



oSEC2 - Advanced Embedded Systems Security

Create secure connected embedded systems

Objectives

- How to manipulate files and directories in a secure manner
- Discover how to protect your programs from malicious user input
- Secure System Software Consideration
- Embedded system hardware features for security
- Secure Software Development methodology and framework
- Apprehend the context and the use of Hypervisors and System Virtualization
- Discover Security checks and Tools

Prerequisites

- Some programming concepts are desirable (whatever language)
- Some cryptography and Linux basics

Course environment

- Theoretical course
 - PDF course material (in English)
 - Course dispensed using the Teams video-conferencing system
 - The trainer to answer trainees' questions during the training and provide technical and pedagogical assistance through the Teams video-conferencing system
- Practical activities
 - Practical activities represent from 40% to 50% of course duration
 - One Online Linux PC per trainee for the practical activities
 - The trainer has access to trainees' Online PCs for technical and pedagogical assistance
- Downloadable preconfigured virtual machine for post-course practical activities

Duration

- Total: 12 hours
- 2 sessions, 6 hours each

Target Audience

- Any embedded systems engineer or technician with the above prerequisites.

Course Outline

First Session

System Software Consideration

- The Operating System
- Multiple Independent Levels of Security
 - Information Flow
 - Data Isolation
 - Damage Limitation
 - Periods Processing
 - Tamper Proof
 - Evaluable
- Core embedded Operating system Security Requirements
 - Memory Protection
 - Virtual Memory
- Guard Pages
- Location obfuscation
 - Fault Recovery
 - Impact of Determinism
 - Secure Scheduling
- Hypervisors and System Virtualization
 - Introduction to System Virtualization
 - Applications of System Virtualization
 - Environment Sandboxing
 - Virtual Security Appliances
- Hypervisor Architectures
- Paravirtualization
- Leveraging Hardware Assists for Virtualization
 - ARM TrustZone
- Hypervisor Security
- I/O Virtualization
- Remote Management
- Assuring Integrity of the TCB
 - Trusted Hardware and Supply Chain
 - Secure Boot
 - Static versus Dynamic Root of Trust
 - Remote Attestation

Exercise: Memory Protection (MPU)

Exercise: ARM TrustZone

Exercise: Secure Boot

Second Session

Data Protection Protocols for Embedded Systems

- Data-in-Motion Protocols
 - Generalized Model
 - Choosing the Network Layer for Security
 - Ethernet Security Protocols
 - IPsec versus SSL
 - IPsec

- SSL/TLS
- Embedded VPN Clients
- DTLS
- SSH
- Custom Network Security Protocols
- Secure Multimedia Protocols
- Broadcast Security
- Data-at-Rest Protocols
 - Choosing the Storage Layer for Security
 - Symmetric Encryption Algorithm Selection
 - Managing the Storage Encryption Key

Testing for Security

- Basic Testing Methods
 - White-Box Testing
 - Black-Box Testing
 - Grey-Box Testing
- Fuzz-Testing