DM CRYPTO : A rendre le 08/05/21 à 23h.

À faire : - implémenter en C un chiffrement par bloc +
// du chiffrement en double +
// de l'attaque du milieu => renvoyer un couple de clé qui fonctionne.

format : - Structure projet normal C (src, headers, makefile, Readme)
+ fichier pour expliquer nos choix d'implémentation (préciser quelles données clair/chiffrés ont été utilisé.)
+ Tout ça sous forme d'archive : LYONNET_BOUMALI.zip

Spécifications :
- Chiffrement par bloc : PRESENT 24
- famille SPN
- exemple :

$m = \underbrace{0011}_{m_0}\ \underbrace{1010}_{m_1}\ \underbrace{1100}_{m_2}\ \underbrace{1001}_{m_3}\ \underbrace{0001}_{m_4}\ \underbrace{0000}_{m_5})_2$

Chiffrement

⇓ division en un bloc → fct (1) => diviser en différents messages

⇓ $m \oplus clé_1 = m'$ → fct (2) → XOR

⇓ pour tous les blocs disponibles

m'_0 m'_1 m'_2 m'_3 m'_4 m'_5 → fct (5)

tableau sur le sujet → fct (3) => table S

(P) { ⊕ clé } 10 fois

fct (4) => permutation

(P)

11ème itération

⊕ clé 11

Scanné avec CamScanner

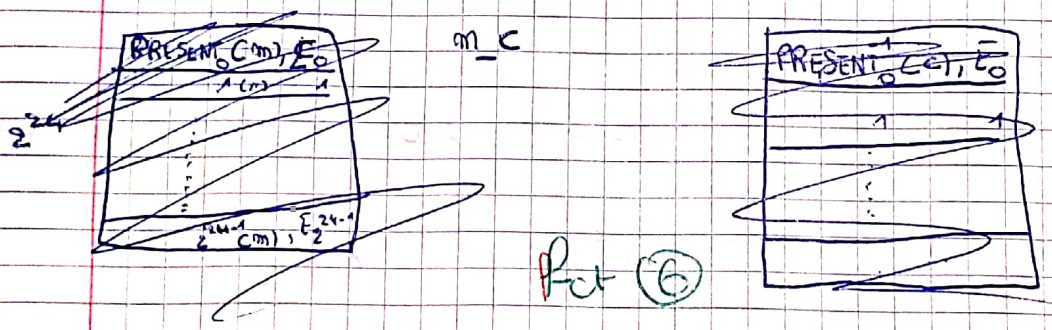
- Répéter ce chiffrement une deuxième fois avec une clé différente appliqué sur le chiffré obtenu précédemment.

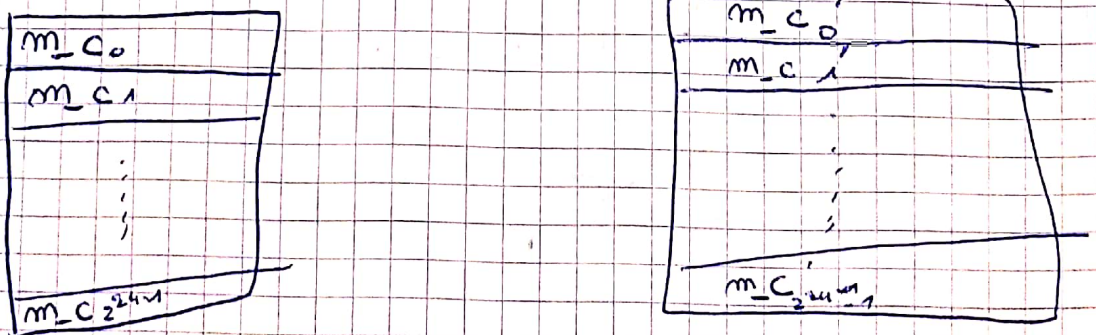
Déchiffrement :

- Refaire le processus précédent à l'envers, avec les clés dans l'ordre inverse, les permutations aussi, et prendre le tableau des boxes S à l'envers (substitution)

Attaque du milieu :

$m \rightarrow$ PRESENT $\xrightarrow{K_1}$ PRESENT $\xrightarrow{K_2} C$





$m\_c_k = ENC_{E_k}(m)$

$m'\_c'_k = DEC^{-1}_{E_k}(c)$

chercher $m\_c_i = m'\_c'_j$ t.q $i \neq j$ ou non fct (7)

Si

Si on trouve => renvoyer (E_i, E_j) et les écrire dans le fichier

Le faire pour nos deux couples (m, c)

Structures :

Tableau de permutation => map ?

Tab Enregistrement des permutations => ?

Stockage de tableau de char* pour l'attaque => ?