellite that:
    - stores the **public** part of root **key**
    - **verifies** signatures of the update files
    - **verifies** the **anti-replay** protection

# AUTHENTICATED INITIALIZATION PROCESS

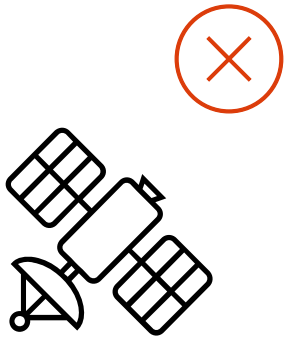
## ANTI-REPLAY MEASURE



**w/ Crypto Unit V1**

# AUTHENTICATED INITIALIZATION PROCESS

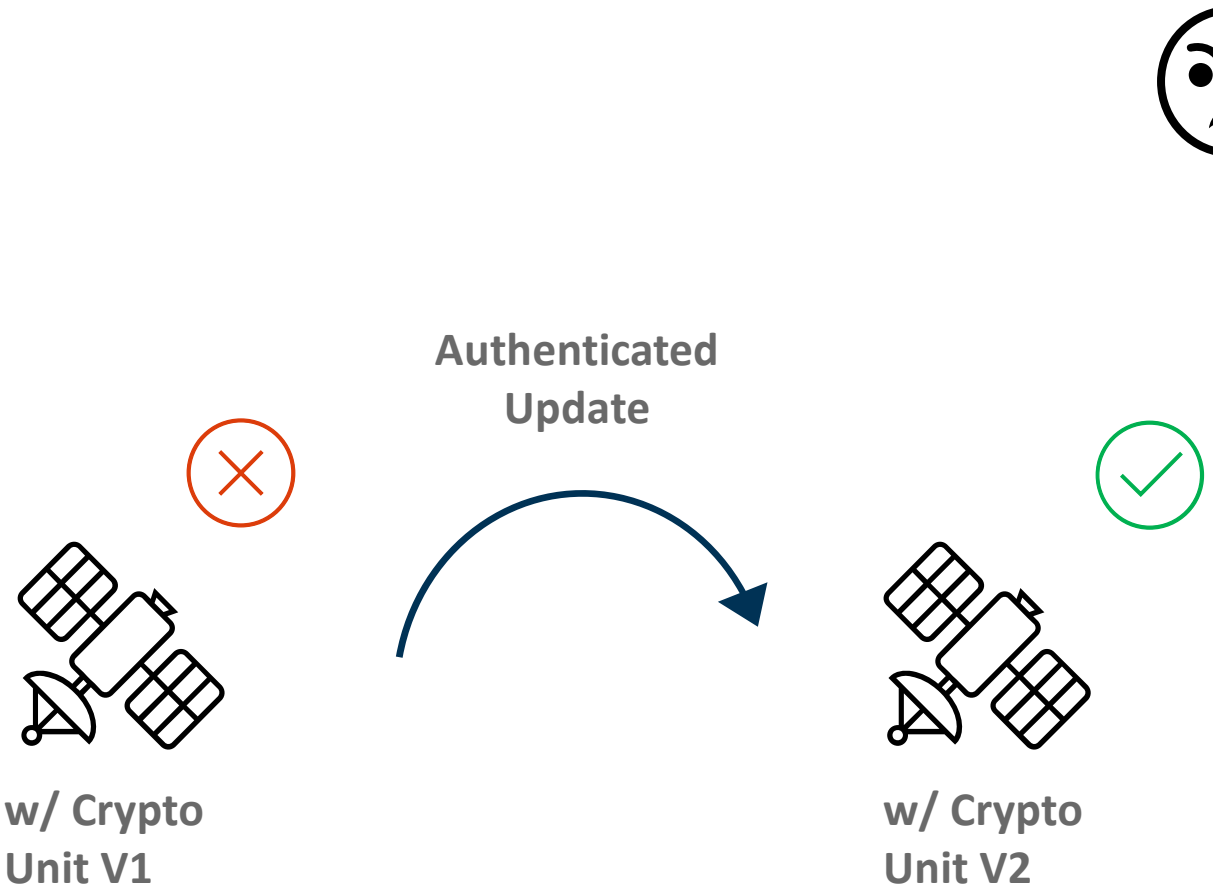
## ANTI-REPLAY MEASURE



**w/ Crypto Unit V1**

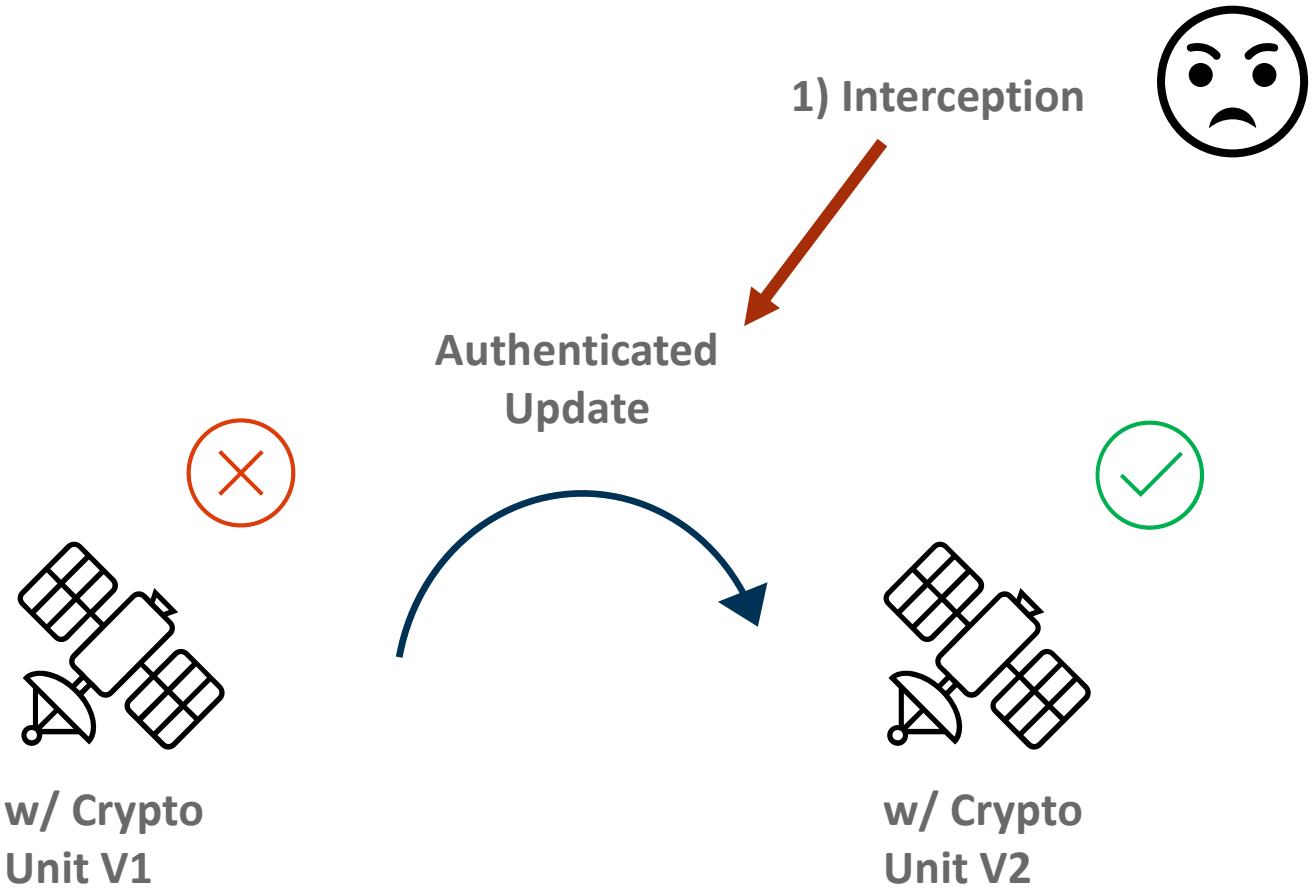
# AUTHENTICATED INITIALIZATION PROCESS

## ANTI-REPLAY MEASURE



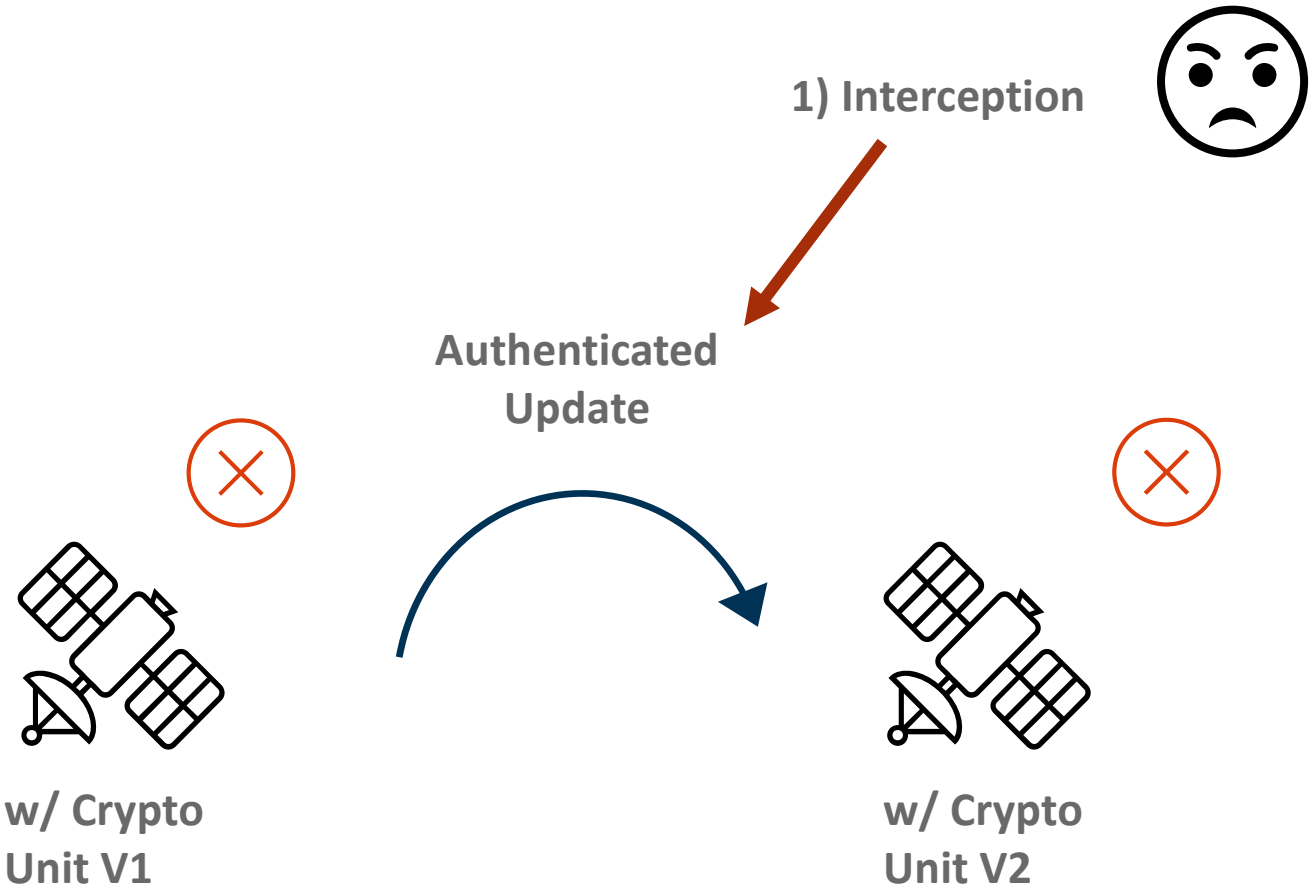
# AUTHENTICATED INITIALIZATION PROCESS

ANTI-REPLAY MEASURE



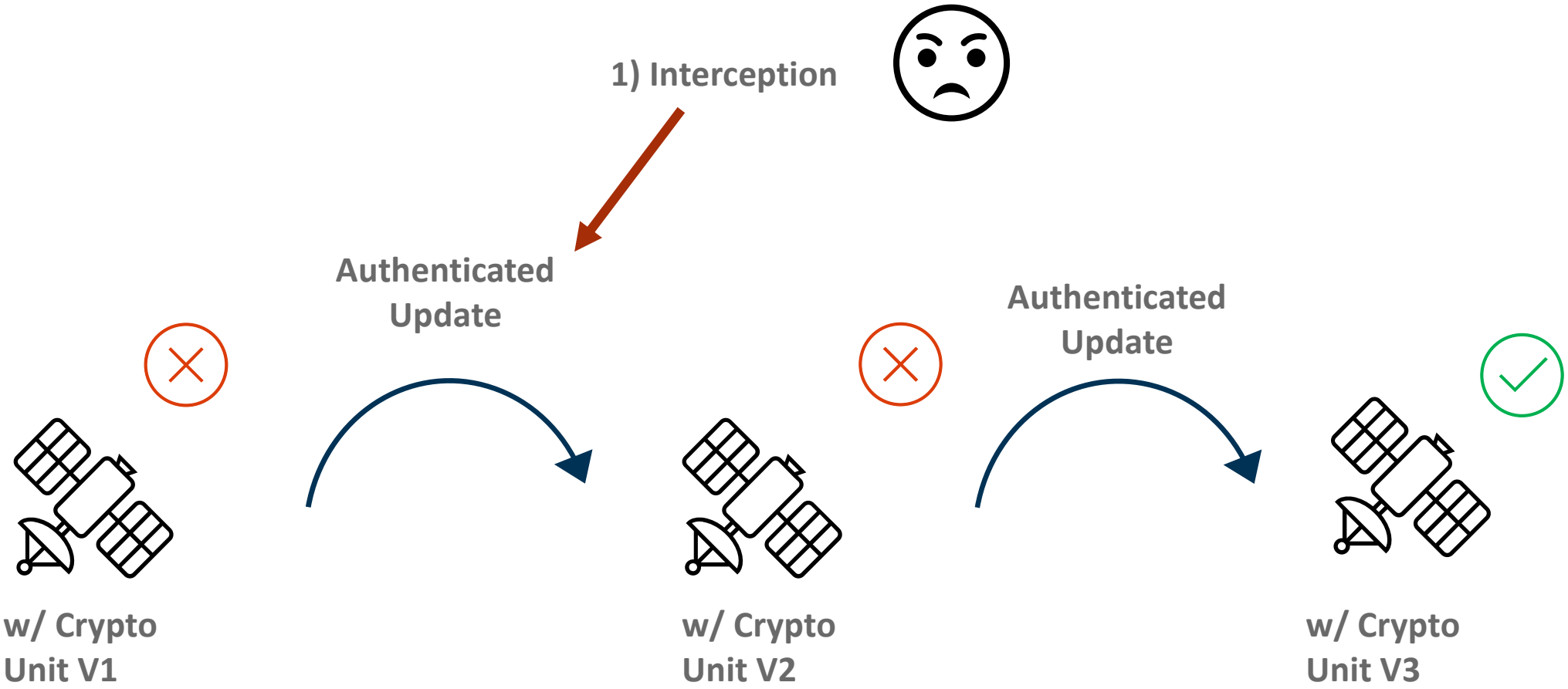
# AUTHENTICATED INITIALIZATION PROCESS

ANTI-REPLAY MEASURE



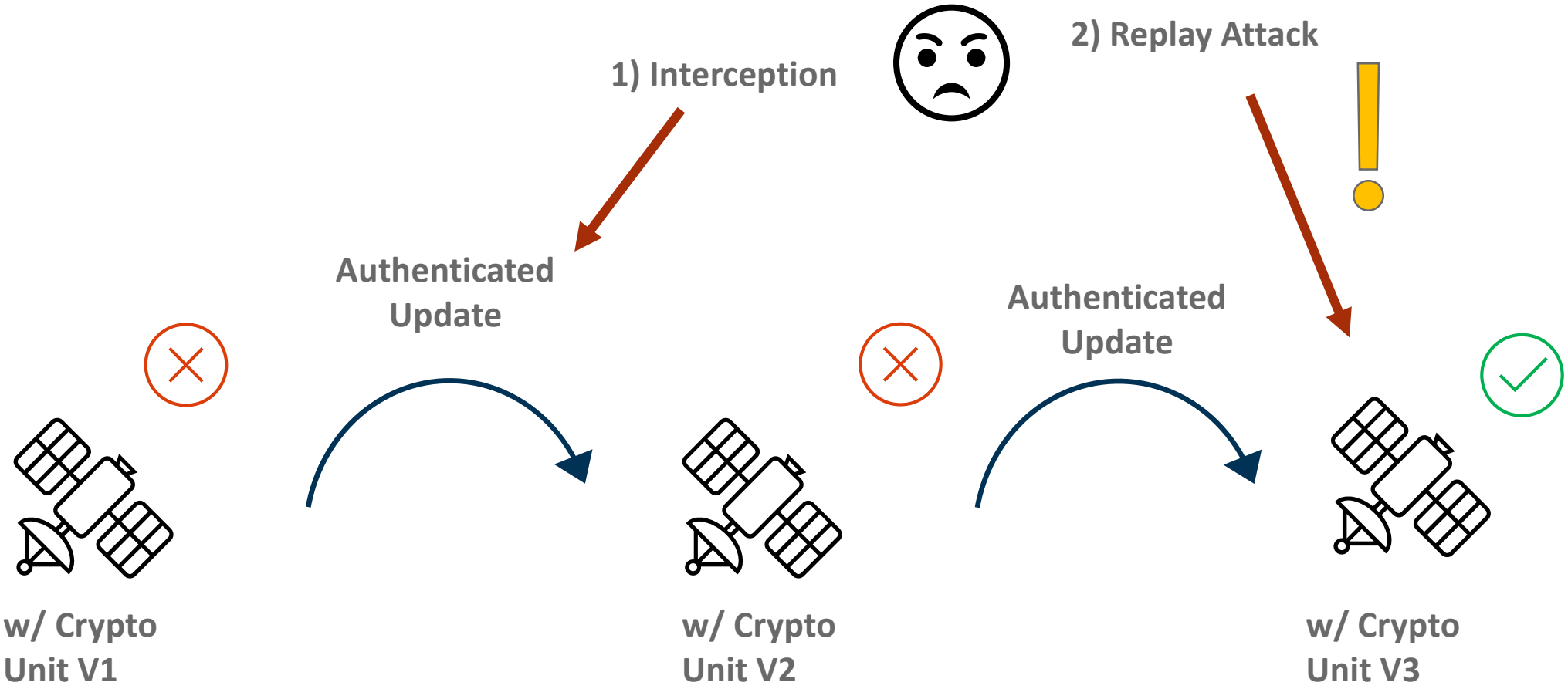
# AUTHENTICATED INITIALIZATION PROCESS

ANTI-REPLAY MEASURE



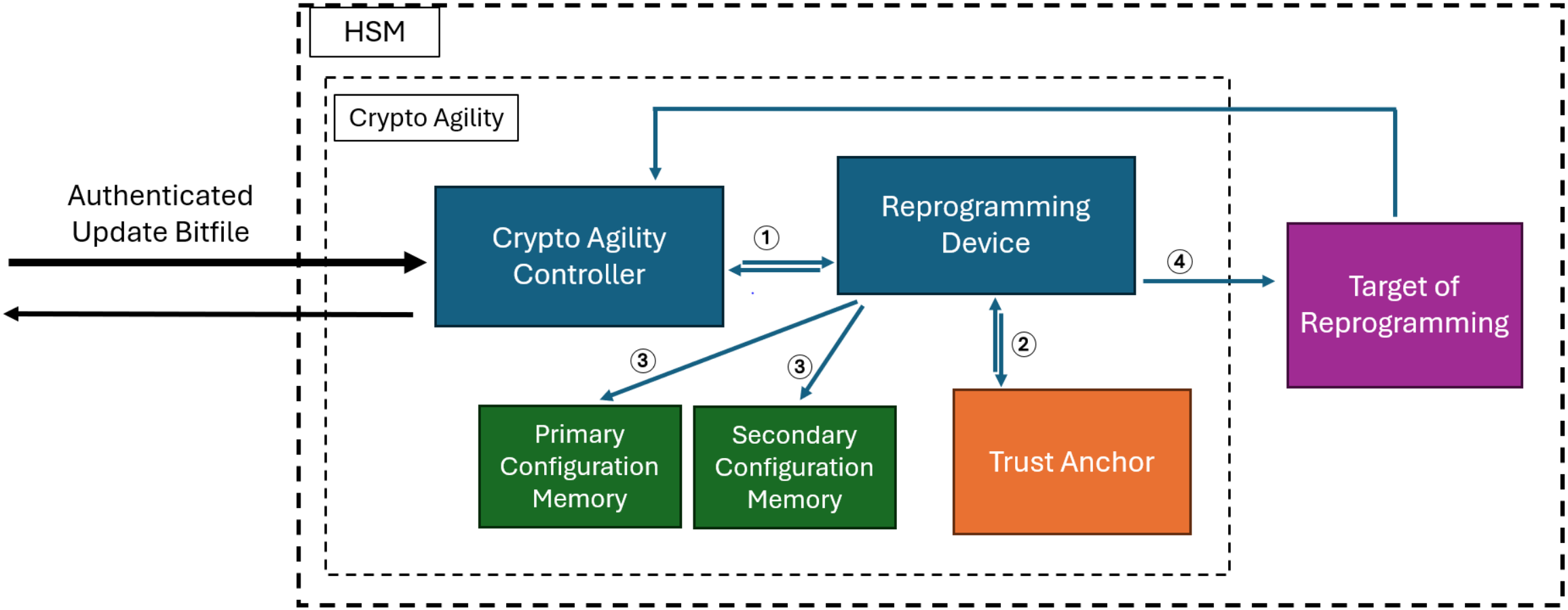
# AUTHENTICATED INITIALIZATION PROCESS

ANTI-REPLAY MEASURE



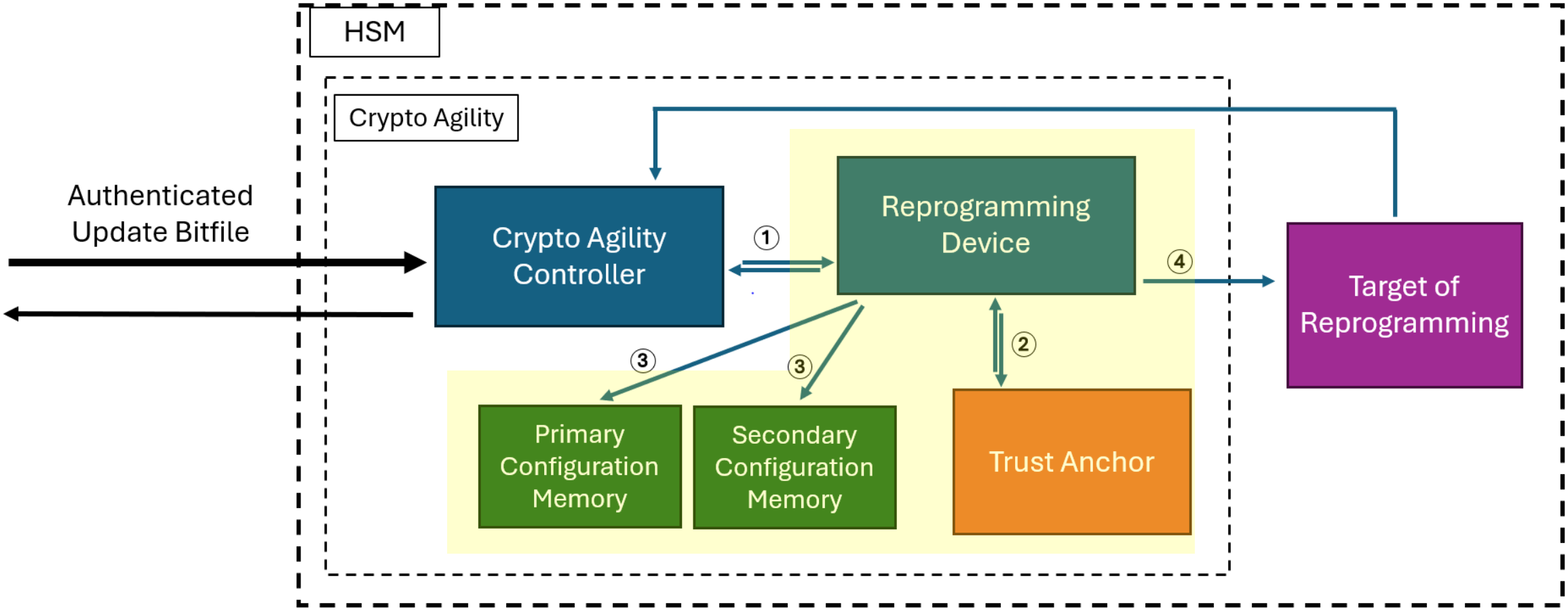
# MINIMAL HARDWARE CONFIGURATION

## FLOW-DIAGRAMM



# MINIMAL HARDWARE CONFIGURATION

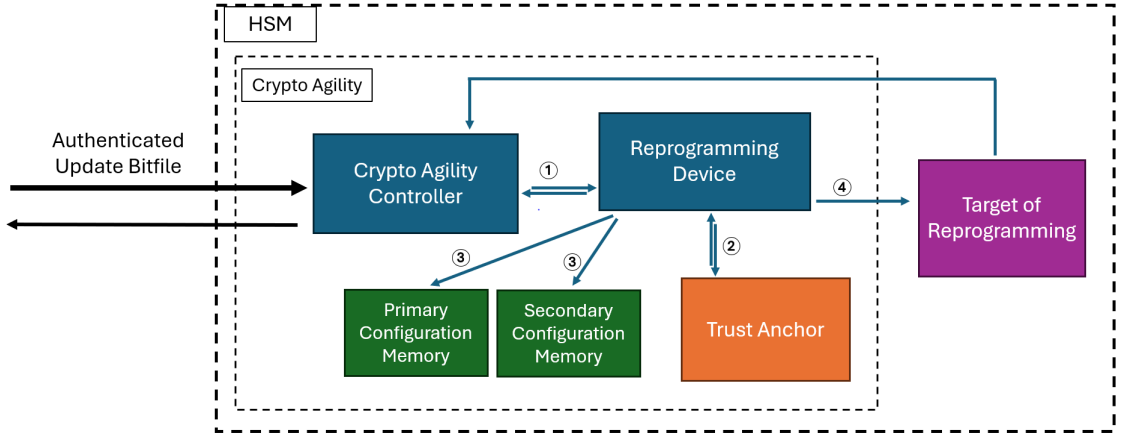
## FLOW-DIAGRAMM



# MINIMAL HARDWARE CONFIGURATION

## DEFINITIONS

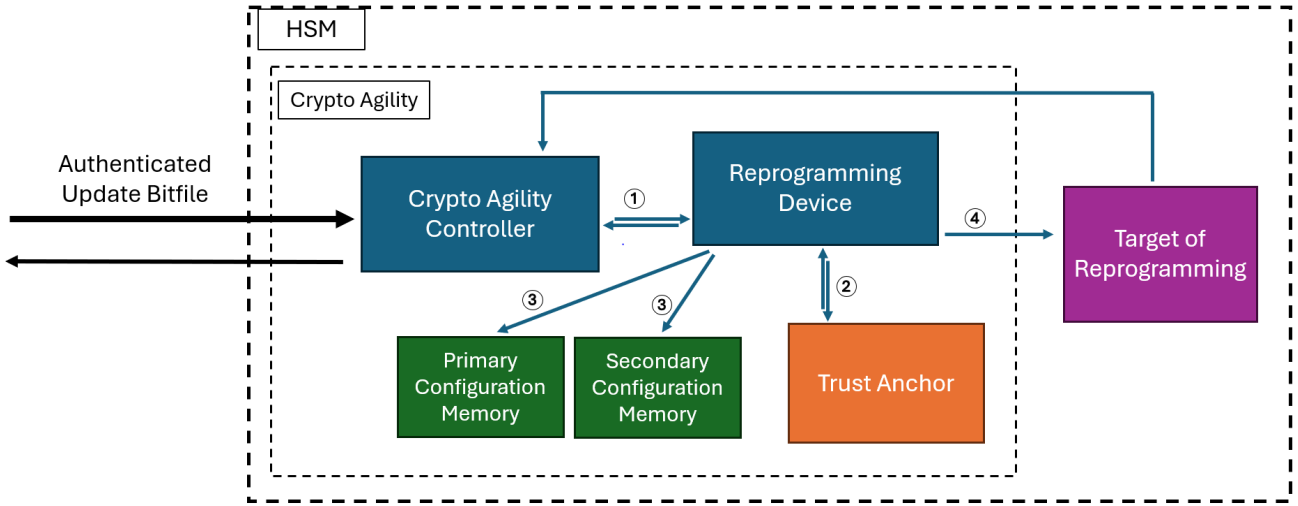
- **Target of Reprogramming**
  - Any **HW/ SW/ Firmware** that is updated
- **Crypto Agility Controller**
  - **Microcontroller / FPGA**
  - Coordinates **Update**
  - Coordinates **Fallback**
- **Reprogramming Device**
  - FPGA that **exclusively reprograms**
  - Access to two **non-volatile memories**
- **Primary & Secondary non-volatile Memory**
  - Primary: stores **update bitstream**
  - Secondary: stores **previous bitstream**
- **Trust Anchor:**



# FALLBACK MECHANISM

## HOW TO RECOVER FROM FAILURE

- **Optimal Scenario:**
  - CA **self-tests** in bitstream
    - new **key-generation**
  - Ground station runs **final tests**
  - Update made **persistent**




- **Fallback Mechanism**
  - **CA-Controller loads old bitstream**
    - needs to generate **authenticated public key**
  - **Self-Induced Fallback**
    - Initiated by CA-Controller
    - Internal time-out
    - Automatic:

      → **No contact** to crypto-unit during

# Questions?

# THANK YOU!

**OHB System AG**
Universitätsallee 27-29
28359 Bremen
Germany

**Phone:** +49 421 2020 8
**Fax:** +49 421 2020 700
**Email:** info@ohb.de
**Web:** www.ohb.de

**Jannik Mähn**