

# • Cryptographie

Car “on a rien à cacher” 🙈 🙉 🙊



## Sommaire

- Définition
- Histoire
  - César
  - Vigenère
  - Enigma
- Concept
  
- Cryptographie moderne
  - Symétrique
  - Condensat
  - Asymétrique
- Authentification
- PGP
- Certification
- Conclusion

“

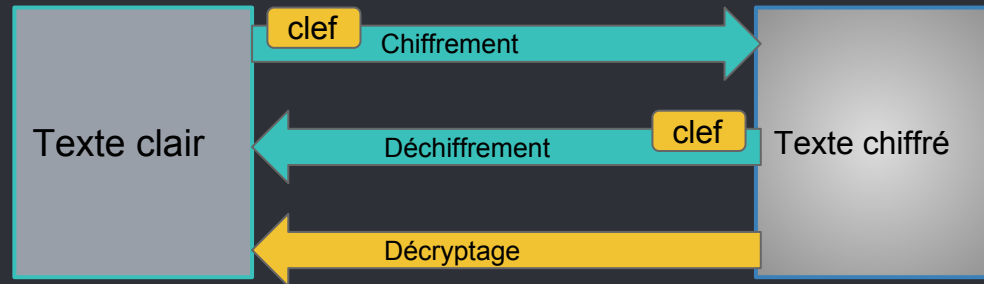
```
$password = 'yourpassword';  
$salt = sha1(md5($password));  
$password = md5($password.$salt);
```

*Les technologies évoluent extrêmement vite, prenez garde aux dates*

```
$password = 'yourpassword';  
$salt = 'randomstring';  
$password = md5($salt.$password);
```

## Définition

- **Texte** : ici il s'agit de tout document (phrases alphabétique/binaires/etc)
- **Chiffrement** : procédé de transformation d'un texte lisible en un autre incompréhensible
- **Déchiffrement** : procédé inverse du chiffrement en respectant le procédé (clef)
- **Crypter**: action de se foutre du monde en prétendant connaître
- **Décryptage** : procédé consistant à transformer le chiffré en clair sans en connaître initialement la clef
- **Cryptologie** : ensemble des techniques permettant de préserver la confidentialité des messages
- **Cryptanalyse** : techniques permettant de déchiffrer un message sans connaître le procédé et/ou la clef



## Fonctions & Utilisations

### - Fonctions :

- Confidentialité : Protection du contenu des messages
- Intégrité : Protection de la cohérence des messages
- Authenticité : Protection de l'auteur des messages

### - Utilisations :

- signature électronique
- messagerie sécurisé
- intégrité d'un téléchargement
- stockage de secret
- vote électronique
- monnaie numérique

Différence à noter, la cryptographie n'est pas de la stéganographie (l'art de masquer)



# Histoire

L'envie de protéger des textes existe depuis très longtemps

Permutation de chaque lettre d'une valeur  $n$  constante.

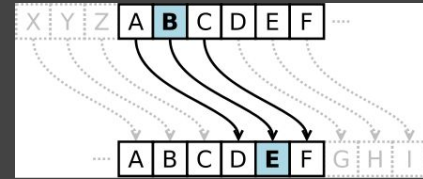
- Déchiffrement :

Permutation de chaque lettre de **n** en arrière.

Possibilités : 26 -> le nombre de lettres dans l'alphabet.

Clef de 3

ABCDZ  $\rightarrow$  A $\gg$ 3, ..., D $\gg$ 3, Z $\gg$ 3 = DEFGC



## Clef de 3

DEFGC  $\rightarrow$  D $\ll$ 3, ... , G $\ll$ 3, C $\ll$ 3 = ABCDZ

## Chiffre Monoalphabétique

- Chiffrement / déchiffrement :

On définit un tableau de correspondance connu de l'expéditeur et du récepteur.

On remplace chaque lettre par sa correspondance et on envoie le texte.

Pour le déchiffrer, on fait l'inverse.

Possibilités :  $4 \times 10^{26}$  -> pour 1 million d'opérations/s, il faut 6 mille milliards d'années en brute force.

A	B	C	D	E	..	Y	Z
T	V	K	A	N	..	D	E

ABCDE -> TVKAN

TVKAN -> ABCDE



## La langue

Chaque langue a ses propres propriétés.

Malgré les grandes possibilités ( $4 \times 10^{26}$ ) aucune protection contre l'analyse de fréquence.

Une simple cryptanalyse sur le texte chiffré suffit à retrouver quasiment toute les correspondances.

Cette vulnérabilité a été découverte au 9e siècle par un cryptologue du nom d'Al-Kindi.

Rang ↕	Caractère ↕	Nombre d'occurrences ↕	Pourcentage ↕
1	e	115 024 205	12,10
2	a	67 563 628	7,11
3	i	62 672 992	6,59
4	s	61 882 785	6,51
5	n	60 728 196	6,39
6	r	57 656 209	6,07
7	t	56 267 109	5,92
8	o	47 724 400	5,02
9	l	47 171 247	4,96
10	u	42 698 875	4,49
11	d	34 914 685	3,67
12	c	30 219 574	3,18
13	m	24 894 034	2,62
14	p	23 647 179	2,49
15	é	18 451 937	1,94
16	...	11 881 118	1,23

TZ ROKT JXT ETKZQOF CTXSTFZ SODOZTK SQ HKGZTEZOGF RTL DTLLQUTL HQK EIOYYKTDTFZ YGKZ

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

T = 13  
Z = 8  
K = 7  
O = 6  
L = 4  
Q = 4  
F = 4  
E = 3  
...

QWVMCNOHPQRSDFIHADLEGCVUFT



# Un problème de taille

Cocorico !

# Chiffre de Vigenère

- Définition d'une clef :
  - prendre un entier définissant la taille  $n$  de la clef :  
-> 4
  - choisir  $n$  lettres :  
-> r-h-i-z  
-> 17-7-8-25

- Chiffrement :

On découpe notre phrase en morceau de  $n$  lettre et on ajoute notre clef a chaque blocs.

- Déchiffrement :

On applique le même algo que pour le chiffrement avec une soustraction.

Possibilités :  $26^n$  clefs différentes.

Chiffrement :

*"j'aime les patates et les crepes"*

**texte à chiffrer** : jaim eles pata tese tles crep es

- $j + 17 = a$
- $a + 7 = h$
- $i + 8 = q$
- $m + 25 = l$

**Texte chiffre** : ahqlvsmrghbzkldksmrtymovz

Déchiffrement :

- $a - 17 = j$
- $h - 7 = a$
- $q - 8 = i$
- $l - 25 = m$

-> jaim eles patates et les crepes

## Encore un problèmes de récurrences ...

Le fait de couper notre texte en morceaux de  $n$  taille crée encore un problème. Encore grâce aux statistiques, on va pouvoir détecter la longueur de la clé puis faire une analyse syntaxique comme pour le chiffre monoalphabétique.

Kasiski définit la méthode suivante:

- En analysant la récurrence des lettres en décalant le texte, on voit clairement des pics de récurrence sur les décalage  $n$  (ici 4) et ses multiples
  - décalage de 4 :

```
gmlizyayrmdwcmdwrtpixmeytcxwcskuhiadomd  
gmilizayrmdwcmdwrtpixmeytcxwcskuhiadomd
```

- décalage de 2 :

```
gmlizyayrmdwcmdwrtpixmeytcxwcskuhiadomd  
gmilizayrmdwcmdwrtpixmeytcxwcskuhiadomd
```

- Il nous reste l'analyse à résoudre sur nos morceaux de texte découpés de la longueur  $n$  :  
notre clef est de 4 : on sépare chaque lettres dans 4 groupes différents.

## ... et de langue

Une autre méthode basé sur l'Indice de Coïncidence (IC) permet de déterminer la longueur de la clef. L'IC détermine la probabilité de répartition des lettres dans un texte.

$$\mathbf{IC} = \frac{\sum_{i=1}^c n_i(n_i - 1)}{N(N - 1)/c}$$

Si un texte est chiffré l'IC sera environ de 0.4

En créant  $n$  textes correspondant à la lecture d'une lettre sur  $n$ , l'IC tendra vers l'IC normal de la langue si  $n$  est la taille de la clef.

Nlgtjtvzggdxolvtlr

ntvgotv

ljzdl

gtgxvr

# Enigma

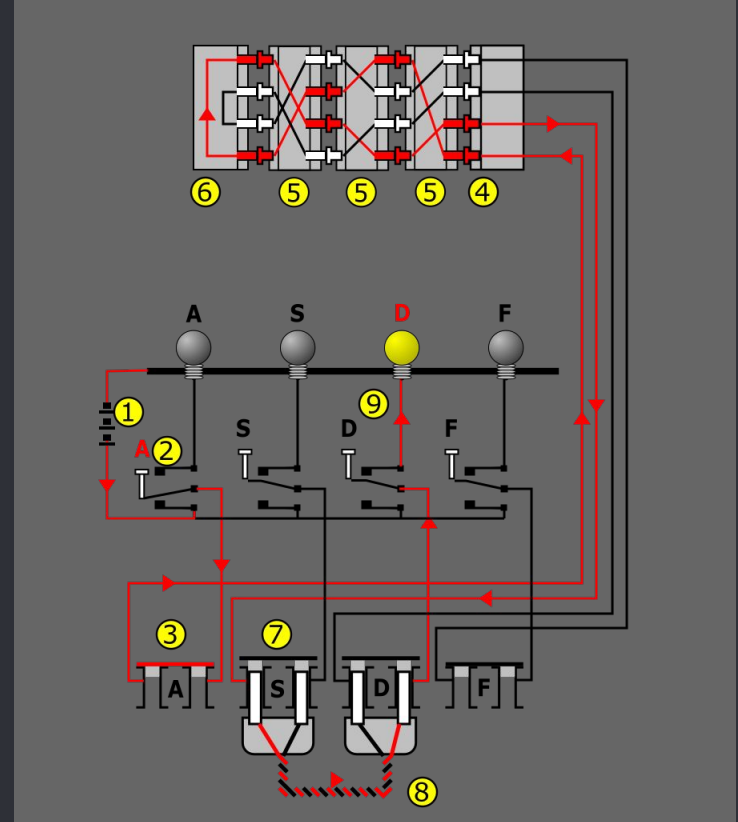
Une avancée importante dans la cryptographie.

Créée en 1918 et perfectionnée avant d'être utilisée par les allemands en 1930.

Composition cryptographique :

- 3 disques de brouillage interchangeables permettant de faire un chiffrement polyalphabétique de 17576 correspondances
- 20 lettres interchangeables qui augmente le nombre de possibilités
- 1 tableau réflecteur pour simplifier l'utilisation mais insère une faiblesse, une lettre ne peut être chiffré par elle même.

Possibilités : 159 milliards de milliards, 2000 fois plus que le DES inventé 40 ans plus tard



## Cruciverbiste

### Redondance :

- Facilite le déchiffrement
- Permet la correction
- Permet la reconstruction avec quelques lettres

Les langues écrite ont une redondance d'environ 75%.

### Utilisation :

- Iban - code insee

*La clef est ajouté volontairement pour ajouter de la redondance et éviter toute erreur*

### Mots probables:

- Permet aussi de deviner facilement les lettres suivantes.
- Très utile pour casser les algorithmes présentés avant, et étroitement liés aux redondances.

### Utilisation:

- Les mot croisés

*Si je découvre les lettres **LUN** il y a de forte chance que je trouve **LUNDI***





# Concepts cryptographiques

Point important pour choisir son algorithme

“

*La sécurité d'un procédé cryptographique ne doit pas dépendre de l'algorithme lui-même mais uniquement de la préservation de la clé.*

*Auguste Kerckhoffs*

## ● Auguste Kerckhoffs

### ○ Traité de cryptographie militaire (1883)

- Solidité mathématique et **pas de sécurité par l'obscurité**
  - Oui c'est à toi que je parle, NSA!
- Clefs multiples et mémorisables
- Applicable aux communications modernes
- Portatif
- Utilisable par un débile

## Claude Shannon

### Communication Theory of Secrecy Systems (1949)

- A partir d'un chiffré donné on ne peut pas associer un clair donné avec une probabilité supérieure à un autre clair.
  - et réciproquement
- Aucun élément du clair ou des données l'accompagnant ne doit permettre de déchiffrer le clair
  - et... on verra ça plus tard!

## ● Concepts supplémentaires

- Avalanche
  - Modifier un octet du clair doit modifier l'intégralité du chiffré
  - Applicable au Hash
- Perfect Forward Secrecy
  - Le secret des messages passés n'est pas compromis par la divulgation de la clef privée à long terme.
- Plausible Deniability and Non Repudiation
  - Concepts opposés,
    - l'un permettant d'affirmer que le message a pu être forgé
    - l'autre qu'il n'a pas pu l'être



# Chiffrement moderne

Le chiffrement à clef secrète □

## Chiffrement moderne

- La sécurité par le nombre
  - La sécurité du chiffrement moderne réside dans la difficulté à résoudre des problèmes mathématiques
  - Tant que le problème reste la sécurité aussi...
  - Le nombre d'opérations possibles augmente chaque année
- Et concrètement
  - On considère que  $2^{128}$  opérations (soit 128 bits de sécurité) est suffisant
  - C'est  $3.4 \cdot 10^{38}$  opérations plusieurs années pour un ordinateur traditionnel.
  - Mais les problèmes ne sont pas toujours insolubles...

## ● Chiffrement symétrique

- Fonctionnement
  - Le chiffrement symétrique consiste à utiliser une même clef pour chiffrer et déchiffrer
  - La sécurité réside dans la taille de la clef pour éviter le bruteforce
- Dangers
  - L'échange de la clef doit être fait de manière sécurisé
  - L'algorithme doit avoir une sécurité équivalente à la taille de la clef
  - L'algorithme doit être sécurisé



## Chiffrement par flot

- Fonctionnement

Le système est des plus simples et sécurisé, un XOR entre la clef binaires et le texte en binaires. Pour le déchiffrement, même chose.

Le système nécessite de se mettre d'accord sur un flot de chiffrement commun. On utilisera pour cela un PRNG (pseudo random number generator)

- Pour être valide :
  - Les deux correspondants doivent avoir le même flot
  - Les flots doivent être différents à chaque fois
  - Les flots ne doivent pas avoir de bits redondants
  - Les flots ne doivent pas se croiser 🌀

## Chiffrement par flot

- Chiffrement

On prend notre texte bit a bit et on ajoute la clef bit a bit aussi.

- Déchiffrement



On prend notre clef que l'on soustrait au message chiffré.


Possibilités : infinie

Fail: infinie

Flot commun : 1001110 ... 0010011101

Message : hello > 1101000 ... 1101111 1010

	T	1	1	0	1	0	0	0	...
	C	1	0	0	1	1	1	0	...
<hr/>									
	>	0	1	0	0	1	1	0	...

	>	0	1	0	0	1	1	0	...
	C	1	0	0	1	1	1	0	...
<hr/>									
	T	1	1	0	1	0	0	0	...

## ● Chiffrement par bloc

- Contrairement au chiffrement par flot qui nécessite un flot aussi grand que le texte pour être sûr, le chiffrement par bloc permet de chiffrer par sous unités de texte tout en gardant une sûreté relative.
- Le chiffrement par bloc permet aussi un partage des opérations des calculs en fonction du **mode de chiffrement** choisi.
- Il sert pour chiffrer :
  - Votre disque dur
  - Votre connexion à Facebook (je vous vois)
  - Vos sextos et vos dickpics sur WhatsApp 📱🔒

## DES Data Encryption Standard

Conception : IBM - NSA en 1976

Chiffrement en blocs de 64 bits avec 16 rondes (étape de modification) et une clef de 64 bits (56 réellement car le dernier octet est réservé pour la détection d'erreur)

Une variante, le triple des permet une augmentation de la fiabilité. avec 3 clefs différentes. Contrairement à ce que l'on peut penser, cette méthode équivaut une sécurité de 112 bits ( $2^{56}$ )

Possibilités :  $2^{56}$  ( $\times 2^{56}$  pour le 3des)

Design Fail : sweet32

Concours DES avec la rapidité de cassage en fonction des années

	Date du défi	Durée
DES Challenge I	mars 1997	95 jours
DES Challenge II	janvier 1998 Juillet 1998	39 jours 3 jours
DES Challenge III	janvier 1999	22h 15 mn

# AES Advanced Encryption Standard

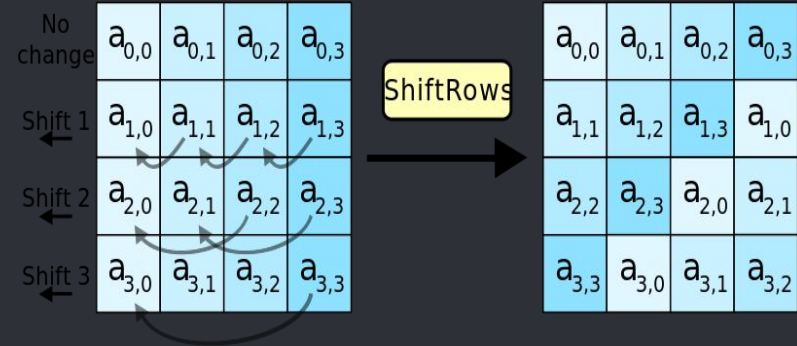
Conception : Rijndael (Vincent Rijmen et Joan Daemen) en 1999

Chiffrement en blocs de 128 bits avec des clefs de 128 / 192 / 256 avec 10 / 12 / 14 rondes.

L'algorithme est fiable, malgré de possibles failles purement mathématiques.

L'AES 256 est utilisée pour la protection de document top secret, par exemple.

Possibilités :  $2^{128}$





# Chiffrement moderne

Le chiffrement à clef publique 🗝️

## ● Principe de sens uniques

○ Les condensats vont permettre de protéger l'intégrité du texte

- Il est basé sur une fonction de transformation à sens unique  $f(x) = y$  tel qu'il soit impossible de déterminer  $x$  sachant  $y$  sauf à tester tous les  $x$  possibles.

### Exemple de sens unique :

J'écris mon message sur une feuille et je la passe au mixeur et envoie mon message (*hashé menu*) à mon correspondants. Si on intercepte mon message il sera très difficile de le recréer ... mais pour mon correspondant aussi.



## ● Empreinte à sens unique (hash)

○ 3 principes:

- Il est facile de calculer l'empreinte d'un document
- Il est difficile de reconstituer le document à partir de l'empreinte
- Si on connaît l'empreinte, il est difficile de soumettre un document différent avec la même empreinte

Les algo de hash :

- Md5, sha1 ... ne sont plus bons pour cacher un secret mais restent pratique pour vérifier l'intégrité d'un document
- Un tas d'autres sont encore valables : blowfish, sha-256, etc ...



## ● Empreinte à sens unique (hash)

### ○ Utilisation:

- Identification ( hash du password stocké au lieu du password en clair)
- Comparaison : comme pour les clefs, si 1 bit du document change, plus de la moitié du hash doit changer
- Détection de modifications / altérations : si mon document n'a pas la même empreinte que celui annoncé, j'ai peut-être eu un problème de téléchargement

### Précautions :

- Un hash seul peut faire l'objet d'une attaque par dictionnaire ou rainbow tables

*Ex: pour éviter cela, il est recommandé d'ajouter un sel!*
- Comme pour le chiffrement, si l'algo doit rester secret, alors l'algo n'est pas utilisable => □

## ● Chiffrement asymétrique

### ● Fonctionnement

- Le chiffrement asymétrique consiste à utiliser une clef pour chiffrer et une autre pour déchiffrer
- Ces deux clefs sont la clef publique et la clef privée
- Ce qui est chiffré par l'une n'est déchiffrable que par l'autre
- La clef privée étant privée, elle peut servir pour s'authentifier

### ● Dangers

- Cela viole la théorie de Shannon
- La clef publique est souvent dérivée de la privée
- Shor va détruire le monde

## RSA :

### Conception :

On choisit deux nombres premiers,  $p$  et  $q$  que l'on garde secrets et on partage le produit de ces 2 nombres.

RSA à plusieurs niveaux de sûreté grâce à une clef qui peut prendre une longueur paramétrable. Plus la clef est longue, plus le chiffrement/déchiffrement/solidité sera long

A savoir que RSA-1024 est le dernier à avoir été factorisé. Actuellement, RSA-4096 est plutôt recommandé

Nos clefs sont composées :

- Pour la privée, de  $n$  (produit de deux nombres premiers  $p$  et  $q$ ) et un nombre  $d$  choisi mathématiquement
- Pour la publique, de  $n$  et un nombre  $e$

- Chiffrement:

Je chiffre mon message avec la clef publique.

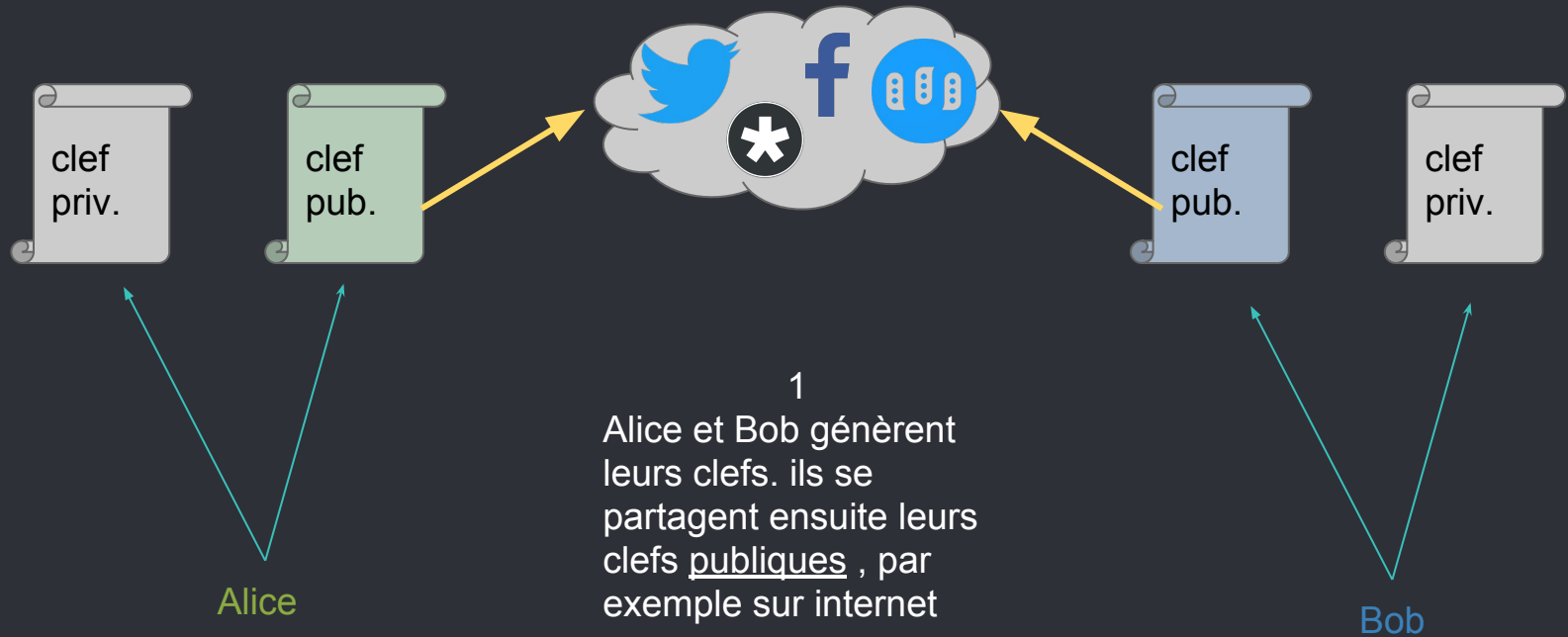
- Pour déchiffrement

Je prend ma clef privée et je déchiffre

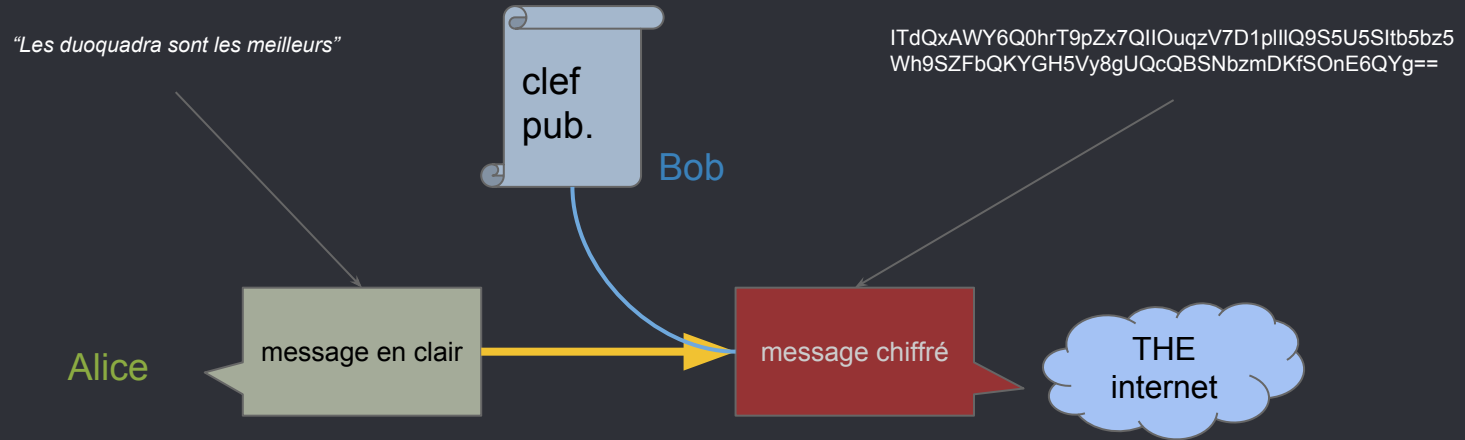
-----BEGIN RSA PRIVATE KEY-----  
MIIBOgIBAAJAADbsl8wx+z6S6bnSv0grd5l5r  
OGUaaQsN3uMt8ws+tGJavAnffB8  
fRWAoSPQIsA82+AC5JvfPE+KcdILQpsS0Q  
IDAQABAKBVw/XcS2IK2slvXsWJNGzJ  
X+bIV1SbyEe3dY9uZ7MQyzpRNQDYHT53  
K/Uc7OueEjJCNIHi+6oOvHQDHqxNExSN  
AiEApGWQ3ob2ACzPihnxTrDkxll0rJ3o1NyD  
rx2jTruXegMCIQCiSKWc7Jov0Vdz  
zA3PXCxLITQaluKACDs26e89EO4RmwIhAl  
MsFz+3aCoTlzWWJZioRKKPVi01gkX1  
/YyIIXmI7QCJAiA6wrde1N8XhWN3CGHPvX  
5ETeeBtzPwTAq03YvEvsVFQIhAJ54  
YCnYUN6w57PZd79h8gRpsgjwQtSQUSTFp  
ucsoNku  
-----END RSA PRIVATE KEY-----

-----BEGIN PUBLIC KEY-----  
MFswDQYJKoZIhvcNAQEBBQADSAwRwJAAD  
bsl8wx+z6S6bnSv0grd5l5rOGUaaQs  
N3uMt8ws+tGJavAnffB8fRWAoSPQIsA82+AC5J  
vfPE+KcdILQpsS0QIDAQAB  
-----END PUBLIC KEY-----

## Schéma d'utilisation de Rsa pour un échange de message



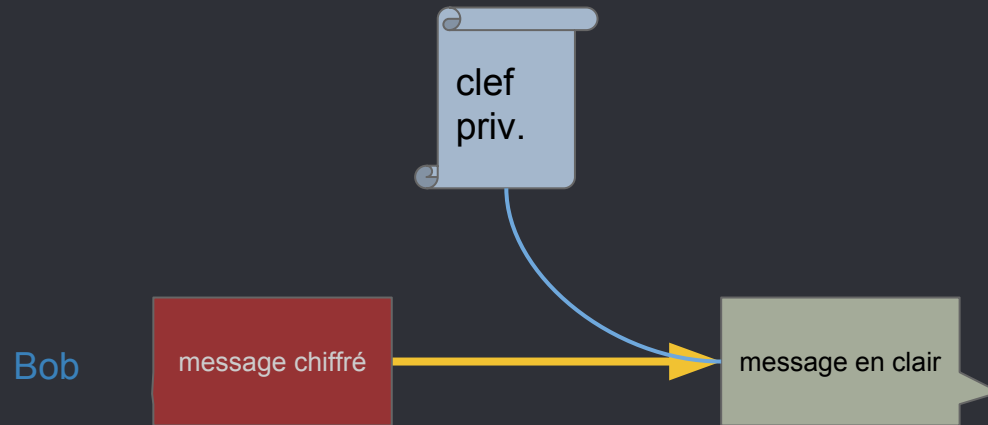
## Schéma d'utilisation de RSA pour un échange de message



2

Alice prend la clef  
publique de Bob et chiffre  
son message

## Schéma d'utilisation de RSA pour un échange de message



3

Bob prend sa clef privée  
déchiffre le message d'Alice

## Quel type choisir ? Symétrique vs Asymétrique

Symétrique :

1 seule clef au total

- Rapide
- Clef généralement limitée dans l'ordre des centaines de bits
- Clef à courte durée de vie, clef «jetable»

Asymétrique:

2 clefs au total

- Permet de supprimer le problème du partage des clefs
- Clef longues, de l'ordre des milliers de bits
- Clef avec une durée de vie longue





# Authentication

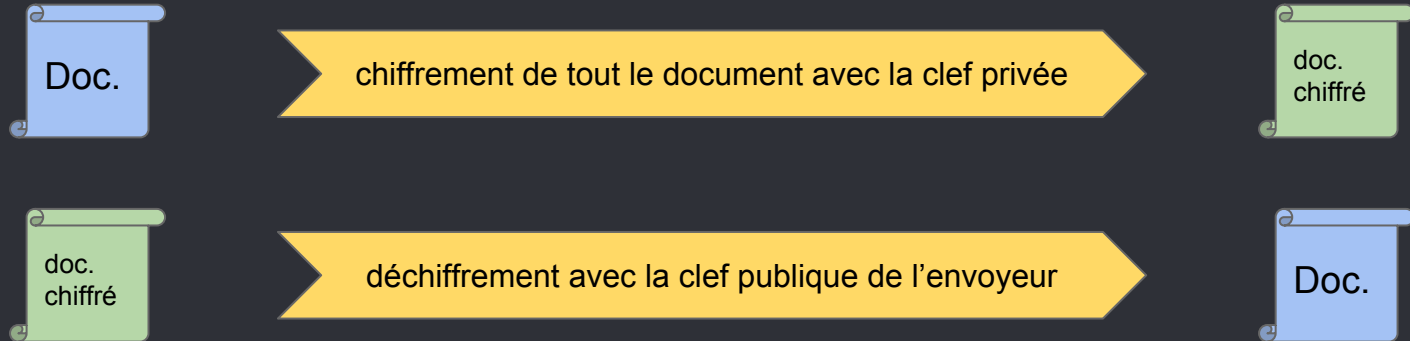
Ou comment savoir si c'est la bonne personne qui me répond

## Signature numérique (**Ce qu'il ne faut pas faire**)

2 garanties:

- Authenticité : la personne est en accord et ne peut pas désavouer ultérieurement le document signé
- Intégrité : le document n'a pas eu de changement après signature.

⚠ La signature en elle même ne protège pas la confidentialité du document.



## Empreinte signé

Signature avec RSA d'une empreinte d'un document



Avantage :

- Même authenticité que de signer le document en entier
- Plus rapide que de chiffrer tout le document



# PGP : un système mixte

Pretty good privacy

## ● Un programme un peu trop sécurisée d'après certain

○ Créé en 1991 par Philip Zimmermann, PGP est un logiciel commercial complet.

Rendu public en juin 1991 après de nombreuses pressions de la part des autorités américaines, il connu un grand succès du fait des circonstances.

3 fonctions, 3 étapes :

- Échange de clef symétrique sécurisée avec le système RSA
- Chiffrement/déchiffrement avec IDEA pour de la rapidité
- Signature avec RSA

Autre point fort du protocole, la gestion des échanges de clefs : Au lieu de se baser sur un serveur central d'autorité (annuaire de clef), il y a une mise en place d'une base locale et un système de confiance entre utilisateurs

## Les Fonctions

1. Création des clefs RSA (public/privée) a minima 2048 : on utilise RSA pour chiffrer la clef de session qui sera utilisée par le chiffrement symétrique. Tant que la clef privé n'est pas compromise, pas de changement de clef
2. Le chiffrement : avec algorithme de chiffrement symétrique par bloc , qui va chiffrer un message. (la clef sera utilisée une seul fois)
3. La signature: le message est hashé et chiffré avec la clef privée de l'expéditeur

## eFail

Utilisation de code HTML pour exfiltrer des données en chargeant du contenu externe.

```
From: attacker@efail.de
To: victim@company.com
Content-Type: multipart/mixed;boundary="BOUNDARY"

--BOUNDARY
Content-Type: text/html


--BOUNDARY--
```

Interpréter HTML automatiquement c'est mal



# Certifier une clef publique

Ou comment être sûr que la clef appartient à Alice



## 1. Autorités de certification

Afin de certifier et d'utiliser une clef publique, on utilise des Autorités de Confiance.

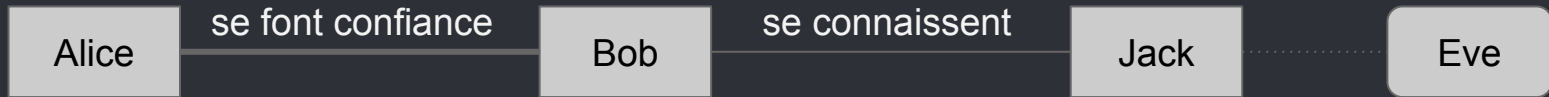
L'autorité signe avec des niveaux de certifications différents: **DV**, **OV**, **EV**.

Dans le cas général, il s'agit d'autorités embarquées sur les terminaux dans un magasin: **Digicert**, **Verisign**, **Diginotar**...

Le problème réside dans le fait qu'il soit nécessaire d'avoir confiance dans les autorités en question!

## 2. Friend-2-Friend

Afin de ne pas se baser sur un tiers, ce système est basé sur la confiance entre membres. PGP est justement basé sur ce système.



Bien sûr, PGP a rendu cette option paramétrable en fonction de la confiance que l'on souhaite accordés à la personne. Dans ce système, chaque noeud est une autorités



## Dans la réalité ...

Certains utilisent encore md4 pour des mdp ... mais ne faites pas ça

## Les vulnérabilités des applications

Il est important de se rappeler que le système cryptographique n'est qu'une des parties sensibles possible

Attaques possibles :

- Le bruteforce: de clef avec logiciel ou matériel spécifique
- La faiblesse d'une chaîne: c'est bien d'avoir un système cryptographiquement sûr mais il faut aussi que le niveau général de sécurité soit haute
- Les utilisateurs: basique, admin:admin n'est pas un bon mot de passe admin
- Les backdoors : efficace surtout quand le code source des algos n'est pas ouvert
- Les zero-day
- ...



## Que conclure

Mis à part que la cryptographie c'est génial et passionnant

## Conclusion

- L'histoire a forgé la cryptographie, utilisée au début à des fins militaires
- L'histoire et la technologie modifieront la cryptographie
- Un système n'est pas sécurisé à 100%, si c'est le cas, c'est qu'on vous a pas tout dit
- L'algorithme doit pouvoir et doit être publique
- Suivre les news, c'est un domaine en progression constante
- Faire attention aux «nouveaux algorithmes» pas encore approuvés et éprouvés
- Oublier le DIY en matière d'algorithmes

“

*Que se passerait-il si tout le monde estimait que les citoyens honnêtes devraient utiliser des cartes postales pour leur courrier ? Si un non-conformiste s'avisait alors d'imposer le respect de son intimité en utilisant une enveloppe, cela attirerait la suspicion. Peut-être que les autorités ouvriraient son courrier pour voir ce que cette personne cache. Heureusement, nous ne vivons pas dans ce genre de société car chacun protège la plupart de son courrier avec des enveloppes. Aussi personne n'attire la suspicion en protégeant son intimité avec une enveloppe. La sécurité vient du nombre. De la même manière, ce serait excellent si tout le monde utilisait la cryptographie de manière systématique pour tous ses e-mails, qu'ils soient innocents ou non, de telle sorte que personne n'attirerait la suspicion en protégeant l'intimité de ses e-mails par la cryptographie. Pensez à le faire comme une forme de solidarité.*

*(Philip Zimmermann, Pourquoi j'ai écrit PGP, 1991, 1999)*

## sources

<https://gchq.github.io/CyberChef/> >\_ couteau suisse pour cryptologie

<http://www.commitstrip.com/fr/> >\_ couteau suisse de blague

<https://fr.wikipedia.org> >\_ car c'est ton deuxieme amis

<http://www.nymphomath.ch/crypto/menu/index.html> >\_ Blog sur la crypto. n'est plus à jours depuis 20016

<http://www.dcode.fr/> >\_ boite a outils bien pratique

<https://piratefurieux.wordpress.com/category/cryptographie/>