Sécurité des SI

Xavier Delannoy

Qu'est-ce que la sécurité ?

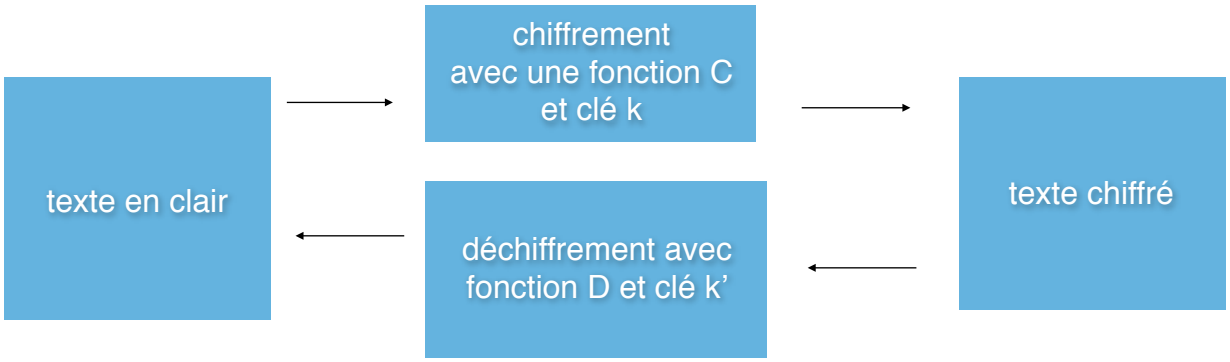
Plusieurs approches, de différents niveau de maturité :

- niveau 0 , c'est du réseau.
- niveau 1 , c'est de la technique ,quand on programme aussi,
- niveau 2 , c'est aussi dans les processus
- dernier niveau, pourquoi fait-on de la sécurité, s'il y a un risque ! La sécurité c'est la réponse aux risques. Des données publiques n'ont pas à être protégées.

Problème de cloud. Est-ce que ça vous embête de stocker des cartes de crédit sur le compte Apple ? oui, non… petit débat.
La sécurité, c'est aussi un problème de l'utilisateur et pas uniquement du RSSI. En voiture, c'est bien au conducteur d'être prudent. De plus, la sécurité est très transverse…! Et est complexe (firewall, base de données, réseaux, …)
On s'intéresse à la sécurité du data processing : sécurité des données, des traitements, des échanges, en commençant par des éléments de cryptographie, de façon pragmatique, sans entrer dans les algorithmes et les méthodes mathématiques en informatique de gestion.

Première partie : Éléments de cryptographie

Quelques éléments de terminologie



(C,D) = crypto système, le chiffre
On peut aussi essayer de casser le texte chiffré.

Attaque texte en clair connu

Hypothèse : l'attaquant dispose de plusieurs couples (texte clair , texte chiffré) : (T1,X1), (T2,X2) On a X3 et on veut T3. Cela a permis à Champollion avec la pierre de rosette, de déchiffrer les hiéroglyphes.

Texte chiffré seul

on a le texte chiffré seul X1 et les fonctions C et D.
Bruteforce : i=1, si Dp(X1) et on voit si c'est un texte intelligible. si c'est ok, on arrête, sinon on passe à i+1.
On peut compliquer les choses pour l'attaquant en changeant les fréquences, en compressant le fichier avant de le crypter (même compresser plusieurs fois) afin de plomber ses temps de calculs. Par exemple, le chiffrement utilise une clé hachée plusieurs fois.

La sécurité

Les points importants de la sécurité :

Disponibilité : ça marche

Intégrité :
Confidentialité
Traçabilité
Preuve, non répudiation. (statique ou dynamique)

HIPS, outil utilisé en sécurité qui permet de bloquer une attaque quand elle apparait. Elle a des signatures d'attaque pour reconnaitre des comportements non standards, des applications qui sortent de leur espace mémoire.

L'antivirus lui recherche des patterns de codes. cependant 70% des malwares sont utilisés sur des cibles uniques et précises. Un antivirus fait du gros œuvre et détecte le bruit de fond. Il fait cependant bien parti de l'arsenal.

La mauvaise sécurité, c'est de mettre un « boîtier » pour chaque problème. Shellshock, Heartbleed, etc…

Il est primordial de désactiver tous les services inutiles, comme un driver usb par exemple. Aujourd'hui, l'avantage est du côté de l'attaquant qui utilise une seule faille, quand la RSSI doit protéger toutes les failles. À la douane, il y a des attaques quotidiennes, et particulièrement les veilles de vacances, de pont etc… Le vrai problème, quand on détecte une attaque, si c'est un antivirus ou un HiPS ça va. Alors que si on trouve quelque chose en fouillant les logs…. ça peut être plus difficile. L'idéal, c'est le réseau distinct ! Mais cela n'est pas possible pour tout le monde.

Retour sur les éléments fondamentaux du chiffrement : Propriétés constitutives d'un bon chiffre

Efficacité : exemple, lorsque les américains ont lancés un concours pour savoir quel chiffre symétrique pourrait remplacer le DES, à utiliser dans l'administration et IBM a été retenu avec lucypher. Puis le DES a été cassé en une journée en force brute et a alors été remplacé, non pas par la plus robuste mais celle qui avait le meilleur compromis entre vitesse et robustesse. Ce chiffre est l'actuel AES.

Confusion : la relation entre d'une part le texte en clair et la clé, et d'autre part le texte chiffré doit être aussi difficile que possible à établir.

Exemple:
(ABC, BCD)
(LMN, MNO)
Peu de confusion, clé k égale à 1.

Diffusion : une modification, même mineure du texte en clair doit se traduire par une modification très importante du texte chiffré.

une bonne diffusion par exemple ABC -> EFG et ABD -> XTA.

Principe de Kerckhoffs

La difficulté de casser un texte chiffré ne doit pas dépendre du secret du cryptosystème mais du secret des clés.

Ce principe apparaît parmi les 6 « desiderata de la cryptographie militaire » énoncés par Kerckhoffs dans son traité, qui sont :

1. Le système doit être matériellement, sinon mathématiquement indéchiffrable ;
2. Il faut qu'il n'exige pas le secret, et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi ;
3. La clef doit pouvoir en être communiquée et retenue sans le secours de notes écrites, et être changée ou modifiée au gré des correspondants ;
4. Il faut qu'il soit applicable à la correspondance télégraphique ;
5. Il faut qu'il soit portatif, et que son maniement ou son fonctionnement n'exige pas le concours de plusieurs personnes ;
6. Enfin, il est nécessaire, vu les circonstances qui en commandent l'application, que le système soit d'un usage facile, ne demandant ni tension d'esprit, ni la connaissance d'une longue série de règles à observer.

Kerckhoffs insiste sur les 3 premiers desiderata, qui sont véritablement originaux à son époque, les 3 derniers n'étant alors pas contestés. Ce qui est appelé aujourd'hui « principe de Kerckhoffs » est essentiellement le second.

« Le principe de Kerckhoffs s'applique au-delà des chiffres et des codes, c'est-à-dire aux systèmes de sécurité en général : tout secret est en fait un point de cassure possible. Par conséquent, le secret est une cause première de fragilité, donc cela même peut amener un système à un effondrement catastrophique. À l'inverse, l'ouverture amène la ductilité. »

Vulnérabilités

De conception
D'implémentation
D'exploitation
Humaines

Typologie des chiffres

Les chiffres symétriques k = k' : ceux par bloc et ceux continue.
Les chiffres asymétriques k ≠ k'

LES CHIFFRES SYMÉTRIQUES PAR BLOC

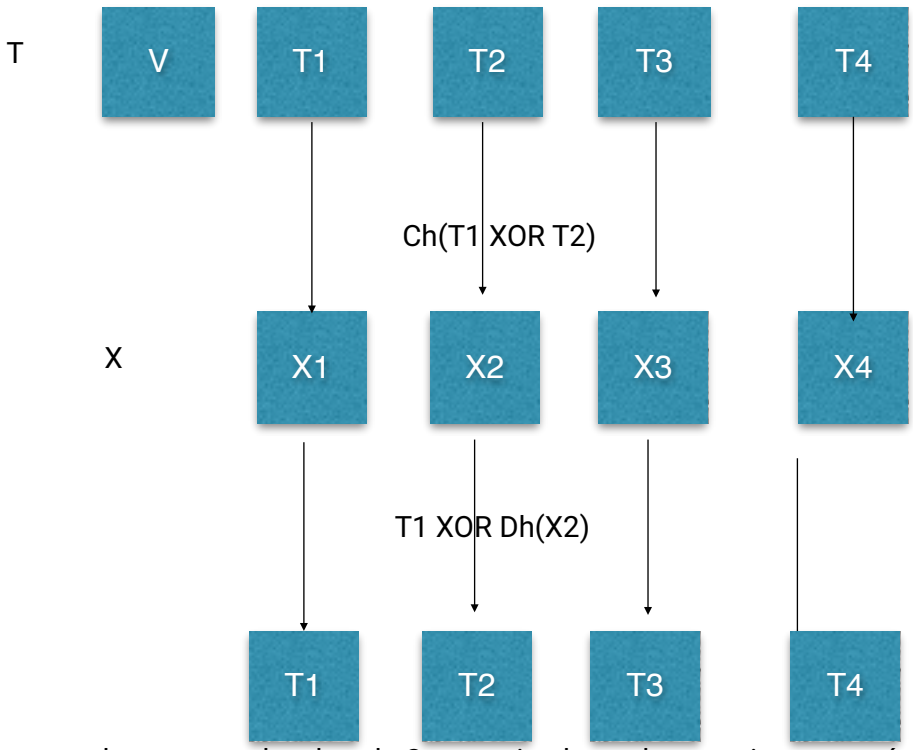
Une liste non-exhaustive de chiffrements par bloc :

- DES, l'ancêtre conçu dans les années 1970, a été passablement étudié
- AES, le remplaçant de DES
- Blowfish, Serpent et Twofish, des alternatives à AES

Le principe

On découpe le texte par bloc et chacun des blocs doit être chiffré
Problème : il y a un invariant, un message identique sera chiffré identiquement. T2 XOR T3. le XOR du XOR est l'identité. T1 XOR Ch(X2) = T1 XOR Dh(Ch(T1 XOR T2)) = T1 XOR T1 XOR T2= T2.
V est le vecteur d'initialisation.

Cypher Bloc Chaining (CBC)



L'asymétrique est beaucoup plus lourd. On essaie donc de revenir au symétrique rapidement, même si on initialise la communication en asymétrique

LES CHIFFRES SYMÉTRIQUES EN CONTINU,

Les chiffrements en continu utilisent un changement de la clé en permanence.

One time pad (le masque jetable) système aujourd'hui le plus robuste. Le one time pad est aussi long que le texte à chiffrer.

1	0	1	0
1	1	0	1
0	1	1	0

xor

1	0	1	1
1	1	0	0
0	1	0	0

=

0	0	0	1
0	0	0	1
0	0	1	0

Pour chiffrer on calcule donc C = A ⊕ B. Le résultat C est le chiffré de A. L'opération est effectuée pour chaque bit du clair avec le bit correspondant de la clé.
Le déchiffrement s'effectue en combinant le chiffré C avec le bit de clé B par la simple opération : C ⊕ B. Il se trouve qu'elle fait retrouver le clair A, comme nous allons le montrer.
Remarquons que l'opération XOR possède les deux propriétés suivantes :

A ⊕ A = 0

A ⊕ 0 = A

ce qu'on vérifie facilement avec le tableau ci-dessus, en considérant les deux valeurs possibles de A, qui sont 0 ou 1.
Le calcul de déchiffrement peut donc s'écrire :

C ⊕ B = (A ⊕ B) ⊕ B = A ⊕ (B ⊕ B) = A ⊕ 0 = A

Il fait bien retrouver le bit de clair A.

Expérimentation concrète d'un algorithme symétrique continu sur un cas concret.