

TD Cours 2 - Vie privé et Cryptographie

Exercice 1 (Vérification d'âge préservant la vie privée selon la CNIL)

Les systèmes préservant la vie privée reposent fortement sur la cryptographie. Il existe une implémentation pratique d'un système de vérification d'âge préservant la vie privée basé sur des primitives cryptographiques telles que les *signatures de groupe* et les *preuves à divulgation nulle de connaissance*. L'objectif est de comprendre comment un utilisateur peut prouver qu'il satisfait une condition d'âge sans révéler son identité ou son âge exact. Le démonstrateur développé par le CNIL LINC repose sur un tiers de confiance, les signatures de groupe et les preuves à divulgation nulle de connaissance pour atteindre cet objectif. <https://linc.cnil.fr/demonstration-privacy-preserving-age-verification-process> fournit le contexte et l'explication du protocole.

Clonez le dépôt du démonstrateur : `git clone https://github.com/LINCnil/SigGroup.git`

Suivez les instructions d'installation fournies dans le dépôt et exécutez le démonstrateur. Alternative-ment, la CNIL fournit une exécution basée sur Docker (les privilèges root sont requis) :

```
docker run -p 9091:9091 -p 9092:9092 -p 9093:9093 cnil/siggroup
```

Ouvrez les trois interfaces web :

- Un service nécessitant une vérification d'âge : <https://localhost:9093>
- Fournisseur de vérification d'âge certifié : <https://localhost:9091>
- Autorité de certification : <https://localhost:9092>

Utilisez le démonstrateur pour réaliser le scénario suivant :

1. Depuis le site du service, créez une demande de vérification d'âge exigeant que l'utilisateur ait au moins 18 ans.
2. Téléchargez le challenge généré.
3. Soumettez le challenge au fournisseur de vérification d'âge certifié.
4. Testez la vérification avec :
 - Bob (16 ans),
 - Alice (25 ans).
5. Observez quel utilisateur obtient une signature valide et renvoyez le challenge signé au site demandeur.
6. Vérifiez que le site n'apprend que si l'utilisateur satisfait la condition d'âge, et non l'identité ou l'âge exact de l'utilisateur.

En vous basant sur vos observations, répondez aux questions suivantes :

1. Quelles sont les trois entités impliquées dans le protocole ?
2. Pourquoi le site demandeur n'a-t-il pas besoin de connaître l'identité de l'utilisateur ?
3. Quelles informations sont cachées au fournisseur de vérification d'âge ?
4. Quel rôle joue l'autorité de certification ?
5. Expliquez pourquoi les signatures de groupe sont utiles dans ce contexte.
6. Quelle propriété cryptographique permet à un utilisateur de prouver une affirmation sans révéler d'informations supplémentaires ?

Comparez cette approche avec un système traditionnel de vérification d'âge où un utilisateur télécharge un document d'identité sur chaque site. Discutez des avantages et des limites de l'approche préservant la vie privée.

Exercice 2 (Faux ou faux.)

Expliquez pourquoi chacune des affirmations suivantes est erronée.

1. Il est sûr de stocker les mots de passe des utilisateurs en clair si le serveur utilise toujours HTTPS (*i.e.*, un canal privé authentifié).
2. Un chiffrement par bloc avec une clé plus grande est toujours plus sûr.
3. Il n'y aura jamais de raison technologique d'utiliser des clés de chiffrement par bloc de plus de 128 bits.
4. Il faut toujours utiliser des clés de chiffrement par bloc de plus de 128 bits.
5. Il faut toujours utiliser le chiffrement par bloc le plus récemment publié.

Exercice 3 (Le chiffrement double augmente-t-il la sécurité ?)

Considérons un schéma de chiffrement symétrique $E_k^2(\cdot)$ avec une clé $k = k_1 || k_2$. Le chiffrement double d'un message m est défini par :

$$c = E_{k_2}(E_{k_1}(m))$$

où k_1 et k_2 sont deux clés indépendantes.

Le chiffrement d'un message deux fois avec deux clés différentes rend-il nécessairement le chiffrement plus sûr qu'un chiffrement unique avec E ? Justifiez votre réponse.

Exercice 4 (MAC et chiffrement)

Vous concevez un protocole de messagerie sécurisée. Un expéditeur souhaite transmettre un message en clair M à un destinataire en utilisant une clé symétrique. La confidentialité, l'intégrité et l'authenticité doivent être garanties de la meilleure manière possible. Pour cela, vous pouvez utiliser une clé partagée K pour un algorithme de chiffrement symétrique E et un code d'authentification de message (MAC). Considérez les trois constructions possibles :

Chiffrer puis MAC : $C = E_k(M)$; $T = MAC_{k'}(C)$; Envoyer (C, T)

MAC puis chiffrer : $T = MAC_{k'}(M)$; $C = E_k(M, T)$; Envoyer C

Chiffrer et MAC : $C = E_k(M)$; $T = MAC_{k'}(M)$; Envoyer (C, T)

1. Comparez les garanties de sécurité de Chiffrer puis MAC, MAC puis chiffrer et Chiffrer et MAC.
2. Pourquoi Chiffrer puis MAC offre-t-il une meilleure protection contre la modification du texte chiffré que les autres approches ?
3. Expliquez pourquoi MAC puis chiffrer peut être vulnérable lorsque le schéma de chiffrement est malléable.
4. Pourquoi Chiffrer et MAC peut-il potentiellement fuiter des informations sur les messages en clair ?
5. Pourquoi Chiffrer puis MAC est-il considéré comme la construction la plus sûre pour le chiffrement authentifié ?

Exercice 5 (Équilibrer sécurité et coût)

Une entreprise constate que, chaque année, elle subit 5 attaques de virus et 3 défigurations de son site web. Chaque remise en état nécessite 2 jours de travail de l'administrateur, pour un coût de 2 000 euros. Le site web peut être réparé en quelques heures, pour un coût de 500 euros. La maintenance d'un antivirus et d'un système de protection coûterait 30 000 euros par an.

1. Calculer le coût annuel des deux menaces et le comparer au coût de maintenance d'un antivirus et d'un système de protection.
2. Critiquer le calcul du coût des menaces et proposer une meilleure méthode.