TD Sécurité Informatique

Série TD 3.2 : Cryptographie Asymétrique (RSA)

À votre avis, quelles sont les **limites** de la cryptographie symétrique ?

Cryptographie asymétrique : L'algorithme RSA

L'algorithme RSA est un algorithme de cryptographie asymétrique largement utilisé. RSA a été publié pour la première fois en 1977 par Ron Rivest, Adi Shamir et Leonard Adleman, d'où son nom RSA (Rivest-Shamir-Adleman).

1. Chaque utilisateur possède deux clés : une clé publique et une clé privée.
2. La clé privée doit être stockée de manière sécurisée et ne doit jamais être divulguée à quiconque.
3. Il est impossible de déduire une clé à partir de l'autre, car elles sont générées à l'aide de formules mathématiques irréversibles.
4. Si vous chiffrez un message avec l'une des clés, vous pouvez le déchiffrer avec l'autre clé correspondante.

RSA : Mode de Fonctionnement

Phase 1 : Génération des clés (publique et privée)

1. **Choisir deux nombres premiers distincts :** Choisissez deux nombres premiers distincts ***p*** et ***q***. Ces nombres doivent être suffisamment grands pour assurer la sécurité du système. Plus ces nombres sont grands, plus la sécurité est renforcée. *(P=3, Q=11)*

2. **Calculer *n* :** Calculez le produit des deux nombres premiers : $n = p \times q$.

3. **Calculer la fonction d'Euler $\varphi(n)$:** Calculer la fonction d'Euler de ***n***, qui est le nombre d'entiers positifs inférieurs à ***n*** qui sont premiers avec ***n***. Pour deux nombres premiers distincts ***p*** et ***q***, $\varphi(n) = (p-1)\times(q-1)$.

4. **Choisir *l'exposant de chiffrement public* (e):** Sélectionnez un entier ***e*** tel que $1<e<\varphi(n)$ et ***e*** est premier avec $\varphi(n)$. ***e*** est généralement choisi petit.

5. **Calculer l'exposant de déchiffrement privé (*d*) :** en utilisons l'algorithme d'Euclide étendu, calculez l'exposant de déchiffrement ***d*** : ~~$d \times e \equiv 1 \bmod \varphi(n)$~~ $e \times d + \varphi(n) \times (-1) = 1$.

6. **Clés générées :** Les clés publiques sont **(*n*, *e*)** et la clé privée est ***d***.

Une fois que vous avez généré les clés, vous pouvez utiliser la clé publique pour chiffrer les messages et la clé privée pour les déchiffrer.

Phase 2 : Chiffrement du message

Si **M** est un **entier** naturel strictement inférieur à n représentant un message, alors le message chiffré sera représenté par : $\boxed{C = M^{e} \bmod (n)}$ *(prenons Exemple : M=4 -> C=31)*

Phase 3 : Déchiffrement du message

$\boxed{M = C^{d} \bmod (n)}$

Exercise 01:

Alice souhaite envoyer le message "Hello" à Bob de manière sécurisée en utilisant RSA. Les paramètres de chiffrement sont les suivants : p = 17, q = 11, e=7

- Calculez N, l'entier de module RSA.
- Calculez $\phi(N)$, la fonction d'Euler de N.
- Vérifiez que e est premier avec $\phi(N)$.
- Trouvez l'exposant de déchiffrement d.

1. Bob choisit le mot "Hello" à envoyer à Alice. Utilisez le code ASCII pour transformer les lettres en chiffres entiers, puis utilisez la clé publique (N, e) pour chiffrer le message.
2. Alice utilise sa clé privée pour déchiffrer le message.