Sécurité informatique

CHAN CHUN TIM Alan 36001954, COUTIN Matthias 36000206

17 avril 2020

1 Introduction

Développement de l'application sous android de différentes méthodes de cryptage ;
— César
— Vigenère
— Hill
— Transposition Rectangulaire
— DES
— RSA

2 Classe ASUtil

Nous avons créé la classe ASUtil qui permet de gérer les caractères ASCII.
La fonction getText() prend en parametre un entier est retourne le caractere.
La fonction getCode() prend une lettre et retourn l'entier modulo 256.

3 Cryptographie César

Corresponds à un décalage de 3 dans la table ASCII étendue.
— Crypter prend en paramètre une phrase et décale chaque caractères de +3 dans la table ASCII étendu. Les caractères non affichables sont retranscrit en héxadécimal.

```
int entier = (ASUtil.getCode(mot.charAt(i)))+3;
    if(entier>=0&&entier<=31||entier==127||entier==255){
        resultat+= String.format("\\x%2x",entier);
```

— Crypter prend en paramètre une phrase et décale chaque caractères de -3 dans la table ASCII étendu. Si \est suivit de "x" alors c'est un hexadecimal on le converti en entier -3 puis en caractère.

```
int entier = (ASUtil.getCode(mot_decrypt.charAt(i)));
    if(entier==92 && ASUtil.getCode(mot_decrypt.charAt(i+1))==120){
        char letter1 =  ASUtil.getAscii(ASUtil.getCode(mot_decrypt.charAt(i+2)));
        char letter2 = ASUtil.getAscii(ASUtil.getCode(mot_decrypt.charAt(i+3)));
        String hexString =""+letter1+letter2;
        resulta1+= ASUtil.getAscii((Integer.parseInt(hexString,16)-3)%256);
        i+=3;
```

4 Cryptographie Vigenère

— Crypter : prend en paramètre le texte et la clé . On convertit en entier et additionne chaque caractère de même indice. Retourne le texte Crypté et affiche en hexadécimal les caractères non affichables.
— Décrypter : prend en paramètre le texte(qui peut contenir \x) et la clé. On convertit en entier et soustrait chaque caractère de même indice. Retourne le texte décrypté et affiche en hexadécimal les caractères non affichables.

5 Transposition Rectangulaire :

La cryptographie par transposition rectangulaire se déroule comme suit :

1. Générer une matrice à partir de la clé et de la phrase à crypter.
2. Changer l'ordre des colonne de la matrice par rapport à l'ordre de chaque lettre de la clef dans l'alphabet
3. Lire la matrice pour constitué la phrase crypter

Nous avons implémenter cette technique en utilisant pour alphabet la table ASCII étendue.

Dans la class RectTrans nous avons construit cpkey qui correspond aux indices de la clef dans l'ordre croissant. Puis dans la méthode crypt nous construisons la matrice de la phrase sous forme de tableau en deux dimension dont la taille et de la longueur de la clef par la longueur de la phrase divisé par la longueur la clé et si un des tableaux n'est pas replis entièrement nous complétons par le code ASCII de la lettre 'z'.

Pour finir on utilise les indice dans cpkey pour parcourir la matrice dans l'ordre de la clef. Pour le décryptage nous utilisons la même méthode.

6 Chiffre de Hill :

La cryptographie en utilisant le chiffre de Hill se passe comme suit :

1. Regroupement des lettres de la phrase à crypté par paire, si la longueur est impaire on complète par un caractère rare.
2. On multiplie les groupes de lettres par la matrice qui forme la clef modulo la taille de l'alphabet utiliser.
3. Les résultats de ces multiplications formes la phrase finale cryptée.

Pour le décryptage nous faisons comme suit :

1. Regroupement des lettres de la phrase à décrypté par paire.
2. On multiplie les groupes de lettres par la matrice inverse de la clef.
3. Les résultats de ces multiplications formes la phrase finale décryptée.

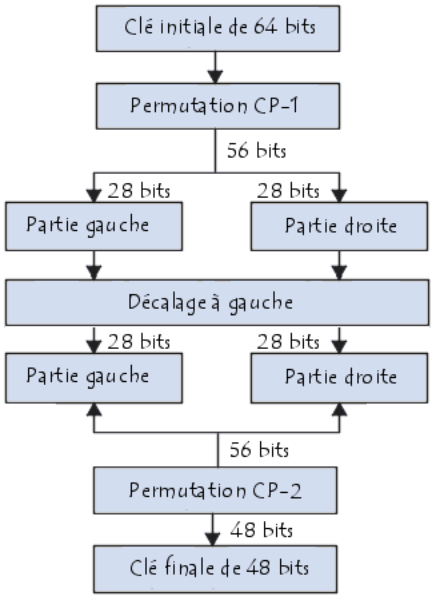
Ici la taille de l'alphabet est de 256 puisque nous utilisons la table ASCII étendue. La difficulté de cette technique de cryptographie réside dans la création de la clef. Dans notre code dans la class HillCrypt nous calculons d'abord la matrice clé en utilisant un mot de 4 lettres, puis nous calculons sons déterminant qui nous permettra de calculer l'inverse de la matrice clef, mais si la matrice clé n'est pas inversible nous retournons un code -1 qu'on considére comme un code erreur. Pour le calcul de la matrice inverse nous cherchons un coefficients qui est un multiple du déterminant de la matrice clef et dont sont modulo avec la taille de l'alphabet (256) est 1.

7 Crytpographie DES

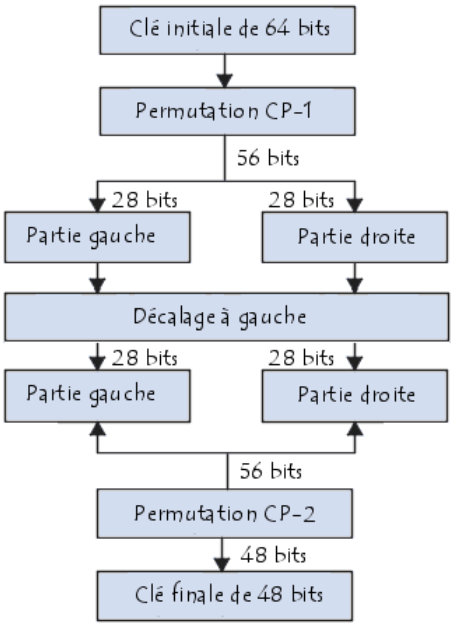
— Crypter : Prend en paramètre un texte(qui peut être en hexadécimal), la clé aussi. Retourne en hexadécimal le résultat.
— Décrypter : prend en paramètre un texte crypté (qui peut être en hexadécimal), la clé aussi. Retourne en hexadécimal le résultat.

Pour plus de précision les iterations sont dans la fenêtre Logcat verbose "coucou" d'android studio.

7.1 Génération des clés



7.2 Algoritme DES



8 chiffrement RSA

La cryptographie en utilisant le chiffrement RSA se passe comme suit :

1. Calculer une clé publique à partir de nombres premiers
2. La clé publique est un couple (e,n), pour crypter le message nous prenons le message M puis on effectue $M^e\%n$
3. Le résultat du calcule nous donne le message crypté

Pour le décryptage nous faisons comme suit :

1. Calculer une clé privée à partir de nombres premiers
2. La clé publique est un couple (d,n), pour crypter le message nous prenons le message M puis on effectue $M^d\%n$
3. Le résultat du calcule nous donne le message crypté.

La difficulté de cette technique de chiffrement réside dans la création des clés publique et privé que nous obtennons en choisissant 2 nombres premiers p e tq puis on calcule leur produit $n = p \times q$ et $\varphi(n) = (p-1)(q-1)$ puis pour la clé publique on prend le couple (e,n) tel que e est premier à n et strictement inférieur à $\varphi(n)$, pour la clé privée nous prenons le couple (d,n) tel que d inverse de *emodn* et strictement inférieur à n. Une particularité de cette technique et aussi la longueur du message à crypter et déterminer par n, pourquoi dans nous avons découper le message par blocs.