Université de Rouen

Département Informatique - Master Sécurité des Systèmes d'Informations

Analyse des risques
Projet de gestion des licences

Auteurs :
Sami Babigeon
Louka Boivin
Noé Dallet
Kaci Hammoudi
Alexis Osmont

Client :
M. Ziadi Djelloul

3 décembre 2021

Revision

Version	Date	Commentaires
0.1	01/12/2021	Création du document
0.2	02/12/2021	Modifcation de la structure du document
0.3	02/12/2021	Mise en forme du document
1.0	03/12/2021	

Table des matières

1 Terminologies

— Le client est le commanditaire du projet.
— Un utilisateur est une personne souhaitant utiliser un logiciel du client.
— Une licence est un droit accordé pour une machine et un utilisateur d'utiliser un logiciel donné.

2 | Liste des points durs

— **Minimisation des risques de failles d'implémentation**
Etant donné notre non experience dans les domaines des gestionnaires de licences et de la greffe de code, la minimisation des failles d'implémentation est un risque élevé.

— **Programmation de l'algorithme de signature ElGamal**
L'un des points les plus important est la signature, la principale difficultés ici est de créer une signature unique.

— **Obfuscation du code**
L'obsfuscation de code est un element essentiel car le but de notre projet est de proteger des logiciel du cracking. Le risque majeur ici est de ne pas suffisement obfusquer le code et donc de le rendre facilement accessible.

— **Utilisation du C pour une DLL ou développement**
L'utilisation du langage C est l'un de nos choix en remplacement du C++ et C# même si celui-ci possède moins d'utilitaires que les autres, et est par consquent plus assujetti aux failles d'implémentation.

— **Injection de Code dans un PE**
L'injection de code peut, si elle est mal implémentée laisser des faille facilement utilisable pour des personnes mals intentionnées.

— **Le temps**
Nous avons pris du retard, dû à des délais de remise de travail demandé ainsi qu'aux difficultés rencontrée quant à l'organisation de réunion entre membre du projet.

3 | Evalutaion du contexte

3.1 Particularité du sujet

Nous possedons des logiciels de référence sur lesquels se baser.

Plusieurs parties à réaliser, sujet pluridisciplinaire :
— Programmation Serveur.
— Interface Web.
— Logiciel client Windows

3.2 Définition du besoin

Le produit final sera utilisé par des personnes non formée à l'informatique.
— Les interfaces decront être pensés pour

Le produit final sera réellement utlisé par le client
— Attente de la part du client

3.3 Disponibilité des acteurs et ressources

Les acteurs
— Difficulté à organiser des réunions entre les membres du groupe.
— Réunion avec Professeur de gestion de projet 1x toutes les 2 semaines
— Réunion avec client 1 fois toutes les 2 semaines.

Les ressources
— Beacoup de documentation sur internet
— Logiciel référence intellilock, etc..
— Référent technique seulement 1x fois toutes les 2 semaines...
— Professeurs de crypto disponnibles.

3.4 Composition de l'equipe

Nous nous connaissions déjà et communication facile.

3.5 Connaissances techniques personnelles

— Obfuscation : 0
— Injection de code : 0
— Programmation Serveur : Sami
— Interface Web : Experience de la licence
— Développement logiciel : Experience de la licence
— Implémentation crypto : 0

3.6 La complexité des solutions techniques

Les solutions techniques n'étant pas encore vraiment décidées.
— ElGamal.

3.7 Perturbations engendrées par les autres activités

— Examens
— Autres Projets
— Travaux personnels (TP)

4 | Ordonner les points durs

— 1 - Injection de Code dans un PE
— 2 - Minimisation des risques de failles d'implémentation
— 3 - Obfuscation du code
— 4 - Gestion du temps
— 5 - Algorithme de signature ElGamal
— 6 - Utilisation du C

5 | Definitions des risques associés

6 | Top cinq des risques (Bilan)

Risque	**Craintes**	**Effets**	**Impacts**	**freq**	**Stratégie**
Failles d'implémentation	Cracking du logiciel et de son injection	Perte de controle du logicel	Critique	Trés forte	Effectuer des testes unitaire sur les logiciels et les exécutables.
Injection de Code	Cracking du logiciel et de son injection	Perte de controle du logicel	Critique	Trés forte	Effectuer des testes unitaire sur les logiciels et les exécutables.
Obfuscation du code	Cracking du logiciel et de son injection	Perte de controle du logicel d'injection de code	Important	Trés forte	Effectuer des testes unitaire sur les logiciels et les exécutables.
Gestion du temps	Manque de temps pour un rendu complet	Prise de retard par rapport aux TP de Gestion de Projet	Important	forte	S'organiser et séparer efficacement les tâches pour bien gérer le travail.
Signature ElGamal	Complications lors de l'accès au logicel	Impossible d'utliser le logicel	Fort	Faible	Effectuer des testes unitaire sur les signatures pour prevenir ce risque.