Formation Writing Secure C/C++ code: This is a Live Online Training

- eLearning et Cours en ligne: Embedded security

oSEC1 - Writing Secure C/C++ code

This is a Live Online Training

Objectives

- Learn how to verify programs are in a secure state on startup and when calling out to other program
- Become familiar with MISRA C guidelines for the use of the C language in critical systems
- Lean ways to use C/C++ safely in critical systems
- Learn interpret the output of MIRS C;2012 checking tools
- how to manipulate files and directories in a secure manner
- Discover how to protect your programs from malicious user input
- How to secure communication with TLS
- Embedded system hardware features for security
- Secure Software Development methodology and framework

Prerequisites

- Some programming concepts are desirable (whatever language)

Plan

First Session

Introduction to embedded security

- Embedded Security Trends
  - Embedded Systems Complexity
  - Network connectivity
  - Reliance on Embedded Systems for Critical Infrastructure
  - Processor consolidation
- Security policies
  - Perfect Security
  - Confidentiality, Integrity, and Availability
  - Isolation
  - Information Flow Control
  - Physical Security Policies
  - Application-Specific Policies
- Security Threats

Writing Secure C/C++ Code

- Safe use of pointers
- Memory allocation and corruption
- Buffer overflow
- Return Oriented Programming
- Core embedded Operating system Security Requirements
- String and format functions
- Integer security
- Concurrency
- File I/O

Exercice : *Memory Overflow Attacks*

Second Session

Secure Coding

- Coding Standards
- Case Study: MISRA C:2012 and MISRA C++:2008
- Embedded C++
- Complexity Control
- Static Source Code Analysis
- Creating a Tailored Organizational Embedded Coding Standard
- Dynamic Code Analysis

Exercice : *Use of static analysis tools*

Cryptography Overview

- Cryptographic Modes
- Block Ciphers
- Authenticated Encryption
- Public Key Cryptography
- Key Agreement
- Public Key Authentication
- Elliptic Curve Cryptography
- Cryptographic Hashes
- Message Authentication Codes
- Random Number Generation
- Key Management for Embedded Systems

Exercice : *Memory Overflow Attacks*

Third Session

Transport Layer Security

- Secure communications
- Authentication
- IoT Protocols
- MQTT
- DTLS
- HTTPS
- CoAP
- TLS Implementation

- Wireless LAN Security and Threats

Exercice : *Installing and using certificates*
Exercice : *Sending secure messages with TLS*

Secure Embedded System Software Architecture

- Secure software architecture goals
- Least privilege, trust and secure processes
- Arm Platform Security Architecture (PSA)

Secure Embedded System Hardware Architecture

- Crypto-Accelerator Overview
- Arm TrustZone
- Secure boot and update
- Hardware options for security

Renseignements pratiques

Durée : 18 heures
Prix : 1850 € HT