MINDSHARE
2025 10–11 SEP
CYBERSECURITY LEADERSHIP FORUM
Securing Identity for our Digital Future

Tobias Frauenschläger,
OTH Regensburg

Using Secure Elements to Improve Crypto-Agility in Operational Technology

Agenda

1. State of the Art
2. Our Work
3. Outlook and Future Work

1

State of the Art

Let’s analyze the title in reverse

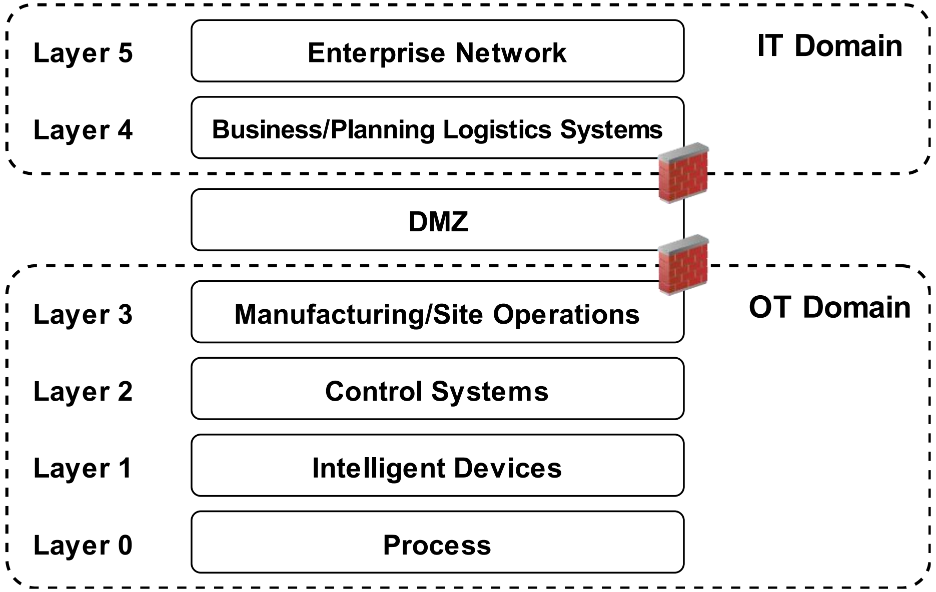
Operational Technology

Control Center

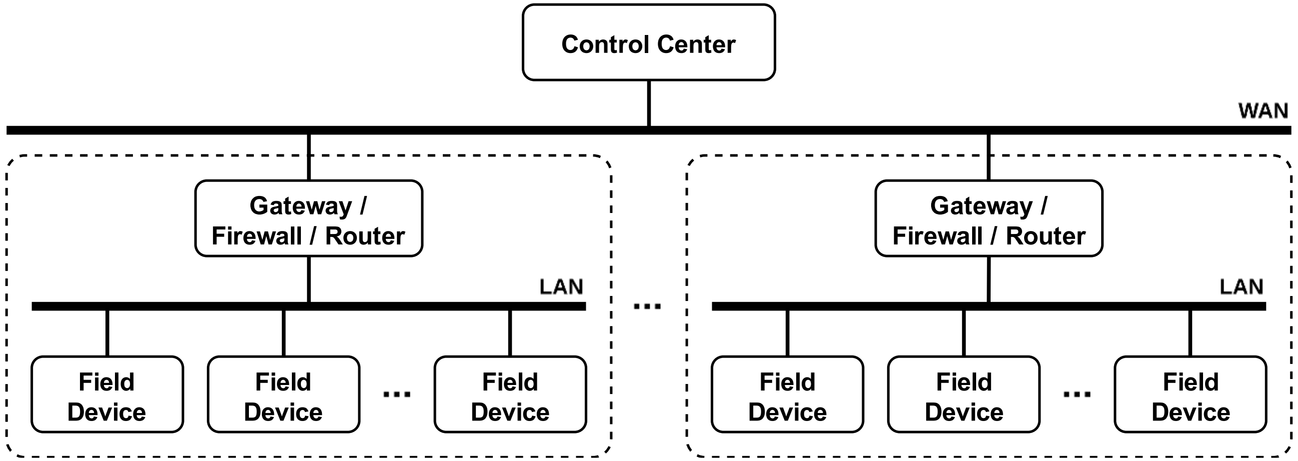
BITSIGHT

Field Devices

OT communication systems



Purdue Enterprise Reference Architecture



Secure communication within OT SCADA systems

OT-Security

- Various **regulations** prescribe thorough security measures with ongoing updates.
- OT communication systems generally support current security measures but feature only **limited update capabilities**.
- **Long mission times** and compatibility **to legacy systems** slow down adoption of new security measures.
- **Security lifetime shorter** than OT mission times.
- Threat of quantum computer leads to **Post-Quantum Cryptography** migration
- Long-term **key-management** oftentimes a hard problem

→ **Update mechanisms** are required

Crypto-Agility

Crypto-Agility describes the capability of updating and replacing security measures during the lifetime of a component:

- Update the **implementation** of existing security measures
- Update the list of **supported cryptographic algorithms** and their security parameters
- Incorporate and adapt to **new functionality** transparently
- Incorporate regional security **regulations** and comply with regional peculiarities
- Create **transition mechanisms** to enable safe and secure migrations to new security measures

Secure Elements

Soldered Chips

- → Secure Elements, TPMs, HSMs, …
- → Intended for Embedded, IoT, OT, …
- ✓ High system security within a device
- × Static, no crypto-agility

Exchangeable Smart Cards

- → Chip Cards, SIM-Cards, SD-Cards, …
- → Intended for personal use
- ✓ High flexibility, high potential for crypto-agility
- × Larger security attack surface

Tongchai Cherdchew / EyeEm/Getty Images

2

Our Work

Approach

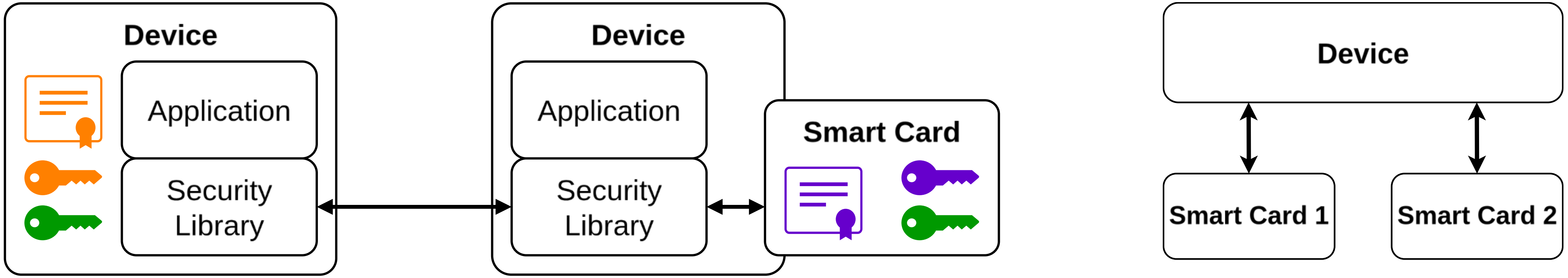
Problem: Management and protection of long-term artifacts (private keys, certificates, PSKs), support for new cryptographic implementation (certified)

Solution: Use exchangeable smart cards in OT devices for artifact storage and to execute cryptographic operations

The deployment of a new smart card:

→ Simplifies the rollout of **new artifacts**

→ Enables support for **new algorithms**/implementations



Implementation

Use Case: Transport Layer Security (**TLS**) handshake (implementation with WolfSSL)

→ Ephemeral key exchange

→ Authentication via handshake signature with long-term private key of entity certificate

→ Signature verification of peer certificate chain (with local root store)

→ Key derivation with long-term pre-shared key (PSK)

Key Exchange	**Signatures**	**PSK Derivation**
ECDHE, **ML-KEM (PQC)**	RSA, ECDSA, **ML-DSA (PQC)**	**HKDF**

Implementation

Requirement: PKCS#11 Version 3.0 for HKDF support
Version 3.2 for PQC support (ML-KEM, ML-DSA, SLH-DSA)

Key Exchange

ECDHE,

ML-KEM (PQC)

Functions:
`C_EncapsulateKey()`
`C_DecapsulateKey()`

Mechanism: `CKM_ML_KEM`

Key type: `CKK_ML_KEM`

Signatures

RSA, ECDSA,

ML-DSA (PQC)

Functions:
`C_Sign()`
`C_Verify()`

Mechanism: `CKM_ML_DSA`

Key type: `CKK_ML_DSA`

PSK Derivation

HKDF

Functions:
`C_DeriveKey()`

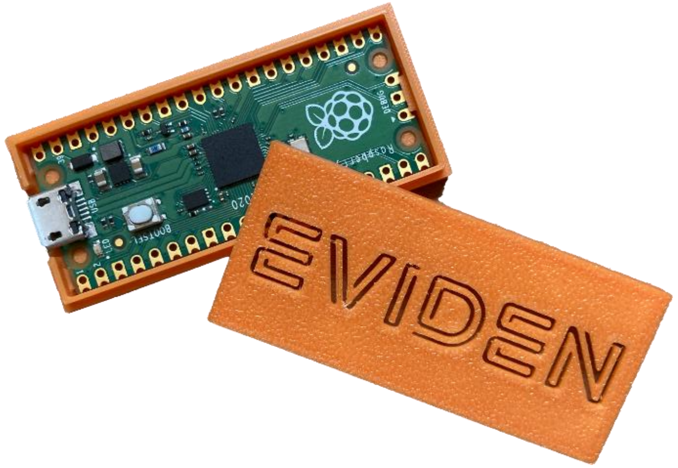
Mechanism:
`CKM_HKDF_Derive`

Key type: `CKK_HKDF`

Implementation

Joint research project with Eviden: KRITIS³M (01/2023 – 12/2025)

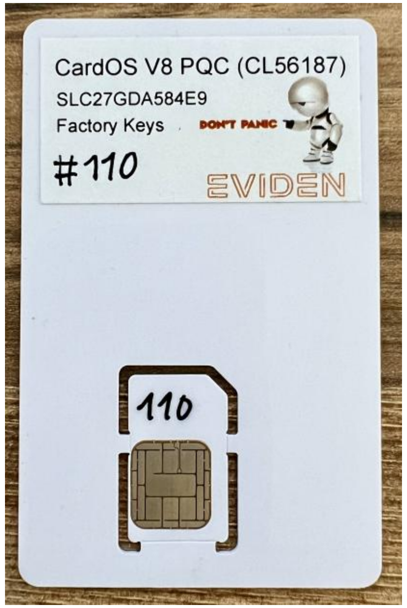
PQCLM: Post-Quantum Crypto Learning Machine



Interfaces:

- USB (CCID)
- I²C

„Real“ **PQC Smart Card**



Interface: ISO7816

Evaluation

Measuring **TLS Handshake duration** (time-to-first-byte) using two Raspberry Pi 4 (one uses smart card via CCID USB reader)

Handshake signature only

Algorithm		Software Only [ms]	V8 [ms]	V5.3 DI [ms]
ECDSA	256	3.77	125.11	209.89
	384	6.89	167.73	301.79
	521	13.62	246.79	478.99
ML-DSA	44	4.03	431.22	-
	65	5.34	648.56	-
	87	7.30	765.68	-

Evaluation

Measuring **TLS Handshake duration** (time-to-first-byte) using two Raspberry Pi 4 (one uses smart card via CCID reader)

Complete Handshake Offloading

Algorithm	Software Only [ms]	V8 [ms]
ECDSA 256	19.80	2959.91
ML-DSA 44	47.22	4127.58

Operations:

- Ephemeral key exchange
- Create the handshake signature
- Verify peer certificate chain

Pre-Shared Key Offloading

Algorithm	V8 [ms]
PSK only	94.16
PSK + ML-DSA 65	711.28

ML-DSA 65: 648.68 ms

4

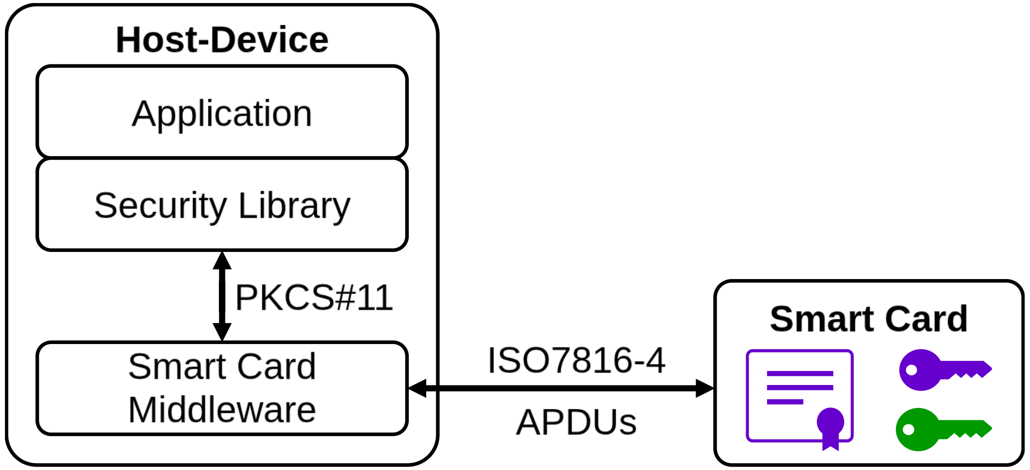
Conclusion and Future Work

Achieved Crypto-Agility

- Physical exchangeability of smart cards enables **partial hardware upgrade**
- PKCS#11 between Security Library and middleware achieves *solid* crypto-agility:
  → No algorithm implementations in Security Library necessary
  → No change for new smartcard with the same set of algorithms
  → **But**: extended API for new algorithms in the future (mechanisms, key types, functions, …)

Future Improvement: *Generic Trust Anchor API* instead of / in addition to PKCS#11

→ More generic and abstract interface for the Security Library **independent of the algorithm**



OT specific Security Considerations

Problem: exchangeable smart card in **unprotected** environment

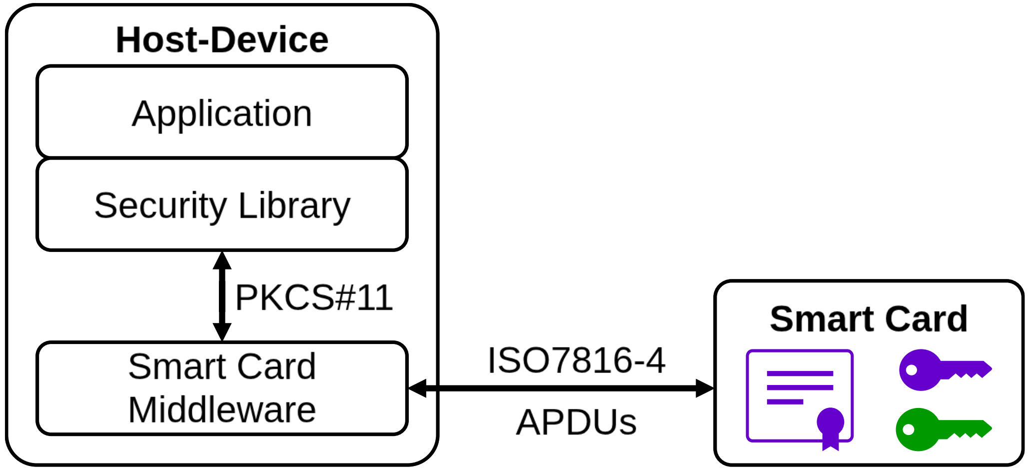
Threats:

- Communication Tampering
- Smart Card Theft
- Malicious Smart Card Insertion

Solution: *Secure Pairing* between Host and Smart Card

- Exchange of symmetric key during pairing
- Mutual Authentication during startup
- Secure Channel Establishment

→ Future Work



Contact

Tobias Frauenschläger, M.Sc.

Research Associate
Laboratory for Safe and Secure Systems, LaS³
University of Applied Sciences Regensburg
OTH Regensburg
Seybothstraße 2
DE, 93053 Regensburg

E-Mail: tobias.frauenschlaeger@oth-regensburg.de
Phone: +49 941 943 9520