DM: Implantation de l'attaque par le milieu contre un chiffrement par bloc

Thanushan PIRABAKARAN & Maya SANTINI

1. Implémentation

Pour l'algorithme de cadencement des clés, nous avons implémenté une fonction qui prend en paramètre un int qui correspond à la clé maître. Pour appliquer l'algorithme donné, on utilise des opérateurs bits à bits et des masques.
Idem pour les fonctions de permutation et de substitution.
Pour l'inverse de ces deux dernières, nous appliquons également des opérateurs et des masques, mais avec le tableau donné inversé.
La fonction de chiffrement prend en entrée le message, un string et la liste des sous clés obtenues pour K; on applique l'algorithme donné en utilisant les fonctions précédemment créées.
La fonction de déchiffrement suit la même logique, on applique l'algo dans le sens inverse.

2. L'attaque par le milieu

Nous avons établi deux listes en parallélisant les opérations (libraire **multiprocessing**):

- Lm, qui contient des tuples (cm1,k1) à la suite du chiffrement de m1 avec toutes les clés k1 possibles (2^24)
- Lc, qui contient des tuples (mc1,k2) à la suite du déchiffrement de c1 avec toutes les clés k2 possibles (2^24)

Ensuite, nous avons trié les listes avec un algorithme de **tri rapide** dans l'ordre croissant.
Puis, on recherche les éléments en commun en parcourant les listes Lm et Lc avec un while :

- Lorsque premier élément du tuple de Lm est égal premier élément du tuple de Lc, on ajoute le deuxième élément de chaque tuple dans un tuple (k1,k2) que l'on ajoute à une troisième liste L3 qui contiendra toutes les clés possibles trouvées.
- Lorsque le premier élément du tuple de Lm est inférieur à celui de Lc, on avance l'indice qui parcourt Lm de 1.
- Lorsque le premier élément du tuple de Lm est supérieur à celui de Lc, on avance l'indice qui parcourt Lc de 1.
- On arrête lorsque les 2 listes ont été parcourues et on renvoie L3.

Nous avons trouvé 16775001 éléments en commun aux deux listes.
On teste par la suite les tuples de clés de L3 avec le second couple clair-chiffré (m2, c2) : on chiffre m2 avec k1, puis on re-chiffre le résultat avec k2, et on compare le résultat final avec c2. Si les résultats concordent, on ajoute les clés (k1,k2) à la liste finale cle_possible, que l'on print.
Avec un ordinateur à 4 cœurs, l'attaque prend 40 minutes. Avec 6 cœurs, 15 minutes.

3. Clés secrètes

Nous avons trouvé 5 clés secrètes pour les couples clair-chiffré de Maya SANTINI : (m1,c1) = (5378ee, d0dd2b) (m2,c2) = (f746a9, c82611) :

- k1 = b99b73, k2 = 2e58a4
- k1 = f68f3a, k2 = bb13f7
- k1 = e573ca, k2 = f37b12
- k1 = a6f1d9, k2 = b20caa
- k1 = 260afa, k2 = defeab