COUR CRYPTOGRAPHIE

Chapitre III :Cryptographie Symétriques
Licence Réseaux Système et Sécurité

Mr. Ahmed KHALIFA
khalifaahmedou@yahoo.fr

December 8, 2015

Sommaire

Introduction

Introduction

Système symétrique

Un système symétrique est un système construit avec une fonction ou un processus facilement reversible. Généralement on entend par systèmes symétriques les systèmes de chiffrement à clés sécrètes.

	A				**B**	
	K →		**canal sûr** →		**K**	
	↓				↓	
M →		**M'** →	**non sûr** →	**M'** →		**M**

Dans un système symétrique, la clé secrète doit être partagée entre les entités en communication d'où la nécessité d'avoir un canal sûr.

Canal sûr:

- Se rencontrer, utiliser la valise diplomatique (avant 1976)
- Utiliser des techniques de cryptographie non symétrique (théoriquement depuis 1976)

Système symétrique

les algorithmes de chiffrements symétriques sont utilisés pour rendre le service de confidentialité et sont composés de deux catégories: les algorithmes de chiffrements par blocs et les algorithmes de chiffrements par flux:

DES, 3DES, Lucifer, FEAL, RC4, RC5, Blowfish, IDEA, AES

2.1:Techniques et outils de base

Techniques et outils de base

Techniques

Les systèmes symétriques utilisent plusieurs techniques ou outils dont les plus essentielles sont:

- Permutation (Transposition)
- Substitution (Trace d'une permutation globale)
- Chiffrement de Vernam ou One Time Pad (1918)
- Chiffrement par blocs
- Chiffrement sequentiel ou par flux (ou flot)

Definition : Permutation

$\Rightarrow$ bijection d'un ensemble donnée;
$\Rightarrow$ changer l'ordre des symboles dans un messages;

Comme on travaille sur des ensembles finis, toute permutation est une bijection de $[0, n-1]$, on le matérialise par la donnée de l'ensemble des images ordonné naturellement c'est à dire $\pi[0, n-1] = \pi_0\pi_1\pi_2....\pi_{n-1}$ et ce tableau est appelé P-Box. Cette suite finie peut être écrite sur une ligne ou sur la forme d'une matrice par exemple $\pi = 102538674$ ou

$$\begin{pmatrix} 1 & 0 & 2 \\ 5 & 3 & 8 \\ 6 & 7 & 4 \end{pmatrix}$$

Techniques et outils de base

Si M est un message et π une permutation, pour caculer la transformé de M par π, on le décompose en morceaux de longueurs égales (**découpage en blocs**) à la longueur de π. Si le dernier morceau est incomplet (longueur plus courte que celle de π) on définit une procédure publique qui permet de faire du padding (= completer le bloc incomplet). On calcule l'image de chaque morceau puis on juxtapose (**concaténation des blocs**) les résultats dans l'ordre du découpage.
NB La concaténation de A et B est noté habituellement $A \parallel B$.

Exemple si M=**Mon premier cours de crypto**" et $\pi = 102538674$, on découpe $M = \underline{monpremie\ rcoursdec\ ryptozzzz}$ puis on calcule les images des blocs et enfin on concatène les résultats
$\pi(M) = \pi(monpremie)\ \pi(rcoursdec)\ \pi(ryptozzzz)$
$\pi(M) =$**OMNEPEMIR CROSUCDER YRPZTZZZO**

Définition: Substitution

$\Rightarrow$ permutation sur l'ensemble des cas possibles appliquées à un nombre fini cas;

$\Rightarrow$ remplacer chaque élément du texte claire (symbole, groupes de symboles) par un autre élément du texte claire (=message);

NB Parfois on utilise des substitutions qui ne sont pas réversibles.
Exemple 1: $S_k =$
décaler les lettres de k rangs vers la droite dans l'ordre alphabétique".
En numérotant les lettres de l'alphabet latins de 0 à 25 on voit que S_k est la permutation (globale) $S_k : \frac{\mathbb{Z}}{26\mathbb{Z}} :\to \frac{\mathbb{Z}}{26\mathbb{Z}} : x :\mapsto x + k \ mod 26$.
S_k transforme symbole par symbole, on dit que c'est une substitution mono-alphabétique.
S_3**=chiffrement de CESAR**

Techniques et outils de base

$S_{k_1,k_2,\ldots,k_m} : (\frac{\mathbb{Z}}{26\mathbb{Z}})^m :\rightarrow (\frac{\mathbb{Z}}{26\mathbb{Z}})^m : (u_1, \ldots, u_m) :\mapsto$
$(u + k_1 \ mod 26, \ldots, u_m + k_m \ mod 26)$ est une substitution polyalphabétique. Dans ce cas, le message est divisé en blocs de longueur m

Si $S_{k_1,k_2,\ldots,k_m}$ est fixé, le m-uplet $(k_1, \ldots, k_m)$ est le mot de passe ou clé: exemple $(PAS) = (15, 0, 18)$

$S_{k_1,k_2,\ldots,k_m}$**= chiffrement de Vigenaire avec un mot de passe de longueur** m

exemple: en utilisant comme mot clé CHIFFRE, le texte: "Ce texte est chiffré par Vigenaire" devient :

$$\begin{array}{rl} & CHIFFRECHIFFRECHIFFRECHIFFRE \\ + & CETEXTEESTCHIFFREPARVIGENERE \\ = & FMCKDLJHACINAKIZNVGJALONTKJJ \end{array}$$

Exemple2 On considère une matrice rectangulaire $S = (S_{i,j})_{i \leq 3, j \leq 15}$ avec $S_{i,j} \in \{0, 1, 2, ..., 15\}$ alors on peut definir ne substitution non bijective $\{0,1\}^6$ dans $\{0,1\}^4$ par $S : \{0,1\}^6 :\to \{0,1\}^4 : b_1 b_2 b_3 b_4 b_5 b_6 \mapsto S_{i,j}$ en base 2 ou $i = b_1 b_6$ et $j = b_2 b_3 b_4 b_5$ en decimal . Par exemple si S est la matrice:

$$\begin{pmatrix} 14 & 4 & 13 & 1 & 2 & 15 & 11 & 8 & 3 & 10 & 6 & 12 & 5 & 9 & 0 & 7 \\ 0 & 15 & 7 & 4 & 14 & 2 & 13 & 1 & 10 & 6 & 12 & 11 & 9 & 5 & 3 & 8 \\ 4 & 1 & 14 & 8 & 13 & 6 & 2 & 11 & 15 & 12 & 9 & 7 & 3 & 10 & 5 & 0 \\ 15 & 12 & 8 & 2 & 4 & 9 & 1 & 7 & 5 & 11 & 3 & 14 & 10 & 0 & 6 & 13 \end{pmatrix}$$

et $B = 101011$ alors $i = b_1 b_6 = 11_{(deux)} = 3_{(dix)}$ et
$j = b_2 b_3 b_4 b_5 = 0101_{(deux)} = 5_{(dix)}$ donc
$S_{i,j} = S_{3,5} = 9_{(dix)} = 1001 = S(B)$.

NB Cette matrice de substitution (appelé **S-box**) est utilisé dans le **DES** qui est le premier algorithme de chiffrement standardisé en 1977 .

2.1.3 Definition: Opérateur XOR

Opérateur XOR=OU Exclusif=$\oplus$:

- $P \oplus Q$ est vrai si P **ou bien** Q est vrai. Si on pose $1 = vrai$ et $0 = faux$, on a: $P \oplus Q = Q \oplus P$, $P \oplus P = 0$ et $(P \oplus Q) \oplus R = P \oplus (Q \oplus R)$
- on a $0 \oplus 0 = 0$, $0 \oplus 1 = 1$, $1 \oplus 0 = 1$, $1 \oplus 1 = 0$.
- Le complément de a est $\overline{a} = 1 + a$. Donc $\overline{0} = 1$ et $\overline{1} = 0$: exemple: $\overline{1101011100} = 0010100011$
- Si on a deux chaines binaires de même longueur a et b on peut les additionner bit à bit, et on retrouve les propriétés: $a \oplus b = b \oplus a$, $a \oplus a = 0$, $(a \oplus b) \oplus c = a \oplus (b \oplus c)$ et $\underline{(a \oplus b) \oplus b = a}$
- Si M est une chaine binaire $|M|$ représente la longueur de M. Par exemple on a $|011001000110001| = 15$.

Techniques et outils de base

Definition: Chiffrement de Vernam (-Mauborgne - Vigenaire

$\Rightarrow$) chiffrer un message M **suffisamment long** avec une **clé parfaitement aléatoire** K **aussi longue que** M;
$\Rightarrow$) **Changer de clé à chaque chiffrement** (clé à usage unique= one time pad);
$\Rightarrow$) $|M| = |K| \geq 160$; chiffrement: $M \oplus K = M'$; déchiffrement: $M' \oplus K = M$ donc celui qui chiffre et celui qui déchiffre effectue la même opération.

Exemple

texte claire M	1000110101111000110101111
clé K	0100111101010100111110101
chiffrement $M' = M \oplus K$	1100001000101100001000010
clé K	0100111101010100111110101
Déchiffrement $M' \oplus K$	1000110101111000110101111

Techniques et outils de base

Algorithme de chiffrement par blocs (Cipher block)

Dans un chiffrement par bloc, le message est divisé en blocs de longueur égale :
$\Rightarrow$) $M = M_1 \oplus M_2 \oplus .... \oplus M_m$, (généralement $l = |M_j| \geq 128$).

$\Rightarrow$) Si $\mathcal{C}_K$ est la fonction de chiffrement, on calcule $M'_j = \mathcal{C}_k(M_j)$ pour tout j.

$\Rightarrow$) Si $\mathcal{D}_K$ est la fonction de déchiffrement, on calcule $\mathcal{D}_k(M'_j) = M_j$ pour tout j

$\Rightarrow$) Il y'a plusieurs façons de combiner les différents blocs chiffrés.

Par exemple, on peut chiffrer **blocs par blocs** ou **utiliser un bloc chiffré dans le suivant**.

Algorithme de chiffrement par blocs (Cipher block)

$\Rightarrow$) Généralement, **une fonction de chiffrement par bloc** contient une sous-fonction (principale) qui est itérée plusieurs fois (**chaque itération est appelée tour**) pour créer une **situation de surchiffrement dans le but d'augmenter la sécurité**.

$\Rightarrow$) S'il y'a r tours à réaliser, il faut r sous clés $K_j, 1 \leq j \leq r$ de taille l qui sont dérivées de la clé de chiffrement K, via un **algorithme de génération de sous clés** .

Si $\overline{\mathcal{C}}_k$ est la sous fonction principale de chiffrement, on calcule par surchiffrement:

$\Rightarrow$) $T' \circ \overline{\mathcal{C}}_{K_r} \circ \overline{\mathcal{C}}_{K_{r-1}} \cdots \circ \overline{\mathcal{C}}_{K_2} \circ \overline{\mathcal{C}}_{K_1} \circ T(M_j)$ où T et T' sont deux fonctions qu'on applique respectivement au début et à la fin (ce choix T et T', dépend des algorithmes).

2.1.5 Algorithme de chiffrement par blocs (Cipher block)

On appelle chiffrement produit un chiffrement par blocs qui combine plusieurs transformations élémentairement (substitutions, transpositions, opérations linéaires ou arithmétiques)
Un chiffrement itératif résulte de l'application itéré d'un chiffrement (en général un chiffrement produit).

Algorithme de chiffrement par blocs (Cipher block)

Exemple: 1: corps de l'algorithme

Entrée (Input): un message claire N de taille 8 (bits) et une clé de clé K de taille 6

Sortie (Output): un message chiffré de taille 8

A) Algorithme de génération de sous clés

1. Entrée: une clé K de taille 6;
2. Appliquer l'expension $E = 53024130$;
3. Découper en deux blocs de longueur 4;
4. Appliquer une permutation circulaire vers la gauche(Left shift) d'ordre 2 sur chaque bloc;
5. Sortie: une paire de clés (K_1, K_2) de longueur 4 chacune.

Algorithme de chiffrement par blocs (Cipher block)

B) Algorithme de chiffrement

1. Entrée un bloc claire N de longueur 8
2. Appliquer la permutation $\pi = 17023564$;
3. Découper N en deux blocs $N = G_0 || D_0$ de longueur 4;
4. **1er tour= 1er round** Calculer $D_1 = G_0 \oplus K_1$ et $G_1 = P(D_0) \oplus G_0$ où $P = 4213$ est une permutation;
5. **2nd tour= 2nd round** Calculer $D_2 = G_1 \oplus K_2$ et $G_2 = P(D_1) \oplus G_1$;
6. Concatener les blocs G_2 et D_2
7. Appliquer la permutation inverse de π
8. Sortie: un chiffré de longueur 8

Exercice 1

1. Donner l'algorithme de déchiffrement de l'exemple précédent.
2. Donner le schémas de l'algorithme de génération de clé.
3. Donner le schémas de l'algorithme de chiffrement.
4. Donner le schémas de l'algorithme de déchiffrement.
5. On donnde $K = 010110$ dériver les deux sous clés K_1 et K_2
6. On donne $M = 11001101$ chiffrer M puis déchiffrer .
7. Généraliser pour n tours sans changer la taille des messages et clés.

Exercice 2

Faire un programme dans le langage de votre choix pour implémenter:

- l'algorithme de génération de clé;
- l'algorithme de chiffrement;
- l'algorithme de déchiffrement.

Techniques et outils de base

Algorithme de chiffrement par flux (Stream cipher)

Le **chiffrement par flux (flot) se fait sequentiellement en générant une clé aussi longue que le message à chiffrer**. Chaque morceau (bit ou byte=octect=8bits) de la clé est composée via la fonction de chiffrement avec la portion de clé correspondante.

$\Rightarrow$) Donc si le message est $M = m_1||m_2|||...||m_{l-1}m_l$ et la clé est $K = k_1||k_2|||...||k_{l-1}k_l$ alors le chiffrement se fait par morceaux: $c_i = \mathcal{C}_{k_i}(m_i)$ (où $\mathcal{C}_{k_i}$ est la fonction de chiffrement) et le déchiffrement se fait par morceaux: $d_i = \mathcal{C}_{k_i}(c_i) = m_i$ (où $\mathcal{D}_{k_i}$ est la fonction de déchiffrement).

$\Rightarrow$) Par exemple: $c_i = m_i \oplus k_i$ et $d_i = c_i \oplus k_i = m_i$.

Algorithme de chiffrement par flux (Stream cipher)

⇒) Ainsi, le chiffrement de Vernam est un exemple de à la fois de stream cipher et de bloc cipher.

⇒) Le chiffrement par flux est adapté à des modes transmission où le message arrive morceaux par morceaux et si les équipements utilisés ont peut de ressource mémoire ou nécessite une transmission rapide par exemple: chiffrements en ligne , qui sont utilisées en particulier par les armées (sécurité), pour la téléphonie mobile (rapidité) GSM et son réseau (systéme $A51$: algorithme de chiffrement), etc..

⇒) Il a un autre avantage sur les chiffrements par flux en ce sens que si une erreur se produit sur m_i où k_i alors cette erreur n'est pas propagée; elle n'affecte que c_i.

⇒) Toute **la sécurité repose sur l'algorithme de génération de clés** qui est généralement couplé avec le chiffrement.

Techniques et outils de base

Algorithme de chiffrement par flux (Stream cipher)

Exemple: 1: corps de l'algorithme
Entrée (Input): un message claire N de taille 4 (bits) et une clé K de taille 4
Sortie (Output): un message chiffré de taille 4

Algorithme de génération des sous clés et chiffrement

1. Entrée: une clé $K = k_0 k_1 .... k_3$ et un message $N = N_0 N_1 .... N_3$ de taille 4;
2. Tour n° i: chiffrement du ième bit de N
   - appliquer une permutation initiale $P = 1032$ à K;
   - Calculer $k'_j = k_0 \oplus k_1 ... \oplus k_j \oplus \overline{k_{j+1}} \oplus ... \oplus \overline{k_3}$ pour $0 \leq j \leq 3$;
   - Chiffer le message N_i en N'_i avec $N'_i = N_i \oplus k_i$;
   - Mettre k'_j dans k_j: $k_j \leftarrow k'_j$ pour $0 \leq j \leq 3$;
3. reprendre (2)
4. Sortie: un message chiffré de longueur 4.

Techniques et outils de base

Exercice 3

1. Donner l'algorithme de déchiffrement de l'exemple précédent.
2. Donner le schémas de l'algorithme de génération de clé.
3. On donnde $K = 0101$ et $M = 1101$ chiffré M en M'.
4. Déchiffrer le message précédent.
5. Reprendre l'exercice précédent en prenant $|M| = |K| = 256$, puis les diviser en 32 octets.

Exercice 4

Faire un programme dans le langage de votre choix pour implémenter:

- l'algorithme de génération de clé et de chiffrement;
- l'algorithme de déchiffrement.

Exercice 5

Généraliser l'algorithme de chiffrement par flot présenté ci-dessus comme suit:

- $Choisir |N| = |K| = 256;$
- découper le message N et la clé K en octets (chaine de 8bits),
- remplacer la permutation P par la substitution $S : \frac{\mathbb{Z}}{256\mathbb{Z}} \to \frac{\mathbb{Z}}{256\mathbb{Z}} : x \mapsto 5x + 7 mod 256$

Exercice 6

Soit F, une fonction quelconque
$F : \{0,1\}^m \times \{0,1\}^n \to \{0,1\}^n : (K,R) \mapsto F(K,R)$.
On considère la fontion $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n \times \{0,1\}^n :$
$(L,R) \mapsto (L', R') = (L \oplus F(K,R), R)$
Montrer que f est bijective et determiner f^{-1}.
NB f est appelé fonction de Feistel (l'algorithme DES)

Avantages et inconvénients des chiffrements symétriques

Avantages et inconvénients

A) Avantages

- les algorithmes symétriques sont rapides (parcequ'ils utilisent de petits entiers et des opérations rapides);
- en général, il semble que les algoritmes symétriques sont plus faciles à fabriquer (plus nombreux !);
- le seul algorithme dont la sécurité est prouvée est un algorithme symétrique à savoir le chiffrement de Vernam;

Avantages et inconvénients

B) Inconvénients

- Confidentialité de la clé sécrète : problème de partage de la clé à travers un canal sûr et problème de stockage de la clé;
- Durée de vie des clés assez courte;
- Peut de service de sécurité sont pris en charge par les systèmes symétriques par exemple: on ne peut déterminer qui entre les deux interlocuteurs légitimes, a chiffré un message;
- Distribution des clés: si n personnes communiquent 2 à 2, il faut $C_n^2 = \frac{n(n-1}{2}$ clés. Pour $n = 1000$ alors $C_n^2 = 499500$