CRYPTOGRAPHIE

3 - ctlf.rpobakotiy.coo

CRYPTOGRAPHIE INFORMATIQUE

CONCEPTS IMPORTANTS

Vocabulaire

Pour pas dire “encrypter”

Message clair (Plaintext): Des données quelconques, lisibles par tous.

Message chiffré/Cryptogramme (Ciphertext): Un message clair qui a été obfusqué pour le rendre illisible par un parti externe - tout en restant lisible à des partis de confiance.

Clé/Secret: Une donnée secrète partagée par les partis de confiance.

Chiffre (Cipher): Une méthode d'obtention d'un cryptogramme à partir d'un message clair, en utilisant une clé.

Chiffrer: Message clair -> Cryptogramme

Déchiffrer: Cryptogramme -> Message clair

BASE64 – PAS UN CHIFFRE

… et sa cousine la base32

- Représentation de données non-affichables
- Plus efficace que l'hexa
- Très utilisé dans le web
- Reconnaissable par les = de padding à la fin

Plaintext:
tiens voila mon zob

Hex:
7469656e7320766f696c61206d6f6e207a6f62

Base64:
dGllbnMgdm9pbGEgbW9uIHpvYg==

XOR – Un chiffre!

eXclusive OR

Opération binaire (bit par bit)

- **Très** utilisé en cryptographie

En particulier, quels que soient A, B et C (strings, nombres, bits):

- Si **A XOR B = C**:
- **B XOR C = A** et **C XOR A = C**

	0	1
0	0	1
1	1	0

HASHS – Pas un chiffre

… Mais importants

Propriétés d'un hash:

- Un hash est la **représentation** d'une donnée selon un certain algorithme.
- La même donnée doit toujours donner le même hash.
- Modifier la donnée d'un octet doit *généralement* complètement modifier le hash complètement.
- **Résistance pré-image**, il doit être très difficile de retrouver la donnée originale à partir du hash.
- **Résistance collisions**: il doit être très difficile de trouver deux données qui donnent le même hash.

```
~|⇒ echo -ne "zob" | md5sum
36f3b2748cea3f5714d849889bb4a0c7  -
~|⇒ echo -ne "zab" | md5sum
46f6e8fb2d9082a6bff9070b7aee619c  -
```


HASHS CONNUS

Hashashins Creed

SHA: Une famille de hash **cryptographiques** reconnue. SHA-256 est le “standard” actuellement.

MD5: Un hash très utillisé pour les fichiers car très rapide, mais pas approprié pour une utilisation cryptographique.

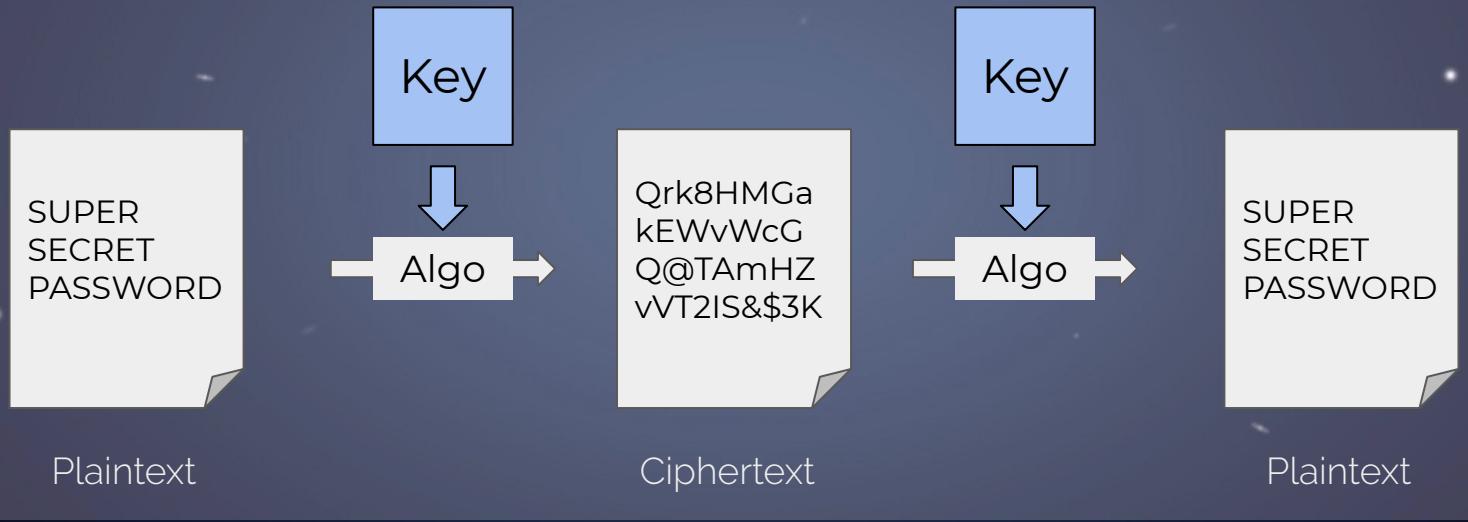
Même pour MD5, généralement, la meilleure façon de craquer est une attaque par dictionnaire, où le hash est comparé à des hashs connus. Sinon, il faut essayer tous les inputs manuellement (brute-force) jusqu’à retrouver le bon hash.

CHIFFREMENT SYMÉTRIQUE

Une clé pour les gouverner tous

L'ENVOI D'UN MESSAGE CHIFFRÉ

Pour envoyer ses dick pics tranquille



L'ENVOI DE LA CLÉ

C'est problémant

ENVOI EN PHYSIQUE

- En personne ou par courrier
- Problèmes de logistique
- Problèmes de sécurité évidents

ex: Enigma

CHIFFRER LA CLEF

- Comment?
- Et surtout, comment on envoie la clé de la clé? Même problème.
- Besoin d'une autre méthode

ex: Utiliser RSA

LES MATHS

- Génération de clé unique
- Communications publiques
- La clé finale est secrète

ex: Diffie-Hellman

LES PREMIERS CHIFFRES

OLD SCHOOL COOL

Rail Fence

Pas mal pour le jardinage aussi

WE ARE DISCOVERED, FLEE AT ONCE

```
W . . . E . . . C . . . R . . . L . . . T . . . E
. E . R . D . S . O . E . E . F . E . A . O . C .
. . A . . . I . . . V . . . D . . . E . . . N . .
```

WECRLTEERDSOEEFE AOCAIVDEN

Chiffrement par transposition (réorganisation)

- La clé est **un nombre**, le nombre de lignes sur lesquels on écrit le message (ici, 3)
- Pas très utilisé historiquement

SUBSTITUTION

Put me in, coach

Table de substitution

La clé ici est une table de substitution qui map chaque caractère à un autre caractère. Par exemple, A devient H, X devient J, etc.

Assez utilisé historiquement, rendu obsolète par les **analyses de fréquence** sur du texte assez large!.

Plaintext:
upon this basis i am going to show you how a bunxh of bright young folks did find a xhampion a man with boys and girls of his own.
Key:
G A B S L Y T E X U C F H I J K Z M N O P Q R D V W
Cryptogram:
pkji oexn agnxn x gh tjxit oj nejr vjp ejr g apide jy amx teo vjpit yjfcn sxs yxis g deghkxji g hgi rxoe ajvn gis txmfn jy exn jri.

LE CHIFFRE DE CÉSAR

L'empire contre-attaque

Chiffrement par décalage (monoalphabétique)

- Une des plus anciennes méthodes
- Chaque lettre est décalée de N lettres dans l'alphabet
- Indéchiffrable pour ces illettrés de barbares

La clé ici est **un nombre** entre 1 et 25, on décale de ce nombre dans l'alphabet.

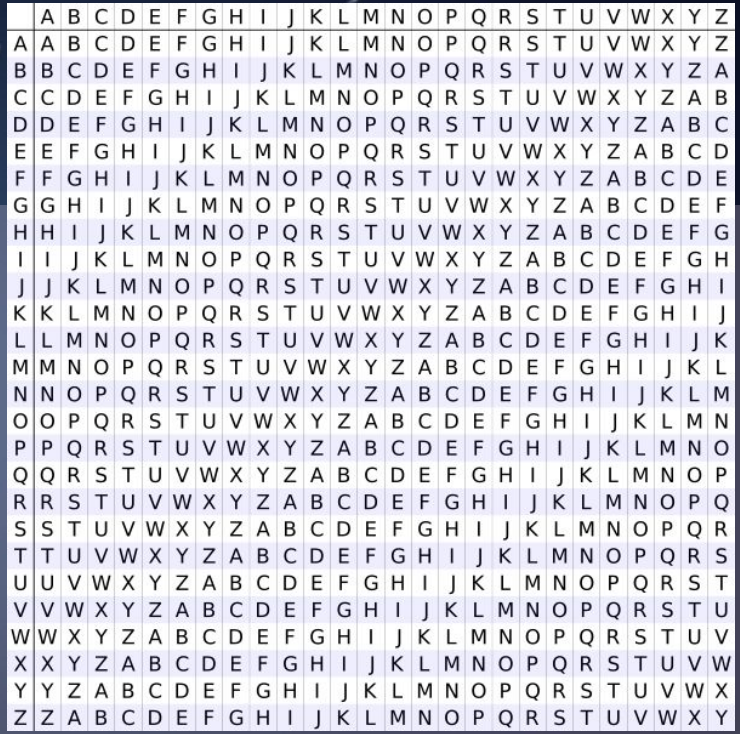
CHIFFRE DE VIGENÈRE

Un César pour les bonhommes

Chiffrement par décalage (polyalphabétique)

- Même principe que César mais le nombre de décalage est **variable**
- Ce nombre est donné par une lettre de l'alphabet, A=0, B=1 etc
- Clés courtes

La clé ici est **une phrase** dont chaque lettre décrit une valeur de décalage.



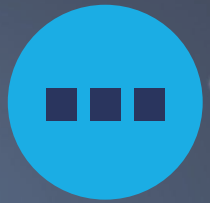
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T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

En informatique moderne

Comment utiliser la clé?

Chiffrement de flux

Les données sont chiffrées en continu, octet par octet par exemple, par la clé.



Chiffrement par bloc

On divise les données en blocs de n caractères et on les chiffre bloc par bloc.

Chiffrement par bloc

Block cipher? I hardly knew ‘er

Un chiffre historique sous stéroïdes

Rien de nouveau sous le soleil

Le chiffrement par bloc est:

- **Symétrique**, il n'y a donc qu'une clé
- Très, très **rapide**
- Très **employé** aujourd'hui.
- Divisible en des **étapes logiques**, comme un chiffre historique.
- On doit appliquer du padding à la fin d'un input qui ne rentre pas tout pile dans les blocs… Un peu comme un Rail Fence!

Par exemple, des permutations de lignes, de colonnes, de caractères…

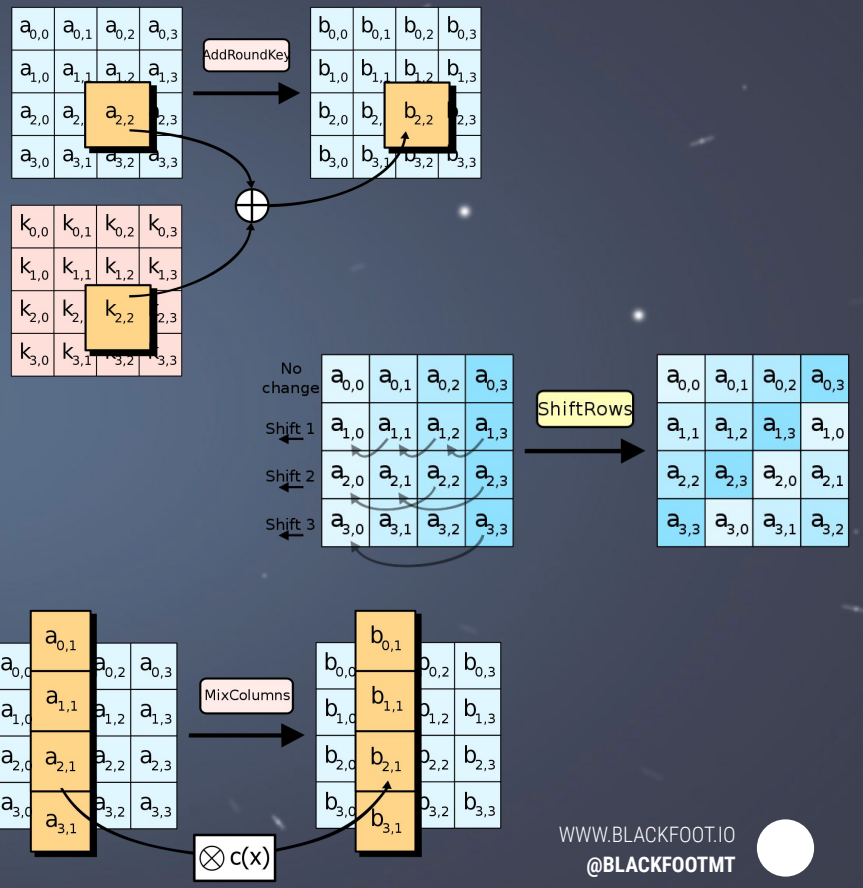
AES – Advanced Encryption Standard

Comment ça marche?

Plusieurs opérations répétées plusieurs fois:

- XOR de la clé et du bloc (*AddRoundKey*)
- Substitution des octets du bloc d'après une table de substitution (*SubBytes*)
- Décalage de rangées (*ShiftRows*)
- Mélange de colonnes (*MixCols*)

Différents algos selon la **taille de la clé**. Le plus utilisé est AES-256.

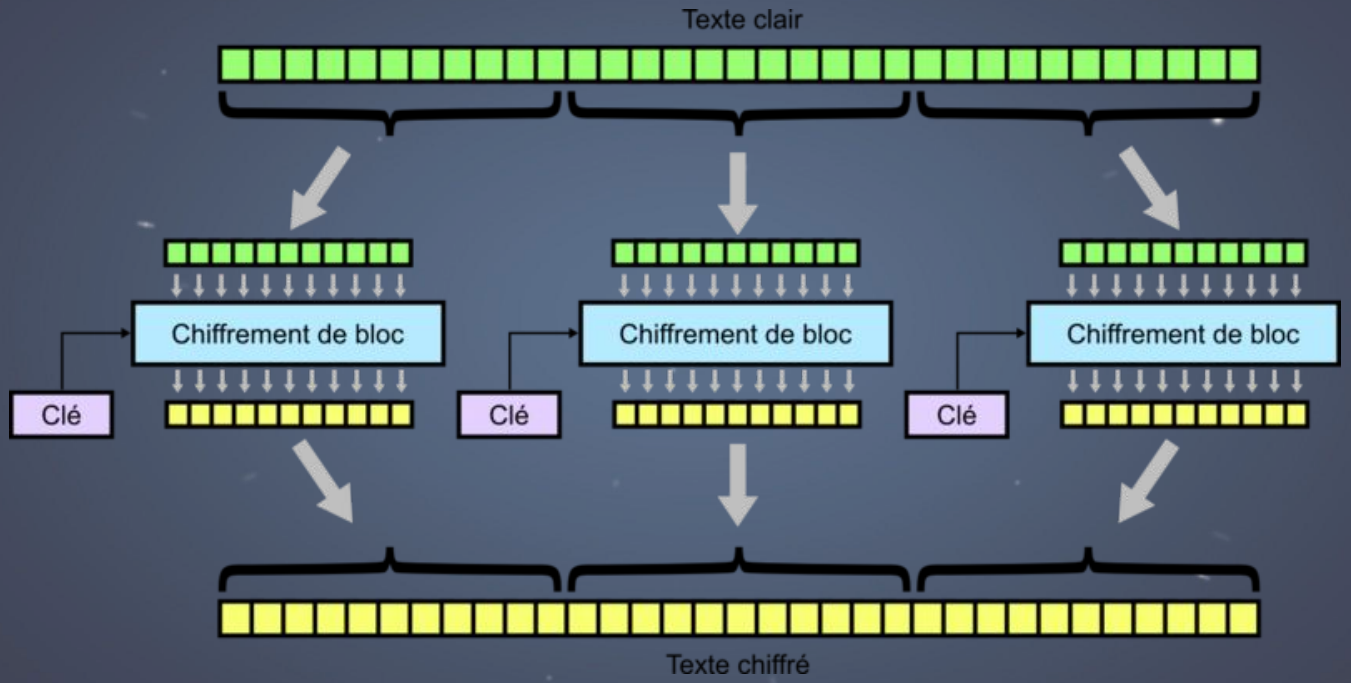


Les modes d'opération

Tu pensais pas que les blocs c'était simple quand même?

ECB – Electronic Codebook

La base



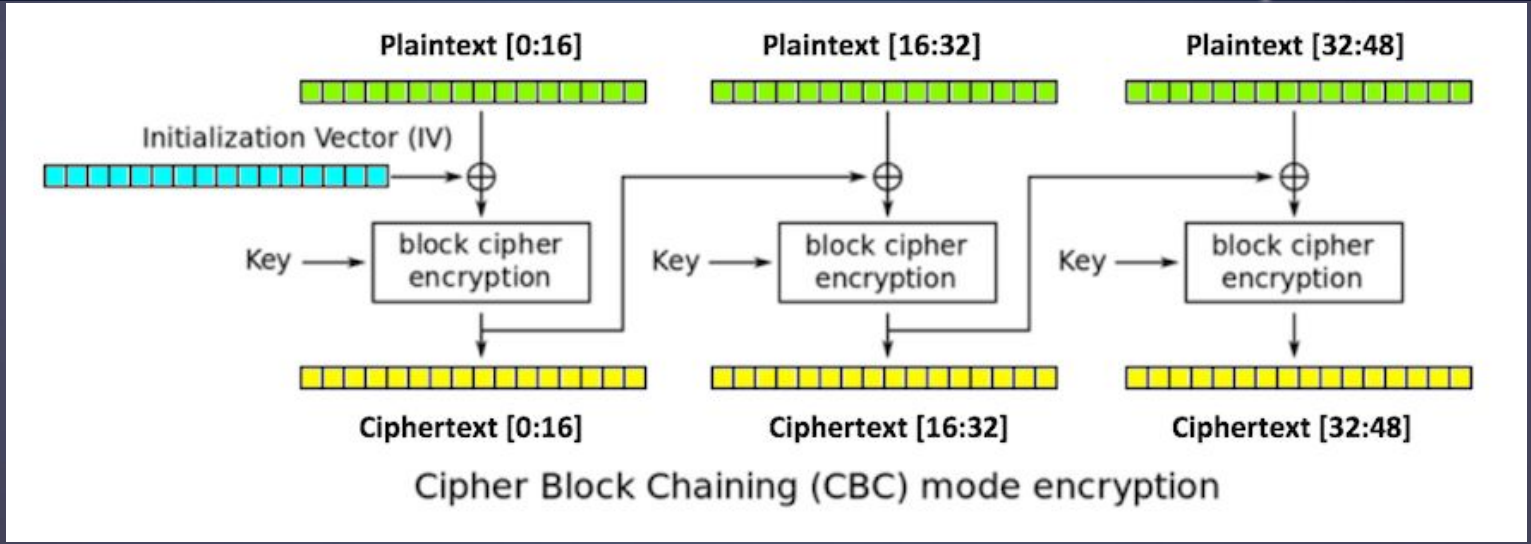
Mais...

C'est limitant, ECB!

- **Pourquoi chiffrer** les blocs indépendamment avec la même clé?
- Un attaquant peut **trouver la clé** en déchiffrant un des blocs!
- On peut aussi discerner des patterns si le message clair est bien plus gros que la clé.
- Une des solutions est de **générer** une "clé de round" unique par bloc - AES fait ça!
- Mais pourquoi s'arrêter là quand on peut faire encore **mieux**?
- Quand on peut… Utiliser chaque bloc pour **chiffrer le bloc suivant**?

CBC– Cipher Block Chaining

Shit gets real

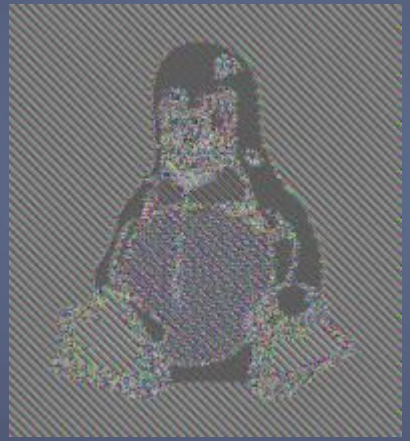


Cipher Block Chaining (CBC) mode encryption

Les effets de CBC

Les docteurs le détestent!

Plaintext



Plaintext -> ECB

Plaintext -> CBC

Et CBC c'est parfait?

lol

Plus lent

CBC empêche le parallélisme et ralentit le chiffrement par bloc. Il reste tout de même très rapide.

HACKERZ!!!!@!

CBC a une faille en particulier, connue et exploitable si un attaquant peut chiffrer les données de façon répétitive:: l'**Oracle Padding Attack**. D'autres approches comme le Bit Flipping existent.

...et l'échange de clé alors?

Ah ben oui, avec tout ça on sait toujours pas comment on va échanger nos clés, nous...

CRYPTO ASYMÉTRIQUE

Jésus, mais pour la crypto

DEUX CLÉS

À DEUX, C'EST MIEUX

PLAINTEXT MESSAGE

CLÉ PUBLIQUE

1]
ᒥQ□E□□r·□b□
□&y□hR□□□□)
,□□□□@□□
{%O□¦□k□u□□
□qV□□□(h

CLÉ PRIVÉE

PLAINTEXT MESSAGE

DESCRIPTION DES CLÉS

LA CLÉ PUBLIQUE (PK)

- Permet de **communiquer** avec le propriétaire.
- Partagée avec **tout le monde**
- Chiffre un message mais **ne peut pas le déchiffrer**.
- **Très difficile** de retrouver la SK.
- Peut **identifier** un message signé par la SK.

LA CLÉ PRIVÉE/SECRÈTE (SK)

- **Identifie** le propriétaire.
- **Peut déchiffrer** un message chiffré par la PK correspondante.
- Peut **signer** un message pour identifier l'auteur.

PRINCIPES DE RSA

YOUPI C'EST L'HEURE DES MATHS

CLÉ PUBLIQUE

Clé publique: 2 nombres, **e** et **N**

- Plaintext transformé en un **nombre M**

“abcdef” ⟶ **M**

- **Chiffrement** du **nombre M** avec la clé publique (**e**, **N**), donne le cipher **C**

C = (**M**^**e**) % **N**

CLÉ PRIVÉE

Clé privée: un nombre: **d**

- **Déchiffrage** du cipher **C** avec la clé privée **d** donne le nombre original **M**.

$$M = (C^d) \% N$$

- Lecture du plaintext à partir du nombre **M**.

M ⟶ “abcdef”

RELATION DES CLÉS

C'EST EN GROS POUR UNE RAISON

$$M = (M^e)^d \% N$$

GÉNÉRATION DE CLÉS

ET OUI JAMIE!

GÉNÉRATION DE PAIRE DE CLÉS

POUR LA DONNER À TES 3 AMIS

• Génération de 2 grands nombres premiers, **p** et **q**	**p**, **q**
• Génération de **N** = p * q	**N** = **p***q
• Calcul de **λ**, le plus petit multiple commun entre (p-1) et (q-1)	**λ** = lcm(**p**-1, **q**-1)
• On choisit un nombre premier **e** plus petit que **λ** (65537 en général)	1 < **e** < **λ**
• On calcule **d**, l'**inverse modulaire** de e par **λ**	(**e***d) % **λ** = 1

VULNÉRABILITÉS

LA PARTIE FUN

FACTORISATION DE CLÉ PUBLIQUE

PARCE QUE FUCK LES RÈGLES

RSA repose sur un simple fait: factoriser N (trouver p et q à partir de N) est **extrêmement difficile**. Donc pas évident, mais faisable dans dans quelques cas:

- N très petit, on peut juste **bruteforce**
- Les facteurs de N sont **connus** (http://factordb.com/)
- d est très petit comparé à N, factorisation par **fractions continues**
- Mauvaise génération de **p** et **q**

ATTAQUE SUR PETIT EXPOSANT PUBLIC

J'AI PAS TROUVÉ DE MOYEN D'EXPLIQUER ÇA GRAPHIQUEMENT

Si plusieurs personnes choisissent un petit exposant public (e=3) et que le même message **M** est envoyé à ces personne, générant 3 ciphers (c1, c2, c3), on a donc:

M^3 = C1 (mod N1) = C2 (mod N2) = C3 (mod N3)

On peut calculer M^3 modulo N1*N2*N3 grâce au **théorème des restes chinois**. Mais comme M est inférieur à N1, N2 et N3, M^3 < N1N2N3 et donc:

M^3 % N1N2N3 == M^3. On retrouve M à partir de M^3

NOTES

- La majorité de ces attaques sont sur le “textbook RSA”, c.a.d sur RSA sans padding appliqué.
- La factorisation difficile, la base de RSA, n’a **jamais été prouvée** mathématiquement.
- RSA reste utilisé surtout pour échanger des clés symétriques qui sont ensuite utilisées pour s’envoyer des données

LIENS / OUTILS UTILES

Why RSA works, un article qui va en profondeur sur les mathématiques impliquées dans RSA: http://doctrina.org/Why-RSA-Works-Three-Fundamental-Questions-Answered.html

numpy: Une lib python qui simplifie la manipulation et les calculs sur les grands nombres.

Sagemath, un outil qui peut faire des calculs de grands nombres très facilement, et qui utilise Python: http://www.sagemath.org/

RootMe: https://www.root-me.org/ Exos de sécurité variés, extrêmement utile pour s'améliorer