TWISNet

Trustworthy Wireless Industrial Sensor Networks

Architecture concept of trustworthy industrial sensor network deployments

Hochschule für Technik und Wirtschaft Dresden
University of Applied Sciences

Outline

1. TWISNet objectives
2. WSN constraints and security requirements
3. Scenarios
4. Security Framework
5. Mediation Layer Agent Examples
6. Security Levels
7. Sensor-side Framework Modules
8. Hardware Building Blocks

Testbed providing the proof-of-concept that effective security is ensured in the WSN

The objective of TWISNet is to develop a platform supporting the integration of sensor networks in a secure, efficient and reliable way, considering the strong technical constraints of sensor networks.

Security framework implementation for home and industrial use

Produce **network protocols**, **architectures** and **middleware**

Wireless Sensors and Actuators

- Limited processing power
  - typically 8-Bit MCU running on 1-16 MHz
- Low memory resources
  - typically 32 to 256 kB Flash memory
  - 4K to 32K SRAM
- Low data rate
  - i.e. IEEE 802.15.4 250 kbit/s on a shared medium
- Capacity limited power source
  - typically 5 mAh in active mode, 20 mAh in send/receive mode, few µAh in sleep mode
  - With the 1000mAh of AAA cells a node would last for about 1 week in active mode
  - Lifetime of many years requires long sleep periods to run on small batteries

WSAN requirements

Technical prerequisite:

- Small form factor
- Low duty cycle
- Multi-hop topology
- Multiple WSANs
- Short radio ranges
- Long sleeping periods
- Hundreds of sensors per WSAN
- Low cost

Security requirements:

(for all scenarios in the project)

- tamper detection
- authentication
- availability
- trustworthiness
- privacy
- confidentiality
- DoS attacks (depleting batteries)
- integrity
- reliability
- multi-owner

Scenarios

Scenario 1: Sensors attached to a person moving from PAN to PAN

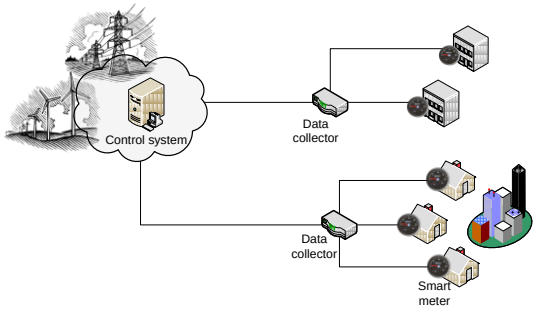
e.g. gathering of medical data in dangerous environments while moving between buildings

Challenges: privacy, secure authentication; integrity and confidentiality; anonymization

Scenario 2: Sensor networks for supply and demand optimization

e.g. smart meters/smart grid enable two-way communication between the nodes and the energy provider for demand optimization

Challenges: confidentiality-protected data; physical attacks; secure remote update; integrity and confidentiality; authentication of commands



Scenario 3: Sensors networks for monitoring and control of industrial processes

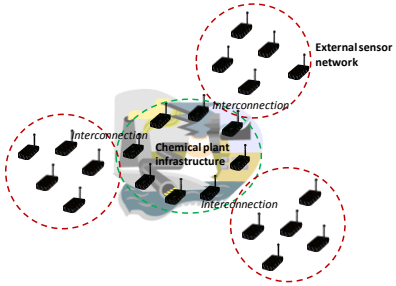
e.g. sensors used to create and monitor a temporary exclusion zone in an industrial process

Challenges: privacy; secure authentication; integrity; confidentiality; anonymization; availability

Scenario 4: Multi-owner sensors networks

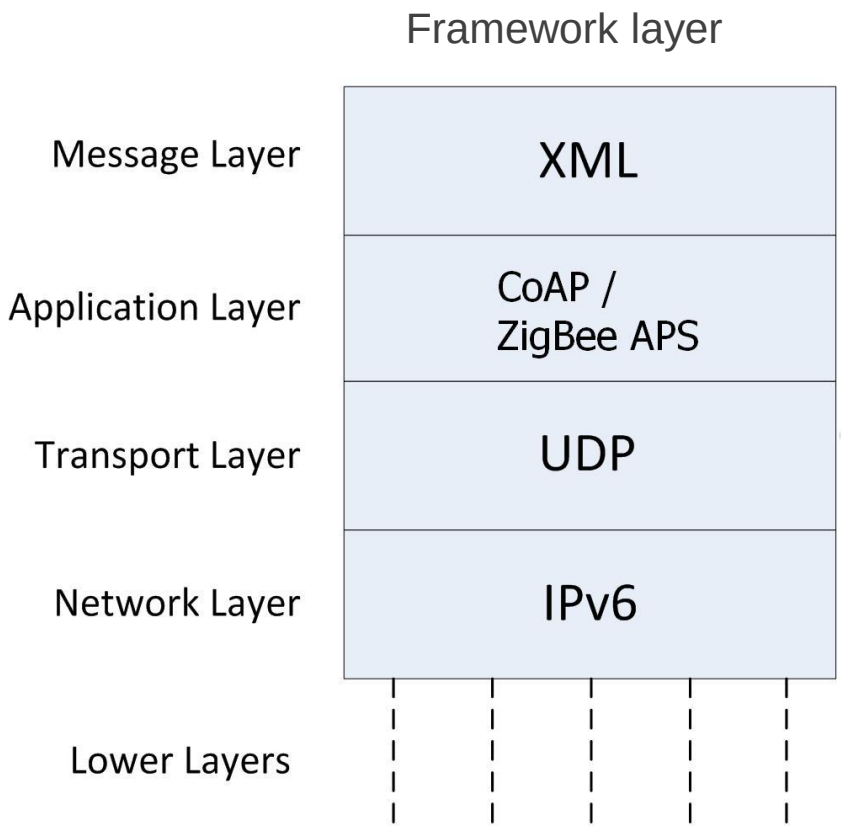
e.g. a sensor network in a (i.e. chemical) plant interconnects with sensor networks in the surrounding

Challenges: traceability of the data; physical attacks (outside sensors); admission control for screening of external sensors



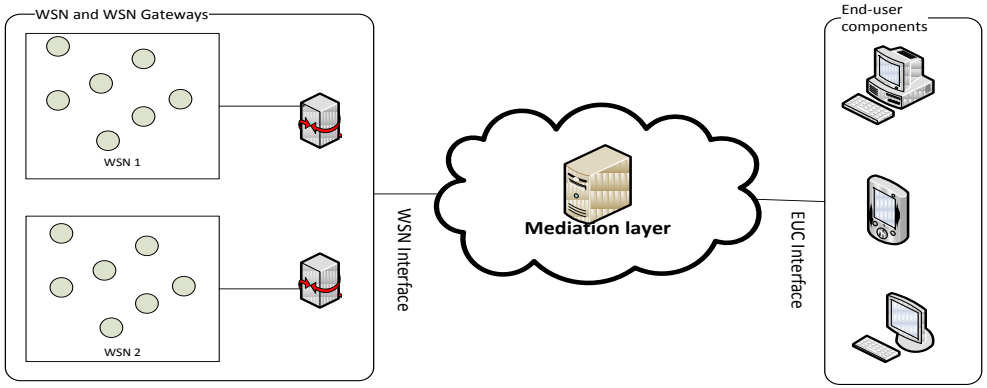
Security Framework

- Currently sensor nodes communicate with IPv6 using 6LoWPAN
  BUT: framework shall not be limited to IP connectivity
- **Question:**
  Would IPSec just solve the problem?
  Answer:
  NO
  - due to WSAN constraints mentioned
  - IP is not the only protocol for WSANs
- **Result:**
  - design of a lightweight security framework using state-of-the-art techniques
  - sensor-side and server side components



Security Framework

- 1..n Wireless Sensor Networks
- 1..n Gateways to the wired world
- 1..n Server (-locations)
- 1..n End User Components

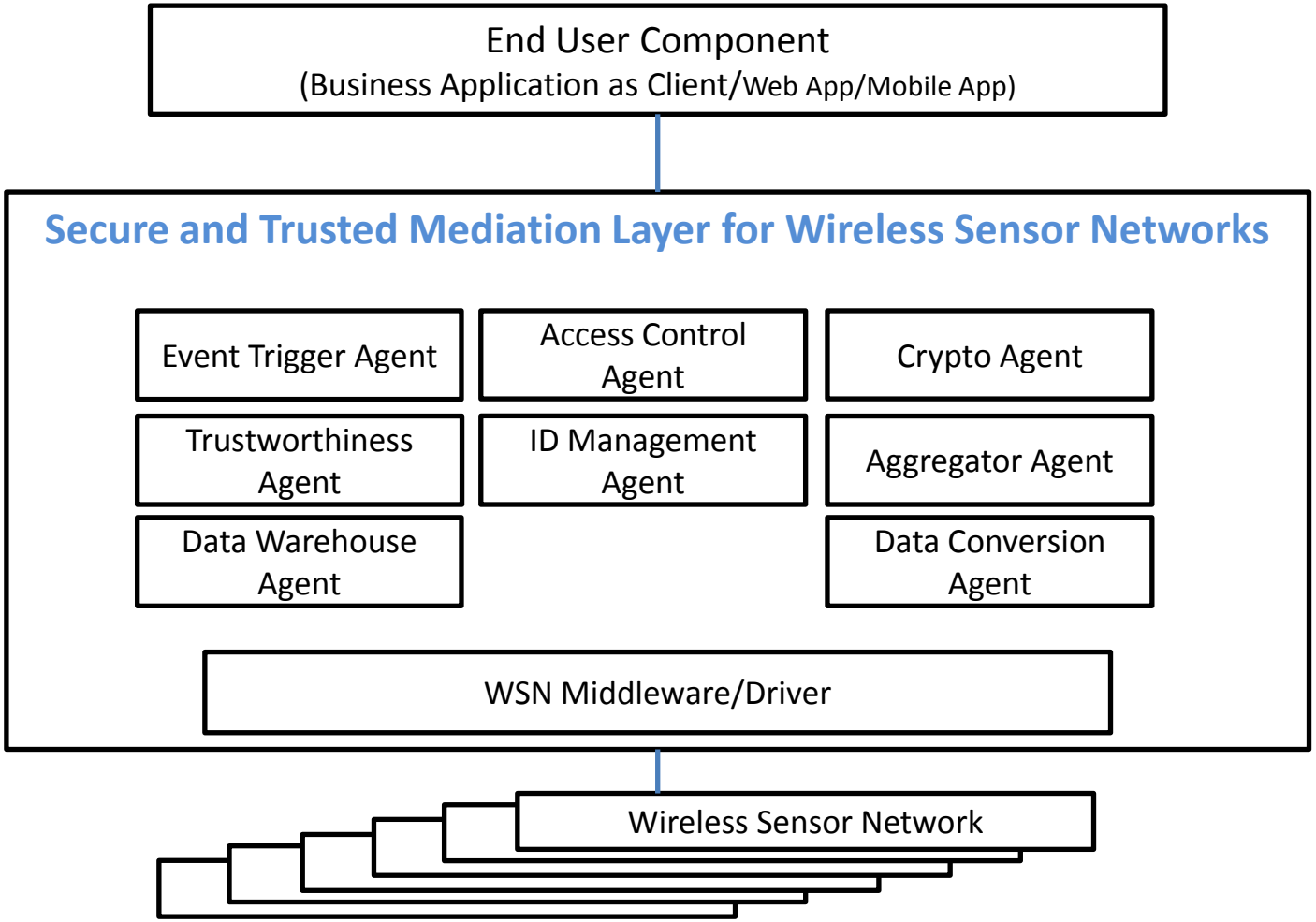


Key element: The Mediation Layer (ML)

The Mediation Layer is:

- a layer between WSN Middleware and Business Application
  Purely server-side component
  Stack: Business-Application/**Mediation-Layer**/Middleware/WSN/…
- a framework/container for many different mediation-agents
  defines synchronous and asynchronous communication between agents
  Modular architecture, supports hot-swapping

Mediation Layer Structure



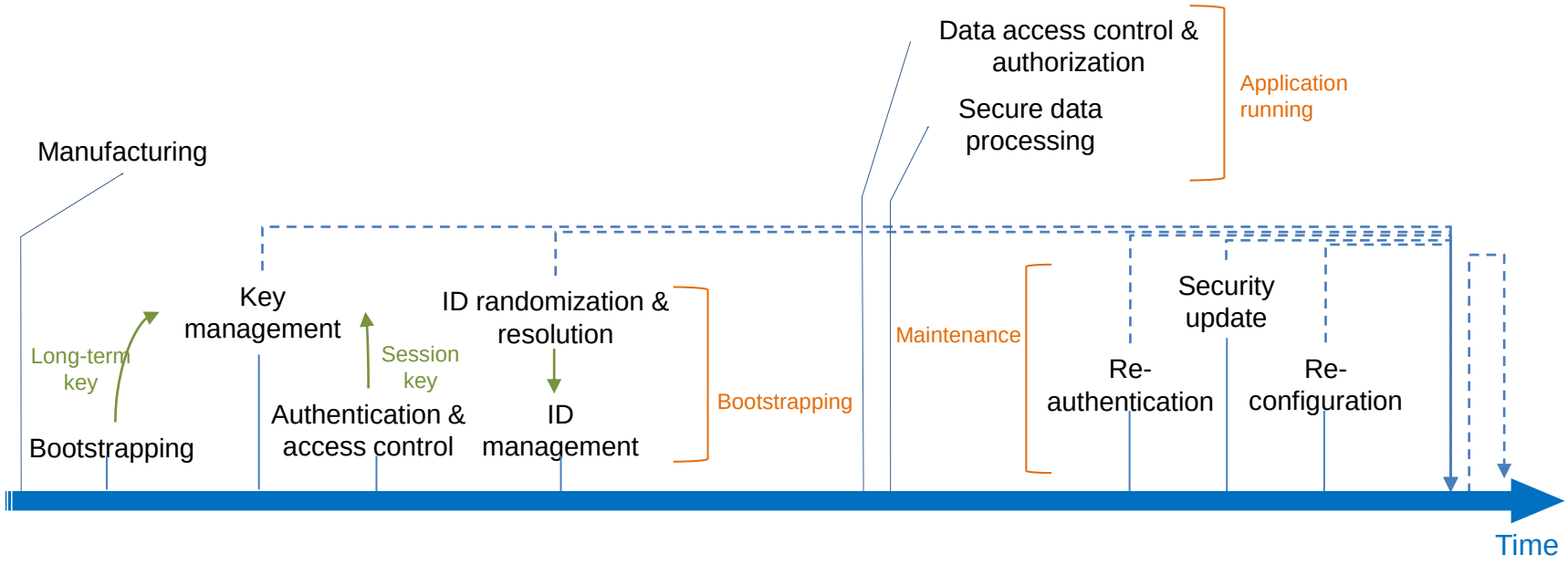
Security Levels

- Default security level selected according to requirements and resources
- The framework can move upwards on certain events depending on the node properties (supported security levels)
- Different services can have different security levels
  e.g. key exchange requites level 4 but data transport level 1

Lvl	Confidentiality	Cryptography	Trust
0	No confidentiality	None	All entities in E2E-path trusted
1	H2H security	Symmetric	All hops have to be trusted (WSN, ML)
2	E2E security from WSN to ML (ML has access to plain data)	Symmetric	Trusted ML
3	E2E security from WSN to EUC	Symmetric, not homomorphic, complex key distribution needed	No complex trust requirements
4	E2E security from WSN to ML (ML has access to plain data)	Asymmetric	Trusted ML
5	E2E security from WSN to EUC	Asymmetric, not homomorphic	No complex trust requirements
6	E2E security from WSN to EUC	Asymmetric, homomorphic	No complex trust requirements

Sensor lifecycle (excerpt)

- Different keys for the different stages of a sensor lifecycle
- Management and updates shall be done automatically
- Re-establishment of the sensor status after a power failure
- Re-establishment of the sensor status after a firmware update (OTAU)

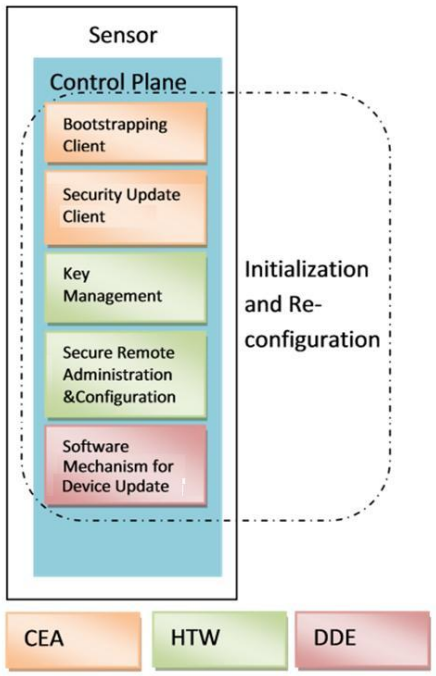


Sensor modules

- Most modules on the sensor have a server-side counterpart
- Some modules are obligatory
  e.g. Failure anticipation client
- Some modules offer node based services for the other modules
  e.g. Crypto service
- Each module has “capabilities” that the framework (network/gateway/mediation layer have to respect

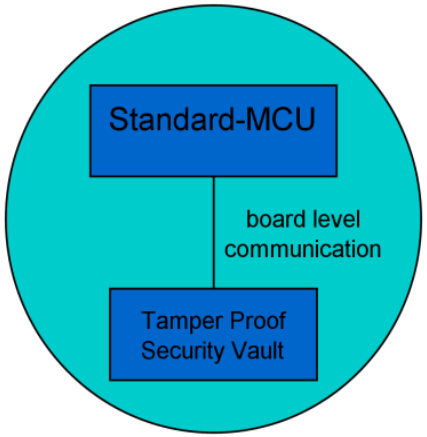
- The sensor/gateway/server components are developed/implemented by all partners

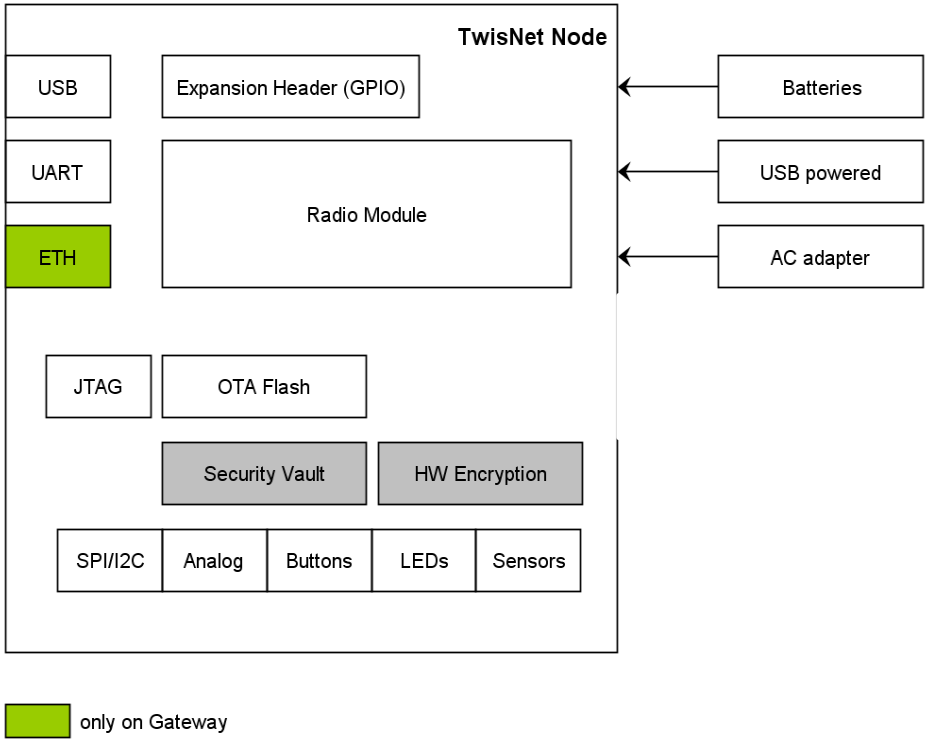
Modules
Bootstrapping client
Authentication client
ID randomization/resolution client
Access control client
Failure anticipation client
Network monitoring client
Data monitoring client
Threat detection client
Trust management cli
Secure remote admini
Security update client
Device (firmware) upd
Crypto service
Key management serv



Hardware Building Blocks

- TWISNet node contains sensors and interfaces for a wide variety of proof-of-concept demonstrations
- Standard MCU + Security Vault allows more flexibility compared to Secure MCU





Current TWISNet platform

- Dedicated node and gateway platforms
- The framework uses the micro controller of the radio module
- Radio modules can be changed to support different ARM (32 Bit) and AVR (8 Bit) controller
- Low power battery operation possible for nodes only
- Security related hardware must be attached via the expansion header

All partners use the same platform for the implementation of the framework!

This work is supported by the EC under grant agreement FP7-ICT-258280 TWISNet project.