La cryptologie

Emmanuel CONCHON
(emmanuel.conchon@univ-jfc.fr)

La cryptologie

- La cryptologie est la science des messages secrets et se décompose en deux disciplines
  - La cryptanalyse
  - La cryptographie
- La cryptanalyse est la discipline qui vise à retrouver un message à partir d'un message crypté
  - Déchiffrement: obtenir le message d'origine en connaissant la méthode de chiffrement et les clés
  - Décryptement: obtenir le message sans avoir les clés
- La cryptographie est une discipline visant à cacher des messages
  - Vient du grec kruptos (caché) et graphein (écrire)
  - La stéganographie est une forme particulière de la cryptographie qui vise à cacher un message dans un autre support pour masquer sa présence

La cryptographie

- La cryptographie est utilisée depuis l'antiquité pour la transmission de secret ou plus simplement comme marqueur social
  - En Egypte, les hiéroglyphes n'étaient connus que de la haute aristocratie
  - L'écriture dans son ensemble peut être vue comme une technique cryptographique rudimentaire
- Les champs d'applications sont très variés
  - Militaire
  - Santé
  - Commerce
  - …
- Pour assurer la protection d'une information, on va devoir utiliser des méthodes de chiffrement
  - symétrique
  - asymétrique

Quelques définitions

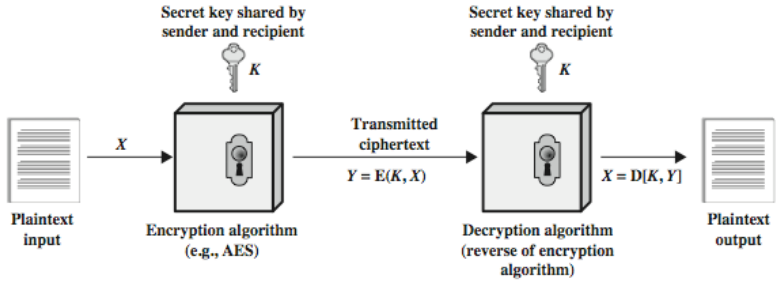
- Le fait de coder un texte en clair en un texte chiffré s'appelle chiffrement
  - L'opération inverse est le déchiffrement
- L'algorithme mis en œuvre pour réaliser le chiffrement est appelé cryptosystème
- Le texte chiffré en sortie s'appelle un cryptogramme (cyphertext en anglais)

Comment protéger le chiffrement ?

- Dans le cas de communications d'un cryptogramme
  - Il peut être intercepté et cryptanalysé (espionnage passif)
  - Dans le pire cas, il peut également être modifié ou étendu (espionnage actif)
- Comment faire pour préserver le secret ?
  - Cacher le principe de chiffrement (i.e. le cryptosystème) ?
  - Pbm: si jamais le cryptosystème est divulgué il faut changer tout le système
- Le principe de Kerckhoffs (La cryptographie militaire, 1883)
  - Ce principe explique que tout le secret doit uniquement reposer sur la clé
  - Le cryptosystème doit pouvoir être divulgué sans risque
  - Reformulé par Shannon en: « L'adversaire connait le système ! »

Le chiffrement symétrique

- L'alphabet a été une première forme de chiffrement qui permettait de ne partager le secret de la connaissance qu'entre lettrés
  - Rapidement, l'alphabétisation a rendu ce système peut fiable et il a fallu mettre en place de nouvelle techniques
- Ces techniques se regroupent dans la famille des techniques de chiffrement symétrique basées sur une clé secrète partagée



Les techniques de transpositions

- Toutes les lettres du message d'origine sont toujours présentes mais dans un ordre différent
  - Repose sur le principe des permutations mathématiques
- Un anagramme est une forme de transposition simple d'un mot
- Dans le cas de transpositions complexes, il faut définir une clé partagé
- Une des formes les plus ancienne vient des grecs en -600
  - Repose sur l'utilisation de cylindre de bois appelés scytale
  - Le texte est écrit longitudinalement sur la bandelette
  - La clé est donc la largeur du scytale
    - Pour déchiffre un message il faut un scytale de même diamètre

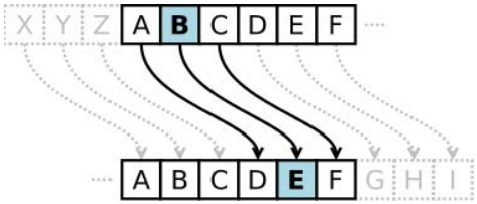
Source Schéma: wikipédia

Les techniques de substitution

- A partir de -200 apparaissent des techniques plus évoluées de cryptage reposant sur la substitution
  - Substitution mono-alphabétique
    - Remplace une lettre par une autre lettre de l'alphabet
  - Substitution homophonique
    - Remplace une syllabe par une autre ayant le même son
    - Exemple: le langage SMS
  - Substitution poly-alphabétique
    - Utilise une suite mono-alphabétique (clé) réutilisée périodiquement
  - Substitution basée sur des polygrammes
    - Substitue un groupe de caractère par un autre groupe de caractère

Le chiffre de César

- Cette méthode de substitution mono-alphabétique est la plus ancienne connue (Ier siècle avant JC)
  - Elle est relativement simple est consiste à décaler les lettres de n rang dans l'alphabet pour obtenir le cryptogramme
  - Lorsque l'on arrive à la fin de l'alphabet on repart au début



Source Schéma: wikipédia

  - La clé est donc le rang de décalage
- Cette technique est assez faible car seules 25 substitutions sont possibles
  - Encore utilisée par les russes en 1915
  - Utilisée sur Internet sous le nom de ROT-13 pour masquer des solutions à des jeux

Les techniques par substitution

- Une méthode classique de cryptanalyse pour déchiffrer un cryptogramme reposant sur des techniques de substitution et l'étude de la distribution statistique des symboles
- Dans une langue, les lettres n'ont pas toutes la même probabilité d'occurrence
  - En anglais les lettres les plus fréquents sont e, t, o, a ,n…
  - Les combinaisons de 2 lettres les plus fréquentes (digramme): th, in, er…
  - De trois lettres: the, ing…
- Un code par substitution ne modifie pas les distribution statistiques
- Il suffit de rechercher les lettres, les digrammes et les trigrammes les plus fréquent dans un cryptogramme pour pouvoir émettre des suppositions
  - Une supposition a de bonne chance d'être juste si des mots commencent à émerger

Les techniques par substitution

- Fréquence des lettres en Français

Lettre	Fréquence	Lettre	Fréquence
a	8,25	n	7,25
b	1,25	o	5,75
c	3,25	p	3,75
d	3,75	q	1,25
e	17,75	r	7,25
f	1,25	s	8,25
g	1,25	t	7,25
h	1,25	u	6,25
i	7,25	v	1,75
j	0,75	w	0,00
k	0,00	x	0,00
l	5,75	y	0,75
m	3,25	z	0,00

Les techniques par substitution

- Les digrammes composés d'un voyelle et d'une consonne les plus fréquents (sur 10 000)

es	305	te	163	ou	118	ec	100	eu	89	ep	82
le	246	se	155	ai	117	ti	98	ur	88	nd	80
en	242	et	143	em	113	ce	98	co	87	ns	79
de	215	el	141	it	112	ed	96	ar	86	pa	78
re	209	qu	134	me	104	ie	94	tr	86	us	76
nt	197	an	30	is	103	ra	92	ue	85	sa	75
on	164	ne	124	la	101	in	90	ta	85	ss	73
er	163										

Les techniques par substitution

- Pour améliorer les faiblesses des techniques mono-alphabétique, l'idée principale a été de faire évoluer l'alphabet pendant le chiffrement
- La technique la plus célèbre est connue sous le nom du Chiffre de Vigenère en hommage à son concepteur Blaise de Vigenère (1586)
  - Cet algorithme est plus résistant à l'analyse fréquentielle
  - Une même lettre d'un message peut être codée différemment en fonction de sa position dans le texte
  - On utilise une clé de chiffrement pour déterminer les alphabets à utiliser conjointement avec le Carré de Vigenère

Les techniques par substitution

- Le carré de Vigenère

B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Les techniques par substitution

- Exemple:
  - Texte en clair: BONJOUR TOUT LE MONDE
  - Clé: SALUT
  - Alphabet de chiffrement:

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S

  - Le codage s'obtient en faisant la substitution suivante:

B	O	N	J	O	U	R	T	O	U	T	L	E	M	O	N	D	E
S	A	L	U	T	S	A	L	U	T	S	A	L	U	T	S	A	L
T	Q	Y	D	H	M	R	E	I	N	L	L	P	G	H	F	D	P

Les techniques par substitution

- Le chiffre de Hill (1929) est une méthode de chiffrement par bloc utilisant l'algèbre linéaire
  - Elle s'appuie sur la représentation de l'alphabet en valeur numérique modulo 26 (A=0, B=1,…,Z=25)
  - Pour chiffrer un message, chaque bloc de n lettre est multiplié par une matrice inversible de n * n
    - Si la matrice n'est pas inversible, on ne peut pas décoder
  - La matrice est la clé de chiffrement
    - Plus n est grand plus fort est le secret

Les techniques par substitution

- Exemple
  - Supposons que nous voulions coder ELECTION
  - Prenons n=2 et la matrice $\begin{pmatrix}3 & 5\\1 & 2\end{pmatrix}$
  - cela revient à coder EL EC TI ON ou $\binom{4}{11}\binom{4}{2}\binom{19}{8}\binom{14}{13}$
  - Ce qui donne:
    - $\binom{4}{11} * \begin{pmatrix}3 & 5\\1 & 2\end{pmatrix} = \binom{67}{26} \; mod\; 26 = \binom{15}{0}$
    - $\binom{4}{2} * \begin{pmatrix}3 & 5\\1 & 2\end{pmatrix} = \binom{22}{8} \; mod\; 26 = \binom{22}{8}$
    - $\binom{19}{8} * \begin{pmatrix}3 & 5\\1 & 2\end{pmatrix} = \binom{97}{35} \; mod\; 26 = \binom{19}{9}$
    - $\binom{14}{13} * \begin{pmatrix}3 & 5\\1 & 2\end{pmatrix} = \binom{107}{40} \; mod\; 26 = \binom{3}{14}$
  - D'où le cryptogramme: PAWITDJO

Le chiffrement de Vernam ou la sécurité inconditionnelle

- On parle de sécurité inconditionnelle lorsque la connaissance du message chiffré n'apporte aucune information sur le message de départ
  - Résistant aux techniques de cryptanalyse
  - Seule attaque possible: la force brute (ou recherche exhaustive)
- Le chiffrement de Vernam (One Time Pad)
  - On parle également de chiffrement à usage unique
  - Le crypto-système emploie une clé de la même longueur que le message à crypter
    - On effectue un OU exclusif (XOR) entre la clé et le message
  - Cet algorithme est connu comme parfaitement sur
    - Le flux provenant de la clé doit bien sur être imprédictible et ne doit pas être réutilisé

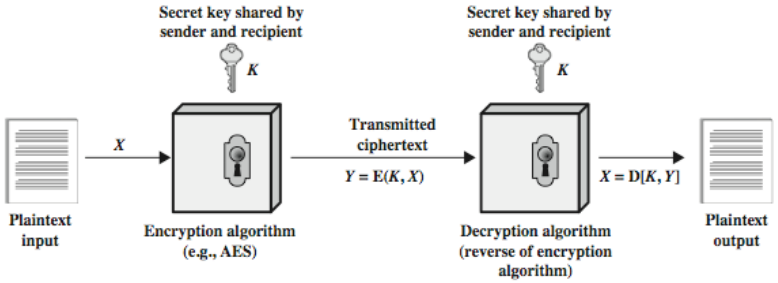
LA CRYPTOGRAPHIE POUR LA SÉCURITÉ DES SI

Les formes modernes de la cryptographie

- La cryptographie moderne repose sur le principe de Kersckhoffs
  - Le cryptosystème doit être publiable et connu de tous
  - La sécurité repose uniquement sur la clé
- Avec les algorithmes par transposition et substitution vus précédemment les algorithmes étaient relativement simples et les clé longues
  - C'est la longueur de la clé qui donne la confiance dans le système
- Dans la forme moderne, le principe retenu est plutôt des algorithmes plus complexes mais permettant d'avoir des clés plus courtes
  - Les algorithmes doivent pouvoir résister longtemps tout en étant connus
- On distingue deux sortes de chiffrement
  - Les chiffrements à clé symétrique
  - Les chiffrements à clé asymétrique

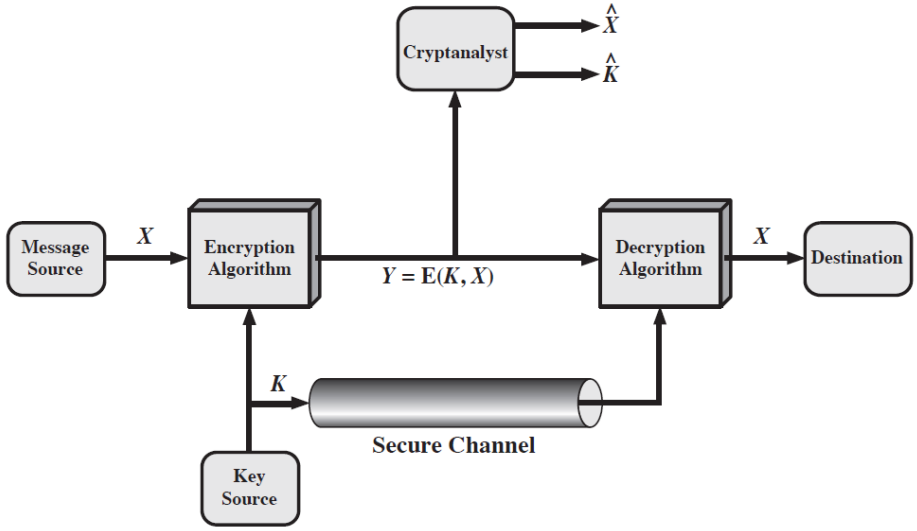
Le chiffrement à clé symétrique - Rappel

- La même clé est utilisé pour le chiffrement et pour le déchiffrement
  - On parle de clé secrète partagée



  - La faiblesse repose justement sur ce partage de clé
    - Comment faire pour s'assurer que la clé n'est pas interceptée ?

Le chiffrement à clé symétrique - Rappel



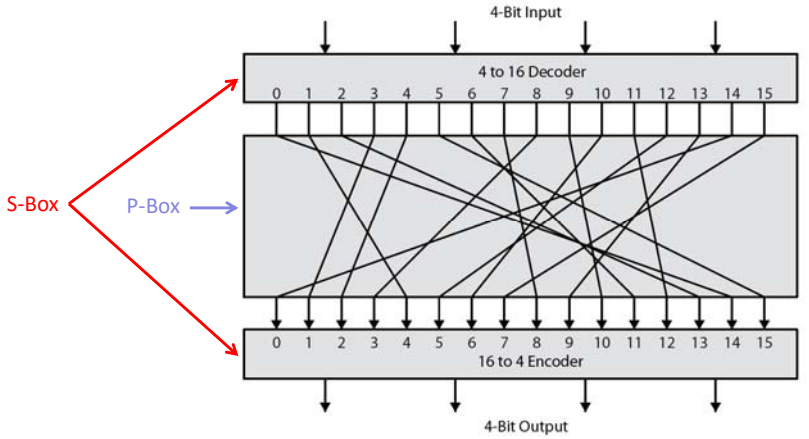
Le chiffrement par bloc

- Le message est converti en chaine binaire
- La chaine binaire est ensuite découpée en n blocs
- On crypte successivement les n blocs
  - XOR entre le bloc et la clé
  - Permutation/substitution de certains bits à l'intérieur du bloc
  - On recommence l'opération plusieurs fois (on parle de ronde)
- On concatène les n blocs chiffrés pour obtenir la chaîne binaire chiffrées

- Les substitutions et permutations ont été introduites par Shannon pour ajouter de la confusion dans le message
  - Substitutions réalisées à l'aide de S-Box
  - Permutations réalisées à l'aide de P-Box
  - L'algorithme de cryptage est réalisé à partir d'une succession de P-Box et de S-Box

Le chiffrement par blocs

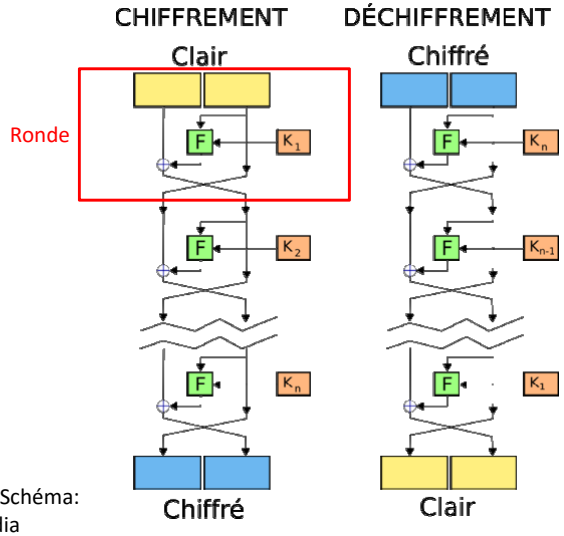
- Exemple d'utilisation de S-Box et de P-Box



Le réseau de Feistel

- Proposé par Horst Feistel (1973), il est à la base de la plupart des algorithmes modernes à clés secrètes (DES en particulier)
- Système de chiffrement par blocs
  - Division d'un bloc en 2 parties égales
  - Chiffrement de la première partie par une fonction F avec une clé K
  - Modification de la seconde partie par un XOR avec la première partie chiffrée
  - Permutation des 2 parties
  - On répète l'opération n fois
- Chaque itération s'appelle une ronde
  - A chaque ronde, la clé va changer pour renforcer le processus
- Le chiffrement et le déchiffrement s'effectuent suivant le même principe

Le réseau de Feistel



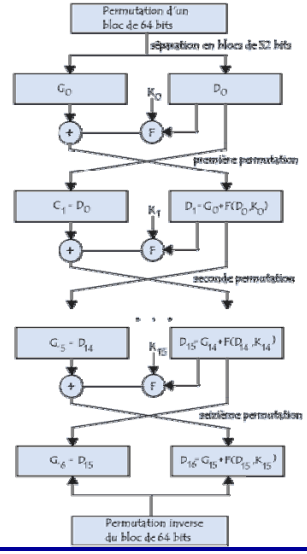
Source Schéma: wikipédia

Le standard DES (Data Encryption Standard)

- L'algorithme DES fut développé au début des années 70 par IBM et fut retenu comme standard par le gouvernement Américain (NSA)
- Il devait répondre aux critères suivants
  - Etre assez simple
  - Reposer sur des clés de petite taille
  - Offrir un haut niveau de sécurité tout en ne nécessitant pas de confidentialité
- L'algorithme DES repose sur
  - un chiffrement par blocs de 64bits (8octets)
    - Chaque bloc est codé séparément puis concaténé aux autres
  - Il n'utilise que des permutations, des XOR et des substitutions
  - Une clé secrète de 64bits (56bits utiles + 8 bits pour le contrôle d'intégrité)

Le standard DES (Data Encryption Standard)

- Algorithme
  - Fractionnements du texte en blocs de 64bits
  - Chaque bloc subit
    - Une permutation initiale
    - Un découpage en deux parties G_0 et D_0
    - Les blocs G et D sont ensuite soumis à 16 rondes
      - ★ $D_{n+1} = Gn \oplus F_{K,n}(Dn)$
      - ★ $G_{n+1} = Dn$
    - Les deux parties sont ensuite recollées
    - Pour finir le bloc subit une permutation inverse de la permutation initiale



Source Schéma: http://www.commentcamarche.net

Le standard DES (Data Encryption Standard)

- La permutation initiale se passe de la manière suivante
  - Le 52ème bit est positionné en première position
  - Le 50ème en seconde position
  - …
- On obtient ensuite G_0 et D_0

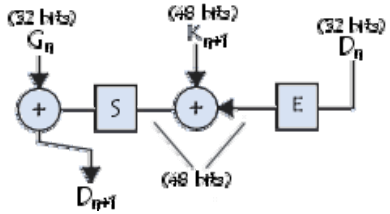
PI							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

G_0							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8

D_0							
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Le standard DES (Data Encryption Standard)

- A chaque ronde on obtient Gn+1 grâce à une fonction F



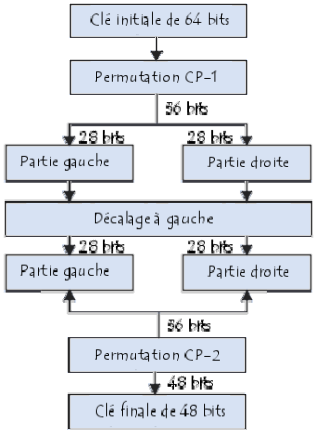
- Cette fonction va démarrer par une expansion E des 32 bits du blocs D_n à 48bits

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	

Le standard DES (Data Encryption Standard)

- Une fois l'expansion réalisée, on peut faire un XOR avec la clé
  - A chaque ronde la clé va changer
  - Algorithme
    - ★ *A partir de la clé de 64bits d'origine on extrait une clé de 56 bits grâce à une permutation*
    - ★ *La clé est ensuite séparée en 2 blocs de 28bits*
    - ★ *Les blocs subissent un décalage de bits vers la gauche*

Ronde	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Nbre de décalage	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1



  - ★ *On effectue de nouveau une permutation*
  - ★ *Et on obtient la clé finale pour la ronde*

Le standard DES (Data Encryption Standard)

- Le bloc D obtenu est ensuite découpé en 8 blocs de 6 bits
- Chaque bloc de 6 bits est passé dans une fonction de substitution S
  - Cette fonction permet de passer de 6bits à 4 bits
  - Pour cela les premiers et derniers bits de chaque bloc nous permet de déterminer la ligne
  - Les bits 2,3,4,5 servent à trouver la colonne
  - Avec la ligne et la colonne on peut trouver une valeur qu'il suffit de coder en binaire
- Attention: La substitution effectuée va changer à chaque ronde

Le standard DES (Data Encryption Standard)

- Ex pour D_{01} = 110111
  - ★ *N° de ligne = $(11)_2 = (3)_{10}$*
  - ★ *N° de colonne = $(1011)_2 = (11)_{10}$*

S_1	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	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

  - ★ *Le bloc en sortie sera donc $(14)_{10} = (1110)_2$*

Le standard DES (Data Encryption Standard)

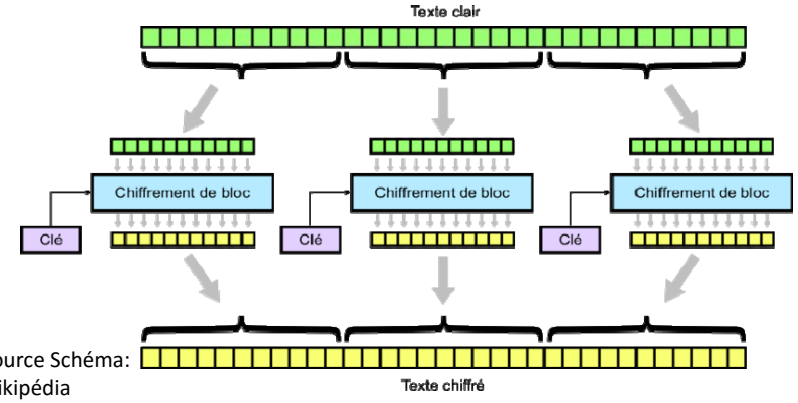
- Une fois les 16 rondes effectuées, les deux blocs G et D sont réunis
- Le bloc résultant subit une permutation inverse de la permutation initiale

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

- On obtient ainsi le bloc crypté et on répète l'opération pour tous les blocs

Le standard DES (Data Encryption Standard)

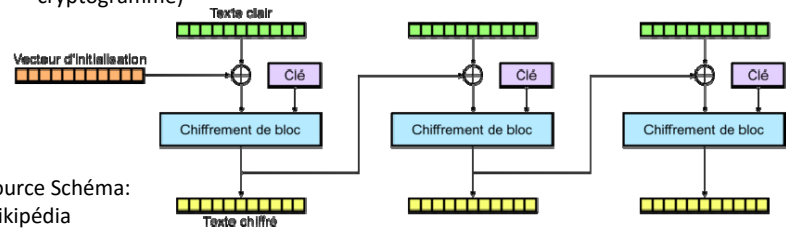
- Historiquement la gestion des différents blocs composants le message se faisait suivant le mode ECB (Electronic CodeBook)
  - Le message est découpé en blocs et chaque bloc est crypté par la méthode DES



Source Schéma: wikipédia

Le standard DES (Data Encryption Standard)

- Le mode ECB est très rapide mais présente quelques limites avec DES
  - Une même message sera crypté de la même manière
  - Sensible aux attaques par rejeu
- On privilégiera en général le mode CBC (Cypher Block Chaining)
  - Dans ce mode on applique à chaque bloc un XOR avec le bloc crypté précédent
  - Pour augmenter la diversité des messages codés, le message en clair subit un XOR avec un Vecteur d'Initialisation (générés aléatoirement et transmis dans le cryptogramme)



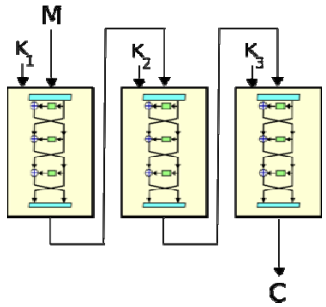
Source Schéma: wikipédia

Le standard DES (Data Encryption Standard)

- Les limitations de DES
  - La taille de clés est assez faible : 2^{56} valeurs $<10^{17}$
    - A l'origine IBM avait recommandé 112 bits pour les clés mais la NSA a ramené cette valeur à 56bits
  - La manière dont ont été conçues les S-Box est assez opaque
    - Une étude basée sur de la cryptanalyse différentielle a montré dans les années 90 qu'elles avaient été bien conçues
  - Depuis les années 70 la puissance de calcul a considérablement progressée
    - En 1998, une machine spéciale appelée Deep Crack a été conçue spécialement pour casser DES
      - ★ *Elle mettait moins d'une semaine pour trouver une clé*
      - ★ *Elle aura tout de même couté 200 000 dollars*
    - Le calcul distribué à grande échelle rend cette technique de force brute beaucoup moins coûteuse
      - ★ *Si l'on considère 1000 PCs à 1Ghz qui fonctionnent en parallèle, il faut compter une trentaine d'heures pour craquer une clé DES par force brute*

Le standard DES (Data Encryption Standard)

- Une amélioration du système consiste à appliquer trois fois l'algorithme DES avec 2 ou 3 clés différentes
  - On parle alors de triple DES (3DES) ce qui revient à avoir une force de 112bits lorsque l'on a 3 clés
    - On ne peut pas obtenir 168bits car le fait d'utiliser plusieurs clé est sensible aux attaques par le milieu



Source Schéma: wikipédia

Le standard AES (Advanced Encryption Standard)

- Aussi connu sous le nom de Rijndael cet algorithme a remporté en 2000 le concours AES lancé par le NIST pour remplacer DES avec trois critères
  - L'algorithme devait résister à toutes les attaques connues
  - Conception simple
  - Code rapide et fonctionnant sur un maximum d'architectures logicielles et matérielles
- Contrairement à DES cet algorithme n'est pas basé sur des réseaux de Feistel et est beaucoup plus simple à implémenter
  - Il travaille sur des blocs de 128bits
  - Autorise des clés de 192 et 256bits

Le standard AES

- Un lien pour mieux comprendre: http://www.formaestudio.com/rijndaelinspector/archivos/Rijndael_Animation_v4_eng.swf

Rijndael_Animation_v4_eng.swf

Les limites de la cryptographie symétrique

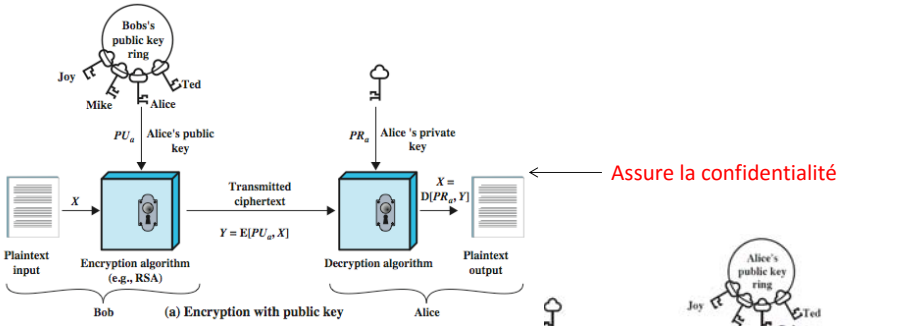
- Les clés ont tendance à se multiplier
  - Il faut une clé différente par canal de communication entre deux entités
  - Comment faire pour échanger les clés ?

- Il n'y a pas de contrôle sur l'origine des messages
  - Si un message est intercepté par une personne possédant la clé secrète, celle-ci peut modifier le message sans que le destinataire n'en soit conscient
  - Il n'y a pas de signatures des messages cryptés avec un simple chiffrement symétrique

Le chiffrement asymétrique

- Pour remédier à ces différents problèmes la cryptographie asymétrique propose d'utiliser un couple de clé plutôt qu'une simple clé secrète
  - Principe découvert par James Ellis (1969) et Witfield Diffie (1975)
- Le couple va se composer
  - D'une clé publique que l'on peut diffuser
  - D'une clé secrète que l'on ne communique jamais
- Le premier algorithme proposé est l'algorithme de Diffie-Hellmann en 1976
  - Objectif de l'algorithme: Echange de clés secrètes
  - Utilisé dans SSL/TLS
  - Scénario type: Alice et Bob veulent s'échanger des informations mais Oscar veut les intercepter
    - Communication sur un canal non sûr:
      - ★ *Oscar va forcément voir passer les échanges*

Le chiffrement asymétrique



Le chiffrement asymétrique

- La construction du couple de clés
  - Alice et Bob choisissent une clé secrète aléatoire qu'ils seront seuls à connaître
  - A partir de cette clé il déduise la clé publique grâce à un algorithme
  - Ils s'échangent leurs clés publiques sur le canal de communication
    - Oscar en a connaissance

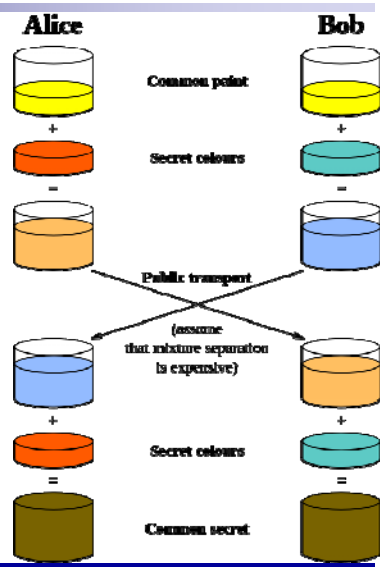
- Le chiffrement du message
  - Lorsqu'Alice souhaite envoyer un message à Bob, elle crypte le message avec la clé publique de Bob
    - On crypte avec la clé du destinataire pour assurer la confidentialité
  - Bob utilisera sa clé privée pour déchiffrer le message

Le chiffrement asymétrique

- La force de l'algorithme repose donc dans la relation entre les deux clés
  - L'objectif est de faire en sorte qu'il soit impossible de retrouver la clé privée à partir de la publique
  - Pour cela on utilise des fonctions unidirectionnelles munies de portes arrières
    - Une fonction unidirectionnelle y=f(x) est une fonction telle que si l'on connaît y il est très difficile voir impossible de retrouver x
      - ★ *Factorisation des grands nombres par exemple*
    - Une fonction est munie d'une porte arrière s'il existe une fonction x = g(y, z) telle que si l'on connaît z, il est facile de calculer x à partir de y
    - Toute la difficulté consiste à trouver f et g ce qui est mathématiquement complexe !
  - Si l'on reprend l'exemple d'Alice et Bob cela veut dire
    - Alice construit un message crypté T à partir d'un message M avec la clé publique de Bob ($cpub_{bob}$) suivant la relation T = f(M, $cpub_{bob}$)
    - Bob retrouve le message grâce à sa clé privée: M = g(T, $cpub_{bob}$, $cpriv_{bob}$)
    - Oscar ne connait que T et $cpub_{bob}$, il ne peut donc pas déchiffrer le message même en connaissant g

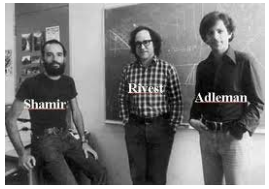
Le chiffrement asymétrique – Diffie Hellmann

- Exemple avec Diffie-Hellmann
  - Repose sur la fonction modulo
    - a et b représentent les clés secrètent d'Alice et Bob
      - ★ *Nombres choisis aléatoirement*
    - g et p le message partagé par Alice et Bob pour générer le secret
      - ★ *Connus d'Oscar*
    - Alice calcule $A=g^a \bmod p$ et le transmet à Bob
    - Bob calcule $B=g^b \bmod p$ et le transmet à Alice
    - Alice calcule ensuite $s = B^a \bmod p$
    - Bob calcule à son tour $s = A^b \bmod p$
    - Alice et Bob possèdent bien le même secret qu'ils peuvent ensuite utiliser comme clé secrète



Le chiffrement asymétrique - RSA

- L'algorithme de Diffie-Hellmann permet donc de partager une clé secrète pour effectuer ensuite un cryptage symétrique
  - Il ne permet pas de crypter une communication de manière asymétrique
  - Il ne fait pas d'authentification
- Le premier algorithme asymétrique permettant de supporter les trois fonctions de cryptage, authentification et d'échange de clés est RSA proposé en 1977
  - Il doit son nom à ses trois inventeurs: Rivest, Shamir et Adleman
  - Il repose la factorisation de deux grands nombres entiers
    - On utilise des nombres premiers
  - Pas de limite sur la taille des clés
    - RSA-512 et RSA-768 ont été cassés en 1999 et 2010
      - ★ *Plus de 5000coeurs utilisés par l'INRIA pour RSA-768*
    - Utilisation de clés d'au moins 1024 ou 2048 bits



Le chiffrement asymétrique - RSA

- L'algorithme de RSA
  - Soit p et q deux nombres premiers distincts
  - On calcule n le produit de p et q : $n = p * q$
  - On tire aléatoirement une clé de chiffrement e telle que

$$e \bmod \Phi(n) \equiv 1$$

e et $\Phi(n)$ sont premiers ce qui revient à dire que pgcd(e, $\Phi(n)$) = 1 avec 1< e < $\Phi(n)$

avec

$$\Phi(n) = (p-1)(q-1)$$

  - La clé pour décrypter notée d est calculée grâce à l'équation suivante

$$e * d \equiv 1 \bmod \Phi(n)$$

  - On obtient ainsi les deux clés:
    - Clé publique: $\{n, e\}$
    - Clé privée: $\{n, d\}$

Le chiffrement asymétrique - RSA

- Pour le chiffrement on utilise la clé publique $\{n, e\}$
  - A partir d'un message M tel que $M < n$
  - On calcule le message chiffré T tel que $T = M^e \ mod \ n$
- Pour le déchiffrement on utilise la clé privée $\{n, d\}$
  - On retrouve M grâce à l'opération $M \ = \ T^d \ mod \ n$
  - Cette relation peut se retrouver en appliquant le théorème d'Euler ou le petit théorème de Fermat

Le chiffrement asymétrique - RSA

- Un petit exemple
  - Prenons p=7 et q=19
  - n = 19*7=133
  - Φ(n)=(19-1)*(7-1)=18*6=108
  - Prenons e tels qu'il soit premier à 108 par exemple 5
  - On obtient d = e^{-1} mod 72 = 5^{-1} mod 72 = 65
    - En effet 65*5 mod 108 = 1
  - Ce qui donne
    - la clé publique(n=133, e=5)
    - la cléprivé (n=133, d=65)
  - Si l'on prend M=6 on obtient T= 6^5 mod 133 = 62

Le chiffrement asymétrique

- L'algorithme RSA est un standard dans le domaine de la cryptographie asymétrique
  - Fiable si la taille des clés est suffisante
    - Peu s'adapter à la loi de Moore en jouant avec cette taille de clé
  - Cependant, un brevet limitait son utilisation jusqu'au début des années 2000
- Une alternative à RSA est l'algorithme d'ElGamal
  - Algorithme non breveté publié en 1987
  - Utilisé dans PGP (Pretty Good Privacy) et GPG (GNU Privacy Guard)
  - Utilisé pour les signatures électroniques dans DSA (Digital Signature Algorithm)

Le chiffrement asymétrique - ElGamal

- Il repose sur
  - une clé secrète s
  - une clé publique (p, g, y) avec
    - g un entier premier de grande taille
    - p un entier premier avec g
    - y un entier obtenu par la relation $g^s \ mod \ p$
- Pour le chiffrement
  - Le message est découpé en blocs compris entre 0 et $p - 1$
    - Chaque bloc est représenté par un entier m
  - On génère au hasard un entier k premier avec $p - 1$
  - On calcule $a \ = \ g^k \ mod \ p$ et $b \ = \ m * y^k \ mod \ p$
  - Le message chiffré est (a, b)
- Pour le déchiffrement
  - $\frac{b}{a^s} = m$

Le chiffrement asymétrique - RSA

- Le chiffrement asymétrique basé sur RSA est donc universel si tout le monde publie sa clé publique
  - On stocke les clés publiques dans des annuaires hébergés chez des tiers de confiance
  - Il ne faut pas que l'annuaire soit compromis sinon tout le dispositif s'effondre
- Un autre propriété de RSA: l'authentification
  - Chiffrement (déchiffrement (message)) = déchiffrement(chiffrement(message))
    - Cela signifie qu'un message chiffré avec sa clé privé peut être déchiffré par la clé publique associé
    - Permet de prouver la source d'un message
- Notion de challenge
  - Pour authentifier Bob, Alice peut lui envoyer un challenge que celui-ci va chiffrer avec sa clé privée
  - Alice vérifie que la clé publique de Bob redonne bien le message d'origine

Symétrique vs Asymétrique

- Comme nous venons de le voir le cryptage asymétrique offre plus de possibilité que le cryptage symétrique
  - Authentification de l'émetteur
  - Echanges sécurisés de clés
- Mais….le chiffrement à clé symétrique est BEAUCOUP plus rapide que le chiffrement asymétrique
  - Avec RSA en utilisant des clés de 1024 on crypte à 300kb/s avec un matériel dédié et à 21,6kb/s avec un logiciel
  - Avec DES en utilisant des clés de 56bits on peut crypter 300Mb/s avec un matériel dédié et 2,1Mb/s avec un logiciel

La cryptographie hybride

- Dans une communication réseau on va privilégier une approche hybride pour le cryptage
  - Méthode asymétrique pour convenir d'une clé secrète
  - Passage à une méthode symétrique une fois la clé secrète établie
- On parle de l'établissement d'une clé de session
- Algorithme
  - Alice génère une clé secrète symétrique qu'elle chiffre avec la clé publique de Bob
  - Bob reçoit le message chiffré et récupère la clé symétrique grâce à sa clé secrète
  - La clé secrète symétrique est maintenant commune à Alice et Bob
    - Cela impose néanmoins qu'Alice possède la clé publique de Bob
- On peut changer de clé de session à chaque communication
- Par contre comment peut-on garantir l'authentification ?
  - On utilise le challenge !

La cryptographie hybride: Authentification + échange sécurisé

- On suppose qu'Alice et Bob possèdent la clé publique de l'autre
- Algorithme
  - Alice chiffre son identité et un nombre n avec la clé publique de Bob
  - Alice envoie le message chiffré à Bob qui peut retrouver n avec sa clé secrète
    - Bob ne sait pas si le message vient bien d'Alice
  - Bob répond à Alice avec un message chiffré grâce a la clé publique d'Alice qui contient le nombre n, un nombre aléatoire p et une clé de session s
  - Alice reçoit le message et le décrypte avec sa clé privée
    - Elle retrouve n, elle est donc assurée de l'identité de Bob
    - Elle obtient la clé de session s
  - Alice renvoie le nombre p chiffré avec la clé de session s
  - Bob reçoit le message et retrouve p, il est donc assuré de l'identité d'Alice

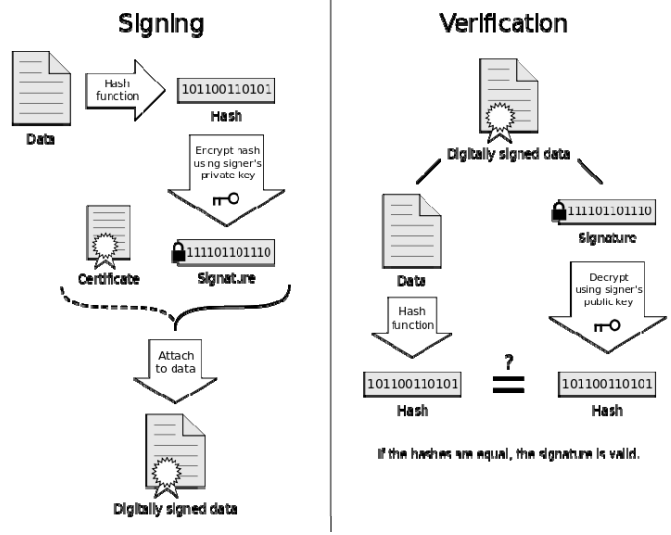
L'authentification des documents

- Comme nous l'avons vu
  - le cryptage symétrique ne permet pas d'assurer l'authenticité de l'émetteur du document
  - Le cryptage asymétrique le permet mais il est très lent
- La solution consiste à compresser le document avant de le signer pour réduire les informations à chiffrer
  - On va produire un résumé du texte d'origine (digest en anglais)
- Pour obtenir le résumé nous allons utiliser une fonction de hachage
  - Elle doit assurer qu'elle associe un et un seul résumé à un texte en clair
    - On parle de fonction sans collision
  - Elle doit être à sens unique
    - Y = f(x) mais impossible de retrouver x à partir de y

Les fonctions de hachage

- Ces fonctions sont très largement utilisées pour assurer l'authentification et l'intégrité d'un message
  - Dans le cas de génération de signature électronique on parle d'authentification
  - Pour vérifier si un document a été modifié on génère son empreinte et on parle alors de contrôle d'intégrité
- La signature électronique (ou sceau) va donc consister à chiffrer le résumé d'un document à l'aide de sa clé privé
  - Le sceau ainsi produit est ensuite ajouté au message à signer
  - Elle doit assurer:
    - L 'authentification
    - La non répudiation
    - L'intégrité
  - Elle doit également être infalsifiable et non réutilisable

Les fonctions de hachage

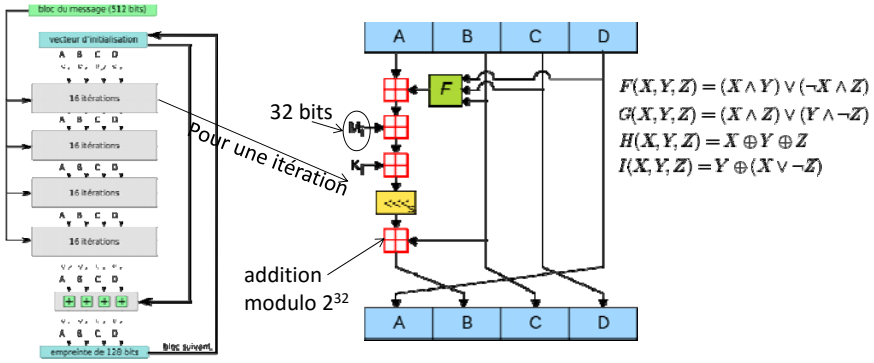


Les fonctions de hachage

- Les principaux algorithmes de hachage
  - MD2, MD4, MD5 (Message Digest)
    - Développé par Ron Rivest de RSA Security en 1991
    - Utilise une empreinte de 128 bits pour les documents
    - Très utilisé pour l'échange de document sur internet même s'il est maintenant considéré comme non sûr
  - SHA1, SHA2, SHA3 (Secure Hash Algorithm)
    - Standard permettant de créer des empreintes max de
      - ★ *160 bits pour SHA-1*
      - ★ *512 bits pour SHA-2*
  - HMAC (Message Authentication Code)
    - Combinaison du hachage avec du chiffrement symétrique

Les fonctions de hachage – MD5

- MD5 travaille sur des blocs de 512bits et va produire en sortie une empreinte de 128bits quelle que soit la taille du message à l'origine
  - Lorsque la taille n'est pas un multiple de 512 on fait du padding



  - Au début de l'algorithme A, B, C, D sont initialisés avec des constantes

Les fonctions de hachage

- Les algorithmes MD5, SHA-0 et SHA-1 sont dorénavant considérés comme non sûrs
  - MD5 n'est pas résistant aux collisions
    - Il est possible de générer à partir de deux fichiers différents des signatures identiques
  - Des problèmes de sécurité de SHA-1 ont été mis en évidence en 2005
  - Bien que SHA-2 repose sur le même principe que SHA-1 il est encore résistant aux attaques
    - Il se décline en deux versions SHA-256 et SHA-512
      - ★ *Attention 256 et 512 n'indiquent pas des tailles de clés mais la taille des blocs sur lesquels l'algorithme fonctionne*

- SHA-3 a été standardisé par le NIST en Octobre 2012
  - Il est recommandé de l'utiliser mais l'utilisation de SHA-2 n'est pas pour le moment remise en cause