Technical Article

Security versus Functional Safety: a View from the Processor Software Development Kit

Elvita Lobo

What Is the Difference between Security and Functional Safety?

At its core, the difference between security and functional safety can be summed up in one word – intent. Security protects against threats from malicious intent. Functional safety on the other hand protects against hazards from malfunction, the cause (or intent) of which may or may not be malicious. As an example, consider an industrial motor drive. Preventing the drive from supplying torque-generating energy and restarting unexpectedly is a safety function. If this same drive can be controlled remotely, sending data and receiving commands over the network, enabling only authenticated access to the drive is an example of a security feature.

So What Is Common between the Two?

The goal of both, functional safety and security is that **system integrity** is preserved in the face of the hazard or threat, no matter whether the source is malicious intent or malfunction. What this means is minimizing the likelihood of unacceptable risk from the hazard or threat to assets or human life. Since functional safety and security have the common goal of avoiding or mitigating impact from hazards and threats, there can be overlaps in the mechanisms used to achieving this goal.

System Integrity as It Applies to Security

To prevent system integrity from being compromised, you need to be able to monitor and control the software that runs on a system via some kind of approval mechanism. This mechanism should check the software running has not been altered (intentionally or unintentionally) between "there" and "here," or between "then" and "now". This is **integrity** as it applies to security.

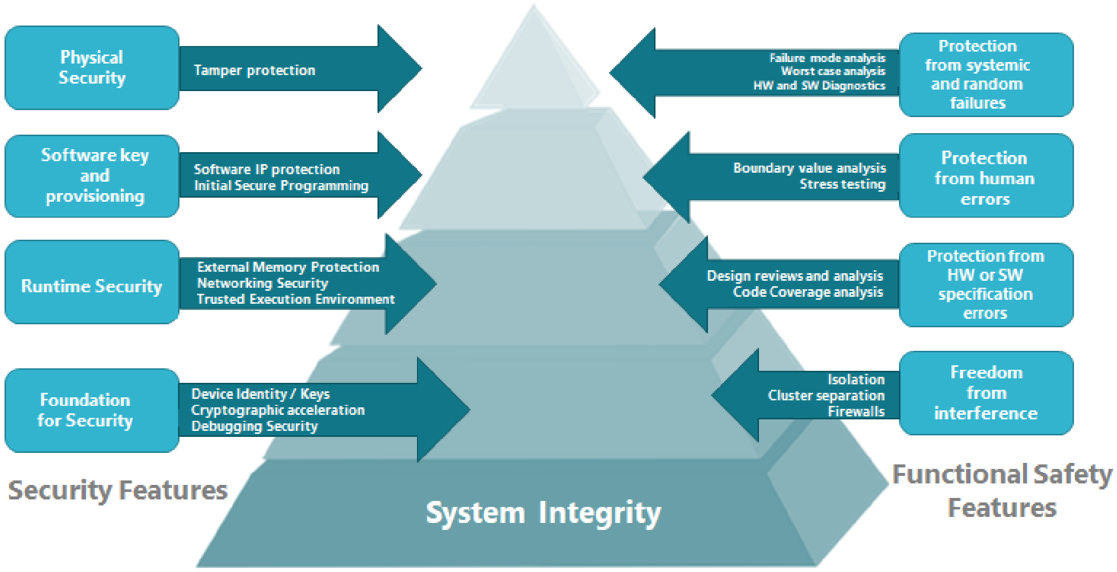
To prevent system integrity from being compromised, we also need to verify that the source of the software is trusted (i.e. you are who you say you are). This is **authentication,** as it applies to security.

Although crucial building blocks, integrity and authentication are just some of the security features a system may need. Access control, confidentiality, availability and non-repudiation are other crucial security features.

System Integrity as It Applies to Functional Safety

Functional safety standards mandate following a **rigorous software development process.** This process includes **traceability** from the software product specification through final test and validation. Developing software targeting functional safety standards requires **multiple cross functional reviews** such as failure mode analysis and design reviews and analysis. Hardware and software **diagnostics testing** is deployed to detect faults during system operation.

To summarize, the process of quantifying risk from hazards or threats and then deploying techniques to mitigate impact from them is the common goal that drives the development of a system with security and functional safety features.



Security and functional safety features in Processor SDK AM65x

Texas Instruments' Processor SDK

portfolio of devices supports a host of security features as part of both Linux and TI-RTOS operating systems, and is available today via TI's software portal.

TI's next generation **AM6x device family** will support both functional safety and security features in hardware and software. Typical functional safety features of this device include cluster separation, hardware and software diagnostics and a dual-core lock step cortex R5F. Functional safety features target automotive and industry standards ISO 26262[1] and IEC 61508[2]. Typical security features include whitelist firewalls, debug security, cryptographic acceleration, initial secure programming, secure boot and runtime security.

The AM654x family coupled with the Processor SDK portfolio leverage decades of TI's expertise in the areas of security and functional safety. With automation and cloud-based communication becoming commonplace in applications, these devices are designed keeping in mind the systems of the future.

References and further reading:

- [1]ISO 26262 Road vehicles – Functional safety
- [2]IEC 61508 Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems
- TI's security whitepaper
- The state of functional safety in Industry 4.0
- AM654x board tour

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265