



INFO0603

COMPRESSION ET CRYPTOGRAPHIE

COURS 2 ET 3

GÉNÉRALITÉS CRYPTOGRAPHIQUE
ET CHIFFREMENTS SYMÉTRIQUES



UNIVERSITÉ
DE REIMS
CHAMPAGNE-ARDENNE

Goéry Valance
Département de Mathématiques et Informatique
Février 2023

Plan du cours

- Rappels sécurité informatique
- Chiffrements de Base
- Chiffrement de Vigenère
- Schéma de Feistel
- Le DES

Les différents problèmes de sécurité informatique

Introduction à la sécurité Sécurité

Confidentialité, intégrité et authentification

Confidentialité

- Concerne toutes les parties du système : matériel, logiciels, données
- Accès uniquement aux parties autorisées
- Sécurisation des données

Intégrité

- Modifications suivant autorisation
- S'assurer de la non modification des données/messages

Authentification

- S'assurer de l'identité de l'acteur
- Associée à des autorisations

• Contrôler une identité

• Chiffrer les données

• Contrôler les données

| | |
|---------|-----|
| Théorie | 300 |
| 1.1 | 301 |
| 1.2 | 302 |
| 1.3 | 303 |
| 1.4 | 304 |
| 1.5 | 305 |
| 1.6 | 306 |
| 1.7 | 307 |
| 1.8 | 308 |
| 1.9 | 309 |
| 1.10 | 310 |
| 1.11 | 311 |
| 1.12 | 312 |
| 1.13 | 313 |
| 1.14 | 314 |
| 1.15 | 315 |
| 1.16 | 316 |
| 1.17 | 317 |
| 1.18 | 318 |
| 1.19 | 319 |
| 1.20 | 320 |
| 1.21 | 321 |
| 1.22 | 322 |
| 1.23 | 323 |
| 1.24 | 324 |
| 1.25 | 325 |
| 1.26 | 326 |
| 1.27 | 327 |
| 1.28 | 328 |
| 1.29 | 329 |
| 1.30 | 330 |
| 1.31 | 331 |
| 1.32 | 332 |
| 1.33 | 333 |
| 1.34 | 334 |
| 1.35 | 335 |
| 1.36 | 336 |
| 1.37 | 337 |
| 1.38 | 338 |
| 1.39 | 339 |
| 1.40 | 340 |
| 1.41 | 341 |
| 1.42 | 342 |
| 1.43 | 343 |
| 1.44 | 344 |
| 1.45 | 345 |
| 1.46 | 346 |
| 1.47 | 347 |
| 1.48 | 348 |
| 1.49 | 349 |
| 1.50 | 350 |

Les différents problèmes de sécurité informatique

- **Interception :**
 - Accès à des parties non autorisées
 - Récupération de données (écoute, copie de données)
- **Interruption :**
 - Service inaccessible : déni de service (distribué)
 - Données corrompues
- **Modification :** modification non autorisées
- **Ajout :** ajout d'informations non autorisées (injection)

Chiffrement

La cryptographie peut être utile :

Mes communications (par exemple courrier électronique)

Mes données (par exemple mon disque dur)

Ceci relève de la confidentialité.

Les données peuvent être chiffrées pour ne pas être déchiffrables par l'adversaire.

Merci à Cécile Pierrot pour son cours de Telecom Nancy

| | |
|--------------|-----|
| Phrygane | 300 |
| la | 301 |
| | 302 |
| | 303 |
| dit | 304 |
| | 305 |
| | 306 |
| faute | 307 |
| des quilles | 308 |
| | 309 |
| pro. le | 310 |
| alors | 311 |
| hard | 312 |
| afin | 313 |
| qu'on | 314 |
| re. vs | 315 |
| elle/plus | 316 |
| cr. | 317 |
| Prismite | 318 |
| | 319 |
| la | 320 |
| au point | 321 |
| | 322 |
| | 323 |
| re. | 324 |
| Arrière | 325 |
| avec | 326 |
| implore | 327 |
| sur | 328 |
| est | 329 |
| de. | 330 |
| franch | 331 |
| capitulation | 332 |
| multitude | 333 |
| et le | 334 |
| tu | 335 |
| et on | 336 |
| implore | 337 |
| et/le | 338 |
| sur | 339 |
| son. son | 340 |
| | 341 |
| | 342 |
| | 343 |
| unbe | 344 |
| | 345 |
| ge final | 346 |
| devant | 347 |
| elle. | 348 |
| avec | 349 |
| tre. | 350 |

Bien définir le problème

Souvent, il n'y a pas de réponse unique à un besoin de cryptographie.
Tout dépend des hypothèses faites sur l'espion et sur les garanties qu'on souhaite obtenir.

Il faut être réaliste. Mon mail passe par gmail, donc la NSA l'écoute.

https://www.lemonde.fr/pixels/article/2021/05/30/comment-des-dirigeants-europeens-ont-ete-espionnes-par-la-nsa-depuis-le-danemark_6082102_4408996.html

En général, on suppose que l'attaquant est très fort, et on voit ce qu'on peut garantir.

Chiffrement

| | |
|----------------|-----|
| Phrygane | 300 |
| la | 301 |
| | 302 |
| | 303 |
| dit | 304 |
| | 305 |
| | 306 |
| faute | 307 |
| des guillemets | 308 |
| | 309 |
| pro. le | 310 |
| alors | 311 |
| hard | 312 |
| afin | 313 |
| quand | 314 |
| re. vs | 315 |
| elle/leur | 316 |
| ce. | 317 |
| Prismoir | 318 |
| | 319 |
| la | 320 |
| au point | 321 |
| | 322 |
| | 323 |
| re. | 324 |
| Arrière | 325 |
| avec | 326 |
| implément | 327 |
| sur | 328 |
| est | 329 |
| de. | 330 |
| tranché | 331 |
| capitulation | 332 |
| multitude | 333 |
| et le | 334 |
| tu | 335 |
| et on | 336 |
| impair | 337 |
| et/ce | 338 |
| | 339 |
| son. son | 340 |
| | 341 |
| | 342 |
| | 343 |
| unbe | 344 |
| | 345 |
| ge final | 346 |
| derrière | 347 |
| elle. | 348 |
| avec | 349 |
| tre. | 350 |

Le chiffrement n'est qu'un des moyens de protection,

Il y a aussi la sécurité informatique, la protection contre les rayonnements, les gens...

Pour se protéger efficacement, il faut combiner les techniques.

Tout cela a bien sûr de nombreuses applications dans la vie courante : paiement, bluetooth, wifi, téléphone etc...

| | |
|---------|-----|
| Théorie | 300 |
| 1.1 | 301 |
| 1.2 | 302 |
| 1.3 | 303 |
| 1.4 | 304 |
| 1.5 | 305 |
| 1.6 | 306 |
| 1.7 | 307 |
| 1.8 | 308 |
| 1.9 | 309 |
| 1.10 | 310 |
| 1.11 | 311 |
| 1.12 | 312 |
| 1.13 | 313 |
| 1.14 | 314 |
| 1.15 | 315 |
| 1.16 | 316 |
| 1.17 | 317 |
| 1.18 | 318 |
| 1.19 | 319 |
| 1.20 | 320 |
| 1.21 | 321 |
| 1.22 | 322 |
| 1.23 | 323 |
| 1.24 | 324 |
| 1.25 | 325 |
| 1.26 | 326 |
| 1.27 | 327 |
| 1.28 | 328 |
| 1.29 | 329 |
| 1.30 | 330 |
| 1.31 | 331 |
| 1.32 | 332 |
| 1.33 | 333 |
| 1.34 | 334 |
| 1.35 | 335 |
| 1.36 | 336 |
| 1.37 | 337 |
| 1.38 | 338 |
| 1.39 | 339 |
| 1.40 | 340 |
| 1.41 | 341 |
| 1.42 | 342 |
| 1.43 | 343 |
| 1.44 | 344 |
| 1.45 | 345 |
| 1.46 | 346 |
| 1.47 | 347 |
| 1.48 | 348 |
| 1.49 | 349 |
| 1.50 | 350 |

Cryptologie

C'est l'étude de la protection de l'information sous forme numérique contre des accès ou manipulations non-autorisés.

cryptologie = cryptographie + cryptanalyse

cryptographie : conception des algorithmes cryptographiques

cryptanalyse : évaluation de la sécurité des algorithmes cryptographiques

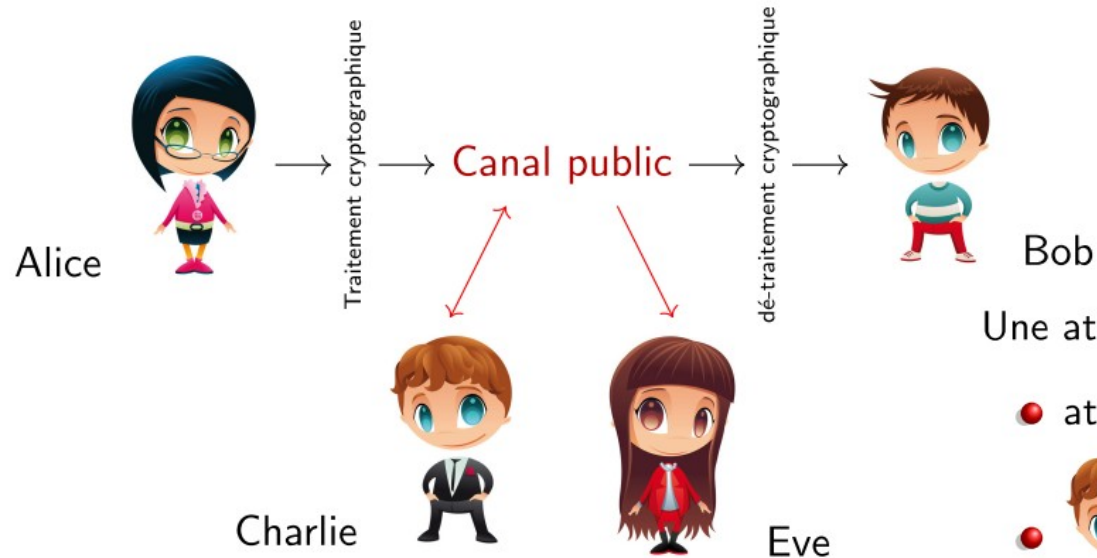
Vocabulaire

| | |
|---------|-----|
| Phonème | 300 |
| la | 301 |
| 302 | |
| 303 | |
| 304 | |
| 305 | |
| 306 | |
| 307 | |
| 308 | |
| 309 | |
| 310 | |
| 311 | |
| 312 | |
| 313 | |
| 314 | |
| 315 | |
| 316 | |
| 317 | |
| 318 | |
| 319 | |
| 320 | |
| 321 | |
| 322 | |
| 323 | |
| 324 | |
| 325 | |
| 326 | |
| 327 | |
| 328 | |
| 329 | |
| 330 | |
| 331 | |
| 332 | |
| 333 | |
| 334 | |
| 335 | |
| 336 | |
| 337 | |
| 338 | |
| 339 | |
| 340 | |
| 341 | |
| 342 | |
| 343 | |
| 344 | |
| 345 | |
| 346 | |
| 347 | |
| 348 | |
| 349 | |
| 350 | |

| Anglais | Français | Commentaire | Anglais | Français |
|------------------|---------------|---|---------------|-----------------------|
| Cipher | Chiffre | Rare (en français) | Cryptanalysis | Cryptanalyse |
| Cryptosystem | Cryptosystème | | Block cipher | Chiffrement par blocs |
| Encrypt | Chiffrer | Crypter | Stream cipher | Chiffrement à flot |
| Encryption | Chiffrement | Chiffrage Cryptage | Plaintext | Message clair |
| Decrypt | Déchiffrer | | Ciphertext | Message chiffré |
| Decryption | Déchiffrement | Décryptement Décryptage | Hash function | Fonction de hachage |
| (Ad.) decrypt | Décrypter | L'adversaire décrypte, | Digest | Haché, ou empreinte |
| (Ad.) decryption | Attaque | le correspondant déchiffre. | | |

Modèle de communication et menace

Modèle simplifié d'un système de communication cryptographique



Une attaque peut être

• attaque passive : **espionnage**

• : attaque active ;

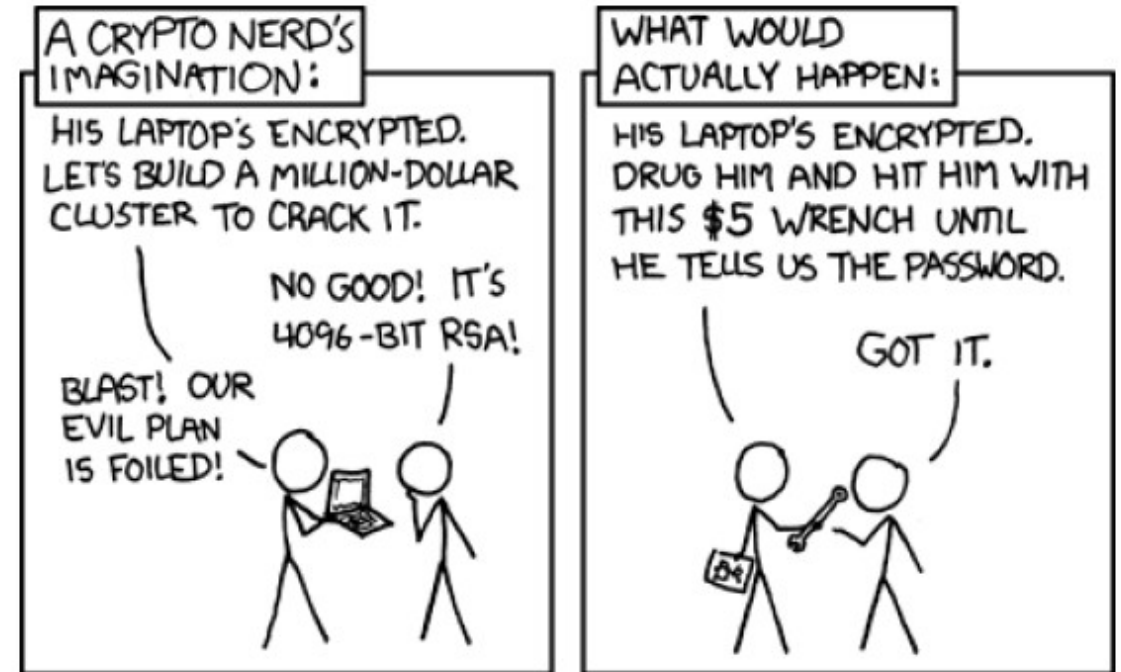
- **usurpation d'identité** (de l'émetteur ou du récepteur)
- **altération des données** = modification du contenu du message
- **répudiation du message** = l'émetteur nie l'avoir envoyé
- **répétition du message**
- **retardement de la transmission**
- **destruction du message**

Motivations et cibles

- Des attaques aux motivations très variées
 - Ludique : amusement, curiosité, défi, réputation
 - Idéologie (voire terrorisme) : vandalisme, déni de service
 - Cupidité : vol de données bancaires, extorsion (*ransomware*)
 - Espionnage : industriel (concurrence) ou étatique (surveillance)
- Tout système d'information est une cible potentielle
 - Infrastructures «vital» : réseaux électrique, de communications, de transports, centrales nucléaires, hôpitaux
 - États : sites gouvernementaux et militaires
 - Entreprises : cyber-espionnage, vengeance
 - Entités académiques : universités, laboratoires de recherche
 - Individus : cibles vulnérables, peu sensibilisées, ne maîtrisent pas toutes les données qu'elles produisent ; leurs machines peuvent aussi servir de relais (*botnet*)

Limites de la cryptographie

- Chaque couche du système présente des vulnérabilités.
- Un plan de sécurité global est nécessaire.
- La cryptographie n'est qu'un élément de réponse au besoin global de sécurité
- L'attaquant attaquera toujours le point le plus faible...



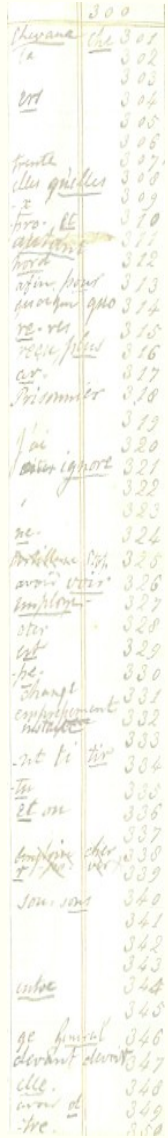
Les primitives cryptographiques

Algorithmes fournissant une fonctionnalité cryptographique élémentaires :

- contrôle d'intégrité → fonction de hachage
- génération de clés → générateur d'aléa
- authentification → algorithme de signature
- confidentialité → chiffrement

Rappel : Une fonction de hachage est un algorithme (efficace) qui calcule une valeur de taille fixe, appelée empreinte ou haché à partir de messages de taille quelconque.

Les desiderata de Kerckhoffs (1883)



| | |
|-----------|-----|
| Théorème | 300 |
| la | 301 |
| | 302 |
| | 303 |
| est | 304 |
| | 305 |
| | 306 |
| | 307 |
| les qu'il | 308 |
| | 309 |
| | 310 |
| | 311 |
| | 312 |
| | 313 |
| | 314 |
| | 315 |
| | 316 |
| | 317 |
| | 318 |
| | 319 |
| | 320 |
| | 321 |
| | 322 |
| | 323 |
| | 324 |
| | 325 |
| | 326 |
| | 327 |
| | 328 |
| | 329 |
| | 330 |
| | 331 |
| | 332 |
| | 333 |
| | 334 |
| | 335 |
| | 336 |
| | 337 |
| | 338 |
| | 339 |
| | 340 |
| | 341 |
| | 342 |
| | 343 |
| | 344 |
| | 345 |
| | 346 |
| | 347 |
| | 348 |
| | 349 |
| | 350 |

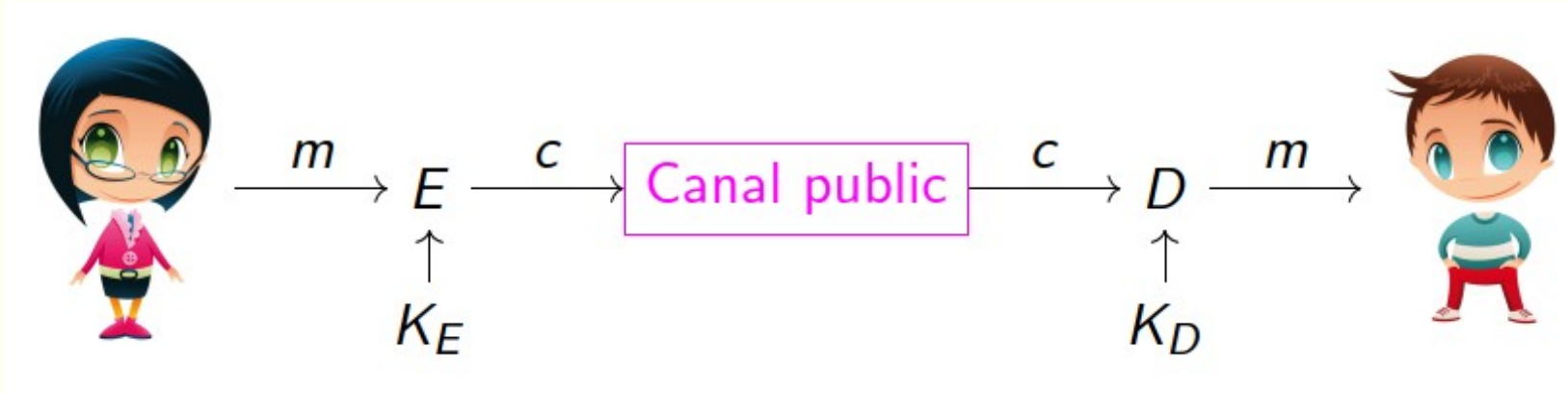
- 1) Le système doit être matériellement, sinon mathématiquement, indéchiffrable
- 2) Il n'exige pas le secret**
- 3) La clef doit pouvoir être retenue sans le secours de notes écrites, et être changée : *(que dire alors des wallet)*
- 4) Il faut qu'il soit applicable à la correspondance télégraphique (ou d'internet...)
- 5) Il faut qu'il soit portatif
- 6) Il faut que le système soit d'un usage facile

Quelques leçons de l'histoire

| | |
|----------|-----|
| Théorème | 300 |
| 1a | 301 |
| 302 | |
| 303 | |
| 304 | |
| 305 | |
| 306 | |
| 307 | |
| 308 | |
| 309 | |
| 310 | |
| 311 | |
| 312 | |
| 313 | |
| 314 | |
| 315 | |
| 316 | |
| 317 | |
| 318 | |
| 319 | |
| 320 | |
| 321 | |
| 322 | |
| 323 | |
| 324 | |
| 325 | |
| 326 | |
| 327 | |
| 328 | |
| 329 | |
| 330 | |
| 331 | |
| 332 | |
| 333 | |
| 334 | |
| 335 | |
| 336 | |
| 337 | |
| 338 | |
| 339 | |
| 340 | |
| 341 | |
| 342 | |
| 343 | |
| 344 | |
| 345 | |
| 346 | |
| 347 | |
| 348 | |
| 349 | |
| 350 | |

- 1) Les desiderata de Kerckhoffs : la sécurité ne doit pas reposer sur le secret des spécifications
- 2) La loi de Moore : la puissance des processeurs double tous les 18 mois, gare à la recherche exhaustive
- 3) La loi de Murphy : un trou de sécurité finira toujours par être découvert... au pire moment
- 4) Le principe de réalité : un procédé inadapté (cher, contraignant, lent, etc.) ne sera pas utilisé
- 5) Ne pas réinventer la roue : utiliser un standard / une librairie existante plutôt que faire son propre algorithme cryptographique (surtout si on n'est pas expert...)

Cryptographie avec clé



On parlera de cryptographie symétrique ou de cryptographie à clé secrète) dans le cas $K_D=K_E$

Notations générales

Un système de chiffrement est un 5-uple $(\mathcal{P}, \mathcal{C}, \mathcal{K}, E, D)$ avec :

- \mathcal{P} l'ensemble des messages en clair (plain text)
- \mathcal{C} l'ensemble des messages chiffrés (ou cryptogrammes)
- \mathcal{K} l'ensemble (ou espace) des clés
- E l'ensemble des fonctions de chiffrement $E_k: \mathcal{P} \rightarrow \mathcal{C}$
- D l'ensemble des fonctions de déchiffrement $D_k: \mathcal{C} \rightarrow \mathcal{P}$

Pour nos Tds TPs

Dans nos TD on prendra toujours $P=C=[0..255]^N$ (ou $[0..255]^N$)

Les éléments de P et C seront des instances de Binaire603

Chaque méthode de chiffrement donnera lieu à la création d'une classe dérivée de `codeurCA` (par exemple `ChiffreurAffine`).

K sera donc dans les attributs

E_k et D_k seront dans les méthodes `binCode` et `binDecode`

Pour nos Tds Tps exemple ChiffreurAffine

```
from arithmetiqueDansZ import ElmtZnZ, PGCD
from Binaire603 import Binaire603
from CodeurCA import CodeurCA
class ChiffreurAffine(CodeurCA):
    """Théorie des codes p 150"""
    def __init__(self, a=13, b=5):
        self.a=ElmtZnZ(a, 256)
        self.b=ElmtZnZ(b, 256)
    def __str__(self):
        return f"Chiffreur affine de avec a={self.a} et b={self.b}"
    def __repr__(self):
        return f"ChiffreurAffine({self.a}, {self.b})"
```

Pour nos Tds Tps exemple ChiffreurAffine

| | |
|---------------|-----|
| Phrygane | 300 |
| la | 301 |
| | 302 |
| | 303 |
| dit | 304 |
| | 305 |
| | 306 |
| faute | 307 |
| des quilles | 308 |
| | 309 |
| pro. de | 310 |
| alors | 311 |
| hors | 312 |
| afine pour | 313 |
| auquel | 314 |
| re. vs | 315 |
| elle/plus | 316 |
| ar. | 317 |
| Primmier | 318 |
| | 319 |
| loi | 320 |
| bin ighor | 321 |
| | 322 |
| | 323 |
| re. | 324 |
| Arrière | 325 |
| avec | 326 |
| implément | 327 |
| sur | 328 |
| est | 329 |
| de | 330 |
| Changé | 331 |
| compagnement | 332 |
| multitude | 333 |
| et le | 334 |
| tu | 335 |
| et on | 336 |
| | 337 |
| implément | 338 |
| et le | 339 |
| son. son | 340 |
| | 341 |
| | 342 |
| | 343 |
| unbe | 344 |
| | 345 |
| ge final | 346 |
| devant devant | 347 |
| elle. | 348 |
| avec et | 349 |
| tre. | 350 |

```
def binCode(self, monBinD: Binaire603) -> Binaire603:
    lbc = [(self.a * x + self.b).a for x in monBinD]
    return Binaire603(lbc)

def binDecode(self, monBinC: Binaire603) -> Binaire603:
    assert self.a.estInversible(), ""a doit être inversible
    et donc doit être premier avec 256""
    lb = [(y - self.b) // self.a).a for y in monBinC]
    return Binaire603(lb)
```

Pour nos Tds Tps exemple ChiffreurAffine

```
def demo():
    monBin=Binaire603([0x00,0x01,0x02,0x010,0x20,0x40,0x80])
    for monCodeur in [ChiffreurAffine(3,5),ChiffreurAffine(1,1),
                      ChiffreurAffine(1,0),ChiffreurAffine(2,5)]:
        print(f"Codage avec {monCodeur} :")
        print("    Bin:",monBin)
        monBinC=monCodeur.binCode(monBin)
        print("    Bin Codé:",monBinC)
        monBinD=monCodeur.binDecode(monBinC)
        print("    Bin Décodé:",monBinD)
        print("    monBinD (décodé) est égal à Monbin ?",monBinD==monBin
              )

if __name__ == "__main__":
    import doctest
    doctest.testmod()
    ChiffreurAffine.demo()
```

Différents types d'attaques :

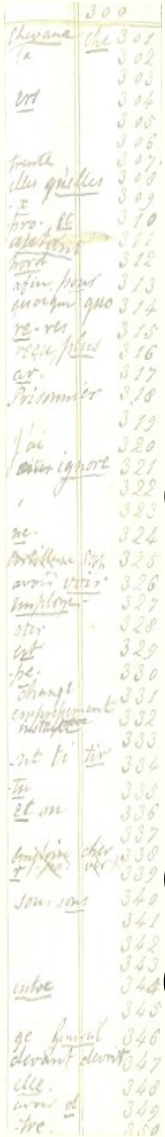
| | |
|---------|-----|
| Théorie | 300 |
| 1.1 | 301 |
| 1.2 | 302 |
| 1.3 | 303 |
| 1.4 | 304 |
| 1.5 | 305 |
| 1.6 | 306 |
| 1.7 | 307 |
| 1.8 | 308 |
| 1.9 | 309 |
| 1.10 | 310 |
| 1.11 | 311 |
| 1.12 | 312 |
| 1.13 | 313 |
| 1.14 | 314 |
| 1.15 | 315 |
| 1.16 | 316 |
| 1.17 | 317 |
| 1.18 | 318 |
| 1.19 | 319 |
| 1.20 | 320 |
| 1.21 | 321 |
| 1.22 | 322 |
| 1.23 | 323 |
| 1.24 | 324 |
| 1.25 | 325 |
| 1.26 | 326 |
| 1.27 | 327 |
| 1.28 | 328 |
| 1.29 | 329 |
| 1.30 | 330 |
| 1.31 | 331 |
| 1.32 | 332 |
| 1.33 | 333 |
| 1.34 | 334 |
| 1.35 | 335 |
| 1.36 | 336 |
| 1.37 | 337 |
| 1.38 | 338 |
| 1.39 | 339 |
| 1.40 | 340 |
| 1.41 | 341 |
| 1.42 | 342 |
| 1.43 | 343 |
| 1.44 | 344 |
| 1.45 | 345 |
| 1.46 | 346 |
| 1.47 | 347 |
| 1.48 | 348 |
| 1.49 | 349 |
| 1.50 | 350 |

On suppose toujours que la méthode de chiffrement/déchiffrement, c.a.d. (P, C, K, E, D) est connue.

De la plus faible à la plus puissante :

- Attaque sur texte chiffré seule sans autres connaissances
- **Attaque à clair connu.** L'attaquant connaît un ou plusieurs couples de messages en clair et leur cryptogramme.
- Attaque à clair choisi. L'attaquant est capable de chiffrer mais pas de déchiffrer.
- Attaque à cryptogramme choisi. L'attaquant est capable de déchiffrer mais pas de chiffrer (systèmes d'authentification).

Attaque sur texte chiffré seule sans autres connaissances



| | |
|-------------|-----|
| Phyvana | 300 |
| la | 301 |
| | 302 |
| | 303 |
| dit | 304 |
| | 305 |
| | 306 |
| franch | 307 |
| des quilles | 308 |
| | 309 |
| pro. de | 310 |
| alors | 311 |
| hors | 312 |
| afin de | 313 |
| auquel | 314 |
| re. de | 315 |
| de. de | 316 |
| ce. | 317 |
| Primitif | 318 |
| | 319 |
| de | 320 |
| auquel | 321 |
| | 322 |
| | 323 |
| de. | 324 |
| Arrière | 325 |
| auquel | 326 |
| auquel | 327 |
| de. | 328 |
| de. | 329 |
| de. | 330 |
| de. | 331 |
| de. | 332 |
| de. | 333 |
| de. | 334 |
| de. | 335 |
| de. | 336 |
| de. | 337 |
| de. | 338 |
| de. | 339 |
| de. | 340 |
| de. | 341 |
| de. | 342 |
| de. | 343 |
| de. | 344 |
| de. | 345 |
| de. | 346 |
| de. | 347 |
| de. | 348 |
| de. | 349 |
| de. | 350 |

C'est l'attaque que l'on voit dans les films....

Elle n'est pas très réaliste dans la réalité.

On commencera en général par une étude statistiques pour engranger un minimum d'informations.

Cryptographie simpliste : le chiffrement de César

C'est le chiffrement des films et du commun des mortels :

César : Bonjour les amis → Cpokpvs mft bnjs

Bonjour les amours → Cpokpvs mft bnpvst

Il ne résistera qu'à votre neveu de 6 ans....

- C'est bien de décaler, mais pas toujours de la même façon !
- Ou alors coder des blocs de lettres : ce sont les méthodes d'aujourd'hui.

Introduction à la Cryptographie — 1. Généralités et concepts de base

Texte en clair :

- + «U»
- + «O»
- + «I»
- + «H»
- + «0»
- + «B»
- + «I»
- + « »
- + «\$»
- + «I»
- + «m»
- + «a»

Texte codé :

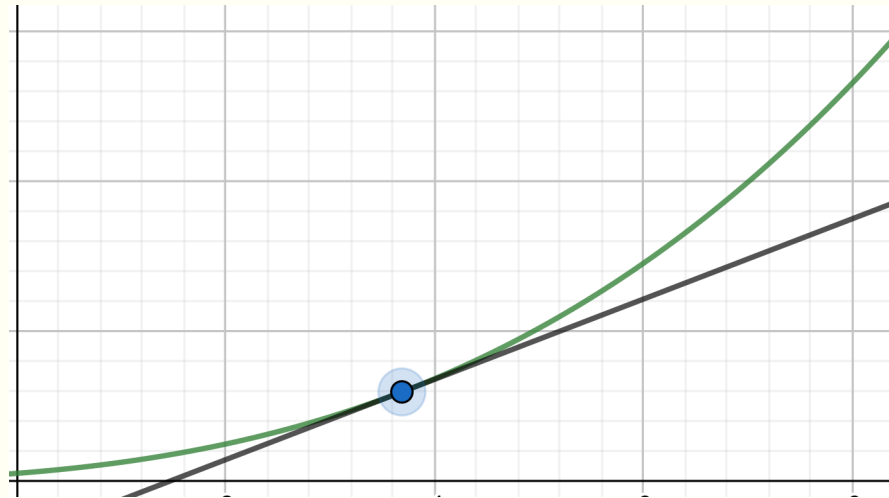
Pour chaque longueur de clé possible, on mène une étude statistique.

Il suffit d'avoir un message assez long.

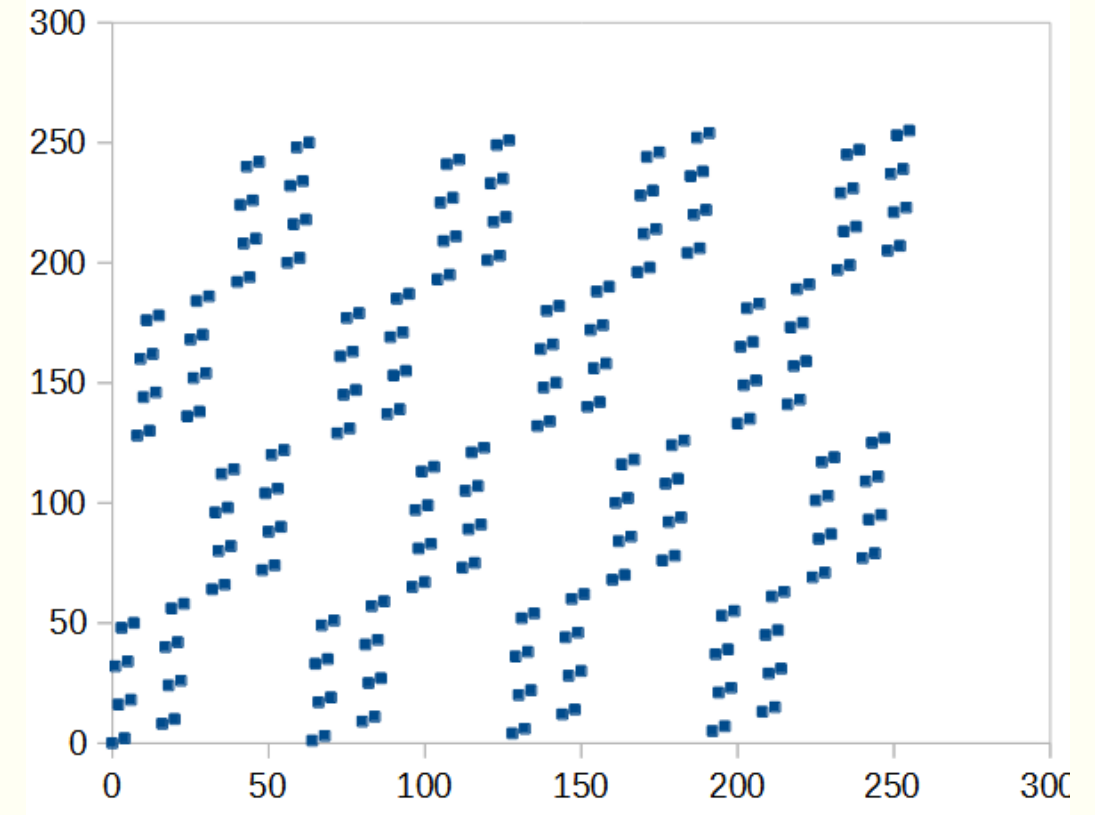
Remplacer un gros bloc par un autre : la difficulté est de ne pas le faire de manière simpliste

Rappel fonctions affines et fonctions dérivables

| | |
|--------------|-----|
| Phrygane | 300 |
| la | 301 |
| | 302 |
| dit | 303 |
| | 304 |
| | 305 |
| | 306 |
| faute | 307 |
| des gâches | 308 |
| | 309 |
| pro. le | 310 |
| alors | 311 |
| hors | 312 |
| afin que | 313 |
| qu'on | 314 |
| re. vs | 315 |
| elle/plus | 316 |
| as. | 317 |
| Primmier | 318 |
| | 319 |
| la | 320 |
| au point | 321 |
| | 322 |
| | 323 |
| re. | 324 |
| Arrière | 325 |
| avec | 326 |
| implément | 327 |
| sur | 328 |
| est | 329 |
| le | 330 |
| tranché | 331 |
| capitulation | 332 |
| multitude | 333 |
| et le | 334 |
| tu | 335 |
| et on | 336 |
| implément | 337 |
| et le | 338 |
| son | 339 |
| son | 340 |
| | 341 |
| | 342 |
| | 343 |
| un | 344 |
| | 345 |
| ge final | 346 |
| devant | 347 |
| elle | 348 |
| avec | 349 |
| tre | 350 |



Codage d'un octet par permutations de bits



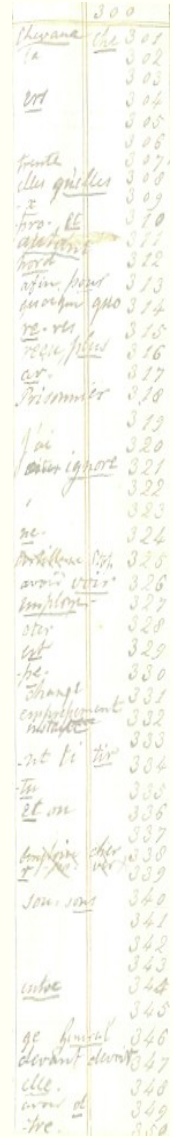
Chiffrement affine : une mauvaise idée !!

$$k=(a,b) \text{ et } c=E_k(m)=a.m+b \text{ et } D_k(c)=a^{-1}.(c-b)$$

Les chiffrements précédents se généralisent en chiffrement affine n'a qu'un petit défaut : il ne résiste pas au traitement à clair connu.

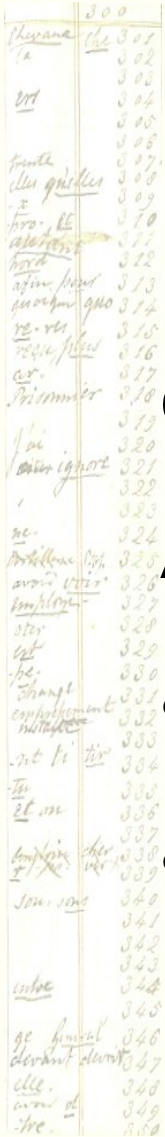
Aucune méthode de chiffrement moderne ne peut donc être linéaire ou même « pas loin » d'être linéaire (penser à « non dérivable »...)

Multiplier les chiffrements linéaires ne sert à rien non plus...



| | |
|---------|-----|
| Therana | 300 |
| la | 301 |
| | 302 |
| | 303 |
| dit | 304 |
| | 305 |
| | 306 |
| | 307 |
| | 308 |
| | 309 |
| | 310 |
| | 311 |
| | 312 |
| | 313 |
| | 314 |
| | 315 |
| | 316 |
| | 317 |
| | 318 |
| | 319 |
| | 320 |
| | 321 |
| | 322 |
| | 323 |
| | 324 |
| | 325 |
| | 326 |
| | 327 |
| | 328 |
| | 329 |
| | 330 |
| | 331 |
| | 332 |
| | 333 |
| | 334 |
| | 335 |
| | 336 |
| | 337 |
| | 338 |
| | 339 |
| | 340 |
| | 341 |
| | 342 |
| | 343 |
| | 344 |
| | 345 |
| | 346 |
| | 347 |
| | 348 |
| | 349 |
| | 350 |

Attaque à clair connu :



| | |
|-------------|-----|
| Phosana | 300 |
| la | 301 |
| | 302 |
| | 303 |
| dit | 304 |
| | 305 |
| | 306 |
| franch | 307 |
| des quilles | 308 |
| | 309 |
| pro. le | 310 |
| alors | 311 |
| hors | 312 |
| afine plus | 313 |
| auquel | 314 |
| re. re | 315 |
| elle plus | 316 |
| or. | 317 |
| Prismier | 318 |
| | 319 |
| la | 320 |
| auquel | 321 |
| | 322 |
| | 323 |
| re. | 324 |
| Arrière | 325 |
| auquel | 326 |
| auquel | 327 |
| or. | 328 |
| et | 329 |
| le. | 330 |
| franch | 331 |
| auquel | 332 |
| auquel | 333 |
| et le | 334 |
| le. | 335 |
| et | 336 |
| auquel | 337 |
| et le | 338 |
| auquel | 339 |
| le. | 340 |
| le. | 341 |
| le. | 342 |
| le. | 343 |
| le. | 344 |
| le. | 345 |
| le. | 346 |
| le. | 347 |
| le. | 348 |
| le. | 349 |
| le. | 350 |

L'attaquant connaît un ou plusieurs couples de messages en clair et leur cryptogramme.

Assez courante :

- La lettre de Charles Quint (mais dont on ne connaissait pas la méthode)
- Midway....

Chiffrement randomisé

| | |
|----------------|-----|
| Phrygane | 300 |
| la | 301 |
| | 302 |
| | 303 |
| dit | 304 |
| | 305 |
| | 306 |
| franch | 307 |
| des guillemets | 308 |
| z | 309 |
| pro. le | 310 |
| alors | 311 |
| hors | 312 |
| afin que | 313 |
| qu'on | 314 |
| re. vs | 315 |
| elle/plus | 316 |
| ce. | 317 |
| Primitif | 318 |
| | 319 |
| la | 320 |
| au point | 321 |
| | 322 |
| | 323 |
| ne. | 324 |
| Arrière | 325 |
| avec | 326 |
| implément | 327 |
| des | 328 |
| est | 329 |
| tranch | 330 |
| capitulum | 331 |
| multiflor | 332 |
| et le | 333 |
| tu | 334 |
| et on | 335 |
| | 336 |
| impair | 337 |
| et/le | 338 |
| | 339 |
| son. son | 340 |
| | 341 |
| | 342 |
| | 343 |
| unbe | 344 |
| | 345 |
| ge final | 346 |
| derrière | 347 |
| elle. | 348 |
| avec | 349 |
| tre. | 350 |

Pour éviter une attaque à clair connu, un chiffrement doit donc être en général **randomisé** afin que deux messages identiques ne produisent jamais le même cryptogramme.

| | |
|--------------|-----|
| Théâtre | 300 |
| la | 301 |
| | 302 |
| | 303 |
| dit | 304 |
| | 305 |
| | 306 |
| faute | 307 |
| des quilles | 308 |
| | 309 |
| pro. le | 310 |
| alors | 311 |
| hard | 312 |
| afin | 313 |
| qu'on | 314 |
| re. ve | 315 |
| elle. l'air | 316 |
| ce. | 317 |
| Primitif | 318 |
| | 319 |
| la | 320 |
| au. igne | 321 |
| | 322 |
| | 323 |
| re. | 324 |
| Arrière | 325 |
| avec | 326 |
| implant | 327 |
| sur | 328 |
| est | 329 |
| de. | 330 |
| Chant | 331 |
| capitulation | 332 |
| multitude | 333 |
| et le | 334 |
| le | 335 |
| et on | 336 |
| | 337 |
| impair | 338 |
| et le | 339 |
| son. son | 340 |
| | 341 |
| | 342 |
| | 343 |
| unbe | 344 |
| | 345 |
| ge. l'air | 346 |
| devant | 347 |
| elle. | 348 |
| avec | 349 |
| tre. | 350 |

Chiffrement de Vernam

La clé K est de même longueur que le message m et $c = m \oplus K$

Le chiffrement de Vernam possède un avantage unique :
Si K est purement aléatoire, alors il n'y a aucun espoir de retrouver (m) à partir de (c).

En effet les probabilités de sortie de chaque bit de c sont alors indépendantes de celles de m.

Le chiffrement de Vernam possède un inconvénient : pour chiffrer 1Mo de messages, il faut 1Mo de clé.

Quand on est à court de clé... on recycle : mauvaise idée !

Cryptographie mécanique

1883 Kerckoffs formalise les systèmes de chiffrement :
« la sécurité d'un cryptosystème ne doit pas reposer que sur le secret de la clef »

1926 Chiffrement de Vernam (masque jetable)

1940 : Enigma



| | |
|----------------|-----|
| Phonema | 300 |
| la | 301 |
| | 302 |
| | 303 |
| dit | 304 |
| | 305 |
| | 306 |
| | 307 |
| faute | 308 |
| des quilles | 309 |
| F | 310 |
| pro. le | 311 |
| aléatoire | 312 |
| hard | 313 |
| afine plus | 314 |
| quatre | 315 |
| re. vs | 316 |
| re. plus | 317 |
| cr. | 318 |
| Prismier | 319 |
| | 320 |
| la | 321 |
| au. ignoz | 322 |
| | 323 |
| re. | 324 |
| Arrière | 325 |
| av. 122 | 326 |
| implom | 327 |
| ster | 328 |
| let | 329 |
| be. | 330 |
| Thaupt | 331 |
| expériment | 332 |
| nutte | 333 |
| nt. le. tir | 334 |
| tu | 335 |
| et. m. | 336 |
| | 337 |
| implom. cher | 338 |
| et. le. | 339 |
| son. son | 340 |
| | 341 |
| | 342 |
| | 343 |
| unbe | 344 |
| | 345 |
| ge. finial | 346 |
| derant. elevat | 347 |
| elle. | 348 |
| un. d. | 349 |
| tre. | 350 |

Enigma



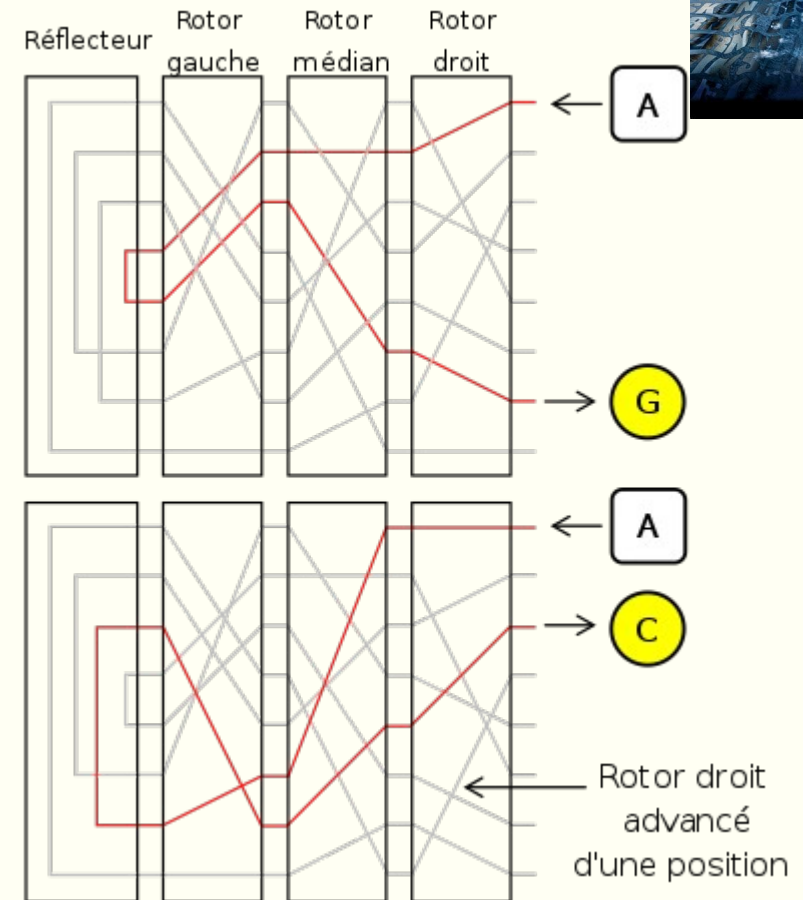
La machine Enigma reprends le codage de Vernam mais la clé est générée par les positions successives de rotors.

On peut chiffrer facilement mais déchiffrer nécessite une puissance de calcul inatteignable à la main....

Les cryptanalystes britanniques, dont Alan Turing, continuèrent les travaux du mathématicien polonais Marian Rejewski. Ils furent par la suite, dans des circonstances favorables et pendant des intervalles de temps plus ou moins longs, capables de déchiffrer les messages Enigma en perfectionnant les « bombes électromécaniques », inventées et mises au point par Rejewski.

Simulateur :

http://www.museedelaresistanceenligne.org/pedago_espace.php?atelier=d&popin=true



On estime que requérir 2^{128} opérations représente aujourd'hui un niveau raisonnable de sécurité (« limite de l'infaisable »). Elle était de 2^{80} en l'an 2000...

En 2021 le superordinateur Fugaku développe 418PFlops ($\sim 2^{84}$ opérations en virgule flottante par an)

<https://www.parismatch.com/Vivre/High-Tech/Voici-Fugaku-l-ordinateur-le-plus-puissant-du-monde-1694491>

La NSA utilise et développe des superordinateurs, elle investit aussi dans les ordinateurs quantiques...

Confidentialité parfaite : La connaissance du message chiffré n'apporte aucune information sur le message clair.

Pour qu'un chiffrement soit inconditionnellement sûr, il faut que la clé soit aléatoire et aussi longue que le texte clair.

Mais un algorithme de chiffrement exigeant, pour être déchiffré, une puissance de calcul inateignable est lui aussi assez sûr...

Par exemple, nécessiter deux fois toute l'énergie disponible sur Terre, est une puissance de calcul inateignable !

Attaque à clair choisi :

| | |
|----------|-----|
| Théorème | 300 |
| 1a | 301 |
| 302 | 302 |
| 303 | 303 |
| 304 | 304 |
| 305 | 305 |
| 306 | 306 |
| 307 | 307 |
| 308 | 308 |
| 309 | 309 |
| 310 | 310 |
| 311 | 311 |
| 312 | 312 |
| 313 | 313 |
| 314 | 314 |
| 315 | 315 |
| 316 | 316 |
| 317 | 317 |
| 318 | 318 |
| 319 | 319 |
| 320 | 320 |
| 321 | 321 |
| 322 | 322 |
| 323 | 323 |
| 324 | 324 |
| 325 | 325 |
| 326 | 326 |
| 327 | 327 |
| 328 | 328 |
| 329 | 329 |
| 330 | 330 |
| 331 | 331 |
| 332 | 332 |
| 333 | 333 |
| 334 | 334 |
| 335 | 335 |
| 336 | 336 |
| 337 | 337 |
| 338 | 338 |
| 339 | 339 |
| 340 | 340 |
| 341 | 341 |
| 342 | 342 |
| 343 | 343 |
| 344 | 344 |
| 345 | 345 |
| 346 | 346 |
| 347 | 347 |
| 348 | 348 |
| 349 | 349 |
| 350 | 350 |

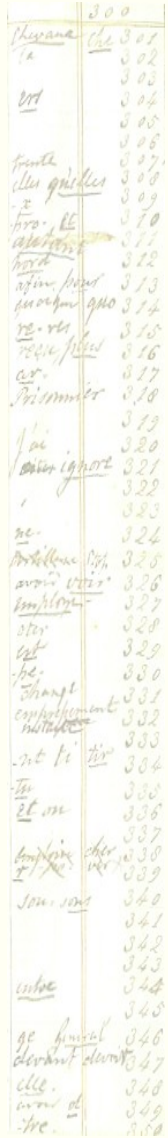
L'attaquant est capable de chiffrer mais pas de déchiffrer :

- Enigma
- SSL : Les données échangées sont chiffrées avec la « clé de session » commune au client et au serveur. (https://www.f5.com/fr_fr/services/resources/glossary/ssl-tls-encryption)

C'est celle de beaucoup des échanges actuels puisque les algorithmes d'internet sont open source.

Si la clé est simple ou mal choisie, l'attaquant n'a qu'à tester des clés jusqu'à déchiffrer le message.

- Echanges de clés
- Transactions du BitCoins



| | |
|------------|-----|
| Therana | 300 |
| la | 301 |
| | 302 |
| | 303 |
| dit | 304 |
| | 305 |
| | 306 |
| | 307 |
| faute | 308 |
| des gâches | 309 |
| | 310 |
| pro. le | 311 |
| alors | 312 |
| hors | 313 |
| afin que | 314 |
| qu'on | 315 |
| re. vs | 316 |
| elle. plus | 317 |
| cr. | 318 |
| Primitif | 319 |
| | 320 |
| au. igne | 321 |
| | 322 |
| | 323 |
| re. | 324 |
| Arrière | 325 |
| avec | 326 |
| imp. l'or | 327 |
| sur | 328 |
| est | 329 |
| de. | 330 |
| Thant | 331 |
| exp. l'or | 332 |
| l'or | 333 |
| et le | 334 |
| le | 335 |
| et on | 336 |
| | 337 |
| imp. l'or | 338 |
| et le | 339 |
| son. son | 340 |
| | 341 |
| | 342 |
| | 343 |
| un. son | 344 |
| | 345 |
| ge. l'or | 346 |
| l'or. l'or | 347 |
| elle. | 348 |
| un. son | 349 |
| tre. | 350 |

Chiffrement par bloc

Si le chiffrement de Vernam n'est pas utilisable on peut essayer de chiffrer par bloc.

Mais on l'a vu, il n'est pas question d'un chiffrement des blocs presque linéaire, Il faut un chiffrement avec beaucoup de **Confusion** et de **Diffusion** (Shannon) .

Confusion : Distribution statistique complexe des blocs chiffrés

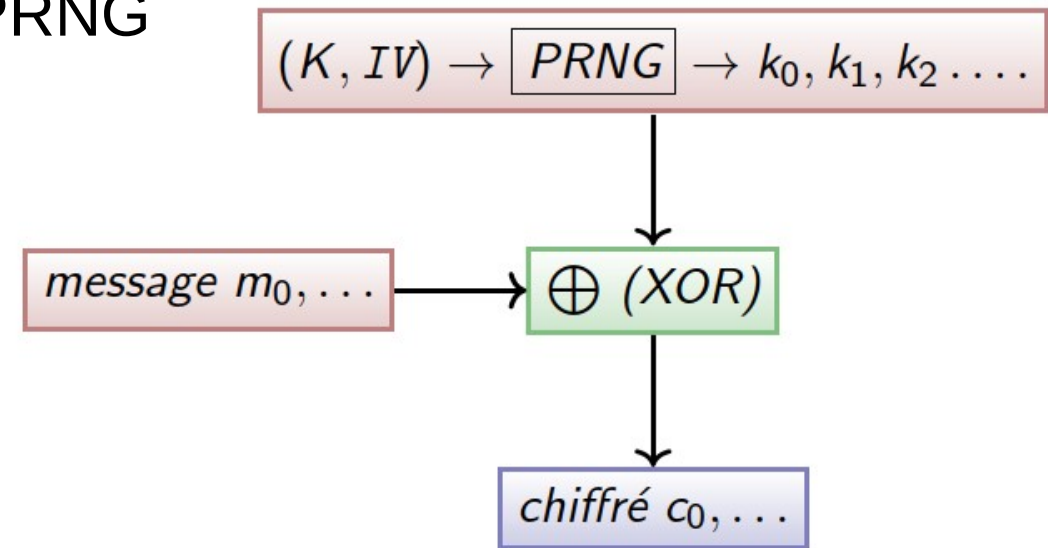
Diffusion : Chaque bit du message et chaque bit de clé a une influence sur chaque bit du cryptogramme.

Chiffrement par Bloc

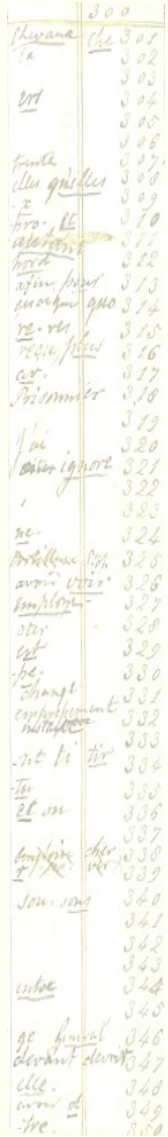
Première méthode : On utilise un générateur de nombre pseudo-aléatoire (PRNG).

La clé de chiffrement est alors utilisée comme graine.

Le message est découpé en bloc et chaque bloc est chiffré par un XOR avec un nombre généré par le PRNG



Chiffrement par bloc



| | |
|-------------|-----|
| Therese | 300 |
| la | 301 |
| | 302 |
| | 303 |
| dit | 304 |
| | 305 |
| | 306 |
| franch | 307 |
| des quilles | 308 |
| | 309 |
| | 310 |
| pro. de | 311 |
| alors | 312 |
| hard | 313 |
| afine pour | 314 |
| au sein | 315 |
| re. de | 316 |
| re. de | 317 |
| cr. | 318 |
| Primitif | 319 |
| | 320 |
| la | 321 |
| au sein | 322 |
| | 323 |
| re. | 324 |
| Arrière | 325 |
| au sein | 326 |
| Am. de | 327 |
| dit | 328 |
| dit | 329 |
| dit | 330 |
| dit | 331 |
| dit | 332 |
| dit | 333 |
| dit | 334 |
| dit | 335 |
| dit | 336 |
| dit | 337 |
| dit | 338 |
| dit | 339 |
| dit | 340 |
| dit | 341 |
| dit | 342 |
| dit | 343 |
| dit | 344 |
| dit | 345 |
| dit | 346 |
| dit | 347 |
| dit | 348 |
| dit | 349 |
| dit | 350 |

Les fonctions de chiffrement par bloc sont des permutations

Mais il est difficile de représenter et évaluer une permutation arbitraire. On peut par exemple se contenter d'utiliser un chiffrement par permutation des positions.

Dans certains cas validés mathématiquement, on peut répéter le chiffrement pour agrandir l'espace des clés (mais pas valable pour les permutations de position !!)

| | |
|--------------|-----|
| Phéonax | 300 |
| la | 301 |
| | 302 |
| | 303 |
| dit | 304 |
| | 305 |
| | 306 |
| | 307 |
| faute | 308 |
| des quilles | 309 |
| | 310 |
| pro. le | 311 |
| alors | 312 |
| hard | 313 |
| afine pour | 314 |
| auoqun 940 | 315 |
| re. ve | 316 |
| elle/plus | 317 |
| or. | 318 |
| Prismier | 319 |
| | 320 |
| l'air ignore | 321 |
| | 322 |
| | 323 |
| re. | 324 |
| Arrière | 325 |
| avon 1722 | 326 |
| implon | 327 |
| ster | 328 |
| let | 329 |
| be. | 330 |
| Shant | 331 |
| capitulation | 332 |
| nutritions | 333 |
| nt le tir | 334 |
| tu | 335 |
| et on | 336 |
| | 337 |
| impair | 338 |
| et/le | 339 |
| son. son | 340 |
| | 341 |
| | 342 |
| | 343 |
| unbe | 344 |
| | 345 |
| ge final | 346 |
| devant d'au | 347 |
| elle. | 348 |
| mon. d | 349 |
| tre. | 350 |

Chiffrement par bloc : Feistel

Chiffrement de Feistel

avec r tours

et les clés K_1, K_2, \dots, K_r

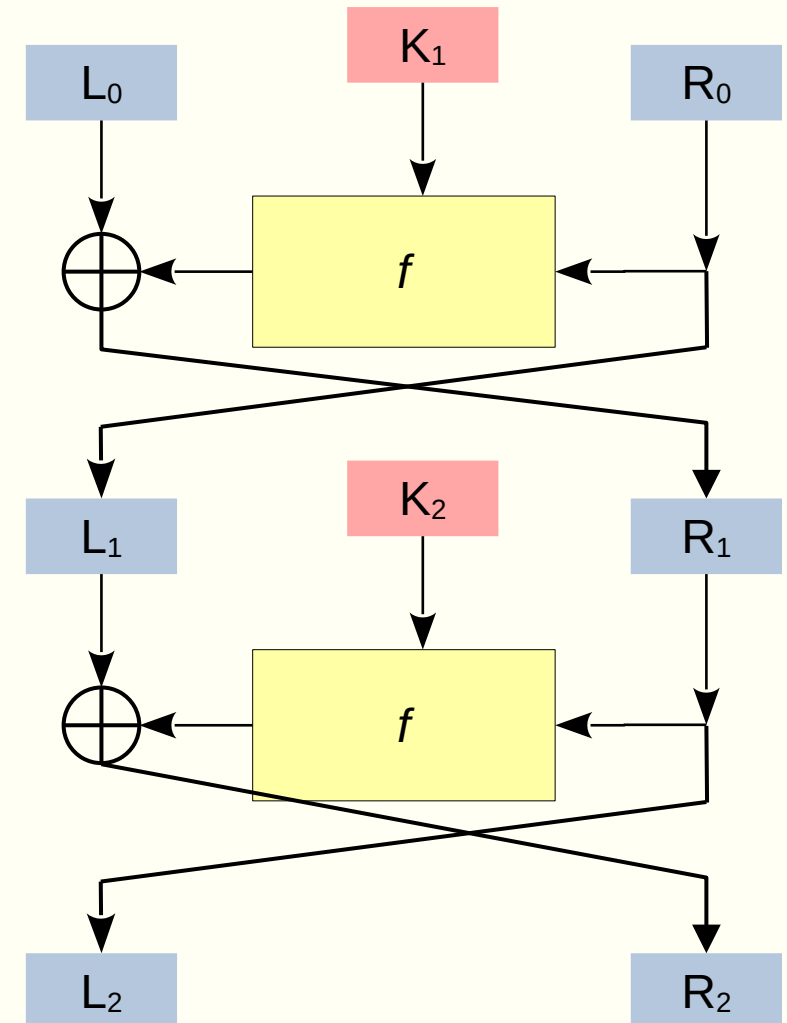
générées par la clé initiale K :

Octet $i = (L_i, R_i) = (R_{i-1}, L_{i-1} \oplus f_{K_i}(R_{i-1}))$

et donc $E_K(m) = E_K(L_0, R_0) = (R_r, L_r) = c$

Et pour le déchiffrement :

$(R_{i-1}, L_{i-1}) = (L_i, R_i \oplus f_{K_i}(L_i))$



Le DES

| | |
|---------|-----|
| Théorie | 300 |
| 1.1 | 301 |
| 1.2 | 302 |
| 1.3 | 303 |
| 1.4 | 304 |
| 1.5 | 305 |
| 1.6 | 306 |
| 1.7 | 307 |
| 1.8 | 308 |
| 1.9 | 309 |
| 1.10 | 310 |
| 1.11 | 311 |
| 1.12 | 312 |
| 1.13 | 313 |
| 1.14 | 314 |
| 1.15 | 315 |
| 1.16 | 316 |
| 1.17 | 317 |
| 1.18 | 318 |
| 1.19 | 319 |
| 1.20 | 320 |
| 1.21 | 321 |
| 1.22 | 322 |
| 1.23 | 323 |
| 1.24 | 324 |
| 1.25 | 325 |
| 1.26 | 326 |
| 1.27 | 327 |
| 1.28 | 328 |
| 1.29 | 329 |
| 1.30 | 330 |
| 1.31 | 331 |
| 1.32 | 332 |
| 1.33 | 333 |
| 1.34 | 334 |
| 1.35 | 335 |
| 1.36 | 336 |
| 1.37 | 337 |
| 1.38 | 338 |
| 1.39 | 339 |
| 1.40 | 340 |
| 1.41 | 341 |
| 1.42 | 342 |
| 1.43 | 343 |
| 1.44 | 344 |
| 1.45 | 345 |
| 1.46 | 346 |
| 1.47 | 347 |
| 1.48 | 348 |
| 1.49 | 349 |
| 1.50 | 350 |

Le DES est un standard aujourd'hui obsolète.
Afin « de mettre les mains dans le cambouis »,
nous allons cependant détailler cet algorithme à partir d'un excellent livre.

Source : Introduction à la cryptographie Dunod

5.2 ALGORITHME DU DES

Le DES est un chiffre de Feistel légèrement modifié avec l'alphabet $\{0, 1\}$ et la longueur des blocs 64. Dans cette section, nous donnons le détail de son fonctionnement.

5.2.1 Espaces des messages en clair et des cryptogrammes

L'espace des messages en clair et l'espace des cryptogrammes du DES sont $\mathcal{P} = \mathcal{C} = \{0, 1\}^{64}$. Les clés du DES sont les mots binaires de longueur 64 qui ont la propriété suivante : quand une clé de 64 bits du DES est divisée en 8 octets la somme des 8 bits de chaque octet est impaire. Cela fait que 7 des 8 bits de l'octet déterminent la valeur du 8^e bit et on peut donc détecter les erreurs de transmission sur un bit. Il résulte de cette condition que l'espace des clés est

$$\mathcal{K} = \left\{ (b_1, \dots, b_{64}) \in \{0, 1\}^{64} : \sum_{i=1}^8 b_{8k+i} \equiv 1 \pmod{2} \text{ avec } 0 \leq k \leq 7 \right\}$$

Le nombre de clés du DES est $2^{56} \sim 7.2 * 10^{16}$.

Exemple 5.2.1 Une clé hexadécimale valable pour le DES est :

133457799BBCDFF1.

Son développement binaire peut se lire dans le tableau 5.1.

TABLEAU 5.1. Une clé valable pour le DES

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |

5.2.2 Permutation initiale

Pour chiffrer un message p , le DES passe par trois étapes.

Avant le chiffrement de Feistel, le DES applique une *permutation initiale* (PI) à p . C'est une permutation de bits sur un vecteur de bits de longueur 64 qui ne dépend pas de la clé choisie. La permutation PI et son inverse sont définies dans le tableau 5.2. Ce tableau doit être lu de la façon suivante : si $p \in \{0, 1\}^{64}$, $p = p_1 p_2 p_3 \dots p_{64}$, alors $PI(p) = p_{58} p_{50} p_{42} \dots p_7$.

TABLEAU 5.2. PI , la permutation initiale

| PI | | | | | | | | PI^{-1} | | | | | | | |
|------|----|----|----|----|----|----|---|-----------|---|----|----|----|----|----|----|
| 58 | 50 | 42 | 34 | 26 | 18 | 10 | 2 | 40 | 8 | 48 | 16 | 56 | 24 | 64 | 32 |
| 60 | 52 | 44 | 36 | 28 | 20 | 12 | 4 | 39 | 7 | 47 | 15 | 55 | 23 | 63 | 31 |
| 62 | 54 | 46 | 38 | 30 | 22 | 14 | 6 | 38 | 6 | 46 | 14 | 54 | 22 | 62 | 30 |
| 64 | 56 | 48 | 40 | 32 | 24 | 16 | 8 | 37 | 5 | 45 | 13 | 53 | 21 | 61 | 29 |
| 57 | 49 | 41 | 33 | 25 | 17 | 9 | 1 | 36 | 4 | 44 | 12 | 52 | 20 | 60 | 28 |
| 59 | 51 | 43 | 35 | 27 | 19 | 11 | 3 | 35 | 3 | 43 | 11 | 51 | 19 | 59 | 27 |
| 61 | 53 | 45 | 37 | 29 | 21 | 13 | 5 | 34 | 2 | 42 | 10 | 50 | 18 | 58 | 26 |
| 63 | 55 | 47 | 39 | 31 | 23 | 15 | 7 | 33 | 1 | 41 | 9 | 49 | 17 | 57 | 25 |

Un chiffre de Feistel à 16 tournées est appliqué une fois que le message en clair est permuté, après quoi, le cryptogramme est définitivement calculé en appliquant la permutation inverse PI^{-1} :

$$c = PI^{-1}(R_{16} L_{16})$$

5.2.3 Chiffrement interne

Nous décrivons le chiffrement par bloc sur lequel s'appuie le chiffre de Feistel du DES. Son alphabet est $\{0, 1\}$, la longueur des blocs est 32 et son espace des clés est $\{0, 1\}^{48}$. Nous expliquons la fonction de chiffrement $f_K : \{0, 1\}^{32} \rightarrow \{0, 1\}^{32}$ associée à la clé $K \in \{0, 1\}^{48}$ (Figure 5.3).

L'argument, $R \in \{0, 1\}^{32}$, est d'abord allongé au moyen de la *fonction de développement* $E : \{0, 1\}^{32} \rightarrow \{0, 1\}^{48}$, définie dans le tableau 5.4. Pour résumer, si $R = R_1 R_2 \dots R_{32}$, alors $E(R) = R_{32} R_1 R_2 \dots R_{32} R_1$.

Ensuite, on calcule $E(R) \oplus K$ et le résultat est découpé en 8 blocs B_i de longueur 6. ce qui donne

$$E(R) \oplus K = B_1 B_2 B_3 B_4 B_5 B_6 B_7 B_8 \quad (5.3)$$

avec $B_i \in \{0, 1\}^6$. Dans l'étape suivante, on utilise 8 fonctions

$$S_i : \{0, 1\}^6 \rightarrow \{0, 1\}^4 \quad 1 \leq i \leq 8$$

On les appelle les *S-box* et elles seront décrites plus loin. En calculant $C_i = S_i(B_i)$ au moyen de ces fonctions, on obtient $C = C_1 C_2 C_3 C_4 C_5 C_6 C_7 C_8$, un mot binaire de longueur 32. Enfin la permutation P du tableau 5.4 est appliquée à la chaîne de caractères C et le mot binaire de longueur 32 que l'on obtient à cet instant est le cryptogramme $f_K(R)$.

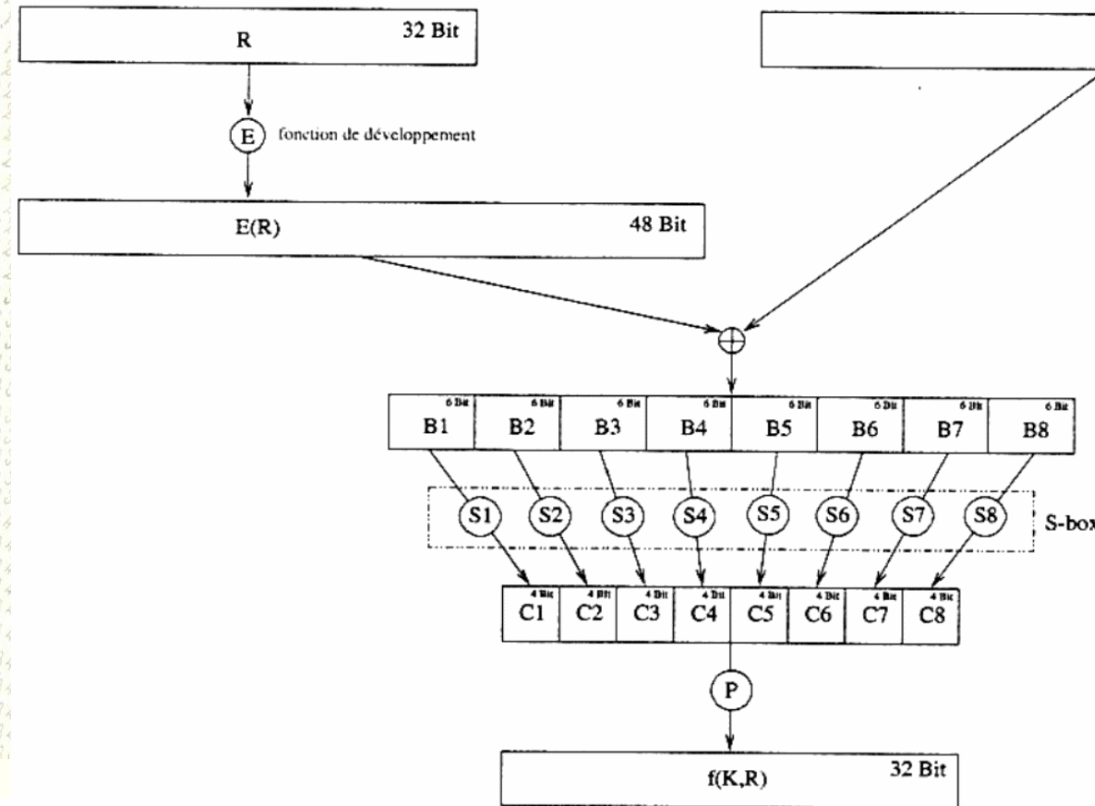


FIGURE 5.3. La fonction f du DES

TABLEAU 5.4. Les fonctions E et P

| | E | | | | |
|----|-----|----|----|----|----|
| 32 | 1 | 2 | 3 | 4 | 5 |
| 4 | 5 | 6 | 7 | 8 | 9 |
| 8 | 9 | 10 | 11 | 12 | 13 |
| 12 | 13 | 14 | 15 | 16 | 17 |
| 16 | 17 | 18 | 19 | 20 | 21 |
| 20 | 21 | 22 | 23 | 24 | 25 |
| 24 | 25 | 26 | 27 | 28 | 29 |
| 28 | 29 | 30 | 31 | 32 | 1 |

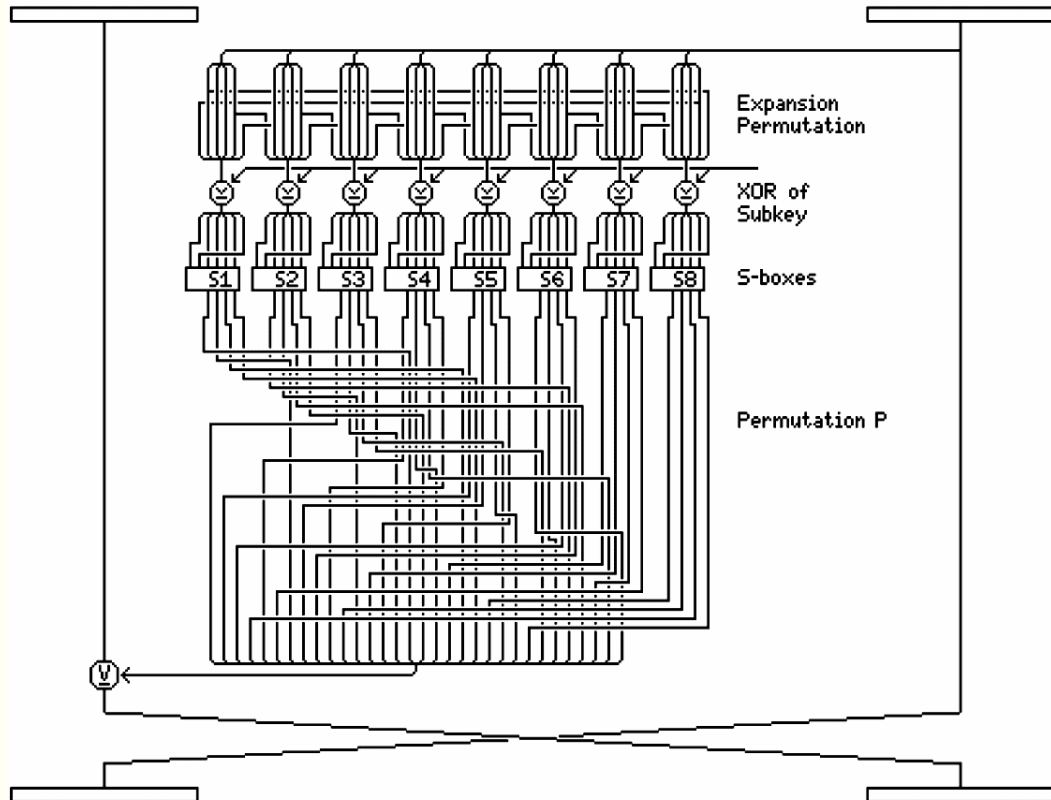
| | P | | | |
|----|-----|----|----|--|
| 16 | 7 | 20 | 21 | |
| 29 | 12 | 28 | 17 | |
| 1 | 15 | 23 | 26 | |
| 5 | 18 | 31 | 10 | |
| 2 | 8 | 24 | 14 | |
| 32 | 27 | 3 | 9 | |
| 19 | 13 | 30 | 6 | |
| 22 | 11 | 4 | 25 | |

5.2.4 S-box

Maintenant nous décrivons les S -box S_i , avec $1 \leq i \leq 8$. Elles sont au cœur du DES parce que ce sont des fonctions hautement non linéaires (exercice 5.6). Le tableau 5.5 en donne la définition. Chaque S -box est représentée par un tableau de 4 lignes numérotées de [0] à [3] et 16 colonnes numérotées de [0] à [15]. Si $B = b_1b_2b_3b_4b_5b_6$, le mot binaire $S_i(B)$ est calculé de la façon suivante. L'entier qui a pour développement binaire b_1b_6 est utilisé comme indice de ligne, celui qui a pour développement binaire $b_2b_3b_4b_5$ est utilisé comme indice de colonne. À l'intersection de cette ligne et de cette colonne figure un nombre entier dans le tableau de S_i ; on écrit cet entier en binaire et s'il le faut, on place des 0 en tête du développement pour obtenir un mot binaire de longueur 4; c'est ce mot binaire qui est $S_i(B)$.

TABLEAU 5.5. Les S -box du DES

| Lignes | Colonnes | | | | | | | | | | | | | | | |
|--------|----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|
| | [0] | [1] | [2] | [3] | [4] | [5] | [6] | [7] | [8] | [9] | [10] | [11] | [12] | [13] | [14] | [15] |
| S_1 | | | | | | | | | | | | | | | | |
| [0] | 14 | 4 | 13 | 1 | 2 | 15 | 11 | 8 | 3 | 10 | 6 | 12 | 5 | 9 | 0 | 7 |
| [1] | 0 | 15 | 7 | 4 | 14 | 2 | 13 | 1 | 10 | 6 | 12 | 11 | 9 | 5 | 3 | 8 |
| [2] | 4 | 1 | 14 | 8 | 13 | 6 | 2 | 11 | 15 | 12 | 9 | 7 | 3 | 10 | 5 | 0 |
| [3] | 15 | 12 | 8 | 2 | 4 | 9 | 1 | 7 | 5 | 11 | 3 | 14 | 10 | 0 | 6 | 13 |
| S_2 | | | | | | | | | | | | | | | | |
| [0] | 15 | 1 | 8 | 14 | 6 | 11 | 3 | 4 | 9 | 7 | 2 | 13 | 12 | 0 | 5 | 10 |
| [1] | 3 | 13 | 4 | 7 | 15 | 2 | 8 | 14 | 12 | 0 | 1 | 10 | 6 | 9 | 11 | 5 |
| [2] | 0 | 14 | 7 | 11 | 10 | 4 | 13 | 1 | 5 | 8 | 12 | 6 | 9 | 3 | 2 | 15 |
| [3] | 13 | 8 | 10 | 1 | 3 | 15 | 4 | 2 | 11 | 6 | 7 | 12 | 0 | 5 | 14 | 9 |
| S_3 | | | | | | | | | | | | | | | | |
| [0] | 10 | 0 | 9 | 14 | 6 | 3 | 15 | 5 | 1 | 13 | 12 | 7 | 11 | 4 | 2 | 8 |
| [1] | 13 | 7 | 0 | 9 | 3 | 4 | 6 | 10 | 2 | 8 | 5 | 14 | 12 | 11 | 15 | 1 |
| [2] | 13 | 6 | 4 | 9 | 8 | 15 | 3 | 0 | 11 | 1 | 2 | 12 | 5 | 10 | 14 | 7 |
| [3] | 1 | 10 | 13 | 0 | 6 | 9 | 8 | 7 | 4 | 15 | 14 | 3 | 11 | 5 | 2 | 12 |
| S_4 | | | | | | | | | | | | | | | | |
| [0] | 7 | 13 | 14 | 3 | 0 | 6 | 9 | 10 | 1 | 2 | 8 | 5 | 11 | 12 | 4 | 15 |
| [1] | 13 | 8 | 11 | 5 | 6 | 15 | 0 | 3 | 4 | 7 | 2 | 12 | 1 | 10 | 14 | 9 |
| [2] | 10 | 6 | 9 | 0 | 12 | 11 | 7 | 13 | 15 | 1 | 3 | 14 | 5 | 2 | 8 | 4 |
| [3] | 3 | 15 | 0 | 6 | 10 | 1 | 13 | 8 | 9 | 4 | 5 | 11 | 12 | 7 | 2 | 14 |
| S_5 | | | | | | | | | | | | | | | | |
| [0] | 2 | 12 | 4 | 1 | 7 | 10 | 11 | 6 | 8 | 5 | 3 | 15 | 13 | 0 | 14 | 9 |
| [1] | 14 | 11 | 2 | 12 | 4 | 7 | 13 | 1 | 5 | 0 | 15 | 10 | 3 | 9 | 8 | 6 |
| [2] | 4 | 2 | 1 | 11 | 10 | 13 | 7 | 8 | 15 | 9 | 12 | 5 | 6 | 3 | 0 | 14 |
| [3] | 11 | 8 | 12 | 7 | 1 | 14 | 2 | 13 | 6 | 15 | 0 | 9 | 10 | 4 | 5 | 3 |
| S_6 | | | | | | | | | | | | | | | | |
| [0] | 12 | 1 | 10 | 15 | 9 | 2 | 6 | 8 | 0 | 13 | 3 | 4 | 14 | 7 | 5 | 11 |
| [1] | 10 | 15 | 4 | 2 | 7 | 12 | 9 | 5 | 6 | 1 | 13 | 14 | 0 | 11 | 3 | 8 |
| [2] | 9 | 14 | 15 | 5 | 2 | 8 | 12 | 3 | 7 | 0 | 4 | 10 | 1 | 13 | 11 | 6 |
| [3] | 4 | 3 | 2 | 12 | 9 | 5 | 15 | 10 | 11 | 14 | 1 | 7 | 6 | 0 | 8 | 13 |
| S_7 | | | | | | | | | | | | | | | | |
| [0] | 4 | 11 | 2 | 14 | 15 | 0 | 8 | 13 | 3 | 12 | 9 | 7 | 5 | 10 | 6 | 1 |
| [1] | 13 | 0 | 11 | 7 | 4 | 9 | 1 | 10 | 14 | 3 | 5 | 12 | 2 | 15 | 8 | 6 |
| [2] | 1 | 4 | 11 | 13 | 12 | 3 | 7 | 14 | 10 | 15 | 6 | 8 | 0 | 5 | 9 | 2 |
| [3] | 6 | 11 | 13 | 8 | 1 | 4 | 10 | 7 | 9 | 5 | 0 | 15 | 14 | 2 | 3 | 12 |
| S_8 | | | | | | | | | | | | | | | | |
| [0] | 13 | 2 | 8 | 4 | 6 | 15 | 11 | 1 | 10 | 9 | 3 | 14 | 5 | 0 | 12 | 7 |
| [1] | 1 | 15 | 13 | 8 | 10 | 3 | 7 | 4 | 12 | 5 | 6 | 11 | 0 | 14 | 9 | 2 |
| [2] | 7 | 11 | 4 | 1 | 9 | 12 | 14 | 2 | 0 | 6 | 10 | 13 | 15 | 3 | 5 | 8 |
| [3] | 2 | 1 | 14 | 7 | 4 | 10 | 8 | 13 | 15 | 12 | 9 | 0 | 3 | 5 | 6 | 11 |



5.2.5 Clés

Finalement, il reste à expliquer comment sont calculées les chaînes de clés des tournées. Soit $k \in \{0, 1\}^{64}$ une clé du DES, on va en déduire les clés des tournées K_i , $1 \leq i \leq 16$, de longueur 48. Pour cela, on définit des valeurs v_i , $1 \leq i \leq 16$, de la façon suivante.

$$v_i = \begin{cases} 1 & \text{pour } i \in \{1, 2, 9, 16\}, \\ 2 & \text{sinon.} \end{cases}$$

Les clés des tournées sont calculées par l'algorithme suivant qui utilise deux fonctions

$$\text{PC1} : \{0, 1\}^{64} \rightarrow \{0, 1\}^{28} \times \{0, 1\}^{28} \quad \text{PC2} : \{0, 1\}^{28} \times \{0, 1\}^{28} \rightarrow \{0, 1\}^{48}$$

qui seront décrites plus loin.

1. On pose $(C_0, D_0) = \text{PC1}(k)$.

2. Pour $1 \leq i \leq 16$:

- On note C_i la chaîne de caractères obtenue à partir de C_{i-1} par un décalage, en permutation circulaire, de v_i positions.
- On note D_i la chaîne de caractères obtenue à partir de D_{i-1} par un décalage, en permutation circulaire, de v_i positions.
- On pose $K_i = \text{PC2}(C_i, D_i)$.

La fonction PC1 associe au mot binaire k de longueur 64 deux mots binaires C et D de longueur 28. Elle est définie par le tableau 5.6. La moitié supérieure de la table décrit C . Si $k = k_1 k_2 \dots k_{64}$, alors $C = k_{57} k_{49} \dots k_{36}$. La moitié inférieure de la table représente D , donc $D = k_{63} k_{55} \dots k_4$. La fonction PC2 associe à un couple (C, D) de mots binaires de longueur 28, c'est-à-dire à un mot binaire de longueur 56, un mot binaire de longueur 48. Cette fonction est définie dans le tableau 5.6. La valeur $\text{PC2}(b_1 \dots b_{56})$ est $b_{14} b_{17} \dots b_{32}$.

TABLEAU 5.6. Les fonctions PC1 et PC2

| PC1 | | | | | | |
|-----|----|----|----|----|----|----|
| 57 | 49 | 41 | 33 | 25 | 17 | 9 |
| 1 | 58 | 50 | 42 | 34 | 26 | 18 |
| 10 | 2 | 59 | 51 | 43 | 35 | 27 |
| 19 | 11 | 3 | 60 | 52 | 44 | 36 |
| 63 | 55 | 47 | 39 | 31 | 23 | 15 |
| 7 | 62 | 54 | 46 | 38 | 30 | 22 |
| 14 | 6 | 61 | 53 | 45 | 37 | 29 |
| 21 | 13 | 5 | 28 | 20 | 12 | 4 |

| PC2 | | | | | |
|-----|----|----|----|----|----|
| 14 | 17 | 11 | 24 | 1 | 5 |
| 3 | 28 | 15 | 6 | 21 | 10 |
| 23 | 19 | 12 | 4 | 26 | 8 |
| 16 | 7 | 27 | 20 | 13 | 2 |
| 41 | 52 | 31 | 37 | 47 | 55 |
| 30 | 40 | 51 | 45 | 33 | 48 |
| 44 | 49 | 39 | 56 | 34 | 53 |
| 46 | 42 | 50 | 36 | 29 | 32 |

Ceci termine la description de l'algorithme de chiffrement du DES.

5.2.6 Déchiffrement

Pour déchiffrer un cryptogramme, on applique DES en utilisant la même suite de clés, mais en ordre inverse.

5.4 SÉCURITÉ DU DES

Depuis son invention, la sécurité du DES a fait l'objet d'études attentives. Des techniques spéciales telles que la cryptanalyse différentielle, ou linéaire, ont été inventées pour attaquer le DES [49] et [70], mais les attaques les plus efficaces proviennent d'une exploration exhaustive de l'espace des clés. Avec des matériels spécifiques ou des grands réseaux de stations de travail, il est maintenant possible de déchiffrer les cryptogrammes venant du DES en quelques jours, voire même quelques heures. Au train où la puissance des PC augmente, on s'attend à ce que le DES puisse bientôt être cassé par un simple PC.

Aujourd'hui, le DES ne peut être considéré comme sûr que si un chiffrement triple, comme dans la section 3.7, est utilisé. Dans ces conditions, il est important de savoir que le DES n'a pas les propriétés mathématiques d'un groupe. En d'autres termes, étant données deux clés quelconques k_1 et k_2 on ne voudrait pas qu'il existe une troisième clé, k_3 , telle que $\text{DES}_{k_1} \circ \text{DES}_{k_2} = \text{DES}_{k_3}$. Si le DES était un groupe, un chiffrement répété ne pourrait pas conduire à une augmentation de la sécurité. En fait, le sous-groupe du groupe de permutations $S_{64!}$ engendré par les permutations du DES est au moins d'ordre 10^{2499} [49].

Cryptanalyse du DES

- Après 16 tours le résultat du DES est statistiquement plat : les caractéristiques générales du document source (fréquences etc..) sont indétectables.
- Une légère modification de la clé ou du message provoque des changements importants.
- Ces opérations sont facilement implémentables en Hard et permettent donc des débits élevés. Il a été notamment utilisé pour chiffrer les paiements par cartes (UEPS)

| Méthode d'attaque | Texte connu | Texte choisi | Stockage | Calculs |
|------------------------|------------------------|--------------|----------|------------------------|
| Recherche exhaustive | 1 | | | 2^{55} |
| Précalcul exhaustif | | 1 | 2^{56} | 1 tableau |
| Crypta. linéaire | 2^{47} puis 2^{36} | | Textes | 2^{47} puis 2^{36} |
| Crypta. différentielle | 2^{55} | 2^{47} | Textes | 2^{47} |

Le DES

Par la suite la taille des clés du DES est devenu insuffisant, on est donc passé à un système de double clé :

$$C = E_1(D_2(E_1(M)))$$

Aujourd'hui le DES a été remplacé par AES, choisi sur concours, avec un code par blocs 128bits, des clés de taille allant jusqu'à 256 bits et des calculs basés sur les polynômes.

Source : Théorie des Codes chez Dunod



PENSER À METTRE À JOUR LA FICHE RÉCAPITULATIVE DU TRAVAIL
RÉALISÉ EN TP

PROCHAIN COURS : HASHAGE ET PROBLÈMES DIFFICILES

