5 – Cryptographie à clef secrète

Cryptologie

- du grec *kryptos* (caché) et *logos* (science), cryptologie signifie littéralement science du secret
- ancienne terminologie : codes secrets
- chiffrer = coder en vue de garder un message secret
- la transmission du message chiffré est publique
- la sécurité du canal n'est pas assurée
- la science du secret revêt 2 aspects :
  1. cryptographie
  2. cryptanalyse
- à distinguer de la stéganographie ou du tatouage numérique *(watermarking)*

- confidentialité : pour garder secrets les messages
- identification : pour reconnaître un utilisateur autorisé d'un système informatique
- authentification : pour éviter l'usurpation d'identité
- intégrité : pour détecter toute altération du message fortuite ... ou volontaire
- non-répudiation : ajout des signatures pour garantir l'origine ou la réception d'un message

(Dé)-chiffrement

- *Alice* veut envoyer un message secret à *Bob*
- *Oscar*, *Eva*, *Melchior* accèdent au canal mais ne doivent pas comprendre le message

Plus précisément :

- M message clair
- E fonction de chiffrement injective
- $C = E(M)$ message chiffré
- D fonction de déchiffrement surjective
- $M = D(C) = D(E(M))$ et pas autrement

En pratique :

- les algorithmes calculant les fonctions cryptographiques utilisent des clefs *(keys)*
- K_e est le jeu de clefs paramètre de E
- K_d est le jeu de clefs paramètre de D
- ainsi :

$$E_{K_e}(M) = C$$

$$D_{K_d}(C) = M$$

Deux systèmes suivant un même principe

- clef secrète : seuls *Alice* et *Bob* connaissent la ou les clefs, aussi appelé chiffrement symétrique

- clef publique : des clefs utilisées respectivement par *Alice* et *Bob* sont en libre accès (pas toutes) : on parle de chiffrement asymétrique

Principes de Kerckhoffs (1883) encore d'actualité :

1. la sécurité repose sur le secret de la clef et non sur le secret de l'algorithme
2. le déchiffrement sans la clef doit être impossible (en un temps raisonnable)
3. trouver la clef à partir d'un *clair* et d'un *chiffré* doit être impossible (en un temps raisonnable)

Un monde peu sûr ...

Outre le bruit, le contenu du canal peut subir différents préjudices :

- attaques passives : écoute du réseau, analyse de son contenu
- attaques actives : modification de ce qui y passe
  - par exemple, avec l'attaque dite de l' ≪ homme du milieu ≫ (MITM)
- les premières attaquent la confidentialité, les secondes l'intégrité et l'authenticité
- la cryptanalyse est l'étude scientifique d'un système en vue de tester sa fiabilité, elle rescense les attaques potentielles de différents niveaux :
  - à chiffré connu
  - à clair connu
  - à chiffré choisi
  - à clair choisi

  et qui utilisent différents algorithmes (*e.g.* sur l'espace des clefs) :
  - par force brute
  - par séquences connues ou forcées
  - par analyse différentielle
  - par analyse linéaire
  - par analyse de la consommation énergétique
  - ...

Chiffrement à clef secrète (symétrique)

- *Alice* et *Bob* se partagent une même clef secrète :

$$K_e = K_d = K$$

- il y a analogie avec le *coffre-fort*
- comme pour les codages, 2 fonctionnements possibles :
  - par flot *(stream cipher)* :
    comme Vernam, RC4 (SSL dont WEP) , E0 (BLUETOOTH), A5/1 (GSM)
  - par blocs *(block cipher)* : comme DES, AES
- en pratique, ces systèmes de chiffrement reposent sur un générateur de nombres aléatoires ou pseudo-aléatoires
  - TRNG : vrai aléatoire
  - PRNG : pseudo-aléatoire (LFSR, *cf.* TD3)
  - CSPRNG : pseudo-aléatoire non prédictif

Exemples *par ordre chronologique :* la scytale, César, Vigenère, chiffres affines, Vernam, machines à rotors (ENIGMA), DES, RC4, TRIPLE-DES, AES-RIJNDAEL ...

Chiffre de Vernam

- chiffrement US datant de 1917 (1re guerre mondiale)
- chiffre parfait breveté par Vernam en 1919 :

$$H(M/C) = H(M)$$

- clef jetable : la clef n'est utilisée qu'une fois (en anglais : one-time pad)
- le chiffrement consiste à faire des additions
- le message M et la clef K sont de même longueur

$$C = E_K(M) = M \oplus K$$

$$M = D_K(C) = C \oplus K = (M \oplus K) \oplus K$$

- la clef doit bien être aléatoire
- la mise en œuvre peut être contraignante (réservé aux états, ambassades, *téléphone rouge*)
- 2 raisons de ne pas réutiliser la clef :
  - l'interception de M et de C révèle la clef :

$$M \oplus C = K$$

  - 2 messages chiffrés avec la même clef K neutralisent dangeureusement son effet (*cf.* TP5)

$$C_1 \oplus C_2 = (M_1 \oplus K) \oplus (M_2 \oplus K) = M_1 \oplus M_2$$

DES

- DES pour *Data Encryption Standard*
- standard de chiffrement développé par IBM en 1977
- premier algorithme à avoir été rendu public dans ses moindres détails
- sans doute le plus utilisé jusqu'à présent
- schéma de *Feistel* à 16 tours
- fonctionnement prévu aussi bien au niveau logiciel que matériel
- c'est le chiffrement des mots de passe sous UNIX
- la fonction f qu'il renferme combine transpositions et substitutions
- avec le temps, émergence de défauts :
  - faiblesses des *S-boxes* ?
  - mais surtout, clef de 56 bits seulement

    insuffisant face à la cryptanalyse par recherche exhaustive de clef
  - le TRIPLE-DES ou 3DES paru en 1999 s'avère lui trop lent :

$$C = E_{des}^{K_3}\left(D_{des}^{K_2}(E_{des}^{K_1}(M))\right)$$

  - en 2001, DES est détrôné par AES - RIJNDAEL

Quelle finalité pour chiffre symétrique ?

Confusion
Cacher la relation entre le clair et le chiffré avec des *substitutions* non-linéaires

Diffusion
Un changement d'un seul bit doit se diffuser à tout le chiffré avec des *permutations* sur les blocs

Clef de tour
Elle permet de générer une collection de clefs assemblées au reste par un $\oplus$

Pseudo-aléatoire
On aurait envie que le texte chiffré ait l'apparence d'une suite aléatoire

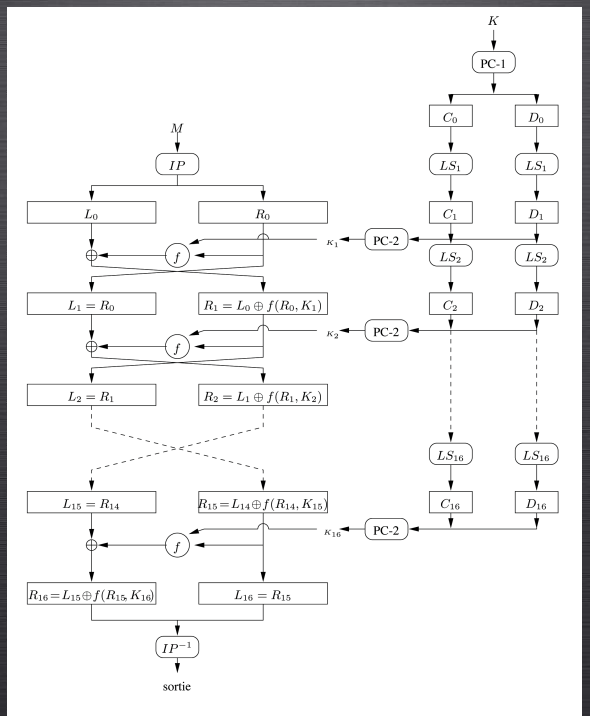
DES : fonctionnement

- soit à chiffrer un message $M = m_1 ... m_{64}$ de 64 bits avec une clef K de 56 bits augmentée de 8 bits de parité
- le cryptogramme C sera lui aussi sur 64 bits

- 3 étapes principales :
  1. $M' = IP(M) = m'_1 ... m'_{64}$ avec $L_0 = m'_1 ... m'_{32}$ et $R_0 = m'_{33} ... m'_{64}$
  2. on effectue 16 itérations utilisant la fonction $f = f_K$ détaillée plus tard :
  $$\begin{cases} L_i = R_{i-1} \\ R_i = L_{i-1} \oplus f(R_{i-1}, K_i) \end{cases}$$
  3. $C = IP^{-1}(R_{16}L_{16})$

- le même algorithme avec la même clef mais réalisant les *tours* dans l'ordre inverse permet de déchiffrer (*cf.* TD3).

DES : schéma



DES : la fonction f

Description du calcul générique $f(R_{i-1}, K_i)$ sur 32 bits :

- R_{i-1} est *expansé* en $E(R_{i-1})$ sur 48 bits
  une table de sélection indique une permutation des 32 bits + 16 répétés

- $B = E(R_{i-1}) \oplus K_i = B_1 B_2 ... B_8$ sur 48 bits
  chaque B_i sur 6 bits donc

- $C_i = S_i(B_i)$ sur 4 bits
  pour tout $1 \leq i \leq 8$ avec S_i est la i^{eme} S-box

- $f(R_{i-1}, K_i) = P(C_1 C_2 ... C_8)$
  la permutation P réordonnant les C_i

DES : TABLE DE SÉLECTION, BOÎTE DE SUBSTITUTION

- exemple d'une table de sélection indiquant la permutation de 32 bits et la répétition de 16 :

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

- la S-box S_1 qui calcule $C_1 = S_1(B_1)$ sur 4 bits :

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	12	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

pour $B_1 = b_1b_2b_3b_4b_5b_6$:
C_1 est l'entier en binaire lu à la ligne b_1b_6 et à la colonne $b_2b_3b_4b_5$

DES : DIVERSIFICATION DE LA CLEF

- la clef initiale K comporte 64 bits dont 1/8 de parité
- en vue d'obtenir les 16 sous-clefs K_i
- K est réordonnée par la permutation initiale $PC\text{-}1$ qui au passage supprime les 8 bits de parité
- on pose $PC\text{-}1(K) = C_0\,D_0$ comptant ainsi 2×28 bits
- pour tout $1 \leq i \leq 16$:

$$\begin{cases} C_i = LS_i(C_{i-1}) \\ D_i = LS_i(D_{i-1}) \\ K_i = PC\text{-}2(C_i\,D_i) \end{cases}$$

avec des décalages circulaires LS_i vers la gauche d'1 ou de 2 positions prédéfinis selon la valeur de i et $PC\text{-}2$ une seconde permutation.

BLOCS ET MODE DE CHAÎNAGE

- ECB (*Electronic Code Book*) : chaque bloc de message est codé de façon indépendante par le même algorithme *(simple mais non utilisé car pas assez sûr)*

$$c_i = E_k(m_i)$$

- CBC (*Cipher Bloc Chaining*) : mode de chiffrement le plus utilisé

$$c_i = E_k(m_i \oplus c_{i-1}) \quad \text{avec} \quad c_0 = IV$$

$$\text{si} \quad D_k = (E_k)^{-1} \quad \text{alors} \quad m_i = c_{i-1} \oplus D_k(c_i)$$

- CFB (*Cipher FeedBack*) : moins sûr que CBC, sans implantation du décodage (cryptages réseaux)

$$c_i = m_i \oplus E_k(c_{i-1}) \quad \text{avec} \quad c_0 = IV$$

$$m_i = c_i \oplus E_k(c_{i-1})$$

- OFB (*Output FeedBack*) : variante de CFB avec codage et décodage symétrique (satellites)

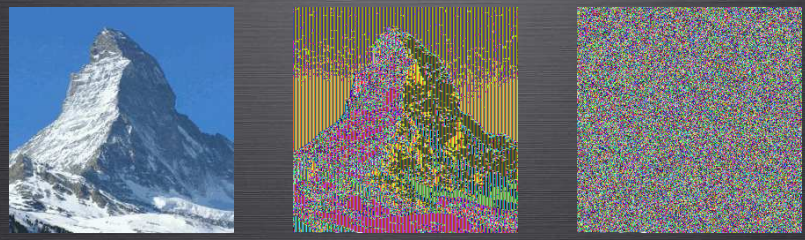
$$z_0 = c_0 \quad z_i = E_k(z_{i-1}) \quad c_i = m_i \oplus z_i$$

$$z_i = E_k(z_{i-1}) \quad m_i = c_i \oplus z_i$$

- CTR (*Counter-mode encryption*) : avec compteur, symétrique et aussi facilement parallélisable

$$c_i = m_i \oplus E_k(T + i)$$

DE L'UTILITÉ DE CHOISIR LE BON MODE ...



(Images Wikipédia)

Mots de passe Unix

- l'identification d'Unix utilise Des comme une fonction à sens unique
- elle en fait une fonction de hachage H qui calcule l'empreinte du mot de passe P et qui sera la seule stockée auprès de l'identifiant
- pour prévenir le cas de mots de passe identiques, il y a ajout d'un sel s aléatoire

  le sel dépend du nombre de secondes depuis l'an 2000 sur 12 bits et est stocké sous la forme de 2 caractères

- le sel modifie la fonction d'expansion E en lui ajoutant une permutation

  si le bit i du sel est à 1, les bits i et $i+24$ de la table de E sont échangés
  le même mot de passe peut ainsi être codé de 4096 manières différentes

- $H(P,s)$ consiste en l'application 25 fois de suite d'un Des ainsi modifié à partir d'une valeur initiale IV (en général que des 0) avec le mot de passe en guise de clef :

$$H(P,s) = E^P_{des'_{25}}\left(\ldots E^P_{des'_2}\left(E^P_{des'_1}(IV)\right)\right)$$

- $H(P,s)$ est enregistré dans le fichier lisible par tous `/etc/passwd` sous forme de 11 caractères sur 6 bits dans l'ensemble

$$\{".","/",0-9,A-Z,a-z\}$$

Contenu de `/etc/passwd`

login	password	salt	encrypted
bwilliam	nutmeg	Mi	qkFWCm1fNJI
espring	ellen1	ri	79KNd7V6.Sk
douglas	Sharon	./	2aN7ysff3qM
mcfields	norahs	am	fIADT2iqjAf
kwoody	norahs	7a	zfT5tIdyh0I

- à chaque connexion, l'identifiant donnant accès au sel s, on teste si le mot de passe P' fourni vérifie bien :

$$H(P',s) \stackrel{?}{=} H(P,s)$$

- `norahs` a été chiffré avec 2 sels différents mais différemment
- un tel fichier est la cible d'attaques par dictionnaire ou par force brute
- `/etc/shadow` en lecture seulement par `root` avec possibilité de varier l'algorithme de chiffrement

AES (Rijndael)

- Rijndael du nom de ses concepteurs belges J. Daemen et V. Rijmen
- choisi par le NIST en 2000 pour le nouvel AES *(Advanced Encryption Standard)*
- nouveau standard plus rapide adopté dès 2001
- chiffrement par blocs de 128 bits et clefs de 128, 192 et 256 bits

- Rijndael a été conçu pour rendre les méthodes classiques comme la cryptanalyse linéaire ou différentielle très difficiles
- plus économe en mémoire, plus rapide il n'en contient pas moins beaucoup de mathématiques

- on estime à 20 ans la robustesse de Rijndael ...

AES : fonctionnement

- l'entrée consiste en un bloc de 128 bits (16 octets)
- la clef comporte 128, 192 ou 256 bits
- le nombre de tours est respectivement 10, 12 ou 14

Description d'un tour :

- l'entrée subit une permutation prédéfinie dans une table
- les 16 octets permutés sont mis dans une matrice 4x4
- un $\oplus$ est appliqué entre cette matrice et celle contenant la clef
- on applique des rotations à droite des lignes de la matrice
- la matrice subit une transformation non-linéaire

  il s'agit d'une multiplication binaire de chaque élément par des polynômes
  cette multiplication est celle de $\mathbb{F}(2^8)$

- un tour se termine par un $\oplus$ entre la matrice obtenue et une autre prédéfinie

A suivre ...