Calcul sécurisé - Contrôle continu

12 mars 2019

Table des matières

1 Question 1 **1**

2 Question 2 **3**
2.1 Décrire précisément ce que vous faites pour retrouver la clé 3
2.2 Donnez les 48 bits de clé que vous obtenez grâce à cette attaque par fautes . . . 4

3 Question 3 **4**
3.1 Expliquer comment on peut retrouver les 8 bits manquants 4
3.2 Faites-le, et donner ainsi la valeur complète de la clé qui vous a été assignée. . . 4

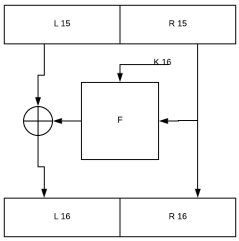
4 Question 4 **4**

5 Question 5 **4**

1 Question 1

Une attaque par faute contre le DES consiste à introduire une modification sur un bit du chiffré durant l'exécution du chiffrement de manière a compromettre son execution.De ce fait si on dispose d'un chiffré correct et d'un chiffré fauté par l'attaque on peut obtenir des informations sur une partie de la clé utilisée pour le chiffrement. Lors d'une attaque par force brute (recherche exhaustive) sur la clé du DES, la complexité est de 2^{56}. l'objectif de l'attaque par faute est de réduire cette complexitée.

En supposant que l'attaquant est capable d'effectuer une faute sur la valeur de sortie R_{15} du $15_{ème}$ tour une attaque par faute peut être décrite de la façon suivante :



Lors d'une utilisation normal de DES (sans attaque par faute), on obtiendrais les resultats suivant pour L_{16} et R_{16}.

— $L_{16} = L_{15} \oplus F(R_{15}, K_{16})$
— $R_{16} = R_{15}$

Maintenant, si on introduit une faute sur la valeur de sortie R_{15} du $15_{ème}$ tour, on obtient les valeurs suivantes :

— $R_{15} = R^*_{15}$
— $L_{16} = L^*_{16} = L_{15} \oplus F(R^*_{15}, K_{16})$
— $R_{16} = R^*_{15}$

On a donc une possibilité de retrouver K_{16} en utilisant L_{16} et L^*_{16}.
On va donc utiliser l'operation XOR ($\oplus$) sur L_{16} et L^*_{16} de façon a obtenir :

$$L_{16} \oplus L^*_{16} = L_{15} \oplus F(R_{15}, K_{16}) \oplus L_{15} \oplus F(R^*_{15}, K_{16})$$

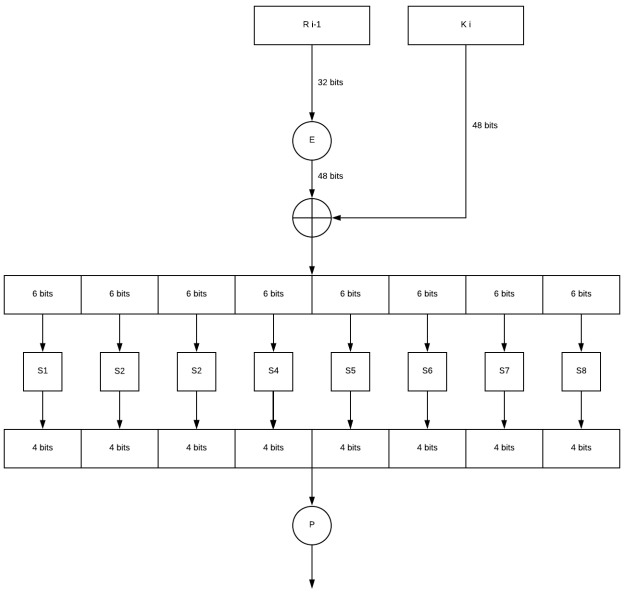
$$= \cancel{L_{15}} \oplus F(R_{15}, K_{16}) \oplus \cancel{L_{15}} \oplus F(R^*_{15}, K_{16})$$

On se retrouve donc avec :

$$L_{16} \oplus L^*_{16} = F(R_{15}, K_{16}) \oplus F(R^*_{15}, K_{16})$$

Pour continuer l'attaque on va devoir étudier la fonction F plus en détail soit :

F



On constate qie la fonction F prend le bloc R_{i-1} de 32 bits en entrée ainsi que la clé K_i. R_{i-1} passe ensuite par la fonction E qui a pour but d'appliquer une expansion sur le bloc, le passant de 32 à 48 bits.
Après l'expansion, l'opération XOR est appliquée entre R_{i-1} et K_i.
On a donc 48 bits , qui vont être "découpés" en 8 blocs de 6 bits. Chaque bloc va ensuite passer par une S-Box. Les S-Box prennent en entrée 6 bits et en renvoient 4, ce qui permet de ramener la nombre de bit à 32 (soit la taille initiale de R_{i-1}).
Ce bloc de 32 bits va finalement subir une permutation, la sortie de cette permutation étant le resultat renvoyé par la fonction F.
On peut donc écrire l'équation $L_{16} \oplus L_{16}^* = F(R_{15}, K_{16}) \oplus F(R_{15}^*, K_{16})$ de la façon suivante :

$$L_{16} \oplus L_{16}^* = P(S(E(R_{15}, K_{16}))) \oplus P(S(E(R_{15}^*, K_{16})))$$

On sait également d'après le schéma d'exécution de la fonction F qu'on peut isoler individuellement le résultat de chaque S-box, ce qui permet de transformer cette équation.
$L_{16} \oplus L_{16}^* = F(R_{15}, K_{16}) \oplus F(R_{15}^*, K_{16})$ devient :

$$L_{16} \oplus L_{16}^* =$$

$$P(S_1(E(R_{15}, K_{16}^{0-5}))||S_2(E(R_{15}, K_{16}^{6-11}))||...)$$

$$\oplus$$

$$P(S_1(E(R_{15}^*, K_{16}^{0-5}))||S_2(E(R_{15}^*, K_{16}^{6-11}))||...)$$

On peut se permettre d'écrire l'équation ainsi car le contenu des différentes S-box est connu, on peut donc à partir des 6 bits d'entrée trouver les bits de sortie correspondants (par ailleurs on peut également retrouver à partir des 4 bits de sortie d'une S-box plusieurs bloc de 6 bits d'entrée possible).

Maintenant on souhaite se débarrasser de P dans notre équation, pour cela il suffit d'appliquer P^{-1} à $L_{16} \oplus L_{16}^*$ soit (possible car la permutation P est connue) :

$$P^{-1}(L_{16} \oplus L_{16}^*) =$$

$$S_1(E(R_{15}, K_{16}^{0-5})) \oplus S_1(E(R_{15}^*, K_{16}^{0-5}))||S_2(E(R_{15}, K_{16}^{6-11})) \oplus S_2(E(R_{15}^*, K_{16}^{6-11}))||$$

$$...$$

On peut maintenant mettre en pratique le fait que l'on puisse retrouver les bits d'entrée des 6 box grâce aux bit de sortie pour utiliser cette équation sur les S-box.
On va diviser $P^{-1}(L_{16} \oplus L_{16}^*)$ en 8 blocs de 4 bits. Chaque bloc correspond donc à une sortie de S-box. On aura donc 8 équations :

- $P^{-1}(L_{16} \oplus L_{16}^*)_{0-3} = S_1(E(R_{15}, K_{16}))_{0-3} \oplus S_1(E(R_{15}^*, K_{16}))_{0-3}$
- $P^{-1}(L_{16} \oplus L_{16}^*)_{4-7} = S_2(E(R_{15}, K_{16}))_{4-7} \oplus S_2(E(R_{15}^*, K_{16}))_{4-7}$
- ...

K_{16} est la seule inconnue de ces 8 équations. Chacune de ces équations va nous permettre de retrouver 6 bits de K_{16}, on va donc devoir faire une recherche exhaustive des bits de K_{16} pour chaque S-box afin de retrouver les 48 bits de la sous-clé (6 bits par S-box).
On fera donc 8 recherches de 6 bits pour une complexité de $8 * 2^6$.

2 Question 2

2.1 Décrire précisément ce que vous faites pour retrouver la clé

Pour trouver la clé on va se servir du chiffré non fauté et le comparer à chacun des 32 chiffrés fauté.
Pour ce faire on va diviser chaque chiffré en 2 blocs de 32 bits chacun.
On aura donc un chiffré non fauté (C) sous la forme suivant :
1E F4 9F 41 || 6D 5D 57 8A
On va pour chaque chiffré fauté (FC) utiliser l'opération XOR entre le premier bloc de C et

celui de FC afin de retrouver 32 bits qu'on nommera E'.
On effectue la même opération entre le second bloc de C et de FC afin de trouver 32 autres bits qu'on nommera E.

2.2 Donnez les 48 bits de clé que vous obtenez grâce à cette attaque par fautes

3 Question 3

3.1 Expliquer comment on peut retrouver les 8 bits manquants

3.2 Faites-le, et donner ainsi la valeur complète de la clé qui vous a été assignée.

4 Question 4

5 Question 5