SAÉ S3.B.01

Création et déploiement de services applicatifs

Objectifs et problématique professionnelle :

La problématique professionnelle est de créer, en équipe, une application en suivant une démarche de développement itérative ou incrémentale.

En partant d'un besoin décrit de manière imprécise ou incomplète par un client, l'objectif est de clarifier, compléter, collecter et formaliser le besoin puis de développer une application client-serveur intégrant la manipulation des données et respectant des paradigmes de qualité (en particulier la sécurité).

Quels sont les livrables attendus ?

- Documents de suivi de projets
- Code de l'application et jeux d'essais
- Schémas d'architecture
- Revue finale du projet
- Guide d'utilisation

Le travail se fait **par groupe de 4 ou 5 étudiants** (pour idéalement avoir 5 groupes).

Objectif

Au cours de cette SAÉ, vous allez concevoir et développer en groupe **une application sécurisée d'enchères à plis fermés**.

Compétence 1

Partir des exigences et aller jusqu'à une application complète

AC 1 Élaborer et implémenter les spécifications
AC 2 Appliquer accessibilité et ergonomie
AC 3 Adopter de bonnes pratiques

Compétence 2

Sélectionner les algorithmes adéquats pour répondre à un problème donné

AC 1 Choisir des structures de données

Compétence 3

Déployer des services dans une architecture réseau

AC 1 Développer des applications communicantes
AC 2 Utiliser la virtualisation

Compétence 4

Optimiser une base de données, interagir avec une application et mettre en œuvre la sécurité

AC 3 Organiser la restitution de données
AC 4 Manipuler des données hétérogènes

Compétence 5

Appliquer une démarche de suivi de projet en fonction des besoins métiers des clients et des utilisateurs

AC 2 Formaliser les besoins
AC 3 Identifier la faisabilité d'un projet
AC 4 Mettre en œuvre un suivi de projet

Compétence 6

Situer son rôle et ses missions au sein d'une équipe informatique

AC 2 Intégrer une équipe informatique
AC 3 Mobiliser les compétences interpersonnelles
AC 4 Rendre compte de son activité professionnelle

Les besoins :

Un client souhaite proposer une application client-serveur sécurisée d'enchères électroniques à plis fermés. Il a opté pour un type particulier d'enchère : les enchères au second prix, ou *enchères de Vickrey*. Celles-ci fonctionnent de la façon suivante : le vainqueur de l'enchère est celui qui a enchéri le prix le plus élevé, mais il va seulement payer le *deuxième prix le plus élevé* pour obtenir l'objet de l'enchère.

Le client décide de faire appel à vous pour développer cette application.

Votre travail :

Pour modéliser ce dispositif, nous pouvons distinguer 3 entités différentes : les enchérisseurs (B_i pour $i = 1, ..., N$, où N est le nombre total d'enchérisseurs), le vendeur ($\mathcal{S}$) et l'autorité de gestion de l'enchère ($\mathcal{A}$).

Chaque entité devra posséder une paire de clés pour un algorithme de signature et une paire de clés pour un algorithme de chiffrement.

Vous allez implémenter un premier protocole qui suppose une confiance en l'autorité de gestion de l'enchère. Ce protocole se déroule de la façon suivante :

L'autorité lance une enchère : un client se connecte et obtient la connaissance de l'objet de l'enchère. Suivent deux phases principales :

— *Phase d'enchère* :

1. L'enchérisseur B_i chiffre son prix b_i à l'attention de $\mathcal{A}$. Il envoie ce chiffré (noté c_i), ainsi que sa signature σ_i de ce chiffré à $\mathcal{S}$ via un canal sécurisé.

2. $\mathcal{S}$ recueille et vérifie les signatures sur les chiffrés (et indique le cas échéant les signatures invalides), puis calcule sa signature z de l'ensemble des chiffrés $\{c_i\}_{i=1,N}$ dans un ordre qui ne dépend pas de i (par exemple, l'ordre lexicographique sur les c_i). Il broadcast la liste $(\{c_i, \Sigma_i\}, z)$ à tous les enchérisseurs, où Σ_i est la signature de $\mathcal{S}$ sur c_i.
3. Chaque enchérisseur obtient la liste précédemment mentionnée, vérifie la signature z et, le cas échéant, fait remarquer l'absence de son chiffré.

— *Phase d'ouverture des enchères*

1. $\mathcal{A}$ vérifie *toutes* les signatures (les Σ_i et z). Il déchiffre chaque c_i pour obtenir l'enchère b_i correspondante. Il calcule le second prix X_2 et envoie à $\mathcal{S}$ cette valeur ainsi que sa signature sur $(X_2, \{c_i\})$.
2. $\mathcal{S}$ vérifie la signature, et broadcast (X_2, z, z') à tous les B_i.
3. Chaque B_i vérifie que la signature z' est correcte (en particulier, calculée sur le même ensemble $\{c_i\}$ que z).
4. $\mathcal{A}$ indique à $\mathcal{S}$ un i_ω tel que c_{i_ω} se déchiffre sur X_1.
5. $\mathcal{S}$ identifie i_ω et le déclare vainqueur.

— Aspects système et réseau :

Vous allez développer une application en Java. Celle-ci va exploiter les *sockets* qui vont permettre aux différentes entités de communiquer.

— Aspects cryptographiques :

Dans un premier temps, vous allez utiliser une librairie java de votre choix qui implémente les différentes primitives cryptographiques nécessaires au protocole décrit ci-dessus.

Améliorations possibles

Cette architecture est une proposition simple d'enchère de Vickrey. Elle souffre de plusieurs problèmes de sécurité. Votre travail consistera également à soulever et démontrer des problèmes de sécurité (en implantant éventuellement des attaques), de proposer des solutions pour les éviter. Vous devrez rendre robuste votre programme, en mettant par exemple en place des tests unitaires, vous pourrez ajouter une interface graphique, ou intégrer une *Public Key Infrastructure*,...