Projet : Mise en place d'une Infrastructure à Clés Publiques (PKI) à Trois Niveaux

Professeur : LECHHAB OUADRASSI Nihad
Master MMSD
Projet de demi-module : Cryptographie et Blockchain
Sous supervision de Mr AZMANI Abdellah

16 mai 2025

Introduction

Une **Infrastructure à Clés Publiques (PKI)** est un système qui permet de sécuriser les communications numériques en garantissant l'authenticité, la confidentialité et l'intégrité grâce à l'usage de **certificats numériques**.

Dans ce projet, nous allons implémenter une PKI à trois niveaux, comprenant :

- **Root CA** : autorité racine, auto-signée, qui émet et signe des certificats.
- **Intermediate CA** : autorité intermédiaire, qui signe les certificats finaux et protège la Root CA.
- **Leaf Certificates** : utilisés par des serveurs ou clients pour s'identifier.

Objectifs du projet

- Comprendre et mettre en œuvre la hiérarchie et la chaîne de confiance dans une PKI.
- Générer, signer et distribuer des certificats numériques dans une architecture à trois niveaux.
- Gérer la sécurité des clés privées et appliquer les bonnes pratiques liées aux certificats.

— Implémenter la gestion des demandes de signature (CSR) et la révocation des certificats (CRL).
— Proposer une interface d'utilisation simplifiée (GUI) pour gérer les opérations clés de la PKI.
— Vérifier la validité et la chaîne de confiance des certificats générés.

Description Fonctionnelle

Architecture

— **Root CA** : Autorité racine, auto-signée, sécurisée, rarement utilisée directement.
— **Intermediate CA** : Autorité intermédiaire, signe les certificats finaux, protège la Root CA.
— **Leaf Certificates** : Certificats finaux pour serveurs, clients ou applications.

Fonctionnalités attendues

— Création des clés RSA/ECDSA pour chaque entité.
— Génération de certificats auto-signés et signés par la CA parent.
— Gestion des CSR (Certificate Signing Requests).
— Stockage organisé des certificats, clés, et bases de données (index.txt, serial).
— Génération et publication de listes de révocation (CRL).
— Vérification de la validité et de la chaîne de confiance des certificats.
— Développer une interface web Flask pour la gestion simplifiée.

Contraintes Techniques et Recommandations

— Utilisation d'OpenSSL en ligne de commande ou via bibliothèques Python (cryptography, PyOpenSSL).
— Respect des normes X.509.
— Sécurisation des clés privées (permissions strictes, stockage isolé).
— Gestion des numéros de série et des bases de données d'émission.
— Documentation complète de chaque étape.

Livrables attendus

- Les commandes utilisées dans la gestion des certificats.
- Fichiers de configuration OpenSSL adaptés (openssl.cnf).
- Documentation technique détaillée incluant :
  - Architecture et schéma de la PKI.
  - Justification des choix (algorithmes, tailles de clés, durées).
  - Procédures d'utilisation.
  - Captures d'écran ou exemples de certificats.
  - Captures d'écran d'interface web avec une description accompagner.
- Présentation orale et démonstration.
- Code source (GitHub).

Environnement et Outils

- Système Linux (Ubuntu recommandé).
- OpenSSL version récente.
- Python 3.x avec bibliothèques cryptography et/ou PyOpenSSL (optionnel).
- IDE (ex : VS Code).
- Terminal et accès aux droits sudo pour installer les outils nécessaires.

Instructions et Développement

Chaque étudiant est encouragé à ajouter d'autres étapes ou à proposer des améliorations pour augmenter la qualité de son projet.

Cela inclut des améliorations comme :

- Ajouter des options supplémentaires pour la sécurisation des clés privées.
- Implémenter des fonctionnalités avancées pour la gestion des certificats.
- Proposer une interface graphique (GUI) plus complexe pour faciliter l'interaction avec la PKI.
- Ajouter des tests de validité et de performance des certificats générés.

Date limite : 4 juin 2025. Aucun travail ne sera accepté après cette date.