# Safety and security

## Secure Embedded Systems

Ensuring the security of embedded systems is important to prevent unauthorized access or manipulation of the system and to protect the confidentiality, integrity, and availability of the system and its data.

There are various approaches to securing embedded systems, including the use of secure processors and specialized security hardware, the implementation of security protocols, and the use of secure coding practices. It is also important to have a system in place for distributing updates and patches to address newly discovered vulnerabilities.

At AC6 Training, we offer a range of courses on embedded security, including courses on secure coding practices, hardware security, and the use of secure processors.

## Main Courses

**oC1 - Effective MISRA C** MISRA C:2023, including guidelines for safety and security supporting all published versions of the C standard  MISRA C:2023, the latest version of the MISRA C standard, which includes guidelines for safety and security supporting all published versions of the C standard. The course has been designed for the smooth and successful adoption of MISRA C into an organization. Lectures, exercises, tests, hands-on sessions and, optionally, a final exam, will significantly strengthen the skills and competences of teams involved in the design, development and verification of critical embedded software systems.

**oC2 - MISRA Compliance for Project Managers** Manage MISRA compliance, ensuring improved project outcomes and code quality in safety-critical systems  MISRA for Project Managers, provides essential insights for managers overseeing projects that require MISRA compliance. It highlights the importance of MISRA standards in safety-critical systems across various industries. The training emphasizes key aspects such as negotiation, planning, execution, and assessment of MISRA compliance, and equips managers with the knowledge to make informed decisions. By enhancing managerial awareness and skills, it ensures better project outcomes, reduced costs, and improved code quality. This training is invaluable for project leaders seeking to streamline MISRA integration and enhance organizational efficiency.

**oSEC1 - Secure C/C++ Development for Embedded Systems**This course provides an introduction to embedded security and covers industry standards such as ISO/SAE 21434, IEC 62443, NIST SP 800-53, Common Criteria, and OWASP. It covers secure coding practices for C/C++ and introduces the RUST programming language with its built-in security features. Students will learn about secure software development methodologies, security testing, and cryptography in embedded systems. The course covers the design and implementation of secure embedded system hardware architecture and communication protocols. Additionally, it provides an overview of security best practices for IoT devices and systems.

**oSEC2 - Advanced Embedded Systems Security** Create secure connected embedded systems The objectives of this course include learning how to manipulate files and directories securely, protect programs from malicious user input, and understand embedded system hardware features for security.

Attendees will also learn about a secure software development methodology and framework, gain an understanding of the context and use of hypervisors and system virtualization, and become familiar with security checks and tools.

These topics are essential for the development of secure systems software and are applicable to a wide range of applications.

**oSEC12 - Comprehensive Secure Systems Programming**The oSEC12 course is designed for software engineers that need to design and program secure systems. This course is a combination of oSEC1 - Secure C/C++ Development for Embedded Systems course and oSEC2 - Advanced Embedded Systems Security course, with a special price when both consecutive sessions are booked at once.

**oSEC3 - wolfSSL for Embedded Security**The oSEC3 course is designed for software/ Hardware engineers to understand how SSL/TLS Works , establish fundamental knowledge about cryptographic, algorithms, and protocols and Learn how to implement secure authentication with wolfSSL

**oSEC4 - Advanced wolfSSL for Embedded Security**The oSEC4 course is designed for software/ Hardware engineers. The aim of this course is to discover how encryption works and how to manage secret keys, learn how to implement secure authentication with wolfSSL, building wolfSSH on standard Platforms, secure boot using wolfBoot (with wolfCrypt and WolfSSL)and understand how to build wolfMQTT on standard platforms and use it in an IoT application

**oSEC34 - Comprehensive Security with WolfSSL**The oSEC34 course is designed for software/Hardware engineers that need to fully understand how SSL/TLS works, establish detailed knowledge about cryptographic algorithms and protocols and how to implement a full secured environment, integrated in a Public Key Infrastructure, with wolfSSL. This course is a combination of oSEC3 - wolfSSL for Embedded Security course and oSEC4 - Advanced wolfSSL for Embedded Security course, with a special price when both consecutive sessions are booked at once.

**oSEC5 - Embedded Security for STM32-based devices**This course is designed to teach you about the security challenges faced by embedded systems and STM32-based devices. You will learn how to identify potential attack vectors and threats and understand the latest security standards and best practices for embedded systems. You will also learn about secure boot and firmware protection mechanisms and how to implement them.

**oSEC6 - Embedded Security for NXP i.MX-based processors**This course teaches the security challenges of embedded systems and NXP-based devices, covers latest security standards and best practices, and explains how to implement secure boot, network protocols, IoT security, and firmware updates.

**oSEC7 - ARM TrustZone for Cortex-M based devices**This course aims to provide an in-depth understanding of the ARM v8-M architecture and its security features. It covers topics such as the Memory Protection mechanism, Security Attribution unit configuration, management of Security access faults, and building and debugging secure and non-secure software. The objective is to equip attendees with the necessary knowledge and skills to develop secure applications for ARM v8-M based systems.

**oSEC8 - Secured Embedded Linux Platform Build**

## Additional Courses

**C8 - Critical Systems Safety**Embedded systems are more and more critical and subject to safety constraints. This training introduces the main concepts and standards applicable to safety-critical systems.