Chapitre 5 Application des codes correcteurs en Cryptographie.

5.1 Notions de Cryptographie.

Définition 5.1. Un *système de cryptographie* ou *cryptosystème* est constitué de:

1- Un alphabet A . En pratique $A = \{0, 1\}$. Les éléments de A sont dits *symboles*.

2- Un ensemble M composé de chaînes de symboles de l'alphabet A appelé *espace de messages clairs*. Un élément de M est appelé *texte clair*.

3- Un ensemble C constitué de chaînes de symboles d'un alphabet B, qui peut être différente de l'alphabet A, appelé *espace de messages chiffrés* ou *cryptogrammes*.

4- Un ensemble K dit *espace des clés*. Un élément e de K est dit clé.

5- Pour chaque clé e de K, on définit une bijection f_e de M dans C, dite *fonction de chiffrement*. Si $m \in M$ alors $f_e(m)=c \in C$.

6- Pour chaque clé d de K, on définit une bijection f_d de C dans M, dite *fonction de déchiffrement*. Si $c \in C$ alors $f_d(c)=m \in M$.

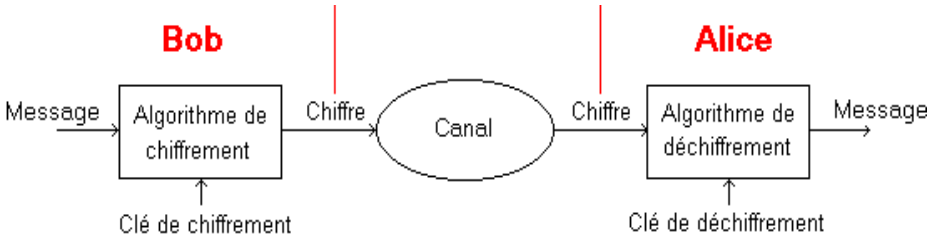

Schéma 5.1 Schéma général d'un cryptosystème.

Il existe deux types de cryptographie :

5.2 Cryptographie symétrique.

Définition 5.2.

En cryptographie *symétrique*, également appelée cryptographie à *clé secrète*, une seule clé suffit pour le chiffrement et le déchiffrement. les clés e et d sont identiques.

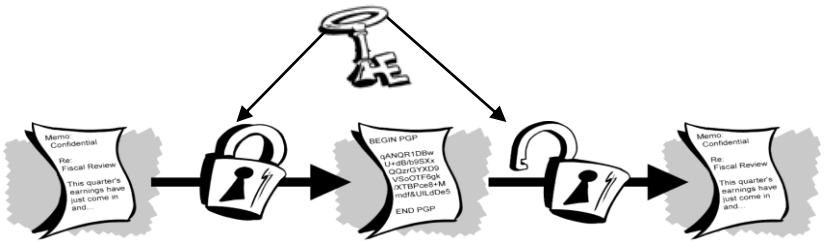

Schéma 5.2 Chiffrement et déchiffrement symétrique.

5.3 Cryptographie asymétrique.

Définition 5.3.

En cryptographie a*symétrique*, également appelée cryptographie à *clé publique*, admet deux clés, une clé *e* (dite *publique*) sert pour le chiffrement et une autre clé *d* différente de la clé *e* (dite *secrète*) sert pour le déchiffrement.

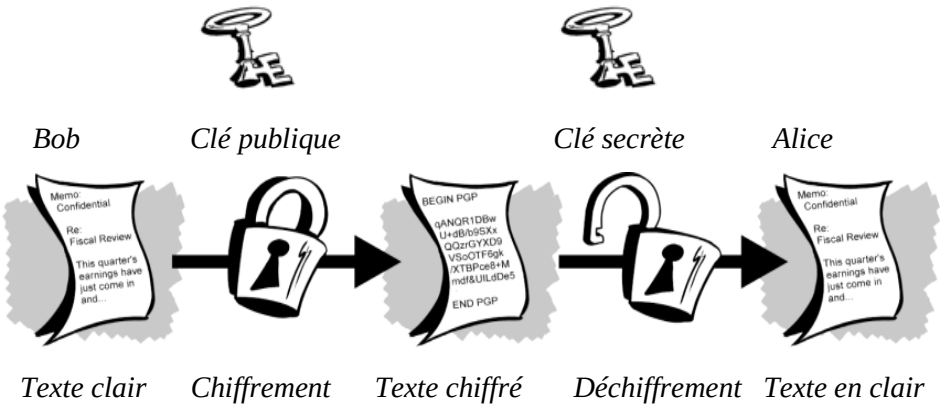

Schéma 5.3 Chiffrement et déchiffrement asymétrique.

5.3.1 Cryptosystème de McEliece.

C'est le plus ancien cryptosystème à clé publique utilisant des codes correcteurs d'erreurs. Il a été imaginé par McEliece en 1978, à peu près en même temps que le chiffrement RSA. Comme tous les cryptosystèmes à clé publique, ce cryptosystème est constitué de trois algorithmes:

1. La génération des espaces de clés,
2. Le chiffrement (utilisant la clé publique) et fonctions de chiffrement.
3. Le déchiffrement (utilisant la clé secrète) et fonctions de déchiffrement.

1. Génération de clé

On commence par générer un code linéaire $C(n, k, d)$ et sa matrice génératrice G de taille $k \times n$. On va mélanger cette matrice pour la rendre indistinguable d'une matrice aléatoire, pour cela on a besoin de :

a. Une matrice de *permutation aléatoire* P_σ de taille $n \times n$ associée à une permutation σ de S_n.

b. Une matrice inversible *aléatoire* S de taille $k \times k$.

La clé publique sera le couple (G', e) tel que $G' = S.G.P_\sigma$ qui est indistinguable d'une matrice aléatoire et e la capacité de correction de C.

La clé secrète est composée des trois matrices S, P_σ et G qui permettent de retrouver la structure du code C et donnent donc accès à l'algorithme de décodage.

2. Chiffrement.

Soit m un message de k bits que l'on veut chiffrer. On ne dispose pour cela que de la clé publique G'. On commence par calculer le mot de code C de longueur n associé à m : c = $m.G'$. Ensuite on génère une *erreur aléatoire* ε de longueur n et de poids $t \leq e$. Le texte chiffré sera simplement le mot bruité : c' = c + ε.

3. Déchiffrement. Pour déchiffrer le texte c', en connaissant P_σ, S et G, il suffit de calculer : r'=c'. P_σ^{-1} = $m.G'$. P_σ^{-1} + ε. P_σ^{-1} = $m\ S.G$ + ε. P_σ^{-1}. Le mot $r = m.S.G$ est un mot du code C et ε'= ε. P_σ^{-1} est une erreur de poids t (car P est une permutation et conserve donc le poids des mots), donc on peut corriger cette erreur et retrouver le message initial $m' = m\ S$ et le message clair $m=m'.S^{-1}$.

Exemple 5.1. Soit C(n, 4, d) un codes linéaire binaire de longueur n=7 de matrice génératrice

$$G=\begin{pmatrix} 1\,1\,0\,1\,0\,0\,0 \\ 0\,1\,1\,0\,1\,0\,0 \\ 0\,0\,1\,1\,0\,1\,0 \\ 0\,0\,0\,1\,1\,0\,1 \end{pmatrix}.$$

En appliquant la méthode de Gausse sur les lignes de G on trouve la matrice génératrice normalisé G_N,

$$G_N=\begin{pmatrix} 1\,0\,0\,0\,1\,1\,0 \\ 0\,1\,0\,0\,0\,1\,1 \\ 0\,0\,1\,0\,1\,1\,1 \\ 0\,0\,0\,1\,1\,0\,1 \end{pmatrix}$$ et donc C admet comme matrice génératrice $H=\begin{pmatrix} 1\,0\,1\,1\,1\,0\,0 \\ 1\,1\,1\,0\,0\,1\,0 \\ 0\,1\,1\,1\,0\,0\,1 \end{pmatrix}$

On déduit la distance d=3 et donc la capacité e=1.

Soit m =1010 $\in \mathbb{F}_2^4$ un message clair. Supposons que Bob veut envoyer ce message à Alice.

1. Génération des clés. Alice génère les clés suivantes :

a. La clés secrète : La matrice normalisée G_N, une matrice de permutation P_σ de type $n \times n$, une matrice inversible et aléatoire S de type $k \times k$ par exemple on choisit:

$$P_\sigma=\begin{pmatrix} 0\,1\,0\,0\,0\,0\,0 \\ 0\,0\,0\,1\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,1 \\ 1\,0\,0\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,1\,0 \\ 0\,0\,0\,0\,1\,0\,0 \end{pmatrix}$$ et $S=\begin{pmatrix} 0\,1\,0\,0 \\ 1\,0\,0\,0 \\ 0\,0\,1\,0 \\ 0\,0\,0\,1 \end{pmatrix}$

b. La clé publique est le couple (G', e) tel que $G'=S.G_N$. $P_\sigma = \begin{pmatrix} 0\,0\,0\,1\,1\,1\,0 \\ 0\,1\,1\,0\,0\,1\,0 \\ 0\,0\,1\,0\,1\,1\,1 \\ 1\,0\,1\,0\,1\,0\,0 \end{pmatrix}$ et la capacité e=1.

2. Le Chiffrement. Bob chiffre le message m en calculant c'= $mG'+\varepsilon$ avec par exemple l'erreur ε=0100000 $\in \mathbb{F}_2^7$ de poids w(ε)=1 $\leq e$=1. Donc le texte chiffré est a c'= $m.G'+\varepsilon$ c'=0111001+0100000 = 0111001.

3. Le déchiffrement

On a c'=$m.G'+\varepsilon=m.S.G_N$. $P_\sigma+\varepsilon \Longrightarrow r'$=c'. $P_\sigma^{-1}=m.S.G_N+\varepsilon$. $P_\sigma^{-1}=m'+\varepsilon'$ tel que $m' \in$C et

w(ε')=1(car P_σ est une permutation et conserve donc le poids des mots)

On calcule r'=c'. P_σ^{-1}= 0111001.$\begin{pmatrix} 0&0&0&1&0&0&0\\ 1&0&0&0&0&0&0\\ 0&0&0&0&1&0&0\\ 0&1&0&0&0&0&0\\ 0&0&0&0&0&0&1\\ 0&0&0&0&0&1&0\\ 0&0&1&0&0&0&0 \end{pmatrix}$ = 1110100. r' est un mot entaché d'erreurs.

En utilisant la méthode de décodage par syndrome. On peut corriger le mots r' d'erreur ε'.

Le syndrome de r' est $h(r')$= C_1+ C_2+ C_3+ C_5=$\begin{pmatrix}1\\1\\0\end{pmatrix}+\begin{pmatrix}0\\1\\1\end{pmatrix}+\begin{pmatrix}1\\1\\1\end{pmatrix}+\begin{pmatrix}1\\0\\0\end{pmatrix}=\begin{pmatrix}1\\1\\0\end{pmatrix}$ = C_1= $h(\varepsilon_1)$ = $h(1000000)$ et w(ε_1)=1. L'erreur est donc ε_1=1000000, et le mot code code est $r= m.S.\ G_N = r'+\varepsilon_1$= 1110100 + 1000000 = 0110100.

Le décodage de r (en enlevant la redondance) on obtient le mot $m'=mS$=0110 et donc le message initial $m= m'.S^{-1}$ = 0110.$\begin{pmatrix} 0&1&0&0\\ 1&0&0&0\\ 0&0&1&0\\ 0&0&0&1 \end{pmatrix}$ = 1010. Qui est le message clair.

Exemple 5.2. (G une matrice génératrice quelconque) Soit C (n=15, k=7, d) un code linéaire binaire de matrice génératrice

$$G=\begin{pmatrix} 1&1&1&0&1&0&0&0&1&0&0&0&0&0&0\\ 0&1&1&1&0&1&0&0&0&1&0&0&0&0&0\\ 0&0&1&1&1&0&1&0&0&0&1&0&0&0&0\\ 0&0&0&1&1&1&0&1&0&0&0&1&0&0&0\\ 0&0&0&0&1&1&1&0&1&0&0&0&1&0&0\\ 0&0&0&0&0&1&1&1&0&1&0&0&0&1&0\\ 0&0&0&0&0&0&1&1&1&0&1&0&0&0&1 \end{pmatrix}$$

et comme matrice de contrôle

$H=(M/I_3)$ tel que $M=\begin{pmatrix}1&0&0&0&1&0&1\\1&1&0&0&1&1&1\\1&1&1&0&1&1&0\\0&1&1&1&0&1&1\\1&0&1&1&0&0&0\\0&1&0&1&1&0&0\\0&0&1&0&1&1&0\\0&0&0&1&0&1&1\end{pmatrix}$

On trouve la distance $d \geq 3$ et donc la capacité $e \geq 1$.

Soit m =1010110 $\in \mathbb{F}_2^7$ un message clair. Supposons que Bob veut envoyer ce message à Alice.

1. Génération des clés. Alice génère les clés suivantes :

La clés secrète : constituée de la matrice G, une matrice de permutation P_σ de type 15×15, une matrice inversible et aléatoire S de type 7×7 par exemple on choisit:

Pour P_σ la matrice de permutation associée à la permutation

σ = (3,11,4,6,13,5,14,2,10,9,12,7,8,1,15) et on choisit $S=\begin{pmatrix}0&1&0&0&0&0&0\\1&0&0&0&0&1&1\\0&1&1&0&0&0&0\\1&0&0&0&0&1&0\\1&0&0&0&1&0&1\\1&1&1&1&0&1&1\\1&1&1&0&1&1&1\end{pmatrix}$

On trouve la clé publique: $G'=\begin{pmatrix}0&0&0&1&1&1&0&0&1&0&1&0&0&0&0\\1&0&1&1&1&0&0&0&1&0&1&1&1&0&1\\0&0&0&0&1&0&0&0&1&0&1&1&1&1&0\\1&1&1&1&1&0&0&0&1&1&1&0&1&1&0\\0&1&1&1&1&0&0&1&0&1&1&1&0&0&1\\1&1&1&1&1&1&1&0&0&0&0&0&1&1&1\\1&0&1&1&1&0&0&1&0&1&0&0&1&0&1\end{pmatrix}$

et on choisit comme vecteur d'erreur par exemple ε=000000000010000 tel que w(ε)=1 ≤ e.

2. Le Chiffrement. Bob chiffre le message m en calculant c'= mG'+ε de poids w(ε)=1.

On a c'= $m.G'$+ε=100100110100000 (c' est le message chiffré).

3. Le déchiffrement.

On a c'=$m.G'$+ε=$m.S.G.P_\sigma$+ε $\Longrightarrow$ c=c'. P_σ^{-1}=$m.S.G$+ε. P_σ^{-1}=m'+ε' tel que $m' \in$C et w (ε′)=1 (car P_σ est une permutation et conserve donc le poids des mots). On a r'=c'. P_σ^{-1}=001000001001110.

En utilisant la méthode de décodage par syndrome. On peut corriger le mot r' d'erreur ε'.

Le syndrome de r' est $h(r')= C_3+ C_9+ C_{12}+ C_{13}+ C_{14}=C_2 = h(\varepsilon') = \mathrm{h}(010000000000000)$

Et w(ε')=1. L'erreur est donc ε'=010000000000000. Et le mot code est $r=m.S.G$ = c+ε',

r = 001000001001110 + 010000000000000 = 011000001001110.

Or $r= m.S.G$=011000001001110. $r= m.S.(A / B)$ d'où $m'=m.S.A$.

Le décodage de m'. On a: $m'=m.S.A$=0110000 et donc et donc $m=m'.A^{-1}.S^{-1}$, le message initial

$m=m'.A^{-1}.S^{-1}$= (0110000). $\begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$ = 1010110. Qui est le mot clair.

5.3.2 Cryptosystème de Niederreiter

Cette variante du cryptosystème de McEliece a été mise au point par Niederreiter en 1986. Elle est exactement équivalente du point de vue de la sécurité et est un peu plus efficace en temps de calcul. Elle fonctionne comme le chiffrement de McEliece, mais au lieu d'utiliser la matrice génératrice G, on utilise la matrice de contrôle H. Ce système est constitué de trois algorithmes:

1. Génération de clé

On commence par générer un code linéaire $C(n, k, d)$ et une matrice de contrôle H de taille $n\text{-}k \times n$. On va mélanger cette matrice pour la rendre indistinguable d'une matrice aléatoire, pour cela on a besoin de :

La clé secrète est composée des trois matrices :

a. Une matrice de contrôle H (et donc la matrice normalisée H_N) du code C.
b. Une matrice de permutation aléatoire P_σ de taille $n \times n$ associée à une permutation σ de S_n.
c. Une matrice inversible aléatoire M de taille $n\text{-}k \times n\text{-}k$.

La clé publique sera le couple (H', t) tel que $t \leq e$ et $H' = M.H_N.P_\sigma$ qui est in distinguable d'une matrice aléatoire et e est la capacité de correction de C.

2. Chiffrement.

Soit y un message de n bits que l'émetteur Bob veut chiffrer et de poids $w(y)= t \leq e$. Le mot chiffré est le mot $z = y.{}^tH'$ de type $1 \times n\text{-}k$.

3. Déchiffrement.

Pour déchiffrer en connaissant P_σ, M et H Alice suit les étapes suivantes :

a. calcule: $s = z.{}^tM^{-1} = (y.{}^tP_\sigma).{}^tH$ de type $1\times n\text{-}k$ qui est un mot syndrome.

b. En utilisant l'algorithme de décodage par syndrome, Alice retrouve le mot $z' = y.{}^tP_\sigma$ correspondant au syndrome s.

c. Le récepteur retrouve enfin le mot $y = z'.{}^tP_\sigma^{-1}$.

Exemple 5.3. Considérons le code (n = 15; k = 7; d = 5) de longueur n = 15 sur le corps de Galois F_{16} de racine primitive α et de polynôme primitif $M_\alpha(X) = X^4+X+1$:

On a $F_{16} = \{0, \alpha^i / 0 \le i \le 14\}$ avec $\alpha^4 = \alpha + 1$. Considérons la matrice de contrôle H:

$$H=\begin{bmatrix}1&1&0&1&0&0&0&1&0&0&0&0&0&0&0\\1&1&0&1&0&0&0&1&0&0&0&0&0&0&0\\0&0&1&1&0&1&0&0&0&1&0&0&0&0&0\\0&0&0&1&1&0&1&0&0&0&1&0&0&0&0\\0&0&0&0&1&1&0&1&0&0&0&1&0&0&0\\0&0&0&0&0&1&1&0&1&0&0&0&1&0&0\\0&0&0&0&0&0&1&1&0&1&0&0&0&1&0\\0&0&0&0&0&0&0&1&1&0&1&0&0&0&1\end{bmatrix}$$ Qu'on peut mètre sous forme systématique :

$H_N=(I_8 /A)$ tel que $A=\begin{bmatrix}1&1&0&1&0&0&0\\0&1&1&0&1&0&0\\0&0&1&1&0&1&0\\0&0&0&1&1&0&1\\1&1&0&1&1&1&0\\0&1&1&0&1&1&1\\1&1&1&0&0&1&1\\1&0&1&0&0&0&1\end{bmatrix}$. On considère la matrice $M=\begin{bmatrix}1&0&0&0&0&0&0&0\\1&1&0&0&0&0&0&0\\1&1&1&0&0&0&0&0\\1&1&1&1&0&0&0&0\\1&1&1&1&1&0&0&0\\1&1&1&1&1&1&0&0\\1&1&1&1&1&1&1&0\\1&1&1&1&1&1&1&1\end{bmatrix}$ et la matrice de permutation $P= P_\sigma$, tel que σ est la transposition τ_{17}.

Aprés calcul on trouve $H' = M.H.P$, $H'=\begin{bmatrix}0&0&0&0&0&0&0&0&1&1&0&1&0&0&1\\0&1&0&0&0&0&0&0&1&0&1&1&1&0&1\\0&1&1&0&0&0&0&0&1&0&0&0&1&1&1\\1&1&1&1&0&0&0&0&1&0&0&1&0&1&1\\1&1&1&1&1&0&0&0&0&1&0&0&1&0&1\\0&1&1&1&1&1&0&0&0&0&1&0&0&1&1\\1&1&1&1&1&1&1&0&1&1&0&0&0&0&1\\0&1&1&1&1&1&1&1&0&1&1&0&0&0&1\end{bmatrix}$.

La clés publique est (H', e=2)

Soit le texte clair qu'on veut chiffrer et envoyer : y=010000000000001 de poids t=2.

Le chiffrement de y est le mot z donné par : $z = y.{}^tH'$. En remplaçant y et H' on trouve: z=10000000

Pour le déchiffrement on calcule: $s = z.{}^tM^{-1}$=11000000 ≠ 0. Théoriquement $s = y.{}^tP.{}^tH_N$. c.à.d. s représente le syndrome du mot $y' = y.{}^tP$ et comme $s \neq 0$ donc y' n'est pas un mot de C.

Pour la correction de y', on va utiliser la méthode de décodage par syndrome. On a $s = C_1+C_2$ (s est la somme de première et la deuxième colonne de H_N) donc $s=h(\varepsilon)$ tel que ε=110000000000000 et le poids $w(\varepsilon)=2=e$. Donc les mots $y' = y.{}^tP$ et ε ont le même syndrome et le même poids alors, d'après la Proposition 4.4.3 on déduit que $y' = \varepsilon$ d'où $y.{}^tP=\varepsilon$ ce qui donne que $y = \varepsilon.{}^tP^{-1}$, comme P est une matrice de permutation symétrique alors,

$P^{-1} = {}^tP = P$ et $y = \varepsilon.P$=010000000000001, qui est bien le mot transmis.

5.3.3 Comparaison des cryptosystèmes McEliece, Niederreiter et RSA.

	McEliece	Niederreiter	RSA
Taille de la clé publique.	*kn*	*k(n-k)*	*2n*
	67072 octets	*32750 octets*	256 octets
Nombre de bits d.information transmis par chiffrement.	*K*	$a=log2(C_n^e)$	*N*
	512	*276*	1024
Taux de transmission.	*k/n*	$Log_2(C_n^e)/n-k$	1
	51,17%	56,81%	100%
Nombre d'opérations binaires du chiffrement par bit d'information.	n/2+ *n/k*	*(n-k)ke/an+n/a*	$125.3^{m-1}/2^m$
	513,9	50,1	2402,7
Nombre d'opérations binaires du déchiffrement par bit d'information.	*B/k*	*C/a*	$25.3^{m-1}/2$
	5140	7863,3	738112,5
$B=n+mnt+4m^2t^2+2mt+mn(2t+1)+k^2/2$ et $C=2n+4m^2t^2+2m^2t+mn(2t+1)+(n-k)^2/2$.			

Tableau 5.1 Comparaison des cryptosystèmes McEliece, Niederreiter et du RSA.

Du tableau ci-dessus on déduit que la taille de la clé au cryptosysteme RSA est meilleur que celle de Niederreiter et cette dernierre et meilleur que McEliece.

Le taux de transmission dans RSA est parait et il est deux fois plus meilleur que celui des deux autres cryptosystemes qui se rapprochent.

Concernant Nombre d'opérations binaires du chiffrement est tres couteux dans RSA est le moins couteux dans Niederreiter.