Utilisation d'automates cellulaires pour la construction de grandes S-Boxes à partir d'un réseau de Feistel

Thomas Prévost - Bruno Martin

I3S, Université Côte d’Azur

Agenda

1. Le chiffrement par blocs
2. Théorème de Luby-Rackoff
3. Boîtes de substitution (S-Boxes)
4. Propriétés mathématiques des S-Boxes
5. Automates cellulaires uniformes
6. Construction de notre S-Box

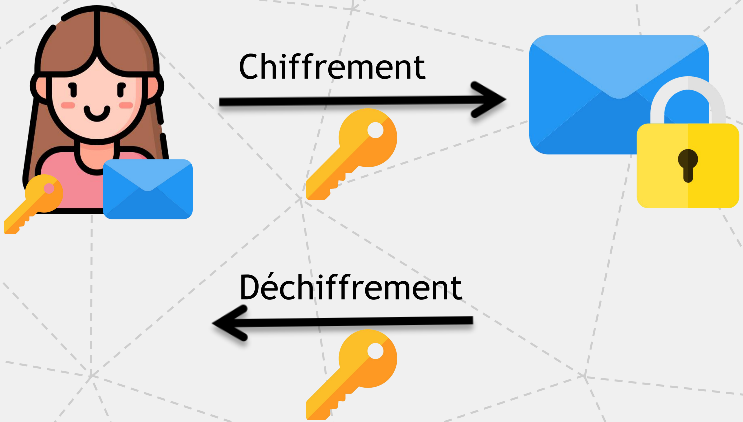
1

Chiffrement par blocs

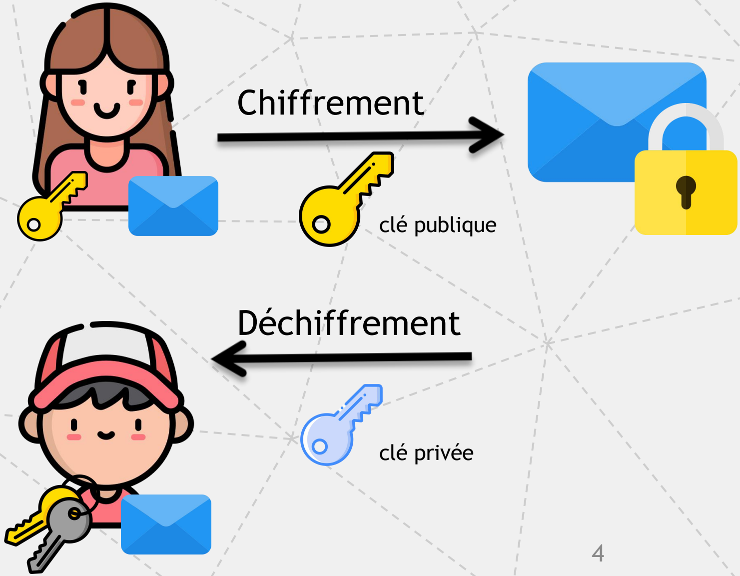
Le chiffrement par blocs

2 modes de chiffrement:

Chiffrement symétrique



Chiffrement asymétrique



Le chiffrement par blocs

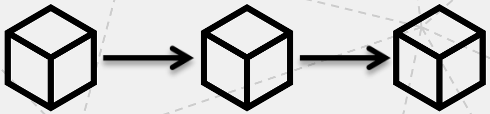
2 modes de chiffrement:

Chiffrement symétrique: Chiffre de gros volumes de données, pour du stockage ou de l'échange de message

Chiffrement asymétrique: Chiffre un petit message, pour échanger un secret entre deux personnes qui ne se sont jamais rencontrées

Le chiffrement par blocs

Chiffrement par blocs:
Découpage du message en blocs de taille égale

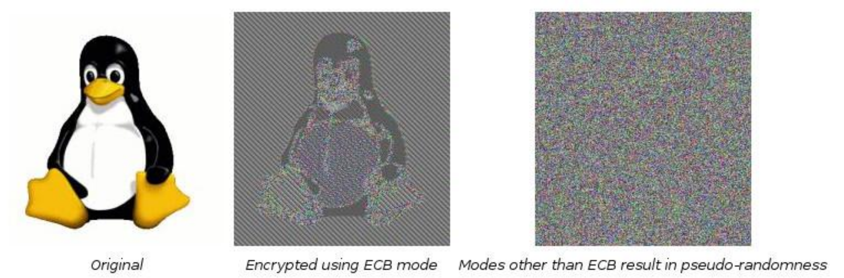


Exemple: Advanced Encryption Standard (AES), algorithme standardisé par le NIST

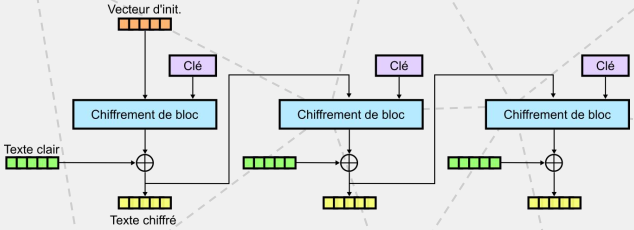
Le chiffrement par blocs

Interdépendance entre les blocs :

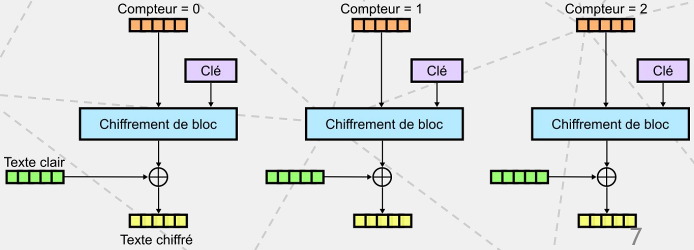
Chiffrement de chaque bloc indépendamment:



Solution: Chaînage des blocs (CBC):



Pour paralléliser: utilisation d'un compteur (GCM, CTR...)

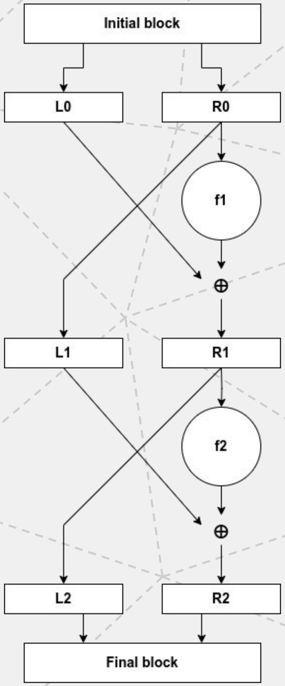


2

Théorème de Luby-Rackoff et constructions de Feistel

Construction de Feistel

Construction de Feistel pour une permutation de bloc



Avec:

- f_1 et f_2 permutations pseudo-aléatoires
- $\oplus$ opération XOR
- Ici, profondeur = 2

Permutation «pseudo-aléatoire»:
Permutation indistingable d'une permutation aléatoire par un adversaire disposant d'une puissance de calcul polynomiale

Théorème de Luby-Rackoff

Profondeurs minimales d'indistingabilité de permutation

Information à la disposition de l'adversaire	Profondeur minimale
Aucune	3
Inversion de la permutation intermédiaire	4
L'adversaire choisit les entrées (IND-CPA*)	7
L'adversaire peut «décoder» des sorties (IND-CCA2*)	10

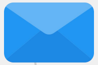
Voir «***Luby-Rackoff: 7 Rounds Are Enough for $2n(1-\varepsilon)$ Security***» *(Jacques Patarin)*

* explications slide suivante

IND-CPA (Indistinguishability under Chosen-Plaintext Attack)

Adversaire probabiliste en temps polynomial (PPT)

m



Envoie m entrée choisie

Oracle

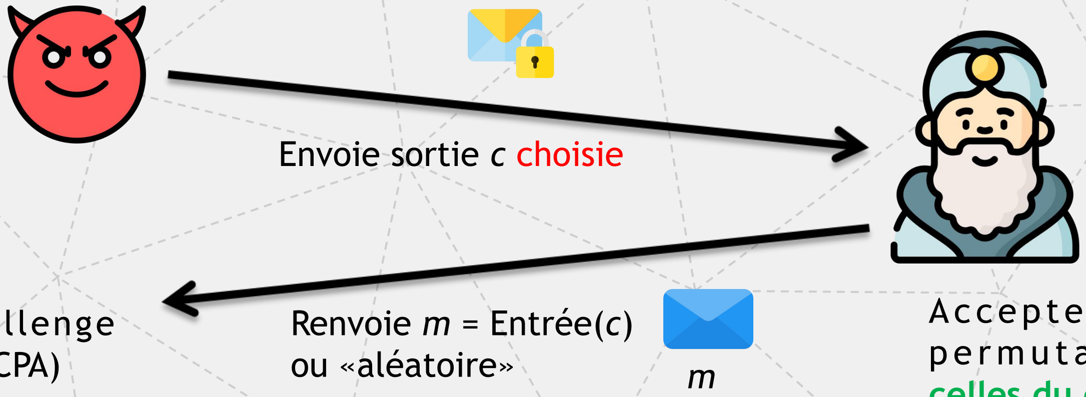
Renvoie c_0 = Permutation_Feistel(m) et c_1 = *aléatoire*

c_0 c_1

Ne doit pas être en mesure de distinguer c_0 et c_1 (c'est à dire deviner lequel correspond à m)

IND-???

IND-CCA2 (Indistinguishability under adaptative Chosen-Ciphertext Attack)



Adversaire PPT, à tout moment

Oracle

Rejoue au challenge précédent (IND-CPA)

Accepte toutes les permutations sauf celles du challenge

Niveau de sécurité intéressant par exemple lorsque l’ensemble des messages clairs possibles est restreint

3

S-boxes

S-boxes

Pourquoi on a besoin de S-boxes ?

Si le chiffrement par blocs était linéaire ?

Connu de l'attaquant

Secret (M) Clé (K) Chiffré (C)

S-boxes

Pourquoi on a besoin de S-boxes ?

Si le chiffrement par blocs était linéaire ?

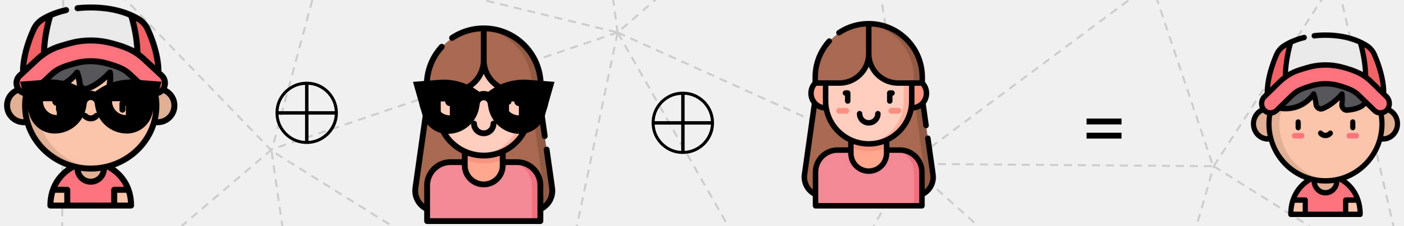
Connu de l’attaquant

Connu de l’attaquant

Clair connu (*M’*) — Clé (*K*) — Chiffré (*C’*)

Pourquoi on a besoin de S-boxes ?

Si le chiffrement par blocs était linéaire ?



Chiffré (C) $\oplus$ Chiffré (C') $\oplus$ Clair connu (M') $=$ Secret (M)

Attaque par clair connu

Pourquoi on a besoin de S-boxes ?

Action de la S-Box

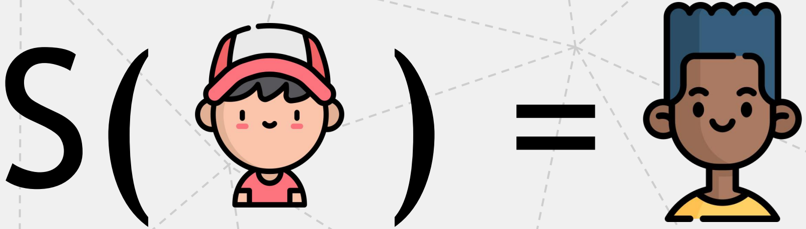


Table de substitution publique

Pour chaque tour: $c_1 = S(m) \oplus k$

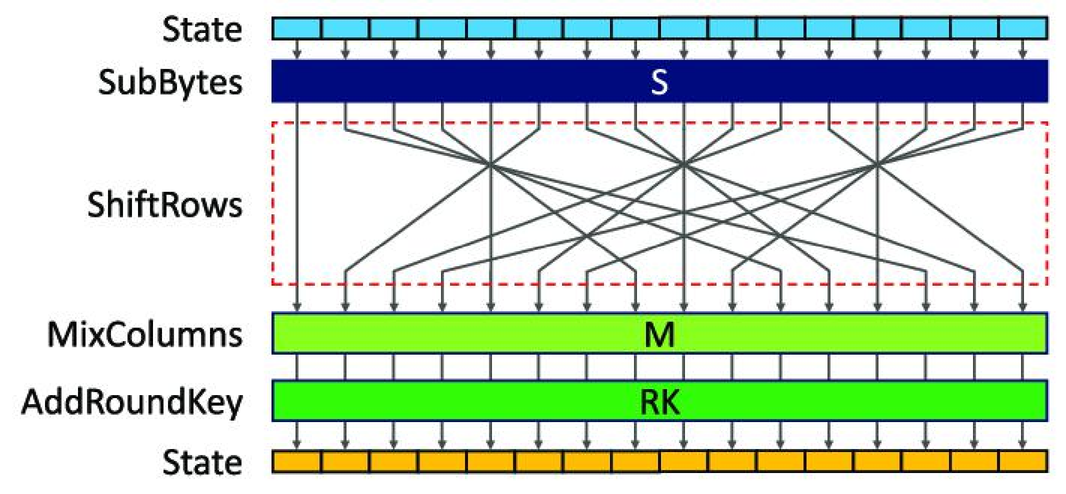
La S-Box doit s'éloigner le plus possible d'une fonction linéaire pour éviter l'attaque précédente... Nous verrons par la suite les propriétés nécessaires.

Exemple: S-box de 4 bits du chiffre PRESENT

x	0	1	2	3	4	5	6	7
$S(x)$	12	5	6	11	9	0	10	13

x	8	9	10	11	12	13	14	15
$S(x)$	3	14	15	8	4	7	1	2

Tour de chiffrement par blocs (ici AES)



1. S-box
2. P-box (pour la diffusion)
3. Mix-column (pour la diffusion)
4. Xor de clé

4

Propriétés mathématiques des S-boxes

Fonctions Booléennes

$f(x_1, x_2, \ldots, x_n) = y$ avec $x_1, x_2, \ldots, x_n$ et y booléens

Forme ANF (Algebraic Normal Form) de f:
$y = x_1 * x_2 * x_3 \oplus x_2 * x_4 \oplus x_5 \oplus 1$

deg(f) = 3 (degré du plus grand monôme)

Si degré = 1, la fonction est **linéaire**

$2^{(2^n)}$ fonctions Booléennes à n variables

Fonctions Booléennes des S-boxes

- On s'intéresse ici seulement aux S-Boxes **bijectives**
- S-Box à n variables: $S(x_1, x_2, \ldots, x_n) = y_1, y_2, \ldots, y_n$
- On étudie 2^n -1 fonctions booléennes **composantes de la S-Box:**
  - $f_1(x_1, x_2, \ldots, x_n) = y_1$
  - $f_2(x_1, x_2, \ldots, x_n) = y_2$
  - …
  - $f_{n+1}(x_1, x_2, \ldots, x_n) = y_1 \oplus y_2$
  - …
  - $f_{2^n-1}(x_1, x_2, \ldots, x_n) = y_1 \oplus y_2 \oplus \ldots \oplus y_n$

Propriétés des S-boxes

Exemple de fonctions composantes

x	00	01	10	11
$S(x)$	10	00	11	01

x	$f_1(x)$
00	1
01	0
10	1
11	0

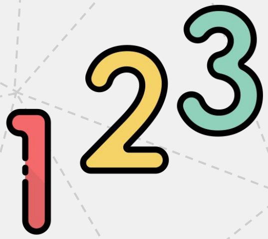
x	$f_2(x)$
00	0
01	0
10	1
11	1

x	$f_3(x) = f_1(x) \oplus f_2(x)$
00	1
01	0
10	1
11	0

Propriétés cryptographiques des S-Box (métriques)

1. Degré algébrique min et max
2. Complexité algébrique
3. Nonlinéarité
4. Critère d'avalanche strict (SAC)
5. Critère d'indépendance de bits (BIC)
6. Probabilité d'Approximation Linéaire (LAP)
7. Probabilité d'Approximation Différentielle (DAP)
8. Uniformité différentielle (DU)
9. Uniformité boomerang (BU)

Propriétés des S-boxes



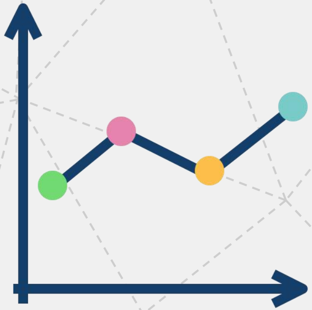
Degré algébrique min et max

- Taille du plus grand monôme de chaque fonction:
- Si $f_1(x_1, x_2, \ldots, x_n) = \boldsymbol{x_1 * x_2 * x_4} \oplus x_1 * x_2 \oplus x_3$ alors $\deg(f_1) = 3$
- Plus grand et plus petit degré de chaque fonction composante

Une grande valeur permet de se prémunir des attaques par «Low order approximation attack» (approximation de la fonction du plus bas degré)

Propriétés des S-boxes

Nonlinéarité



- Pour chaque fonction composante, la distance de Hamming à la fonction linéaire la plus proche
- C’est à dire le nombre de bits de la table de vérité qu’il faudrait changer pour obtenir une fonction linéaire

Une grande valeur permet de résister à la cryptanalyse linéaire (approximation linéaire de la S-Box)

Propriétés des S-boxes

Critère d'avalanche strict (SAC)

- Lorsqu'**un bit d'entrée** est modifié, en moyenne **50% des bits de sortie** doivent être modifiés
- La valeur idéale est donc de 50%

- On définit aussi une table de taille n*n:
  - Lorsque le *i*è bit d'entrée est modifié, dans quelle proportion le *j*è bit de sortie est modifié ?
- Les valeurs de la table doivent être aussi proche que possible de 50%

Propriétés des S-boxes

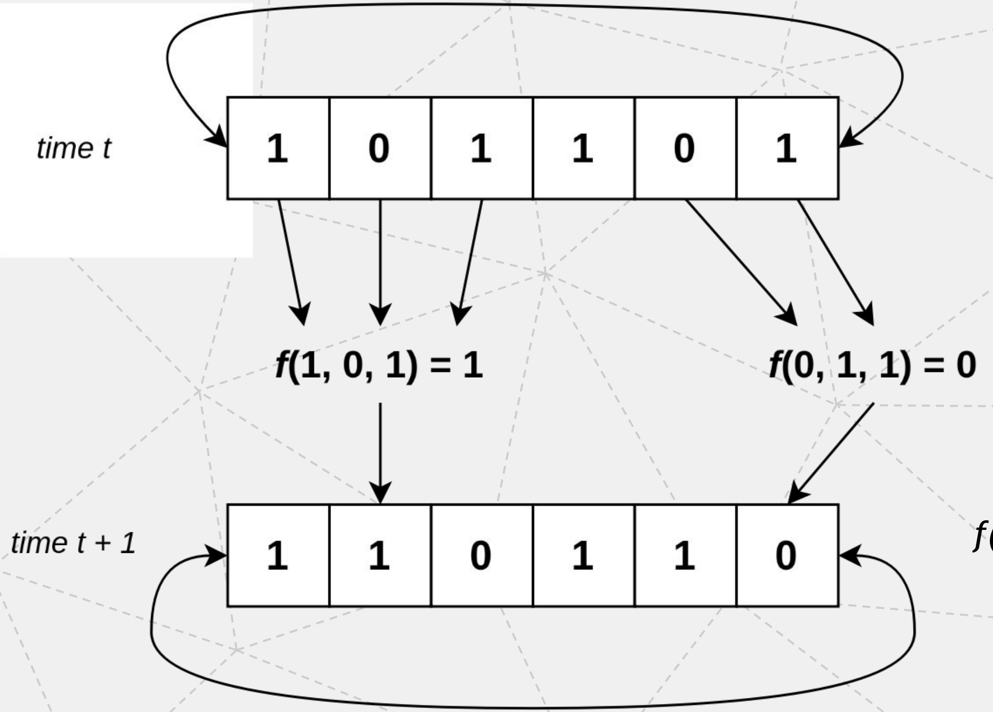
Uniformité différentielle

- Donne la proximité à une S-Box parfaitement non-linéaire
- Pour chaque combinaison *a*, *b*, la table d'uniformité différentielle δ donne le nombre d'entrées *x* tel que $S(x) \oplus S(x \oplus a) = b$
- On a alors U = max(δ)
- La plus petite valeur est la meilleure

5

Automates cellulaires binaires uniformes

Automates cellulaires uniformes



f: local
rule **224**

0 0 0 -> 0
0 0 1 -> 0
0 1 0 -> 0
0 1 1 -> 0
1 0 0 -> 0
1 0 1 -> 1
1 1 0 -> 1
1 1 1 -> 1

$$f(x) = x_0 * x_1 * x_2 + x_0 * x_2 + x_1 * x_2$$

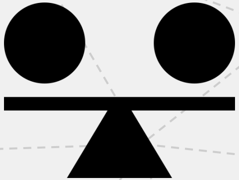
6

Construction de notre S-Box de 10 bits

Construction de notre S-box

Choix d’une fonction de transition locale

- Fonction Booléenne à 5 variables
- Equilibrée
- Non linéaire

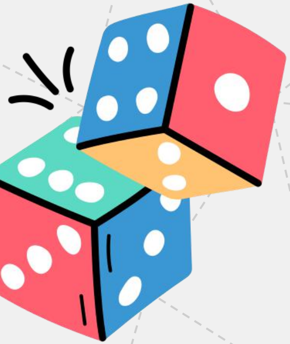


- Valide l’immunité de corrélation (sortie indépendante de n’importe quel sous-ensemble de l’entrée)
- Valide le critère d’avalanche strict

Choix d'une fonction de transition locale

Validation du caractère chaotique de la fonction de transition locale:

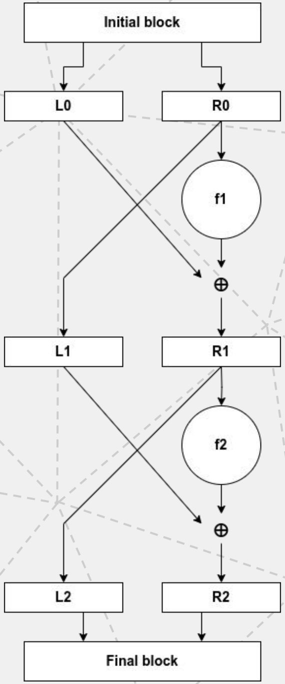
- Construction d'un automate cellulaire à 5 variables
- Evaluation de l'aléatoire généré à chaque génération (NIST FIPS 140-2)



Il reste 1 règle de transition, dont la table de vérité est **1 438 886 595**

Construction de notre S-box

Construction d'un réseau de Feistel à 10 variables



Construction empirique basée sur la cryptanalyse

- f_1: fonction affine: $f(x) = 5x+3$ mod 31
- f_2 à f_5: 1 génération de notre automate
- f_6: fonction affine: $f(x) = 7x+11$ mod 31
- f_7 à f_9: 1 génération de notre automate
- f_{10}: fonction affine: $f(x) = 13x+17$ mod 31
- f_{11}: 1 génération de notre automate

Fonction affine à coefficients premiers: évite $S(0) = 0$ et $S(1023) = 1023$, et garantit la bijectivité de la S-Box

Construction de notre S-box

Comparaison avec la S-Box d'AES

Propriété	Notre S-Box (10 bits)	AES (8 bits)
Degré algébrique min	8	7
Degré algébrique max	9	7
Complexité algébrique	**1023**	255
Nonlinéarité	434 (=108.5x4)	112
Critère d'avalanche strict	**0.44 - 0.5 - 0.57**	0.45 - 0.5 - 0.56

Comparaison avec la S-Box d'AES

Propriété	Notre S-Box (10 bits)	AES (8 bits)
Critère d'indépendance de bits	0.124	0.134
Probabilité d'approximation linéaire	9.28%	6.25%
Probabilité d'approximation différentielle	1.37%	1.56%
Uniformité différentielle	14 (=3.5x4)	4
Uniformité boomerang	24 (=6x4)	6

Construction de notre S-box

Implémentation

https://github.com/thomasarmel/sponges/blob/sbox_10/ascon/src/lib.rs#L100

Merci !

Avez-vous des questions ?

Propriétés des S-boxes

Complexité algébrique

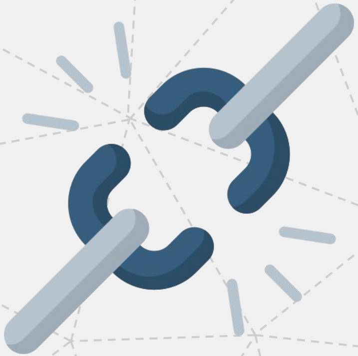
- On représente notre S-Box sur $\mathbb{N}$:
- $S(x) = a_0 + a_1 * x + \ldots + a_{2^n-1} * x^{(2^n)-1}$ mod 2^n avec $x, a_0, a_1, \ldots \in [\![0, 2^n-1]\!]$

La complexité algébrique est le nombre de monômes composants le polynôme univarié

Une grande valeur permet de se prémunir des attaques par interpolation

Propriétés des S-boxes

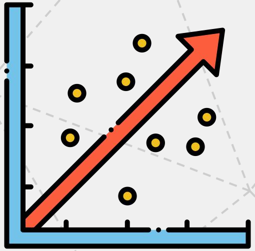
Critère d’indépendance de bits (BIC)



- Une S-Box satisfait le critère d’indépendance de bits lorsque pour tout bit d’entrée *k*, pour toute combinaison i, j, changer le k^{è} bit d’entrée modifie les i^{è} et j^{è} bits de sortie indépendamment
- $i, j, k \in [\![1, n]\!]$ et $i \neq j$

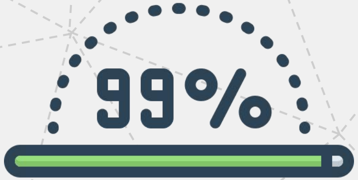
- Une métrique entre 0 et 1 indique à quel point une S-Box est proche de satisfaire le BIC
- 0 est le meilleur, 1 le pire

Probabilité d’approximation linéaire (LAP)



- Donne une indication de la résistance de la S-Box à la cryptanalyse linéaire
- Définie comme la corrélation maximum entre *a*x* et *B*S(x)*, pour tout *a* et *B* $\in [\![1, 2^n]\!]$
- La plus petite valeur est la meilleure

Probabilité d'approximation différentielle (DAP)



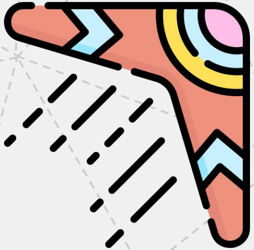
Donné par la distribution XOR entre l'entrée et la sortie:

- Pour chaque combinaison Δx, Δy, la table de probabilité différentielle DP donne le nombre d'entrées x tel que $S(x) \oplus S(x \oplus \Delta x) = \Delta y$
- On a alors DAP = max(DP)

Une valeur basse garantit une forte résistance à la cryptanalyse différentielle

Propriétés des S-boxes

Uniformité boomerang (BU)



- Définit la résistance de la S-Box aux **attaques boomerang** (variante de l'attaque par cryptanalyse différentielle)
- Pour chaque combinaison *a*, *b*, la table de connectivité boomerang BCT donne le nombre d'entrées *x* tel que:

$$S^{\wedge -1}(S(x) \oplus b) \oplus S^{\wedge -1}(S(x \oplus a) \oplus b) = a$$

- On a alors BU = max(BCT)
- La plus petite valeur donne une S-Box de plus grande résistance face aux attaques boomerang