oSEC4 - Advanced wolfSSL for Embedded Security

Objectives

- Establish fundamental knowledge about cryptographic, algorithms, and protocols.
- Discover how encryption works and how to manage secret keys
- Understanding and Implementing Cryptography
- Learn how to implement secure authentication with wolfSSL
- Develop using wolfSSL's cryptography library (wolfCrypt)
- Understand how to build wolfMQTT on standard platforms and use it in an IoT application
- Building wolfSSH on standard Platforms
- Secure boot using wolfBoot (with wolfCrypt and WolfSSL)

Prerequisites

- C programming
- Experience with embedded systems development.
- oSEC3 Embedded Security with wolfSSL

Course Environment

- Theoretical course
  - PDF course material (in English) supplemented by a printed version for face-to-face courses.
  - Online courses are dispensed using the Teams video-conferencing system.
  - The trainer answers trainees' questions during the training and provide technical and pedagogical assistance.
- Practical activities
  - Practical activities represent from 40% to 50% of course duration.
  - Code examples, exercises and solutions
  - For remote trainings:
    - One Online Linux PC per trainee for the practical activities.
    - The trainer has access to trainees' Online PCs for technical and pedagogical assistance.
    - QEMU Emulated board or physical board connected to the online PC (depending on the course).
    - Some Labs may be completed between sessions and are checked by the trainer on the next session.
  - For face-to-face trainings:
    - One PC (Linux ou Windows) for the practical activities with, if appropriate, a target board.
    - One PC for two trainees when there are more than 6 trainees.
  - For onsite trainings:
    - An installation and test manual is provided to allow preinstallation of the needed software.
    - The trainer come with target boards if needed during the practical activities (and bring them back at the end of the course).
- Downloadable preconfigured virtual machine for post-course practical activities
- At the start of each session the trainer will interact with the trainees to ensure the course fits their expectations and correct if needed

Duration

- Total : 12 hours
- 2 sessions of 6 hours

Target Audience

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

Evaluation modalities

- The prerequisites indicated above are assessed before the training by the technical supervision of the traineein his company, or by the trainee himself in the exceptional case of an individual trainee.
- Trainee progress is assessed in two different ways, depending on the course:
  - For courses lending themselves to practical exercises, the results of the exercises are checked by the trainer while, if necessary, helping trainees to carry them out by providing additional details.
  - Quizzes are offered at the end of sections that do not include practical exercises to verifythat the trainees have assimilated the points presented
- At the end of the training, each trainee receives a certificate attesting that they have successfully completed the course.
  - In the event of a problem, discovered during the course, due to a lack of prerequisites by the trainee a different or additional training is offered to them, generally to reinforce their prerequisites,in agreement with their company manager if applicable.

Plan

First Session

Advanced WolfCrypt usage

- Public Key Cryptography
  - RSA
  - DH (Diffie-Hellman)
  - EDH (Ephemeral Diffie-Hellman)
  - DSA (Digital Signature Algorithm)
- PKCS Public Key Cryptography Standards
  - PKCS#1 RSA Cryptography Standard
  - PKCS#3 Diffie-Hellman Key agreement Standard
  - PKCS#5 Password Based Cryptography Standard
  - PKCS#6 Extended-Certificate Syntax Standard Historic
  - PKCS#7 and RFC 3369 : Cryptographic Message Syntax (CMS)
  - PKCS#8 Private Key Information Syntax Standard
  - PKCS#9 Selected Object Classes and Attribute Types
  - PKCS#10 Certification Request Syntax Standard
  - PKCS#11 Cryptographic Token Interface Standard
  - PKCS#12 Personal Information Exchange Syntax Standard
  - PKCS#15 Cryptographic Token Information Syntax Standard
- Cryptographic Certification
  - FIPS 140-2 Certification
  - NSA Certification
- Progressive Cryptography
- Hardware Accelerated Cryptography
- X.509 Certificates
- Key and Certificate genration
- WolfCrypt CertManager API

Exercise: *Sign and Verify data with RSA*
Exercise: *Sign and Verify data with ECC*
Exercise: *Sign and Verify data with Ed25519*
Exercise: *Key Agreement with ECDH and Curve25519*
Exercise: *PKCS#7 and CMS bundle Generation and Verification*
Exercise: *wolfCrypt “FIPS Readyâ€Â•*
Exercise: *Progressive Cryptography*
Exercise: *Creating Keys and Certificates*
Exercise: *Generate RSA/ECC Key and Certificates via API*
Exercise: *WolfCrypt Certificate Manager*

Second Session

wolfBoot

- • Introduction
- Features
- Components
- Wolfboot bootloader
- Integrating wolfBoot
- Upgrading the firmware

Exercise: *Upgrading the firmware using wolfBoot*

wolfSSH

- Overview
- Building wolfSSH
- wolfSSH User authentication Callback
- wolfSSH SFTP
- wolfSSH SCP

Exercise: *wolfSSH examples with passwords and Keys*
Exercise: *SFTP with wolfSSH*
Exercise: *SCP with wolfSSH*

wolfMQTT

- MQTT and MQTT-SN Overview
- MQTT client
- MQTT packet
- MQTT socket
- API reference

Exercise: *MQTT Broker*

Renseignements pratiques

Inquiry : 12 hours