Trusted Cyber Physical Systems (TCPS)

Dave Thaler, Partner Software Engineer

August 2017

Windows

Trusted Cyber Physical System: The Problem

Problem

- Attacks against **safety critical systems** can have devastating consequences
- **Need proof of correct operation** for regulatory, insurance, etc. reasons

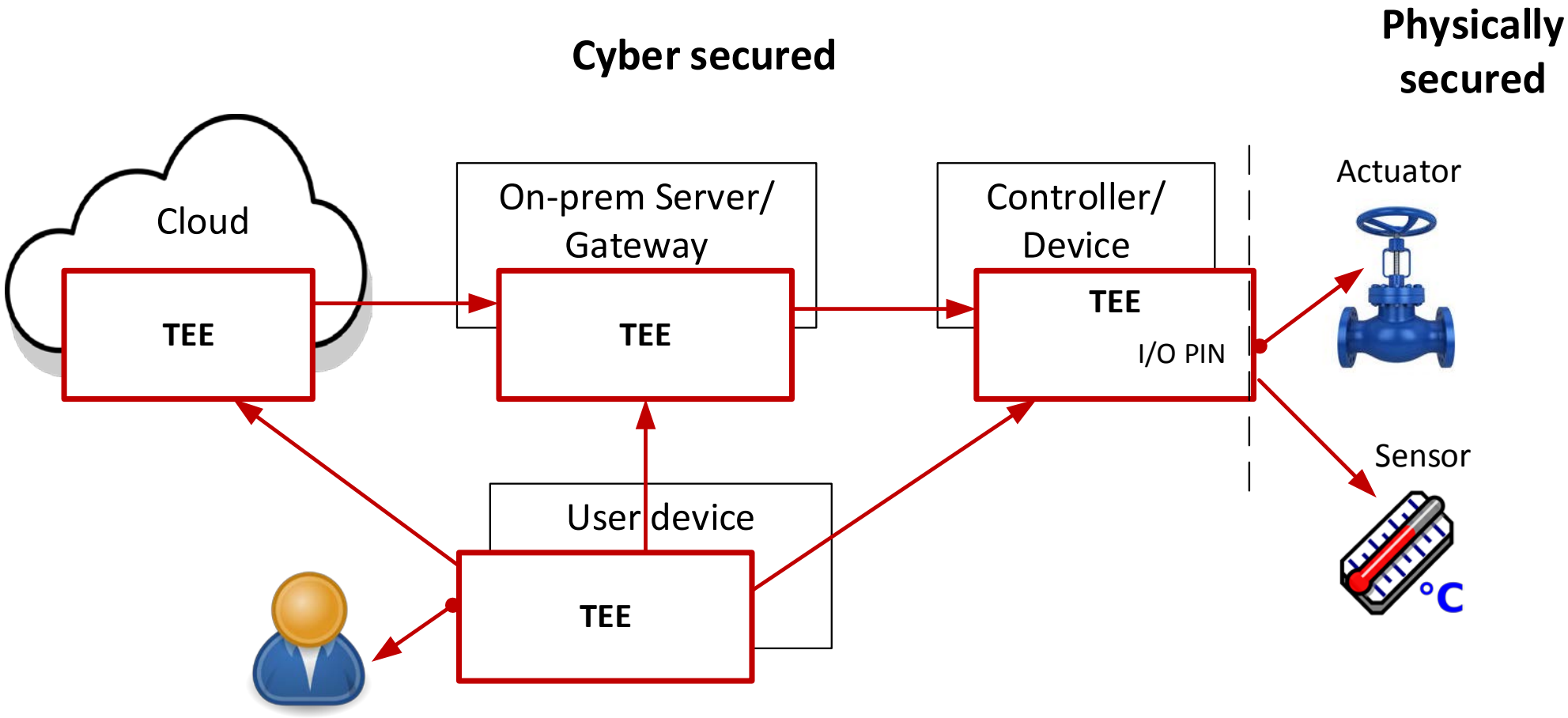
Besides the usual threats, additional threats that must be addressed include:

- **Malware** might has compromised a normally-trusted device
- **Rogue administrator** with access to the hosting system, but without operating rights
- **State actors** might have injected vulnerabilities or do surveillance via software vendors, hardware vendors, cloud hosters, etc. in their jurisdiction

Strong security promises needed

- **Tamperproof authorization and non-repudiated auditing to control and monitor actions**
  - No one (including malware) can execute actions except as authorized by the owner.
- **Data flow and storage throughout the infrastructure is encrypted and integrity protected, only giving entities authorized by the owner access**
  - No one (including malware) can decrypt, alter or replay security-sensitive data without the owner's explicit permission.
  - If a cloud hoster is given a subpoena, FISA order, etc., it cannot give out highly-secure data since all it gets is a random-looking set of bits.
- **Use of well proven industry standards provides transparency and trust in all security-related operations throughout the system**
  - Code in the trusted computing base is vettable by a customer (or their security service).

Trusted Execution Environment (TEE) at each endpoint



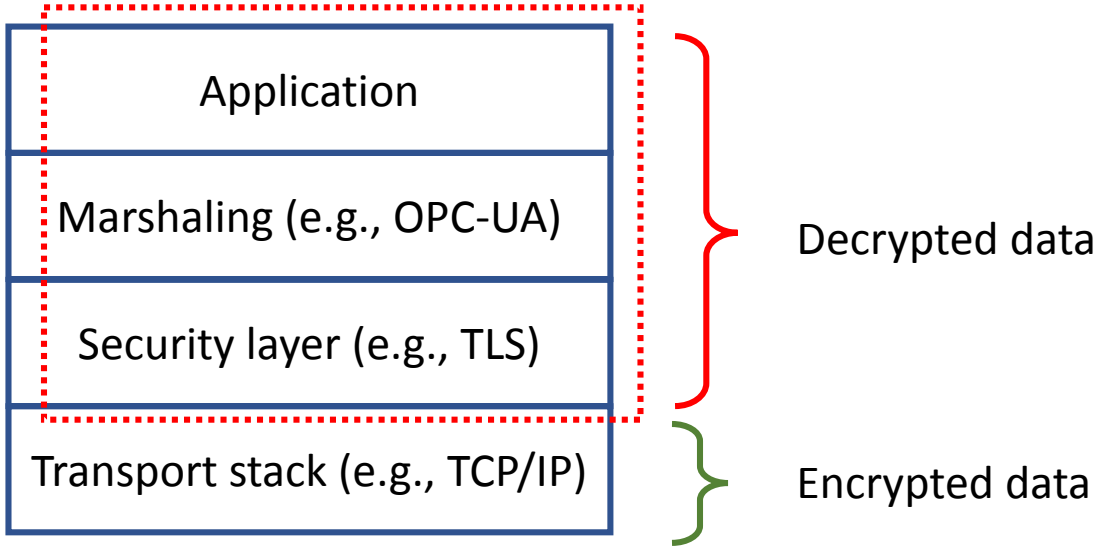
I/O to and from non-TEEs is accessible <u>ONLY</u> to a TEE

Trusted Execution Environments

- A TEE provides hardware-enforcement that
  1. the device has a unique security identity
  2. any code inside the TEE is operator-authorized code
  3. any data inside the TEE cannot be read by code outside the TEE
  4. any trusted peripherals/busses cannot be accessed by code outside the TEE

- Already widely deployed in the payment industry (e.g., chip-and-pin cards)
- Already adopted in some standards bodies (GlobalPlatform, OneM2M, etc.)

Communication stack threat



- Any section of the communication stack which handles decrypted data provides an **attack surface for Malware**
- Moving the communication stack layers which handle unencrypted data into an **Trusted Execution Environment (TEE) mitigate the attack surface** significantly

How can I trust code?

- The code is running on **hardware you trust**
- Any **trusted roots** used by the TEE code are **chosen by the owner**
- The **source code** for TEE code **is available and small** enough for a security lab to evaluate and certify.
  - This is to mitigate attacks where the code contains a back door.
- The **toolchain** used to produce or verify the TEE code **is available** and small enough for a security lab to evaluate and certify.
  - This is to mitigate attacks where the compiler inserts back doors.