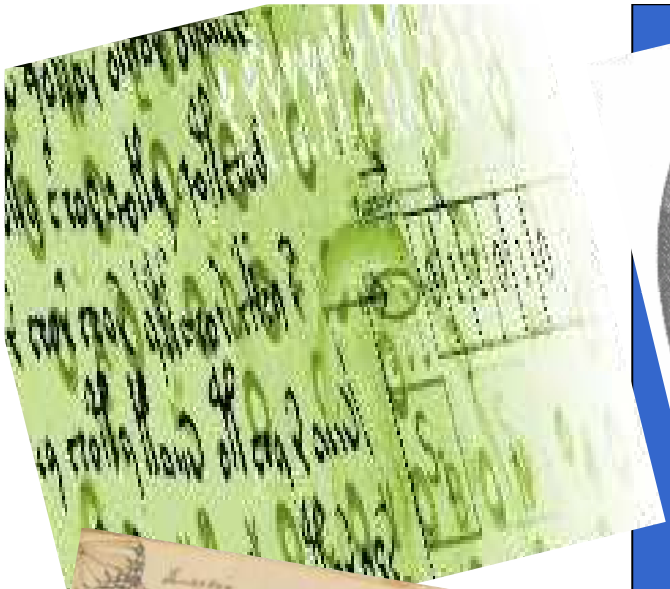


CRYPTANALYSE LINEAIRE:

(méthode d'attaque du DES)

- **Miyoo Tsanang Yves Stéphan**
- **Dorin-Alin Dicu**

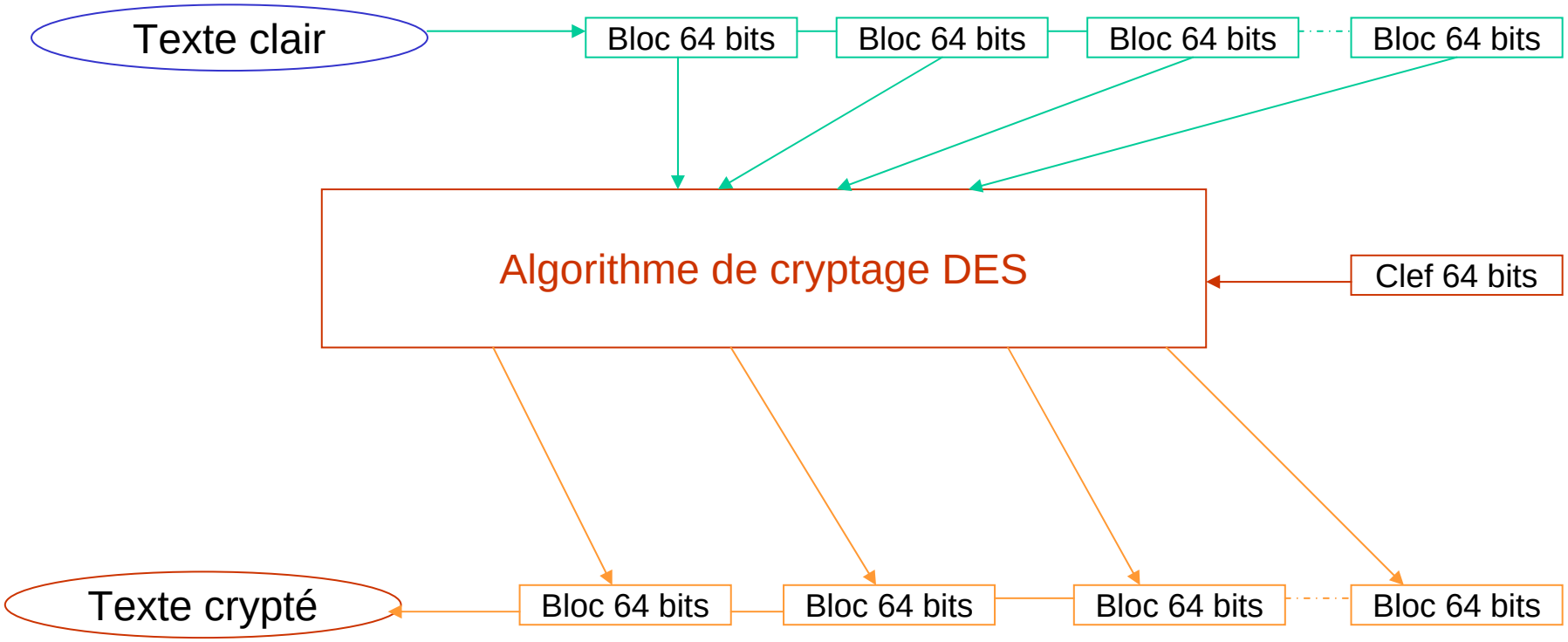
Sommaire

- Definition du DES et son fonctionnement
- Les types d’attaques du DES
- Cryptanalyse linéaire
  - Exemple
  - Principe de fonctionnement
- Securité du DES de nos jours
  - Points faibles
  - Autres techniques de cryptanalyse

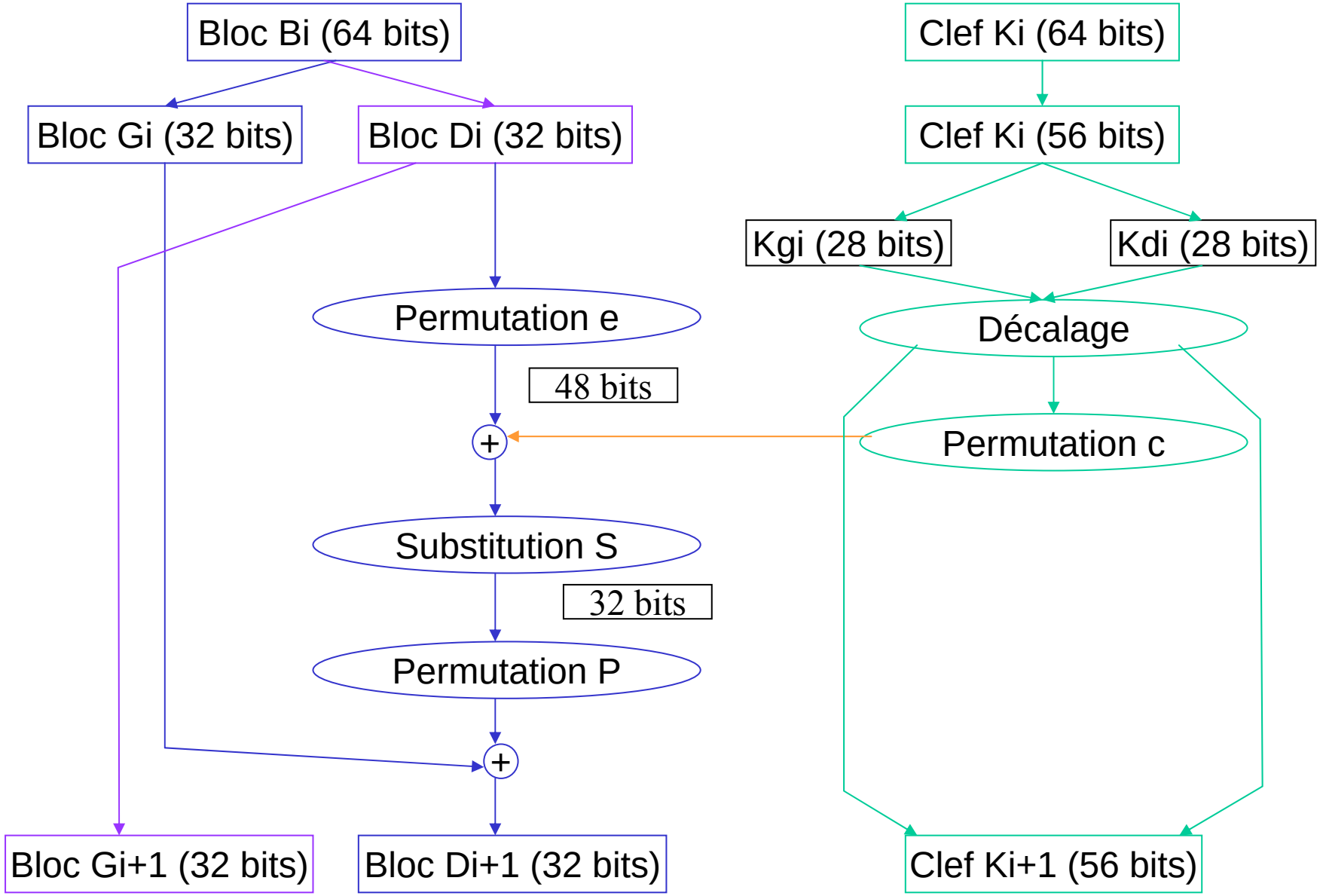
Introduction au DES

- DES = Data Encryption Standard
- En 1977 par IBM
- But : documents classés ou sécrets
- Aujourd'hui : industrie du logiciel et carte à puces
- 200 lignes de code
- Utilisation assez rapide sur des systémes dédiés, comme la carte à puces

Fonctionnement du DES (1)



Fonctionnement du DES (2)



Types d’attaques au DES

1. Recherche exhaustive
2. Machine dédiée
3. Compromis temps-espace (entre 1 et 2)
4. Cryptanalyse lineaire (Matsui)
5. Cryptanalyse differentielle
6. Projet LASEC

La cryptanalyse linéaire

- Techniques inventée par Matsui Mitsuru en 1993 pour casser le DES.
- Il s'est inspiré des travaux de Adi Shamir et Eli Beham, co-inventeur de la cryptanalyse différentielle
- En 1994 il realisa son attaque sur le DES, grace à 12 ordinateurs qui ont travaillé pendant 50 jours d'affilé.
- Sa méthode est plus efficace que la cryptanalyse différentielle
  - Il casse un DES de 16 tours en 2^{47}

Matsui Mitsuru

Exemple d'approche de la cryptanalyse linéaire(1)

- Soit par exemple une substitution avec 8 éléments, la fonction *S* est la fonction *substitution S(X)=Y, S(Y)=S(S(X))*

- cette table est non-linéaire mais **la combinaison de plusieurs substitutions et opérations peut annuler en partie la non-linéarité**; c'est la faille recherchée par la cryptanalyse linéaire

- La cryptanalyse linéaire vise à attribuer des vraisemblances aux équations possibles

X	Y
000	010
001	100
010	000
011	111
100	001
101	110
110	101
111	011

Exemple d'approche de la cryptanalyse linéaire(1)

- On peut déduire logiquement de cette subtition l'équivalence de ces 2 équations (1) et (2)

$$X_1 \oplus X_2 \oplus \ldots \oplus X_n = Y_1 \oplus Y_2 \oplus \ldots \oplus Y_n (1)$$
$$X_1 \oplus X_2 \oplus \ldots \oplus X_n \oplus Y_1 \oplus Y_2 \oplus \ldots \oplus Y_n = 0 (2)$$

- La cryptanalse linéaire vise à attribuer des vraisemblances aux équations possibles

- Exemple: considerons les 2 equations suivantes:

$$\rightarrow \begin{cases} X_1 \oplus X_2 \oplus X_3 = Y_1 \oplus Y_2 \\ X_2 \oplus X_3 = Y_3 \end{cases}$$

- On l'applique ensuite a notre subtitution précedente

X	Y
000	010
001	100
010	000
011	111
100	001
101	110
110	101
111	011

Exemple d'approche de la cryptanalyse linéaire(1)

Première équation

X	Y	$X_1 \oplus X_2 \oplus X_3$	$Y_1 \oplus Y_2$
000	010	0	1
001	100	1	1
010	000	1	0
011	111	0	0
100	001	1	0
101	110	0	0
110	101	0	1
111	011	1	1

Deuxième équation

X	Y	$X_2 \oplus X_3$	Y_3
000	010	0	0
001	100	1	0
010	000	1	0
011	111	0	1
100	001	0	1
101	110	1	0
110	101	1	1
111	011	0	1

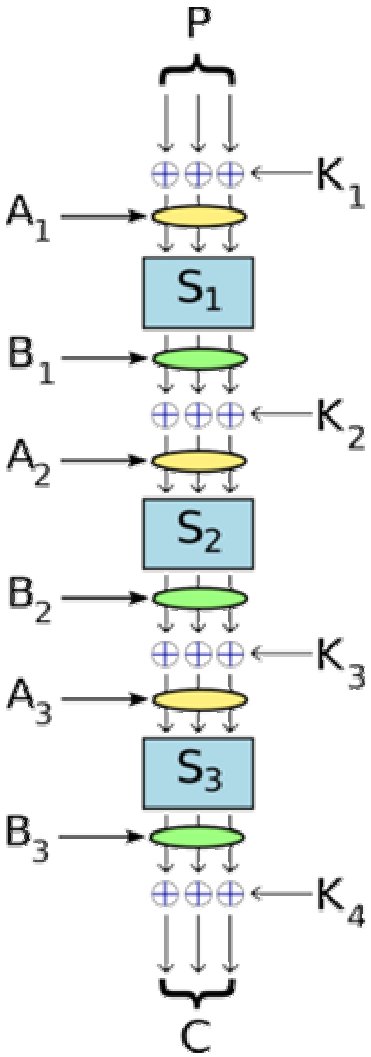
RESULTAT

- **La première équation est satisfaite 4 fois sur 8 (1)**
- **La seconde équation est satisfaite 2 fois sur 8 (2)**
- **Donc l'équation (1) est donc une meilleure approximation de notre substitution.**

Exemple d'approche de la cryptanalyse linéaire(2)

- Considérons maintenant un algorithme de chiffrement très simple qui prend 3 bit en entrée et donne 3 bit *chiffrés* en sortie.

- Soit ***P*** la donnée en clair de 3 bits et soit le résultat final ***C*** chiffré de 3 bits.



Chiffrement

La procédure de chiffrement s'effectue comme suit :

1. $A_1 = K_1 \oplus P$
2. $B_1 = S_1(A_1)$
3. $A_2 = K_2 \oplus B_1$
4. $B_2 = S_2(A_2)$
5. $A_3 = K_3 \oplus B_2$
6. $B_3 = S_3(A_3)$
7. $C = K_4 \oplus B_3$

Exemple d'approche de la cryptanalyse linéaire(2)

Création de l'approximation linéaire

- $S_1 : X_1 \oplus X_2 \oplus X_3 = Y_2$
- $S_2 : X_2 = Y_1 \oplus Y_3$

Première étape du chiffrement

A l'origine, nous avons

$$B_1 = S_1(A_1)$$

Avec l'approximation sur la première substitution S_1, on peut écrire :

$$B_{1,2} = A_{1,1} \oplus A_{1,2} \oplus A_{1,3}$$

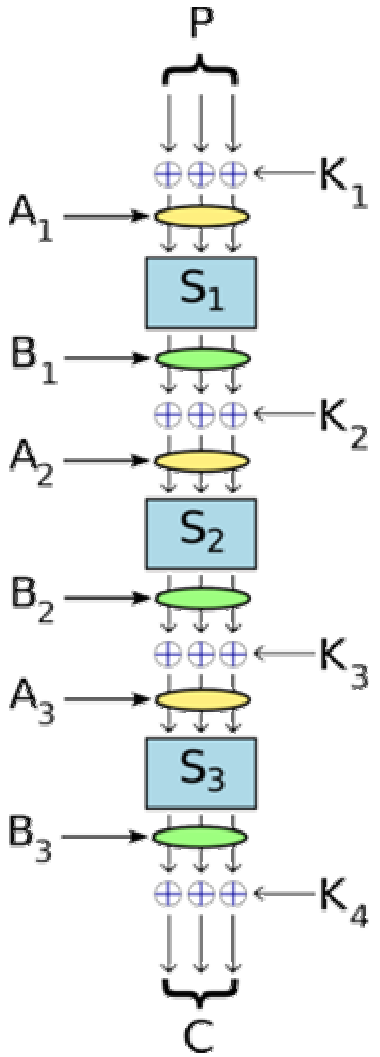
Or A_1 est équivalent à $K_1 \oplus P$, nous remplaçons donc A_1 :

$$B_{1,2} = (K_{1,1} \oplus P_{1,1}) \oplus (K_{1,2} \oplus P_{1,2}) \oplus (K_{1,3} \oplus P_{1,3}) \quad \textbf{(1)}$$

Deuxième étape du chiffrement

L'étape suivante dans le chiffrement consiste à faire un XOR entre B_1 et la clé K_2. Nous intégrons directement ce résultat avec l'équation (1) obtenue à l'étape précédente

$$A_{2,2} = B_{1,2} \oplus K_{2,2}$$
$$A_{2,2} = \Big((K_{1,1} \oplus P_{1,1}) \oplus (K_{1,2} \oplus P_{1,2}) \oplus (K_{1,3} \oplus P_{1,3})\Big) \oplus K_{2,2}$$



Exemple d'approche de la cryptanalyse linéaire(2)

Troisième étape du chiffrement

A ce stade, nous avons l'équation linéaire suivante :

$$A_{2,2} = \Big((K_{1,1} \oplus P_{1,1}) \oplus (K_{1,2} \oplus P_{1,2}) \oplus (K_{1,3} \oplus P_{1,3})\Big) \oplus K_{2,2}$$

Nous appliquons maintenant la 2e substitution $S_2 : X_2 = Y_1 \oplus Y_3$:

$$A_{2,2} = B_{2,1} \oplus B_{2,3}$$

En substituant :

$$\Big((K_{1,1} \oplus P_{1,1}) \oplus (K_{1,2} \oplus P_{1,2}) \oplus (K_{1,3} \oplus P_{1,3})\Big) \oplus K_{2,2} = B_{2,1} \oplus B_{2,3}$$

Quatrième étape

La sortie de l'étape précédente est maintenant chiffrée avec la clé K_3 donc $A_3 = B_2 \oplus K_3$:

Ceci donne finalement :

$$\Big((K_{1,1} \oplus P_{1,1}) \oplus (K_{1,2} \oplus P_{1,2}) \oplus (K_{1,3} \oplus P_{1,3})\Big) \oplus K_{2,2} = (A_{3,1} \oplus K_{3,1}) \oplus (A_{3,3} \oplus K_{3,3})$$

Nous arrangeons cette équation pour regrouper les termes :

$$(K_{1,1} \oplus K_{1,2} \oplus K_{1,3} \oplus K_{2,2} \oplus K_{3,1} \oplus K_{3,3}) \oplus (P_{1,1} \oplus P_{1,2} \oplus P_{1,3}) \oplus (A_{3,1} \oplus A_{3,3}) = 0$$

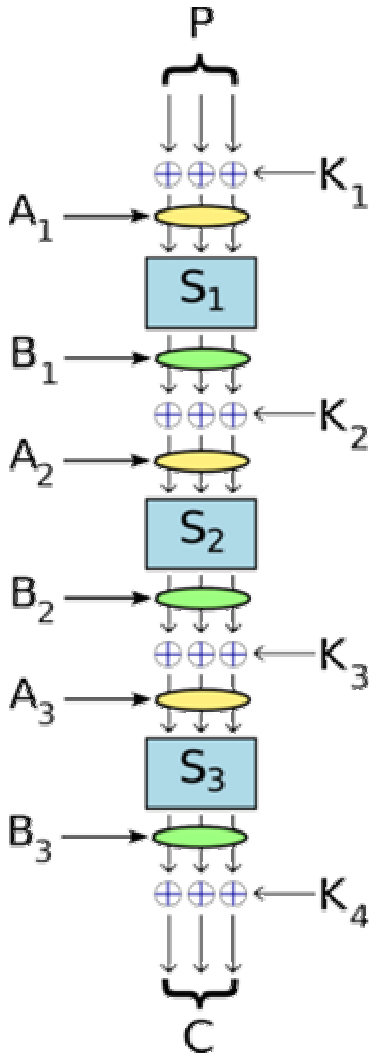
De manière plus condensée :

$$\Sigma K \oplus (P_{1,1} \oplus P_{1,2} \oplus P_{1,3}) \oplus (A_{3,1} \oplus A_{3,3}) = 0$$

avec $\Sigma K = (K_{1,1} \oplus K_{1,2} \oplus K_{1,3} \oplus K_{2,2} \oplus K_{3,1} \oplus K_{3,3})$

Nous avons maintenant une approximation linéaire qui dépend de :

- une partie des trois clés intermédiaires
- le texte en clair
- une partie de l'entrée de la dernière table de substitution

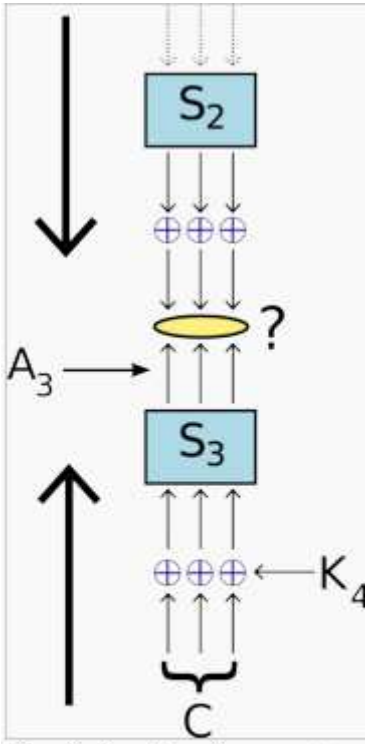


Exemple d'approche de la cryptanalyse linéaire(2)

Récupération des clés

Nous avons sous la main une approximation des 3 premiers tours de notre algorithme de chiffrement mais il manque la clé du dernier tour, soit K_4 dans notre cas. C'est ici qu'interviennent les messages chiffrés à notre disposition. Nous prenons un message et essayons de le déchiffrer en testant toutes les clés K_4 possibles. On s'intéresse plus particulièrement aux résultats à la fin du chiffrement. Plus précisément, nous prenons un message chiffré C et effectuons un XOR avec la dernière clé K_4 : $C \oplus K_4$. Cela correspond à la sortie de la dernière table de substitution. Nous effectuons maintenant la substitution inverse, la table étant connue : $S_3^{-1}(C \oplus K_4)$.

Or cette valeur correspond en fait au membre de gauche de notre équation linéaire ! Nous avons ainsi : $S_3^{-1}(C \oplus K_4) = A_3$. On peut donc avoir une estimation de la validité des clés testées en comparant la valeur exacte retournée par la substitution inverse et notre approximation linéaire sur tout ou une partie des bits. Avec un grand nombre de paires de messages, on peut rendre plus précise les estimations. Pour découvrir les autres clés intermédiaires, on attaque l'algorithme en remontant progressivement dans les tours jusqu'à arriver à la première clé.



Récupération de la clé en commençant par la fin et en confrontant les résultats à l'estimation linéaire

Principe de fonctionnement d'après Matsui

- L'idée de base consiste à estimer une portion de l'algorithme de chiffrage en utilisant des expressions linéaires ou l'opération linéaire est représenté par le ou-exclusif (XOR).

- En d'autres termes on exploite les comportements statistiques (non uniformes) dans le processus de chiffrement

- Il est donc question de déterminer les expressions qui résoudraient cette équation avec la plus grande probabilité ou la plus petite probabilité.

$$\left(\bigoplus_{i \in \{1 \ldots 64\}} \mathcal{P}^{(i)} \right) \oplus \left(\bigoplus_{j \in \{1 \ldots 64\}} \mathcal{C}^{(j)} \right) = \bigoplus_{k \in \{1 \ldots 56\}} \mathcal{K}^{(k)}$$

Principe de fonctionnement d'après Matsui

- Retrouver toutes les expressions linéaires **effectives** (1)
  - Etudes complètes des approximations linéaires des S-Box (2)
- Décrire de manière explicite leur probabilité de succès sur le nombre d'essai total possibles
- Rechercher la meilleure expression linéaire et calculer la probabilité d'occurrence la plus élevée.
  - Utiliser le Piling-up Lemme (3)
- Estimations des clés
  - Algorithme 1 et 2 de Matsui

$$P[i_1, i_2, \ldots, i_n] \oplus C[j_1, \ldots, j_m] = K[k_1, \ldots, k_c] \quad (1)$$

$$P[i_1, i_2, \ldots, i_n] \oplus C[j_1, \ldots, j_m] \oplus F_n(C_l, K_n)[l_1, \ldots, l_d] = K[k_1, \ldots, k_c] \quad (2)$$

$$P[X_i \oplus \ldots \oplus X_n] = \frac{1}{2} + 2^{n-1} \bullet \prod_{i=1}^{n} (p_i - \frac{1}{2}) \quad (3)$$

Sûreté du DES de nos jours

- Reponse facile: de moins en moins sûr, graces aux nouvelles technologies.
- Reponse difficile: rumeurs concernant des nouvelles techniques cryptographiques.

Autres techniques de cryptanalyse

1. Attaque par interpollation
2. Attaque par résolution de systémes d'equations algébriques
3. Attaque par décalage
4. Attaque par saturation

We wish you ...