oSEC3 - wolfSSL pour la sécurité embarquée

Objectifs

- Comprendre le fonctionnement de SSL/TLS
- Établir des connaissances fondamentales sur la cryptographie, les algorithmes et les protocoles
- Apprendre à mettre en Œuvre l'authentification sécurisée avec wolfSSL
- Apprendre à configurer et compiler wolfSSL pour les plateformes cibles (NXP, STMicro, Xilinx SoCs, Ti, ...)
- Apprendre des stratégies efficaces de débogage de wolfSSL
- Ajouter wolfSSL à des applications client et serveur basées sur ANSI-C
- Comprendre comment utiliser la bibliothèque de cryptographie de wolfSSL (wolfCrypt)
- Apprendre à utiliser un TPM pour authentifier les périphériques matériels (wolfTPM)

Pré-requis

- Programmation en C
- Expérience dans le développement des systèmes embarqués
- Quelques notions de sécurité sont souhaitables (voir nos formations OSEC1 et OSEC2)

Environnement du cours

- Cours théorique
  - Support de cours au format PDF (en anglais) et une version imprimée lors des sessions en présentiel
  - Cours dispensé via le système de visioconférence Teams (si à distance)
  - Le formateur répond aux questions des stagiaires en direct pendant la formation et fournit une assistance technique et pédagogique
- Activités pratiques
  - Les activités pratiques représentent de 40% à 50% de la durée du cours
  - Elles permettent de valider ou compléter les connaissances acquises pendant le cours théorique.
  - Exemples de code, exercices et solutions
  - Pour les formations à distance:
    - Un PC Linux en ligne par stagiaire pour les activités pratiques, avec tous les logiciels nécessaires préinstallés.
    - Le formateur a accès aux PC en ligne des stagiaires pour l'assistance technique et pédagogique
    - Certains travaux pratiques peuvent être réalisés entre les sessions et sont vérifiés par le formateur lors de la session suivante.
  - Pour les formations en présentiel::
    - Un PC (Linux ou Windows) pour les activités pratiques avec, si approprié, une carte cible embarquée.
    - Un PC par binôme de stagiaires s'il y a plus de 6 stagiaires.
  - Pour les formations sur site:
    - Un manuel d'installation est fourni pour permettre de préinstaller les logiciels nécessaires.
    - Le formateur vient avec les cartes cible nécessaires (et les remporte à la fin de la formation).
- Une machine virtuelle préconfigurée téléchargeable pour refaire les activités pratiques après le cours
- Au début de chaque session (demi-journée en présentiel) une période est réservée à une interaction avec les stagiaires pour s'assurer que le cours répond à leurs attentes et l'adapter si nécessaire

Audience visée

- Tout ingénieur ou technicien en systèmes embarqués possédant les prérequis ci-dessus

Durée

- Totale : 18 heures
- 3 sessions de 6 heures chacune (hors temps de pause)

Modalités d'évaluation

- Les prérequis indiqués ci-dessus sont évalués avant la formation par l'encadrement technique du stagiaire dans son entreprise, ou par le stagiaire lui-même dans le cas exceptionnel d'un stagiaire individuel.
- Les progrès des stagiaires sont évalués de deux façons différentes, suivant le cours:
  - Pour les cours se prêtant à des exercices pratiques, les résultats des exercices sont vérifiés par le formateur, qui aide si nécessaire les stagiaires à les réaliser en apportant des précisions supplémentaires.
  - Des quizz sont proposés en fin des sections ne comportant pas d'exercices pratiques pour vérifier que les stagiaires ont assimilé les points présentés
- En fin de formation, chaque stagiaire reçoit une attestation et un certificat attestant qu'il a suivi le cours avec succès.
  - En cas de problème dû à un manque de prérequis de la part du stagiaire, constaté lors de la formation, une formation différente ou complémentaire lui est proposée, en général pour conforter ses prérequis, en accord avec son responsable en entreprise le cas échéant.

Plan

Première session

Introduction to wolfSSL

- Introduction to embedded security
  - Embedded Security Trends
  - Security policies
- Secure Embedded System Hardware/Software Architecture Overview
- Securing legacy Systems
- Cryptography Overview
- wolfSSL Products and Library overview

wolfSSL embedded SSL/TLS library (1st part)

- Building wolfSSL
- Features
- Portability
- Callbacks
- Keys and Certificates
- Library Design
- SSL/TLS History and Protocol
- WolfSSL Basic Library usage

Exercise : *wolfSSL TLS integration*
Exercise : *SSL/TLS Tutorial*
Exercise : *wolfSSL Examples*

Deuxième session

wolfSSL embedded SSL/TLS library (2nd part)

- Debugging
- wolfSSL TLS usage
- wolfSSL DTLS Usage
- wolfSSL PSK Usage
- wolfSSL Session Resumption
- wolfSSL with Non-Blocking I/O
- wolfSSL and TLS 1.3

Exercise : *Wireshark*
Exercise : *Convert TCP/IP Client and Server to TLS*
Exercise : *Extracting Certificate Fields via API*
Exercise : *Convert simple UDP Client and Server to DTLS*
Exercise : *Convert simple TCP client and Server to PSK*
Exercise : *Session Resumption Client*
Exercise : *Write a Non-Blocking Client and Server*
Exercise : *TLS 1.3 Client and Server*
Exercise : *TLS 1.3 Early Data*

wolfCrypt (1st part)

- PRNG(Pseudo-Random Number Generator) and RNG(Random Number Generation)
- HASH Functions
- Block Ciphers
  - AES
  - DES and 3DES
  - Camellia
- Stream Ciphers
  - ARC4
  - RABBIT
  - HC-128
  - ChaCha

Exercise : *PRNG and RNG*
Exercise : *Creating a Hash of a File*
Exercise : *Block Ciphers*
Exercise : *Block Ciphers*

Troisième session

wolfCrypt (2nd part)

- Public Key Cryptography
  - RSA
- PKCS Public Key Cryptography Standards
  - PKCS#7 and RFC 3369 : Cryptographic Message Syntax (CMS)
- Cryptographic Certification
  - X.509 Certificates
- Key and Certificate genration

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 Certificate Manager*
Exercise : *Creating Keys and Certificates*

WolfTPM

- Overview of TPM Architecture
- The Root-of-Trust
- Key Hierarchy and Key Management
- TPM command Message Overview
- Command Authorization (Typical/Atypical)
- TPM Signature Command
- TPM Capability and Self-Test
- Building WolfTPM
- wolfTPM Library Design

Exercise : *Building and Testing wolfTPM*

Renseignements pratiques

Renseignements : 18 heures