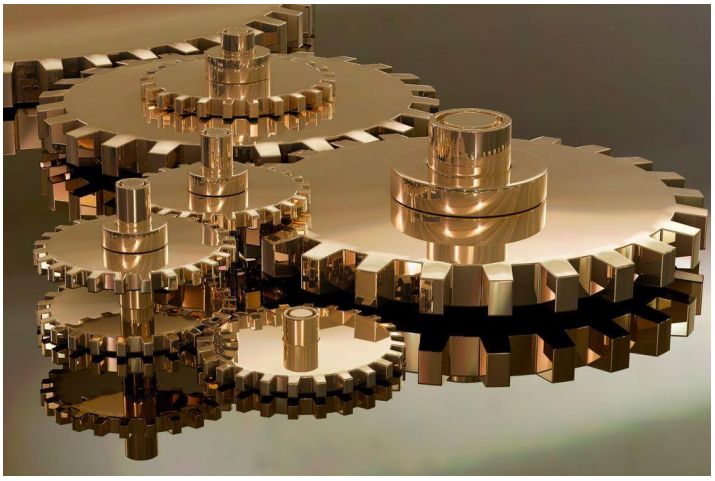
Security Certification of IoT Devices with a Component-based Software Design

28.11.18 | Author / Editor: Sergey Tverdyshev * / Sebastian Gerstl



Different types of IoT applications are subject to different security standards. Whether you want to meet the requirements of the Common Criteria for Information Technology Security (ISO 15408), IEC 62443 for Industrial Control Systems, EDSA (Embedded Device Security Analysis) or J3061 in the automotive sector:

We will show you how you can quickly meet all necessary requirements with a component-based software design.

(Image: Public domain/ CC0)

The reason that IoT approaches are so successful is that open standards and open protocols enable embedded, commercially available off-the-shelf components to be linked together in a comparatively loose manner. This level of modularity is also reflected in the certification standards. In this article, we will show where and how a component-based software design can make it significantly easier to meet certification requirements.

In the Internet of Things, conventional IT security is increasingly being expanded to encompass embedded components. One feature of security requirements shared by many IoT systems is that integrity and availability are significant focal points. This can be seen in certification standards as well: The conventional Common Criteria for Information Technology Security (ISO 15408) are being supplemented by domain-specific security standards, such as IEC 62443 for Industrial Control Systems, EDSA (Embedded Device Security Analysis) or J3061 in the automotive sector, all of which feature a strong focus on "security for safety".

The Internet of Things (IoT) is extremely diverse, highly dynamic and always available – and therefore always vulnerable to attack. The IoT consists of IoT components as modular elements (see also I. Yaqoob, E. Ahmed, I. Hashem, A. Ahmed, A. Gani, M. Imran and M. Guizani, "Internet of Things Architecture: Recent Advances, Taxonomy, Requirements, and Open Challenges").

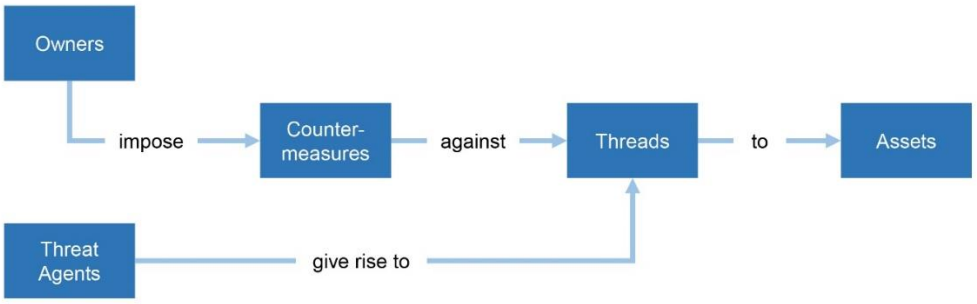


Image 1: Simplified representation of a security model according to the Common Criteria and IEC 62443-1-1.

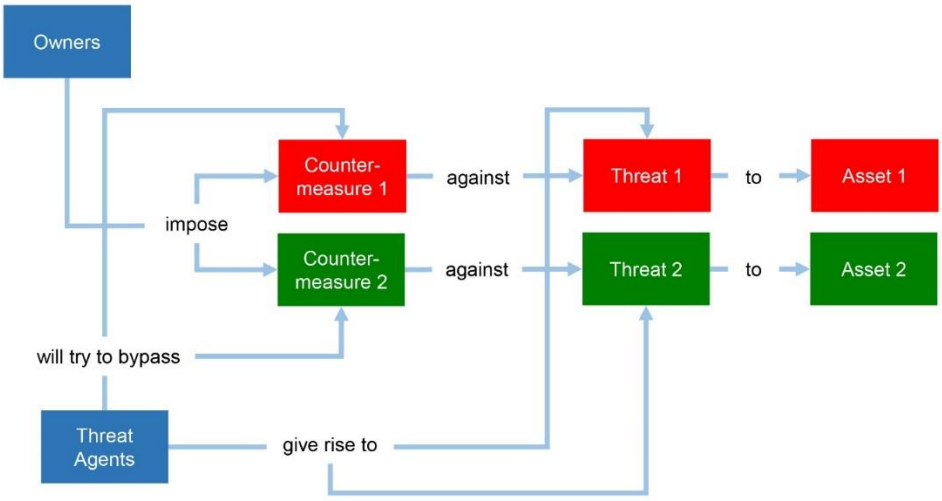


Image 2: Security model CC and IEC 62443-1-1 with two assets.

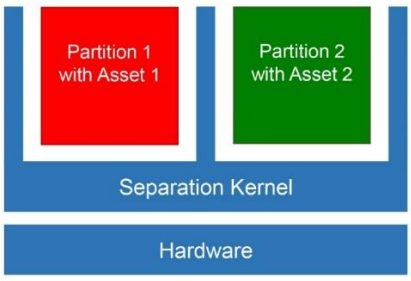


Image 3: The two assets from the example shown in Figure 2 on a MILS system.

Security also tends to be modular

Figure 1 shows the security model in accordance with Part 1 of the Common Criteria (CC) (Section 7.1), which is explicitly incorporated in IEC 62443-1-1 (Section 5.1) as well. This Figure shows all assets, threats and countermeasures as individual boxes. In our experience, however, a closer look reveals that the situation is often closer to that shown in Figure 2. The table below shows an example of a possible allocation with two assets:

Number/Colour	Asset	Thread	Countermeasure
1/red	Engine control	Obstacle detection → Personal injury	Real-time guarantees, separation
2/green	Performance logger	Loss of operational history	Password in the website interface

In the example shown above, the assets and the threats to them have vastly differing levels of criticality. The specific allocation in Figure 2 also makes it clear that an additional threat exists, namely that the attacker could try to bypass the countermeasures ("will try to bypass"), e.g. Ethernet access to the performance logger could be abused to attack the engine control. One means of governing complex IT systems and in particular, IoT components with different levels of criticality at an appropriate abstraction level, is to divide them into security domains ("divide and rule"). A security domain is a zone in which all objects are subject to the same security policy. The boundaries of security domains are also known as "trust boundaries".

Establishing a security architecture for Common Criteria certification

With the Common Criteria for Information Technology Security Evaluation, a manufacturer works together with an examining body to evaluate an IT product put forward by the manufacturer. The product can consist of software or both software and hardware, and thus explicitly includes IoT components. If the evaluation is successful, the certification body, which in Germany is the Federal Office for Information Security (Bundesamt für Informationstechnik (BSI)), will issue a certificate.

The Common Criteria require that the developer provide design documentation as the central component of the documentation to be submitted to the examining body, in which the product must be broken down into subsystems (one-tier) or subsystems and modules (two-tier), depending on the desired evaluation level. The properties of subsystems and modules and their interactions must be described. The design documentation also describes the extent to which the interfaces of the subsystems and modules are directly or indirectly accessible to attackers.

A security architecture is a means of analysing and documenting the security properties of a system with regard to its security domains.

A security architecture (ADV_ARC) in accordance with the Common Criteria elucidate the following points:

- Which security domains does the system have? To what extent are these security domains fully separate, or are they able to communicate with one another (in a controlled manner)? In our example, the performance logger and the engine control were different domains.
- How is the system initialised?
- How does the system itself protect against attempts by attackers to attack it?
- How does the system protect against attempts to bypass it (in our example: Web access to the performance logger cannot bypass the engine control)?

Security architecture in accordance with IEC 62443

IEC 62443 is a standard for the security of industrial control systems as a whole (in particular Parts 3-1 to 3-3) and their components (in particular Parts 4-1 and 4-2). IEC 62443 is referenced by IEC 61508 safety standard for security (IEC 61508 Part 1-1 Section 7.5.2.2: “If security threats have been identified, then a vulnerability analysis shall be undertaken in order to identify security requirements. Note: Guidance is given in the 62443 series”). IEC 62443 is to a large extent still in (advanced) development by IsaSecure.

In IEC 62443, the security domains are known as "Zones" and the system should support partitioning into zones (IEC 62443 Part 3-3 Section SR 5.4) and operate resource management that is well protected against attackers (IEC 62443 Part 3-3 Sections SR 7.1 and SR 7.2, and IEC Part 4-2 Sections CR 7.1 and CR 7.2), including protection against denial-of-service attacks, for instance.

With regard to the development process, IEC 62443 Part 4-1 SR-2 requires the creation of a threat model with trust boundaries which also governs how information flows across these trust boundaries. A defence-in-depth design is recommended (IEC 62443 Part 4-2 SD-2). IEC 62443 Part 4-1 SD-6 also requires that one of the design goals of the system must be to minimise the attack surface.

Security architecture in accordance with IsaSecure EDSA / SDLA / SSA

With its SDLA (processes), EDSA (functional systems for components) and SSA (functional requirements for entire systems) standards, IsaSecure has developed a precise certification scheme for the IEC 62443 series, which also includes suggestions taken from NIST 800-53. For instance, EDSA-311 includes the requirements on the functional level listed in the table below:

FSA-RDF-1	The IACS embedded device shall provide means to enforce assigned authorizations for controlling the flow of information outside the embedded controller zone and between interconnected systems in accordance with user specific policy.
FSA-RDF-2	The IACS embedded device shall separate data acquisition services, from management functionality.
FSA-RDF-3	The IACS embedded device shall isolate security functions from non-security functions by means of partitions, domains, etc., including control of access to and integrity of, the hardware, software, and firmware that perform those security functions.
FSA-RDF-4	The IACS embedded device shall prevent unauthorized and unintended information transfer via shared system resources where it supports connection sessions from users with different levels of access.

Just as in IEC 62443 Part 4-1, SDLA-312 requires a modular design on the process level (SDLA-DSD-1.*) and clear identification of trust boundaries and attack surfaces (SDLA-SAD-*).

Security architecture in J3061

J3061, published by SAE, is the most recent of the standards we take into account, and the version that we reference is the draft published in 2016. In this standard, the creation of software architecture begins with "Refine Functional Cybersecurity Concept into Technical Cybersecurity Concept" (Section 8.4.3), and in turn the isolation of specific functions is important for this. The following is used as an example: “Isolation/partitioning of systems that have external access (e.g. Wi-Fi, Bluetooth, OBD) from safety-critical systems and systems that can have important impacts on the operation of the vehicle.” The software architecture is then subjected to a threat analysis with regard to confidentiality, integrity and availability (Section 8.6.3) and analysed for vulnerability and threats. Section 8.6.4 specifies STRIDE, ASF, and DREAD as possible tools to help with threat categorisation.

Common features and outlook

In this consideration of requirements for software architecture, we have primarily restricted ourselves to the security architecture on the left-hand branch of the V-model. It is clear that compliance with these requirements simplifies not only the design process, but also the testing/vulnerability analysis.

We have noted that all standards relevant to the IoT that we have investigated (Common Criteria/ISO 15408, IEC 62443, EDSA/SDLA/SSA and J3061) require a security architecture that provides isolation, resource management and information flow control between security domains (also known as "zones" or "partitions").

When considering material consumption (e.g. one control unit per security domain), this kind of architecture with clear, bypass-proof separation of tasks seems costlier at first glance. However, material consumption can be controlled via virtualisation as follows:

- With regard to network virtualisation, new developments in load-balanced real-time network standards are particularly interesting (e.g. TSN, IEEE 802.1 Qbu/Qbv), as these enable secure separation of wiring.

- With regard to CPU virtualisation, the MILS concept, which originates from the security-sensitive and material-sensitive avionics sector, has been becoming increasingly popular in recent years.

In a MILS system, the example shown at the beginning in Figure 2 would appear as shown in Figure 3.

When implementing a MILS system as shown in Figure 3, the isolation, resource management and information flow control as required by the Common Criteria, IEC 62443, EDSA, and J3061 is provided by a separation kernel. The partitions created by a separation kernel form the basis for security domains in IoT devices which use a MILS architecture.

*Dr. Inf. Sergey Tverdyshev is Director R&T (Research & Technology) at SYSGO AG.