Licence L2 STS Mention SPI Parcours Informatique
Unité 174EN007
Sécurité Informatique

TD4

Ce TD porte sur différents aspects cryptologiques.

Note : les deux premiers exercices proviennent de l'excellent site web « La cryptographie expliquée » (http://www.bibmath.net/crypto/index.php)

Exercice n°1 : le chiffre de Hill

Les chiffres polygraphiques

Les chiffres polygraphiques sont des chiffres pour lesquels on partage d'abord le message chiffré en groupes d'un certain nombre de lettres. Pour chacun de ces groupes, on opère alors un algorithme de chiffrement (le plus souvent, une substitution) pour chiffrer le message.
Par exemple, on peut couper le message en blocs de deux lettres LE MA TI NL ES OL EI LS EL EV EX, et remplacer chacun de ces blocs de deux lettres par un autre bloc de deux lettres. On parle alors de chiffre digraphique. Le principal avantage est qu'il devient plus difficile de casser le chiffre par une étude statistique. Si tous les blocs de 2 lettres n'ont pas la même fréquence (ES est plus fréquent que XV), il y a désormais 26×26=676 couples à analyser, contre 26 lettres.
De nombreux algorithmes modernes de chiffrement fonctionnent sur ce principe. Comme ils ne sont pas basés sur des lettres, mais sur des données informatiques, on parle plutôt de chiffrement par blocs que de chiffrement polygraphique.

Le chiffre de Hill

Lester Hill, mathématicien cryptographe (1891-1961) publie en 1929 dans la revue *American Mathematical Monthly* un article intitulé *Cryptography in an algebraic alphabet*, où il détaille un nouveau type d'algorithme de chiffrement. Son idée est de continuer à utiliser des décalages du même type que celui du chiffre de César, mais en effectuant ces décalages simultanément sur des groupes de m lettres ! Bien sûr, plus m est grand, plus les analyses statistiques deviennent difficiles !

D'abord, nous remplaçons chaque lettre par son ordre dans l'alphabet : A devient 0, B devient 1,..., Z devient 25. On groupe les nombres ainsi obtenus par m (prenons par exemple m=2). Pour chaque bloc de m nombres à coder $x_1x_2...x_m$, on calcule le texte codé en effectuant des combinaisons linéaires (ici m=2) :

$$y_1 = ax_1 + bx_2$$
$$y_2 = cx_1 + dx_2$$

Si a, b, c, d sont des entiers, y_1 et y_2 seront aussi des entiers. Pourtant, si l'on souhaite les reconvertir en lettres, il faudrait qu'ils soient compris entre 0 et 25 ce dont on ne peut s'assurer. On les y ramène en prenant leur reste dans la division par 26. Si z_1 et z_2 sont les restes respectifs de y_1 et y_2 dans la division par 26, on peut retransformer z_1 et z_2 en lettres, et obtenir le message codé.

Le choix de la clé correspond ici au choix d'un nombre m, et au choix des combinaisons linéaires à effectuer (ce sont toujours les mêmes de blocs en blocs), c'est-à-dire des entiers a, b, c et d.

Exemple

On souhaite coder le mot ELECTION avec le chiffre de Hill, pour m=2, a=3, b=5, c=1 et d=2.
Etape 1 : On partage en blocs de 2 : EL EC TI ON.
Etape 2 : On remplace les lettres par leur nombre associé : 4-11 | 4-2 | 19-8 | 14-13.
Etape 3 : On effectue les combinaisons linéaires pour chaque bloc. Par exemple, pour le premier bloc, où $x_1=4$ et $x_2=11$, on a :

$y_1=3\times4+5\times11=67$
$y_2=1\times4+2\times11=26$

De même, $y_3=22$, $y_4=8$, $y_5=97$, $y_6=35$, $y_7=107$, $y_8=40$.
Etape 4 : On prend les restes modulo 26, et on trouve : $z_1=15$, $z_2=0$, $z_3=22$, $z_4=8$, $z_5=19$, $z_6=9$, $z_7=3$, $z_8=14$.
Etape 5 : On reconvertit en lettres, pour trouver PAWITJDO.

On peut remarquer que le premier E de ELECTION est transformé en P, tandis que le second est transformé en W. Le critère des chiffrements polyalphabétiques est bien respecté : les analyses statistiques directes sur la fréquence des lettres sont impossibles.

Explication mathématique

Le chiffre de Hill est à l'intersection de l'arithmétique et de l'algèbre linéaire. En remplaçant les lettres par des nombres (A-->0,...), on ne traite plus que des entiers compris entre 0 et 25. En outre, un nombre n est identifié avec tous les nombres n+26k, où k est un entier (en clair, si 1 représente B, 27,53,-25... aussi!). Quand les calculs faits par les combinaisons linéaires sortent des entiers de 0 à 25, on s'y ramène en prenant le reste dans la division par 26. On dit que l'on travaille dans $\mathbb{Z}/26\mathbb{Z}$.

Chaque groupe de 2 lettres, ou par identification de 2 nombres x_1, x_2, est représenté par un vecteur colonne $\binom{x1}{x2}$. Les relations de dépendance linéaire sont, comme souvent, représentés par une matrice $A=\begin{pmatrix}a & b\\ c & d\end{pmatrix}$. On a, dans $\mathbb{Z}/26\mathbb{Z}$, la relation $\binom{y1}{y2}=A.\binom{x1}{x2}$ ou $\binom{y1}{y2}$ est le bloc codé et $\binom{x1}{x2}$ est le bloc clair.

Déchiffrement

Pour déchiffrer un message codé connaissant la clé, on procède exactement de la même façon. On découpe donc en blocs de m lettres, mais cette fois il faut inverser les relations données par les combinaisons linéaires : si un système donne y_1 et y_2 en fonction de x_1 et x_2, il faut pouvoir l'inverser et exprimer x_1 et x_2 en fonction de y_1 et y_2.
Toute matrice de chiffrement ne convient pas ! Par exemple, la matrice $A=\begin{pmatrix}1 & 2\\ 2 & 4\end{pmatrix}$ n'est pas une bonne matrice de chiffrement, car par exemple, $A.\binom{0}{0}=\binom{0}{0}$ et $A.\binom{2}{25}=\binom{52}{104}=\binom{0}{0}$ (où on a ramené les calculs dans $\mathbb{Z}/26\mathbb{Z}$). Ainsi, deux vecteurs (ou encore deux couples de deux lettres) différents sont codés de la même façon. Il est donc impossible, même en connaissant la clé, de décrypter.

Pour que le processus soit inversible, il est nécessaire et suffisant que A soit inversible, mais attention, inversible dans $\mathbb{Z}/26\mathbb{Z}$. On montre que cela est vérifié si, et seulement si, det A=a×d-b×c est inversible dans $\mathbb{Z}/26\mathbb{Z}$ (c'est-à-dire qu'il existe k un entier tel que k×det A=1+26n, où n est un entier - cela est équivalent à dire que det A est premier avec 26). Dans ce cas, l'inverse est donné par : $A^{-1} = (\det \quad A)^{-1} = \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$. On déchiffre alors en utilisant le même procédé, mais en utilisant A^{-1}.

Exercice

Coder le message LE TD EST FINI sachant que m=2, a=3, b=5, c=1 et d=2.
Décoder le message EUOCEMYOWITZFT avec les mêmes paramètres.

Exercice n°2 : le chiffre de Vernam et le chiffre du Che

Le chiffre de Vernam

Un chiffre parfaitement sûr est un chiffre tel que, l'adversaire interceptant le message, même ayant à sa disposition une puissance de calcul infinie, ne peut pas retrouver la moindre information concernant le message clair à partir du message chiffré.
Ceci peut se traduire très bien avec des probabilités. Si on intercepte le message crypté DSJMSZERT, on souhaite qu'avec un chiffre parfaitement sûr, il n'y ait pas plus de chances que le message clair soit ORDINATEUR, CAISSIERES ou DKIRJSMOTS. Tous les messages clairs possibles sont équiprobables !

Un tel chiffre parfait existe : c'est Gilbert Vernam, ingénieur au laboratoire de recherche de la compagnie *American Telephone & Telegraph* qui l'a inventé et publié en 1926. Il peut être décrit simplement comme un chiffre de Vigenère, mais où la clé répond aux trois impératifs suivants :

- elle est aussi longue que le texte à chiffrer ;
- elle est parfaitement aléatoire ;
- elle n'est utilisée que pour chiffrer un seul message, puis est immédiatement détruite.

C'est Claude Shannon, lui aussi chercheur dans les laboratoires de AT&T, qui prouva en 1949 le fait que ce chiffre est parfaitement sûr. La seule information dont on dispose, si on intercepte le message chiffré, est la longueur du message clair. De plus, tout chiffre parfaitement sûr est nécessairement une variante du chiffre de Vernam.

De façon moderne, le chiffre de Vernam (on parle aussi de **masque jetable**, pour souligner le fait que la clé doit être à usage unique), est implémenté de la façon suivante. Le message est d'abord converti informatiquement en suites de bits, c'est-à-dire de 0 et de 1. On prend une clé (complètement aléatoire) composée elle aussi d'une suite de 0 et de 1, aussi longue que le message à chiffrer.

On prend ensuite chaque bit du message clair et de la clé, et on en fait le ou exclusif. Rappelons que cette opération, que nous noterons $\oplus$, est définie par :

a	b	a $\oplus$ b
0	0	0
0	1	1

1	0	1
1	1	0

Cette opération (qu'on peut voir comme l'addition en base 2, mais en oubliant la retenue), vérifie notamment les propriétés suivantes : $x \oplus x = 0$ et $x \oplus 0 = x$.
Par exemple, si le message clair est 101110011, si la clé est 011101000, alors le message chiffré est 110011011, comme le montre le tableau suivant :

Message clair	1	0	1	1	1	0	0	1	1
Clé	0	1	1	1	0	1	0	0	0
Message chiffré	1	1	0	0	1	1	0	1	1

À la réception, celui qui reçoit le message fait la même opération à partir du message chiffré : il prend donc chaque bit du message chiffré et fait le ou exclusif avec le bit correspondant de la clé. Il retrouve le message initial, à cause des deux propriétés précédentes.

Hélas, le chiffre de Vernam n'est pas la panacée. D'abord, il exige qu'une clé serve une seule fois. Si vous utilisez la même clé deux fois, alors on peut extraire beaucoup d'informations des messages chiffrés. En effet, si on envoie les deux messages m_1 et m_2 avec la même clé k, on obtient les cryptogrammes $c_1=m_1 \oplus k$ et $c_2=m_2 \oplus k$. Mais si on effectue $c_1 \oplus c_2$, alors on obtient $c_1 \oplus c_2 = m_1 \oplus k \oplus k \oplus m_2 = m_1 \oplus m_2$, c'est-à-dire la somme des deux messages clairs. On peut alors obtenir beaucoup d'informations sur m_1 et m_2.
Ensuite, le chiffre de Vernam exige des clés extrêmement longues, et une parfaite synchronisation des clés. L'échange des clés, qui doit être sécurisé, est donc difficile à réaliser. Enfin, les clés utilisées doivent être parfaitement aléatoires, ce qui n'est pas facile à garantir.
C'est pourquoi ce chiffre n'est mis en œuvre que dans des cas très particuliers. Il fut ainsi utilisé pour sécuriser le téléphone rouge, ligne directe entre la Maison Blanche et le Kremlin du temps de la guerre froide. Les clés circulaient dans les valises diplomatiques, transportées dans des avions bourrés d'agents secrets. Et on raconte que pour produire des clés aléatoires, les Soviétiques employaient des "lanceurs de dés" : leur travail consistait à lancer des dés toute la journée et à noter le résultat.
Les systèmes de chiffrement symétriques utilisés dans la pratique cherchent à imiter le chiffrement de Vernam, en donnant le sentiment (l'illusion?) que le message chiffré est aléatoire. Bien sûr, il est impossible de fabriquer avec un logiciel quelque chose de complètement aléatoire, et ces systèmes sont tous moins performants que le chiffre de Vernam. Cependant ils sont bien plus pratiques à mettre en œuvre et offrent souvent une sécurité suffisante pour les besoins.

Le chiffre du Che

Lorsqu'Ernesto Guevara, dit le Che, fut retrouvé mort par l'armée bolivienne en 1967, après avoir tenté de développer des foyers révolutionnaires en Amérique latine et participer à la guérilla bolivienne, on découvrit sur lui des papiers expliquant comment il chiffrait ses messages qu'il communiquait à Fidel Castro. Il commençait par remplacer les lettres par un nombre à deux chiffres compris entre 01 et 99.

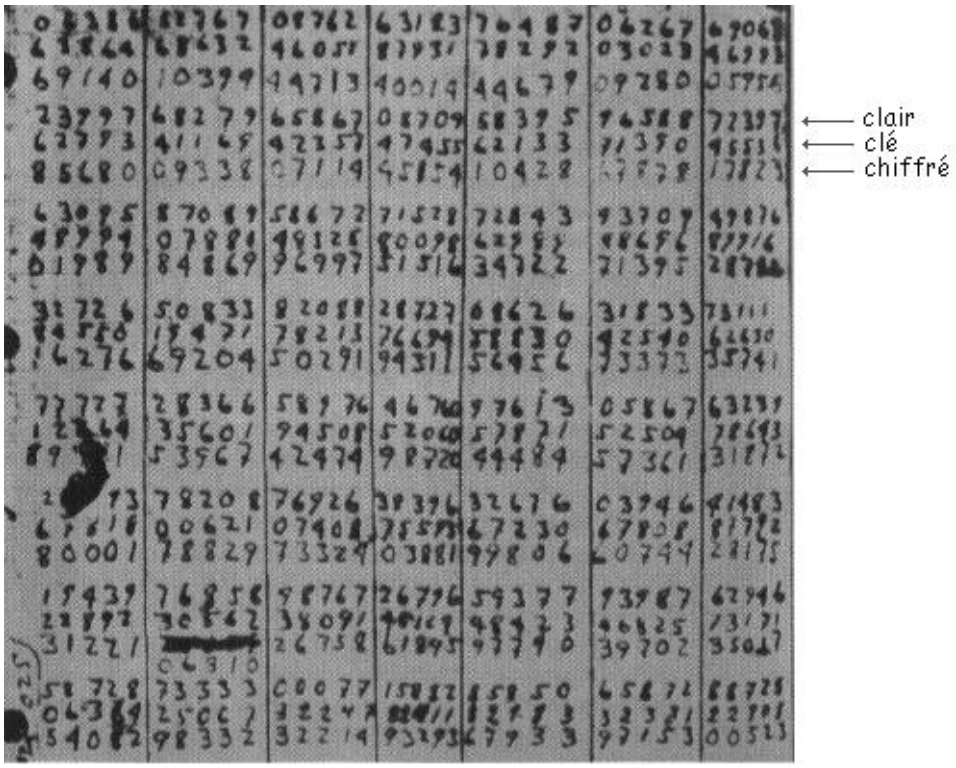
A	06	E	08	I	39	M	70	Q	71	U	52	Y	01
B	38	F	30	J	31	N	76	R	58	V	50	Z	59
C	32	G	36	K	78	O	09	S	02	W	56		

D	04	H	34	L	72	P	79	T	00	X	54	

En soit, cela ne constitue qu'une simple substitution dont on sait qu'elle n'apporte aucune sécurité. Mais le Che et Fidel Castro avaient aussi recours à une forme du chiffre de Vernam, et que l'on sait parfaitement sûr. Che Guevara écrivait sous le message chiffré (transformé en nombres de la façon précédente) une suite de chiffres, aussi longue que le message à envoyer, et connue uniquement de lui-même et de Cuba. Il séparait ensuite les deux lignes de chiffres par groupes de 5, chaque groupe comportait deux couples de 5 chiffres correspondant au texte à envoyer et à la clé. Pour chacun de ces groupes, il effectuait les opérations suivantes :

- il scindait, à la fois pour la ligne du texte à envoyer et pour la ligne de la clé, le groupe de 5 chiffres en deux nombres de deux chiffres (avec respectivement le premier et le deuxième chiffre, et le troisième et le quatrième chiffre), et un nombre de 1 chiffre (le dernier).
- il additionnait le premier nombre (de deux chiffres) du texte à envoyer et de la clé; si le reste dépassait 100, il ne gardait que les deux derniers chiffres;
- il effectuait la même opération pour les deuxièmes nombres du texte à envoyer et de la clé;
- il additionnait le dernier chiffre du texte à envoyer et le dernier chiffre de la clé; si le reste dépassait 10, il ne gardait que le dernier chiffre.

On obtient ainsi, pour chaque groupe de 5 chiffres du message à envoyer et de la clé, un groupe de 5 chiffres correspondant au message chiffré.



Si la clef est parfaitement aléatoire, employée une seule fois, et de longueur aussi longue que le message à envoyer, le chiffre utilisé par le Che est prouvé parfaitement sûr : personne ne peut décrypter sans la clef. Ceci explique la popularité du chiffre de Vernam dans les milieux diplomatiques. Le problème essentiel est le transport des clés, qui doivent être très longues.

On préfère désormais bien souvent utiliser un autre algorithme, avec une clé plus courte, mais qui dans la pratique offre le même niveau de sécurité !

Exercice

Décoder le message 9790327590062385783956 sachant que la clé commence par 1582327192042355486317.

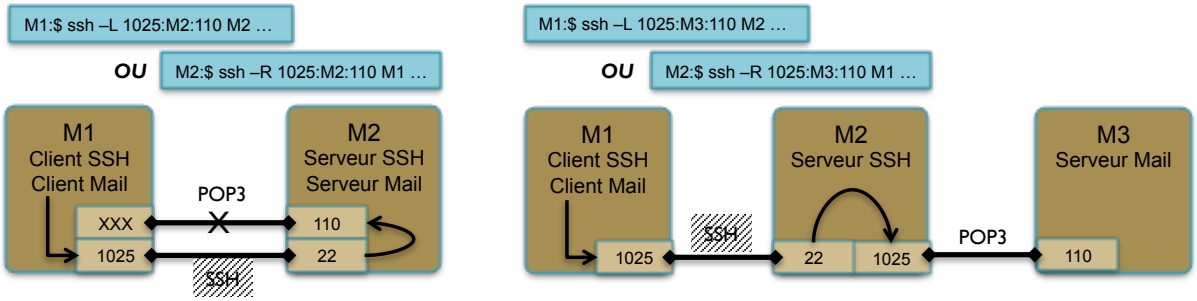
Exercice n°3 : redirection de port avec SSH

Redirection locale d'un port
ssh –L *port_local*:*machine_finale_X*:*port_X machine_redirection*

port_local : nouveau numéro de port pour le flux sécurisé par ssh
machine_finale_X :port_X : machine finale de réception du flux non sécurisé
machine_redirection : machine (serveur ssh) qui fait la redirection depuis le port TCP 22

Redirection distante d'un port
ssh –R port_distant:machine_finale_X:port_X machine_distante

machine_distante:port_distant : machine de provenance du flux sécurisé
machine_finale_X :port_X : machine finale de réception du flux non sécurisé
La redirection est faite par la machine (serveur ssh) sur laquelle on exécute la commande



Schématisez et donnez les commandes *ssh* pour définir un accès sécurisé à un serveur FTP M3 via un serveur SSH M2 depuis une machine M1 sachant que le protocole FTP utilise les ports 20 (ftp-data) et 21 (ftp-control).

Exercice n°4 : étude des messages TLS v1.2

Donnez la suite des messages échangés entre un client et un serveur dans le cas suivant :

- Le serveur est authentifié par certificat.
- Le client n'est pas authentifié.
- DH est utilisé pour partager la clé secrète de chiffrement.
- Le chiffrement sera fait en 3DES.
- L'intégrité sera fait avec SHA.
- La version de TLS sera v1.0.

Vous ajouterez les actions internes du client et du serveur.