

X GIBNEY OSCAR® WINNING DIRECTOR OF GOING CLEAR AND TAXI TO THE DARK SIDE



WORLD WAR 3.0

"A WHITE KNUCKLE THRILLER.  
Clear, urgent, and positively terrifying at times."  
-Peter Debruge, Variety

-Peter Debruge, Variety

ZERO DAYS

A SHOULDER PICTURES AND PARTICIPANT MEDIA PRODUCTION IN ASSOCIATION WITH SOUTHPAW DOCUMENTARY FILMS & GLOBAL PRODUCE / JESSICA PROCTOR "ZERO DAYS"  
BY ANDY GREENE  
PRODUCED BY NELL BATES  
WRITTEN BY ANTHONY BASSI AND BRETT VALLEY  
DIRECTED BY JEFF SKILL, DAVID VERNERIAN, SARAH DOWRICK  
PRODUCED BY MARC SCHAFFER AND REED SODERBERGH  
PRODUCED BY ANDREW BONET

COMING JULY 8

AUX OSCARS®  
FILM DOCUMENTAIRE

ILL RÉALISÉ PAR  
A POITRAS

JOUEUR EXÉCUTIF  
SODERBERGH

ENFOUR

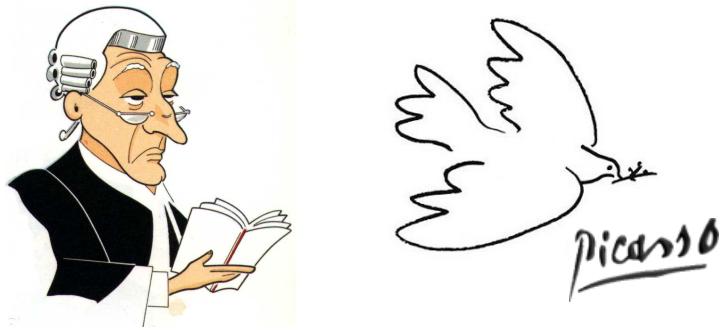
PRODUCTION: JESSICA PROCTOR, DAVID VERNERIAN, SARAH DOWRICK, ANDREW BONET  
PRODUCTION DESIGN: NELL BATES  
EDITOR: DAVID VERNERIAN, DAVID BASSI, ANDREW BONET  
CINÉMATOGRAPHIE: JEFF SKILL, DAVID VERNERIAN, ANDREW BONET  
SONORE: DAVID VERNERIAN, DAVID BASSI, ANDREW BONET  
MUSIQUE: JEFF SKILL, DAVID VERNERIAN, ANDREW BONET  
PRODUCTION DESIGN: NELL BATES  
EDITOR: DAVID VERNERIAN, DAVID BASSI, ANDREW BONET  
CINÉMATOGRAPHIE: JEFF SKILL, DAVID VERNERIAN, ANDREW BONET  
SONORE: DAVID VERNERIAN, DAVID BASSI, ANDREW BONET  
MUSIQUE: JEFF SKILL, DAVID VERNERIAN, ANDREW BONET  
PRODUCTION DESIGN: NELL BATES  
EDITOR: DAVID VERNERIAN, DAVID BASSI, ANDREW BONET  
CINÉMATOGRAPHIE: JEFF SKILL, DAVID VERNERIAN, ANDREW BONET  
SONORE: DAVID VERNERIAN, DAVID BASSI, ANDREW BONET  
MUSIQUE: JEFF SKILL, DAVID VERNERIAN, ANDREW BONET



ONCE YOU'RE IN, THERE'S NO WAY

TEHRA

# La signature électronique sécurisée



F96DE8C227A259C87EE1DA2AED57C9  
3FE5DA36ED4EC87EF2C63AAE5B9A7E  
FFD673BE4ACF7BE8923CAB1ECE7AF2  
DCF7AE29A3DA44F235A24C963FF0DF  
3CA3599A70E5DA36BF1ECE77F8DC34  
BE129A6CF4D126BF5B9A7CFEDF3EB8  
50D37CF0C63AA2509A76FF9227A55B  
9A6FE3D720A850D97AB1DD35ED5FCE  
6BF0D138A84CF8DC34BE129F8DC34B

# ELSI - jeudi matin 09:00-12:00

#1 : Hacking - Crypto 101

#2 : Crypto 101 - Signature - IGC

#3 : Signature - IGC - Certificats

#4 : TP noté : OpenSSL

#5 : Loi - Textes - Règlements - Audits

#6 : Révisions + Examen

EPITA - Cours ELSI 2025 n... 09:00  
12:00

JEUDI 13 NOVEMBRE 2025

EPITA - Cours ELSI 2025 n... 09:00  
12:00

JEUDI 20 NOVEMBRE 2025

EPITA - Cours ELSI 2025 n... 09:00  
12:00

JEUDI 27 NOVEMBRE 2025

EPITA - Cours ELSI 2025 n... 09:00  
12:00

JEUDI 4 DÉCEMBRE 2025

EPITA - Cours ELSI 2025 n... 09:00  
12:00

VENDREDI 12 DÉCEMBRE 2025

EPITA - Cours ELSI 2025 n... 09:00  
12:00

# Les 5 primitives cryptographiques

Chiffrement Symétrique



Chiffrement Asymétrique



Fonctions de hachage



Établissement de clé



Générateurs d'aléa



# Cheat Sheet de la cryptographie moderne

## 1. ☑ Chiffrement symétrique (AES)

**Principe** : même clé pour chiffrer et déchiffrer.

**Propriétés** : rapidité, sécurité fondée sur la confidentialité de la clé, résistance aux attaques différentielles/linéaires.

**Usages** :

- Chiffrement de données en transit ou au repos (TLS, VPN, disque chiffré)
- Authenticated Encryption (AES-GCM, ChaCha20-Poly1305)
- Protection de sessions ou de bases de données

**Exemples** : AES, ChaCha20, Camellia, Twofish

**Modes d'usage** : ECB (à proscrire), CBC, CTR, GCM

**Propriété clé** : Confidentialité

AES



RSA/EC



## 3. ☑ Fonctions de hachage (SHA)

**Principe** : fonction unidirectionnelle produisant un condensat de taille fixe.

**Propriétés** : non-inversibilité, résistance aux collisions et aux préimages.

**Usages** :

- Empreintes numériques (SHA-256, SHA3)
- Signatures électroniques (hachage avant signature)
- Stockage sécurisé de mots de passe (PBKDF2, bcrypt, Argon2)
- Intégrité (Merkle trees, blockchain)

**Exemples** : SHA-2, SHA-3, BLAKE2, BLAKE3

**Propriété clé** : Intégrité

SHA



DH



## 5. ☑ Générateurs d'aléa (PRNG != RNG)

**Principe** : production de bits imprévisibles et uniformes.

**Propriétés** : imprédicibilité, entropie, résistance aux biais.

**Usages** :

- Génération de clés (symétriques/asymétriques)
- IV, nonces, sels, défis de protocole
- Construction de schémas probabilistes (RSA-OAEP, signatures aléatoires)

**Exemples** : Fortuna, /dev/urandom, HMAC-DRBG, ChaCha20-DRBG

**Propriété clé** : Imprévisibilité

PRNG



## 2. ☑ Chiffrement asymétrique (RSA/EC)

**Principe** : clé publique pour chiffrer/vérifier, clé privée pour déchiffrer/signer.

**Propriétés** : non-répudiation, authentification lenteur (s'utilise pour échanger des clés ou signer).

**Usages** :

- Échange sécurisé de clés (RSA, ECC, post-quantum KEM)
- Signature numérique (RSA, ECDSA, EdDSA, Dilithium)
- Authentification (certificats X.509, OpenID, S/MIME)

**Exemples** : RSA, DSA, ECDSA, Ed25519, Dilithium, Kyber

**Propriété clé** : Confidentialité + Authenticité + Non-répudiation

## 4. ☑ Établissement de clé (Diffie-Hellman / ECDH / ECDHE)

**Principe** : accord sur un secret commun

**Propriétés** : confidentialité, éphémérité, protection contre interception passive.

**Usages** :

- Échange de clés dans TLS, SSH, Signal (X25519, X448)
- Protocoles de messagerie sécurisée (ratchets)
- Construction de sessions symétriques temporaires

**Exemples** : DH, ECDH, X25519, Kyber (post-quantum KEM)

**Propriété clé** : Confidentialité partagée

Le reste n'est qu'assemblage

**Calcul  
quantique**



**Cryptographie  
quantique**



**CPQ**

**Cryptographie  
Post Quantique**



Catégorie	Nom officiel	Origine
KEM	ML-KEM	(CRYSTALS-Kyber)
	HQC	(Hamming Quasi-Cyclic)
Signatures	ML-DSA	(CRYSTALS-Dilithium)
	SLH-DSA	(SPHINCS*)
	FN-DSA	(FALCON)



« Le secret ... c'est qu'il  
y a toujours un secret »

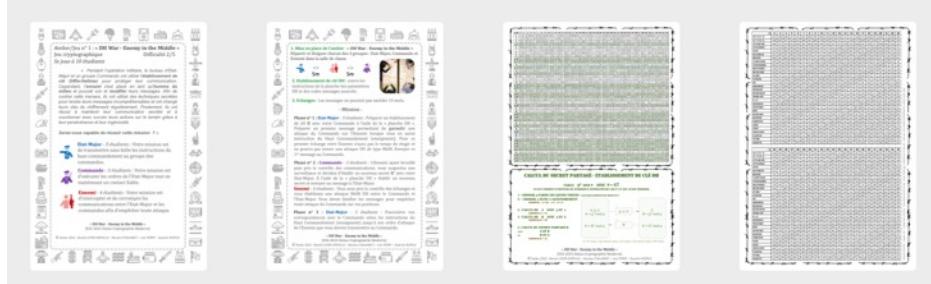
Cryptographie moderne ou pas

« Où est la clé privée ? »



# Ateliers/jeux Cryptographie Moderne

## Atelier 1 : DH War- Ennemy in the Middle



4 groupes  
de 10 étudiants

4 groupes  
de 5 étudiants

## Atelier 2 : Wannacry Nightmare Puzzle

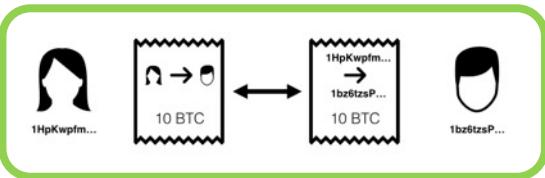
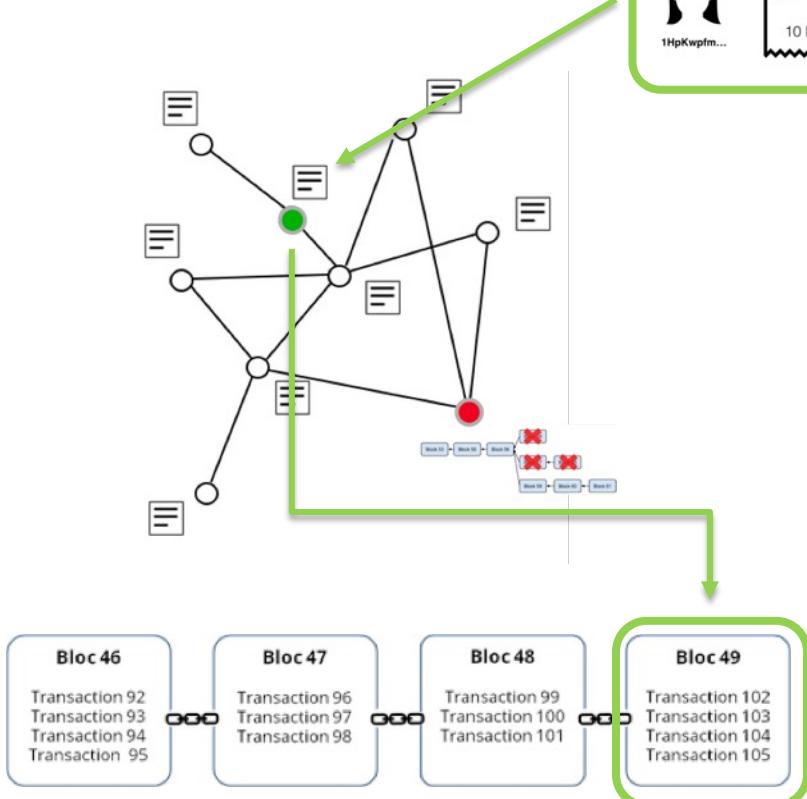


# EX\_MACHINA



**Bitcoin as Frankenstein creature ?**

# Bitcoin



preuve de possession - imputabilité



anonymat - pseudonymat



tracabilité - intégrité - non répudiation



disponibilité - accessibilité



évolutivité - adaptabilité - scalability



autonomie et pérennité de fonctionnement

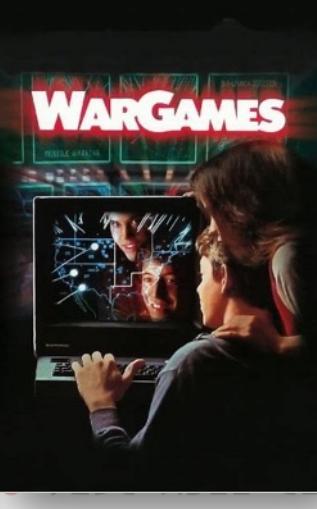
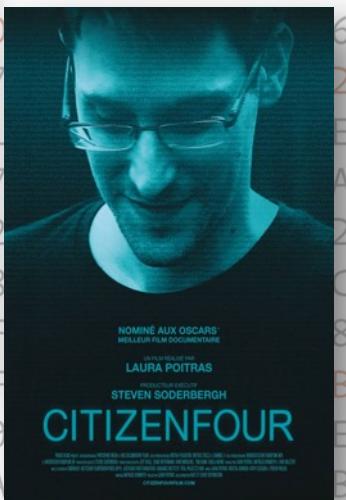


consensus décentralisé et automatisé



unicité de la dépense (anti double spending)

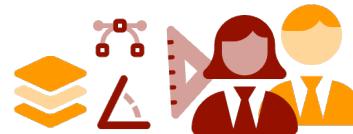




# Voilà ... c'est fini !

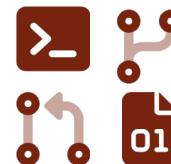
## Ingénieur(e)s

- scientifiques
- design



## Code

- préhistoire
- tout à inventer



## Stage

- pas très grave
- management

1<sup>ère</sup> page du rapport le 1<sup>er</sup> jour + répétitions



# Cabinet d'Experts Cybersécurité



Benoit  
Leger-Derville



Nicolas  
Chalanset



Paul Chabas



Loïc Perry



Quentin Ropele



Hector Colin



Julien Erhard

# reBop.io

## Certificate Lifecycle Management



The screenshot shows the reBop web application interface. At the top, there's a navigation bar with links for 'Features', 'Pricing', 'Docs', 'More', and a user profile. A pink 'launch' button is visible on the right. Below the navigation, there's a search bar and a 'filter host' dropdown. The main area displays a list of certificate status alerts:

- 1 host up • 23/11/2021 • 23:00:06 • REVIVED (green)
- 2 cert locations will expire in 24 days (yellow)
- 14 hours ago • 23/11/2021 • 23:00:06 • EXPIRATION (yellow)
- 3 cert locations will expire in 84 days (yellow)
- 14 hours ago • 23/11/2021 • 22:00:00 • EXPIRATION (yellow)
- 4 cert locations will expire in 64 days (yellow)
- 14 hours ago • 23/11/2021 • 22:00:00 • EXPIRATION (yellow)
- 1 host up • 23/11/2021 • 22:00:06 • EXPIRED (red)

Below this, there are two sections with badges and descriptions:

- All alert logs at the right place**  
Fighting against expiration is a real sport. reBop allows you to play in the big leagues.
- Expired and revoked**  
With clearly and easily identified badges, locations with expired and revoked certificates are obvious.
- Checking alert logs**  
Based on the experience of its customers, reBop offers you to come and check the alert logs at least once every 5 days.

A modal window titled 'Online Error Cert' is open, showing details for a certificate that has gone online:

Online Error Cert	
CN	apim-factoring.societe generale.com
Issuer	QuoVadis Global SSL ICA G3
SN	2C17C3A94AAZ19EIEEE3P1F46A92EC08CF55A38AS
Expires	10 Jul 2022 15:28:00
C	FR
S	Ile-de-France
L	Paris
O	Societe Generale SA
OU	SG120222
O	apim-

**Manage online invalid certificates** R

With reBop quickly detect at a glance your still online expired, revoked or suspended certificates.

It's time to clean up and ask your team why these online hosts still have

The mobile device screen shows a similar interface to the desktop version, with a user profile picture and a 'launch' button. The main content area is mostly blacked out, suggesting a blurred view of the certificate logs or alerts.

# Offre de stage candidats à voir

1. **Expertise SSI** : Consulting, analyse technique Cyber/SSI/Crypto, Lecture/rédaction de cible de sécurité labo CESTI, Lecture de spécifications (Passeport biométrique et Carte d'Identité électronique), Rédactions de notes techniques

=> Objectif n° 1 : devenir **Consultant(e) Expert Cyber**  
+ à terme Auditeur MIE / PVID / eIDAS / 27K1

2. **Code** : Développement d'outils internes et commerciaux Stelau  
(IdCheck, IdFod, reBop, Lib CEV et VDS Verify)

=> Objectif n° 2 : Connaitre et **maitriser** l'assemblage des primitives cryptographiques de cybersécurité

3. **Conception** : Participation à l'élaboration et au maintien des Cours EPITA et EGE (renouvellement et adaptation des TP, TD et jeux cryptographiques).

=> Objectif n° 3 : **Apprendre et enseigner**

# excellents stages missions formatrices



## rémunération ++

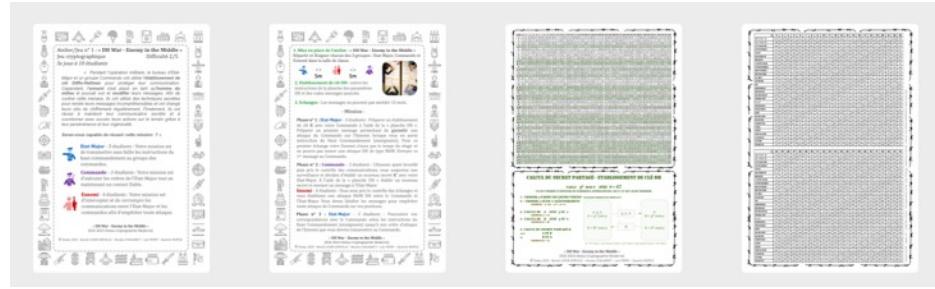
A screenshot of the France Identité mobile application. The top navigation bar includes links for "Accueil", "En savoir plus", "Questions fréquentes", "Actualité", "Justificatif d'identité", "Contact", and "Votre compte". The main content area features a smartphone displaying a digital French ID card (Carte Nationale d'Identité) with a photo of a person named Hélène Martin. Below the phone, there are three buttons: "Scanner un QR Code", "Créer un Justificatif d'identité", and "Télécharger la version Web de l'application". To the right, a large call-to-action button says "Gardez la maîtrise de vos données d'identité". Smaller text below it provides tips: "Prouver votre identité sans divulguer toutes vos données", "Éviter l'usurpation de votre identité", and "Remplacer vos identifiants et mots de passe". At the bottom, there are download links for Google Play and the App Store.



# Ateliers/jeux Cryptographie Moderne

<https://github.com/stelaucconseil/EPITA-ELSI>

## Atelier 1 : DH War- Ennemy in the Middle



4 groupes  
de 10 étudiants

## Atelier 2 : Wannacry Nightmare Puzzle

4 groupes  
de 5 étudiants



# ELSI - jeudi matin 09:00-12:00

#1 : Hacking - Crypto 101

JEUDI 13 NOVEMBRE 2025

#2 : Crypto 101 - Signature - IGC

JEUDI 20 NOVEMBRE 2025

#3 : Signature - IGC - Certificats

JEUDI 27 NOVEMBRE 2025

#4 : TP noté : OpenSSL

EPITA - Cours ELSI 2025 n... 09:00  
12:00

#5 : Loi - Textes - Règlements - Audits

JEUDI 4 DÉCEMBRE 2025

#6 : Révisions + Examen

EPITA - Cours ELSI 2025 n... 09:00  
12:00

VENDREDI 12 DÉCEMBRE 2025

EPITA - Cours ELSI 2025 n... 09:00  
12:00

# Les 3 objectifs

- La cryptographie moderne c'est 5 primitives à connaître

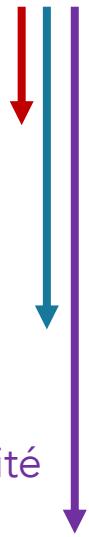


- Essayer de comprendre l'aspect « révolutionnaire » de la cryptographie **asymétrique**
- En déduire le concept de **signature** cryptographique



# Les vraies difficultés de la cryptographie moderne

1. **THEORIE** : Cryptologie
  - failles théoriques = **mathématiques**
  - cryptologue est un métier
2. **CODE** : Implémentations
  - erreurs/failles/vuln. = **informatique**
  - « *Do not implement cryptography yourself!* »
3. **UX** : Usages
  - mauvais usages = ignorance/pusillanimité
  - bons usages = **formation**



# Sondage ?

Chiffrement symétrique



Chiffrement asymétrique



# Vocabulaire cryptographique

- Cryptologie - science (λόγος) du secret (κρυπτός) :
  - cryptographie
  - cryptanalyse
- « *Do not implement crypto yourself* »
- Le bureau du chiffre du quai d'Orsay
- « crypter » est un néologisme
- Cybersécurité = sécurité informatique = SSI
- SSI : sécurité des systèmes d'information
- C'est ~~crypté~~ chiffré donc c'est sécurisé ... ne veut rien dire en SSI.

La cryptologie ne se limite plus aujourd'hui à assurer la confidentialité des secrets

Algo. de chiffrement	Avec Clé	Sans Clé
message clair	chiffrer	<i>crypter</i>
message chiffré	déchiffrer	décrypter



# Les 5 primitives cryptographiques

Chiffrement Symétrique



Chiffrement Asymétrique



Fonctions de hachage



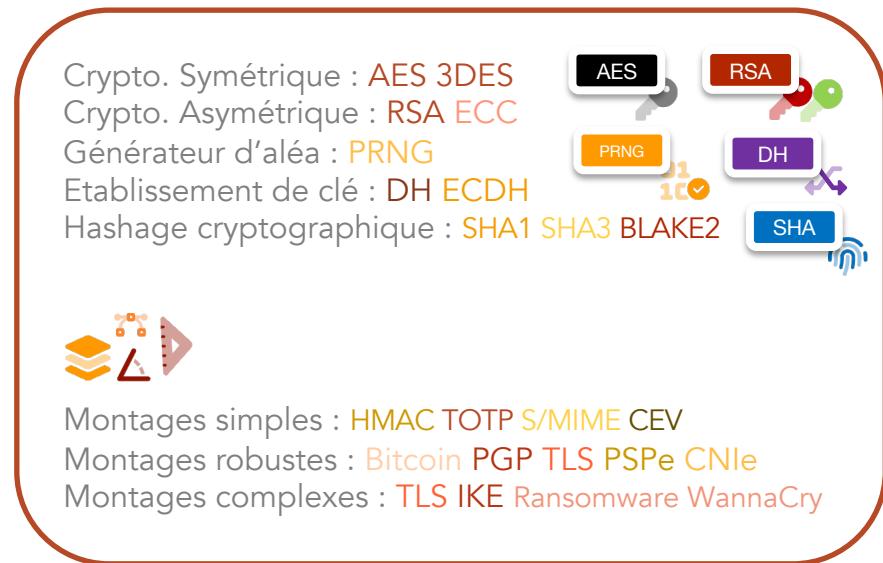
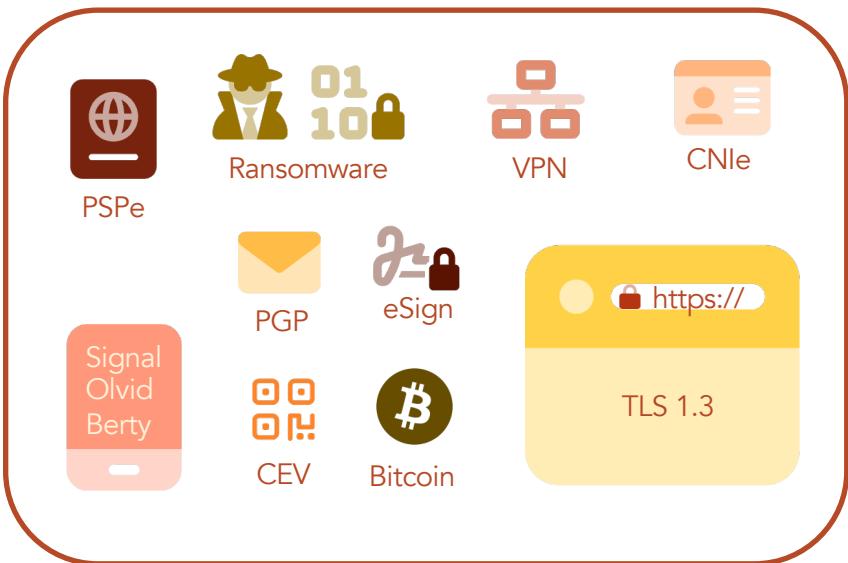
Établissement de clé



Générateurs d'aléa

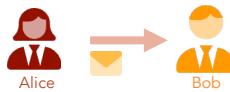


# Cryptographie moderne

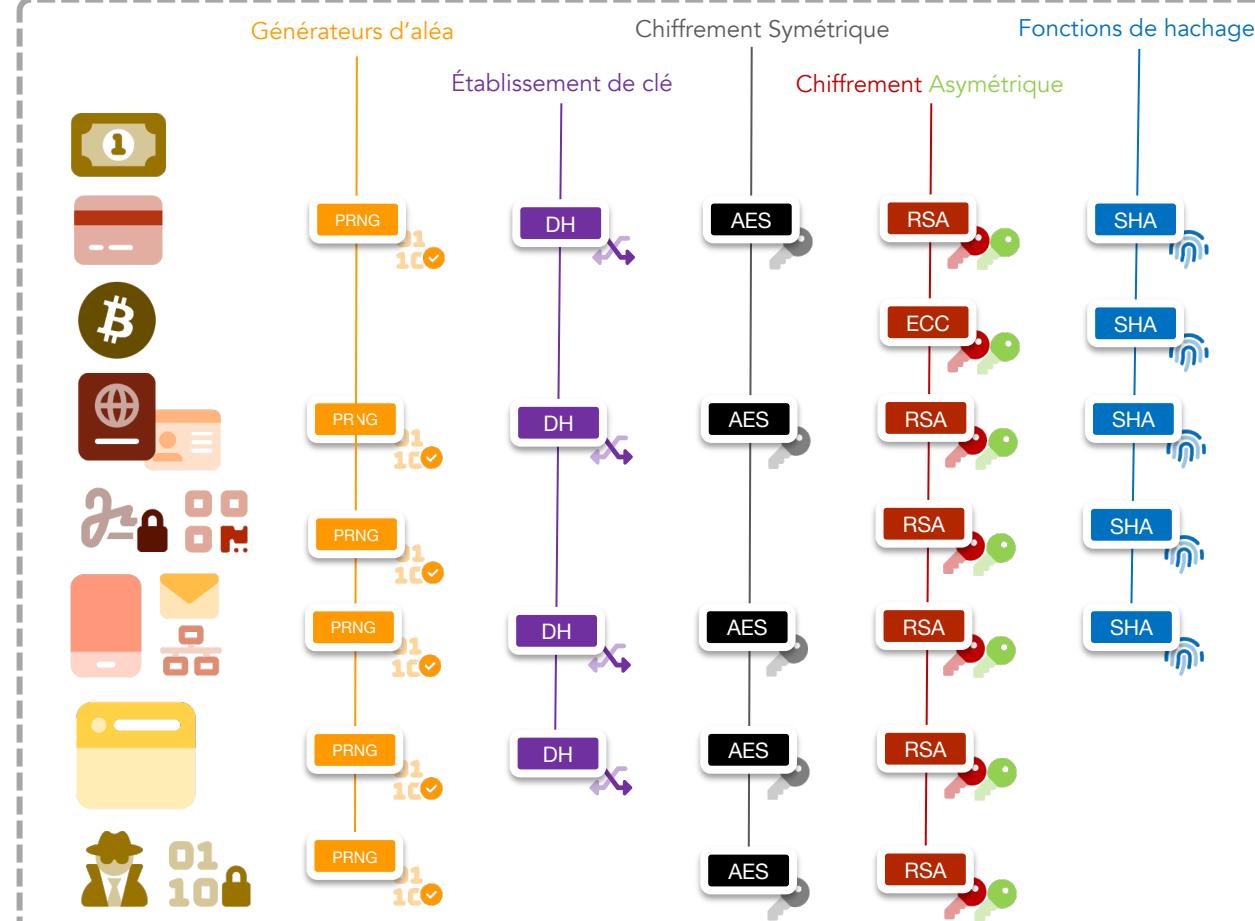


# Cryptographie

La cryptographie ne se limite plus aujourd'hui à assurer la **confidentialité** des secrets



1. Confidentialité
2. Intégrité
3. Authenticité
  
4. Disponibilité
5. Originalité
6. Non-répudiation
  
7. Traçabilité
8. Preuve à divulgation nulle (zero-knowledge)



# Les 5 primitives cryptographiques

Chiffrement Symétrique



Chiffrement Asymétrique



Fonctions de hachage



Établissement de clé

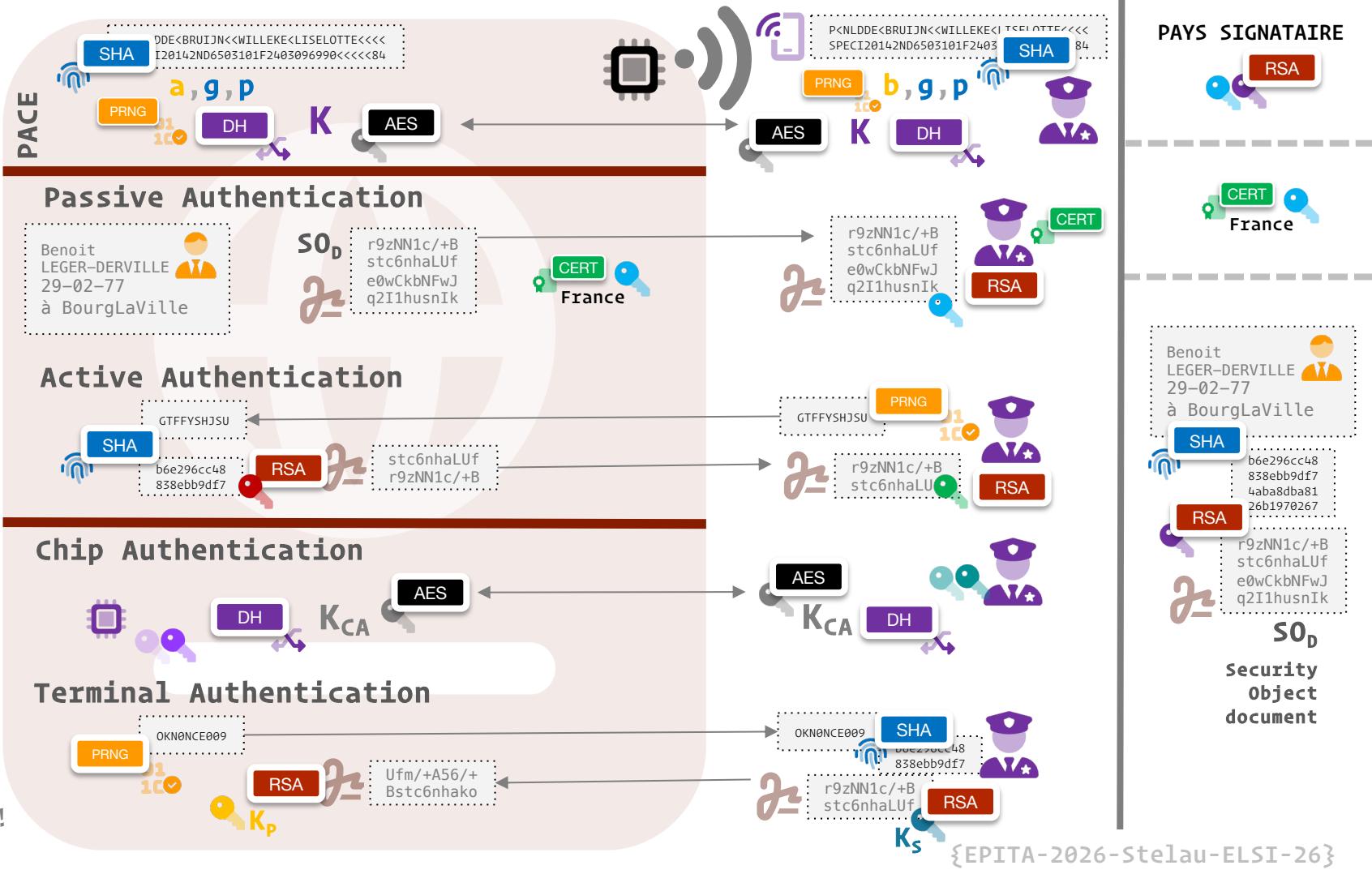


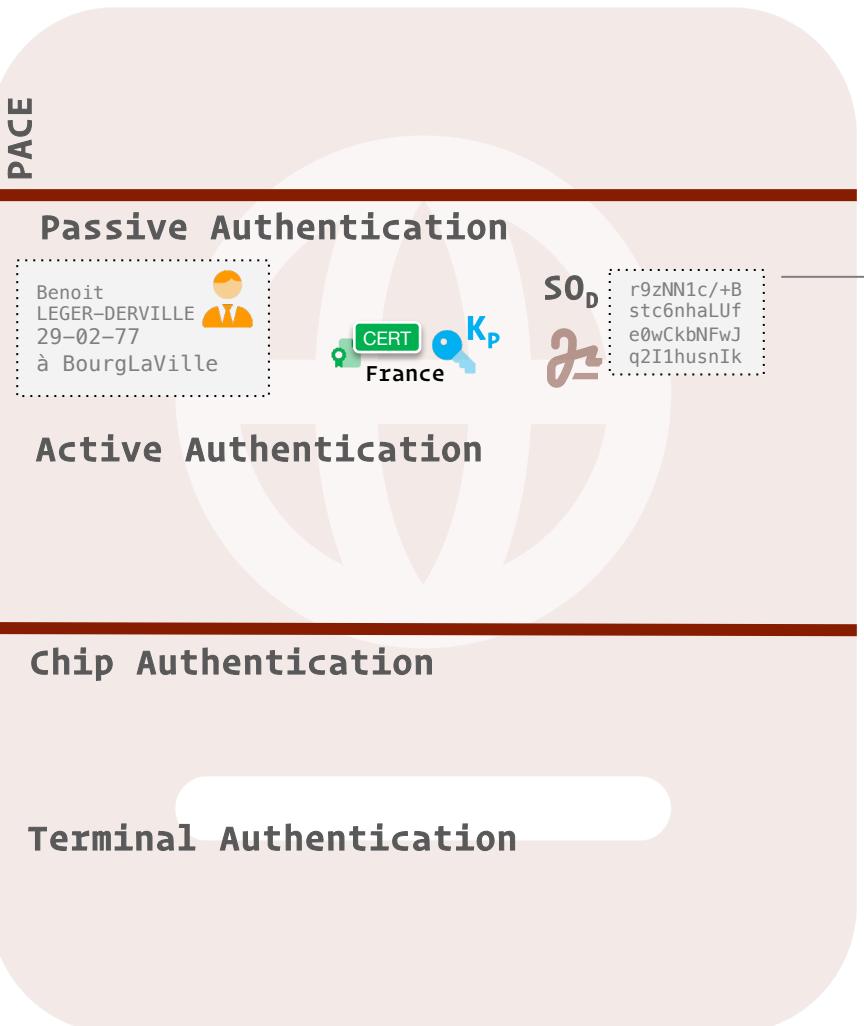
Générateurs d'aléa



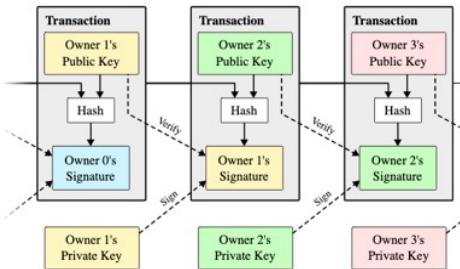
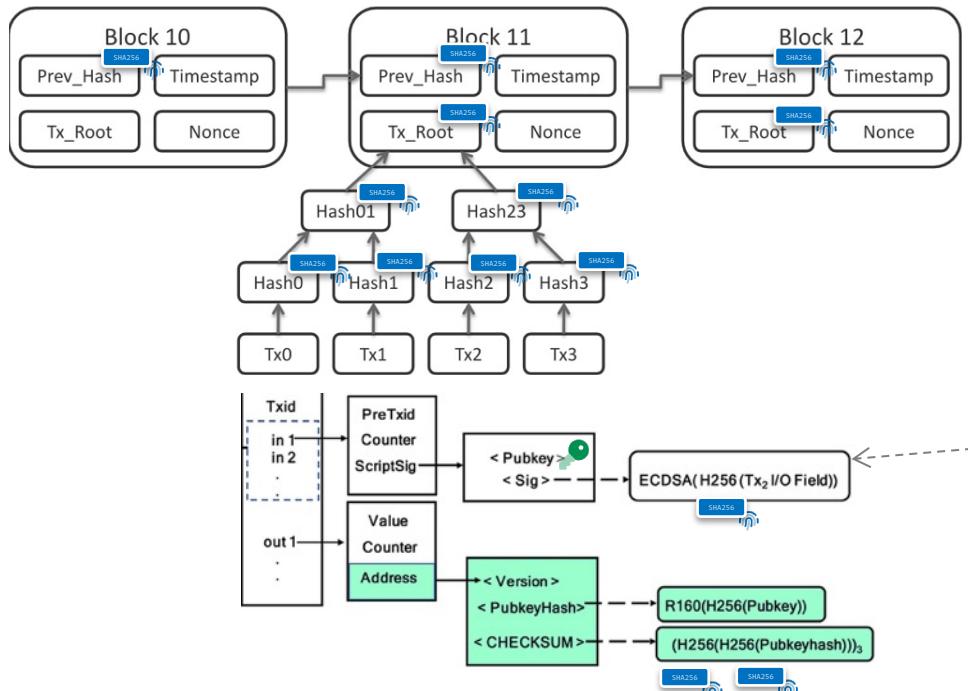
# OACI Passport

Assemblage  
riche et  
complexe  
mais sans  
surprise !

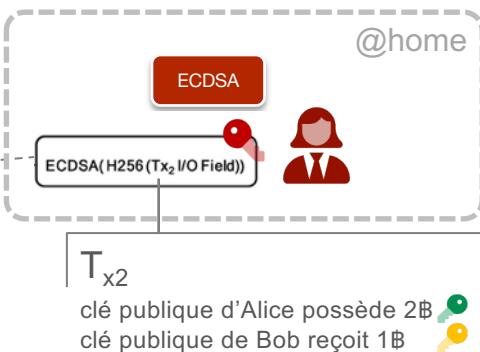




# Bitcoin : hashes et signature



La clé privée de l'expéditeur est utilisée pour signer le hachage SHA-256 de toutes les données de la transaction, y compris les entrées, les sorties et les montants.



Un Tx (ou transaction) en Bitcoin est une opération dans laquelle une ou plusieurs entrées de Bitcoin sont utilisées pour créer une ou plusieurs sorties de Bitcoin. Les entrées de Bitcoin sont des sorties de transactions antérieures qui ont été envoyées à l'adresse Bitcoin du destinataire et qui sont maintenant disponibles pour être dépensées. Les sorties de Bitcoin sont des montants de Bitcoin qui sont envoyés à une adresse Bitcoin.

# CEV : Cachet Electronique Visible



r9zNN1c/+B  
stc6nhaLUF  
e0wCkbNFWJ  
q2I1husnIk

+  
+

RSA

@EDF

ba81264aba  
b6e296cc48  
838ebb9df7  
8db1970267

SHA

**NOUS CONTACTER**

N° client ; Identifiant Internet

**Par Internet et Mobile**  
edf.fr  
sur Smartphone et Tablette  
Télécharger l'app mobile EDF&MOI

**Par téléphone**  
Du lundi au vendredi de 8h et jusqu'à 21h  
09 69 32 15 15  
(appel gratuit, prix appel)

**Par courrier**  
EDF SERVICE CLIENT  
TSA 20012  
75197 Paris Cedex 01

**Nos boutiques**  
Retrouvez la boutique la plus proche de chez vous sur [boutiques.edf.com](#)

**Lieu de consommation**

M. LEGER BENOIT  
Votre contrat



LEGER BENOIT  
2 rue d'Ici  
1<sup>er</sup> Etage  
75001 PARIS

## ATTESTATION TITULAIRE DE CONTRAT

Par la présente, EDF atteste que M. BENOIT LEGER est actuellement titulaire auprès d'EDF pour le logement situé au

Ce contrat a été établi au nom de M. BENOIT LEGER sur la base de ses décl

Pour servir et valoir ce que de droit.



@home

RSA



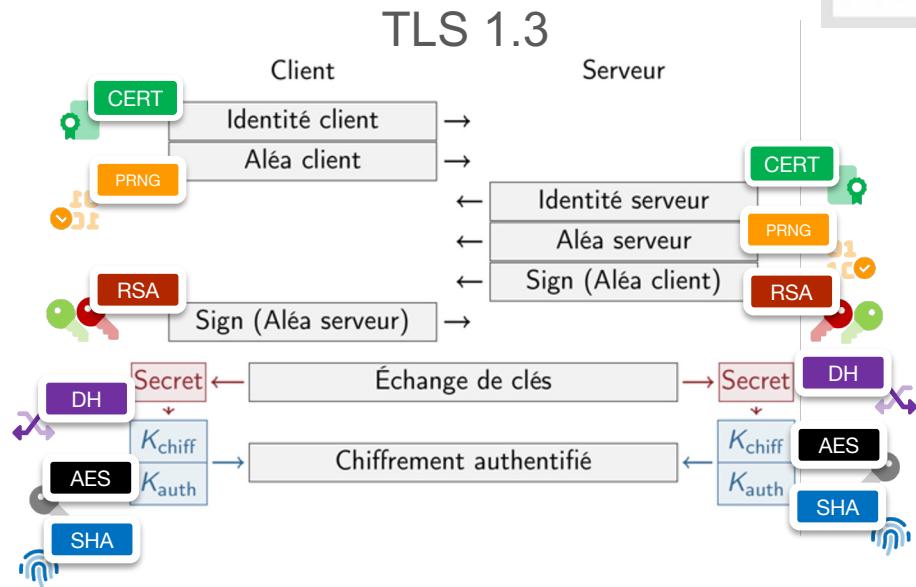
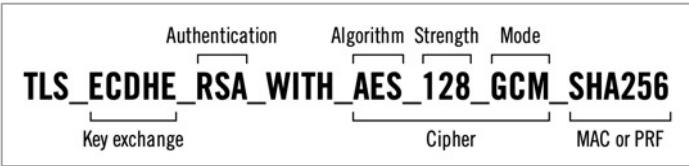
r9zNN1c/+B  
stc6nhaLUF  
e0wCkbNFWJ  
q2I1husnIk

ba81264aba  
b6e296cc48  
838ebb9df7  
8db1970267

ba81264aba  
b6e296cc48  
838ebb9df7  
8db1970267

SHA

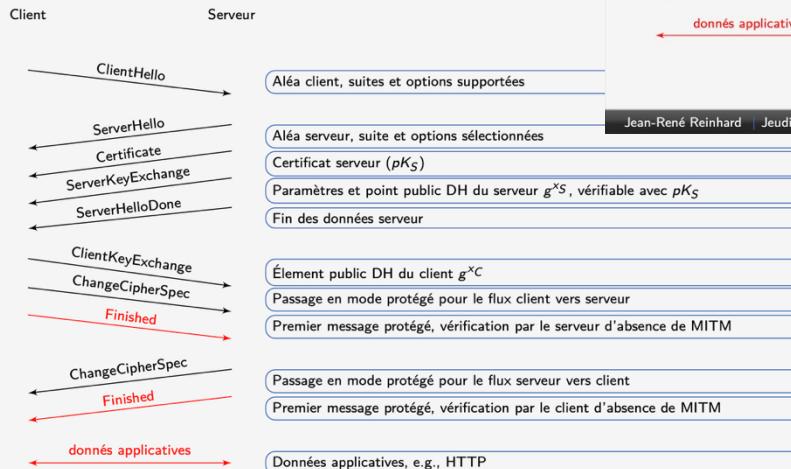
# Assemblage Crypto



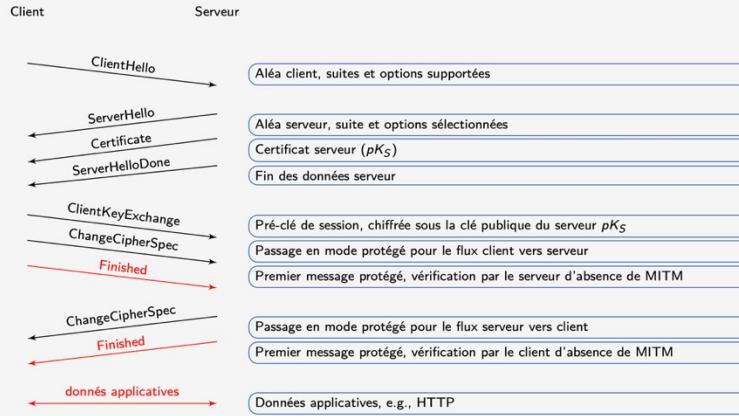
Cipher Suite Name	Auth	KX	Cipher	MAC	PRF
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	RSA	ECDHE	AES-128-GCM	-	SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	ECDSA	ECDHE	AES-256-GCM	-	SHA384
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA	RSA	DHE	3DES-EDE-CBC	SHA1	Protocol
TLS_RSA_WITH_AES_128_CBC_SHA	RSA	RSA	AES-128-CBC	SHA1	Protocol
TLS_ECDHE_ECDSA_WITH_AES_128_CCM	ECDSA	ECDHE	AES-128-CCM	-	SHA256

# Assemblage Crypto

## SSL/TLS : établissement de clé DHE

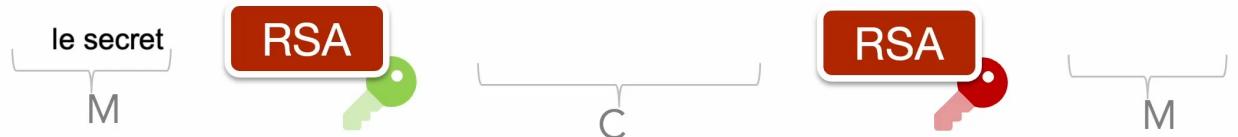


## SSL/TLS : établissement de clé RSA



Hello !

AES



Hello !

SHA



26, 47, 10

DH



91690410bec9  
graine

PRNG



798

aléa

# Very Short Crypto Story

3000 ans de crypto. **symétrique**

*recettes militaro-diplomatiques  
de confusion et de diffusion*

**Confusion et Diffusion**  
*« tant bien que mal »*  
de César à Enigma

100 ans de crypto. **moderne**

*de Kerckhoffs ...  
au crypto-système incassable*



1. Principes de Kerckhoffs - 1883
2. One Time Pad - 1917

50 ans de crypto. **asymétrique**

*LA véritable révolution*

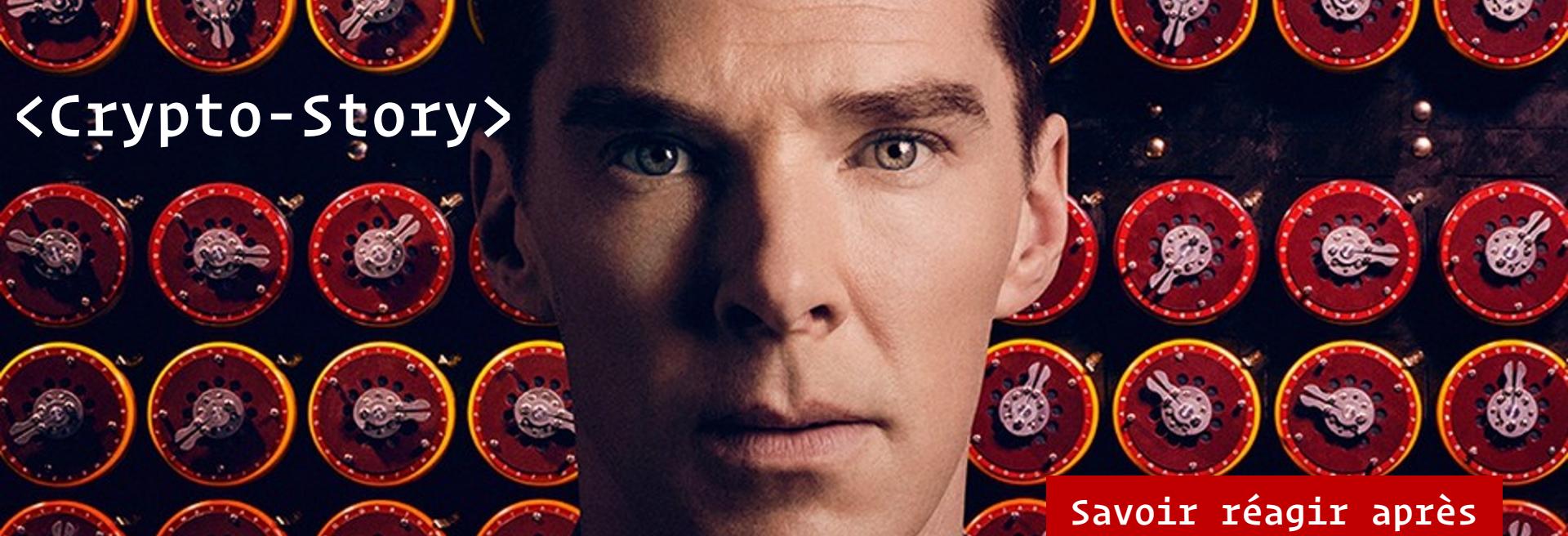


Résout la difficulté de  
l'échange de clé  
+  
Permet l'usage du  
principe de **Signature**

20 ans de crypto. **post quantique**

*révolution ? (ou pas)*





# <Crypto-Story>

Feindre d'ignorer ce qu'on sait,  
de savoir tout ce qu'on ignore, ...  
avoir souvent pour grand secret  
de cacher qu'il n'y en a point, ...

Beaumarchais - Le Mariage de Figaro (1778)

**Savoir réagir après  
avoir cassé le code  
de son adversaire**

# Confusion et Diffusion

## Substitution et Permutation

Cryptographie Symétrique => Confusion et Diffusion

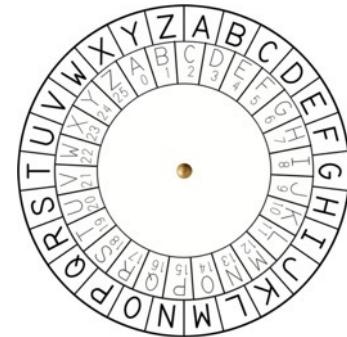
Confusion par Substitution

A → E L → T L → T O → Y ...

Diffusion par Permutation / Transposition

J E S U I S P A S L A => S I U S E J A L S A P

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



# Principes de Kerckhoffs - 1883

Crypto symétrique moderne

Auguste Kerckhoffs - 1883

1. Il faut qu'il n'exige pas le **secret**, et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi.

2. La **clef** doit pouvoir en être communiquée et retenue sans le secours de notes écrites, et être changée ou modifiée au gré des correspondants.

=> ce qui est gardé secret doit être ce qui est le moins coûteux à changer si le secret s'avérait divulgué

Notion de **clé secrète**

Algorithme connu de tous :

- Substitution
- Permutation

mais basées sur la clé secrète

a b c d e f g h i k l m n o p q r s t u x y z  
ø † † # a ☐ ø ☐ i ö n // ø v s m f Δ E C 7 8 9

Nulles ff.—.—.d. Dowbleth σ

and for with that if but where as of the from by  
2 3 4 4 3 7 5 2 3 8 X ☐

so not when there this in wic h is what say me my wyrt  
2 X 7 + 6 x 5 β n n m m o

send lfe receave bearer I pray you Mte your name myne  
1 S 2 T 1 H — R E SS

# Chiffrement Symétrique

## cryptosystème parfait : « incassable »



### Le Masque Jetable - One Time Pad 1917

- La clé doit être une suite de caractères au moins aussi longue que le message à chiffrer
- Les caractères composant la clé doivent être choisis de façon aléatoire
- Chaque clé, ou « masque », ne doit être utilisée qu'une seule fois
  - d'où le nom de masque jetable

One-Time Pad										
• Plain text:	H	O	W	A	R	E	Y	O	U	
	7	14	22	0	17	4	24	14	20	
+										
OTP:	13	2	1	19	25	16	0	17	23	
	N	C	B	T	Z	Q	A	R	X	
<hr/>										
Initial total:	20	16	23	19	42	20	24	31	43	
<hr/>										
Mod 26:	20	16	23	19	16	20	24	5	17	
<hr/>										
Ciphertext:	U	Q	X	T	Q	U	Y	F	R	

# Chiffrement Symétrique

## cryptosystème parfait : Masque jetable ou OneTimePad



### Alphabet utilisé

A=0, B=1, ..., Z=25, 0=26, ..., 9=35, espace=36

Opération : addition modulo 37 (l'espace est chiffré aussi).

### Alphabet complet (37 symboles)

Caractères : ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789% ( = espace)

Index :

0→A | 1→B | 2→C | 3→D | 4→E | 5→F | 6→G | 7→H | 8→I | 9→J | 10→K | 11→L |  
12→M | 13→N | 14→O | 15→P | 16→Q | 17→R | 18→S | 19→T | 20→U | 21→V | 22→W |  
| 23→X | 24→Y | 25→Z | 26→Ø | 27→1 | 28→2 | 29→3 | 30→4 | 31→5 | 32→6 |  
33→7 | 34→8 | 35→9 | 36→%

### Exemple 1

yaml

Clair : RDV%AU%LOUVRE%A%15H

Masque : UJZDE8GXD6NCF10EPF9

Chiffré : AMJCERF8RP8TJ00DF%F

Clair 1 : RDV AU LOUVRE A 15H

Masque : UJZDE8GXD6NCF10EPF9

→ Chiffré 1 : AMJCERF8RP8TJ00DF F

Clair 2 : RDV AU LOUVRE A 16H

Masque : K3M2TQH5Z8L0B7N1C4D

→ Chiffré 2 : 1671T GFCR6GF6N03ZK

### Exemple 2 (heure différente + masque différent)

yaml

Clair : RDV%AU%LOUVRE%A%16H

Masque : K3M2TQH5Z8L0B7N1C4D

Chiffré : 1671T%GFCR6GF6N03ZK

### Exemple 3 (clair totalement différent)

yaml

Clair : BONJOUR%PARIS%2025

Masque : ZYMLOPIFQY78F5M0NX

Chiffré : 0BZU29ZE5YNFX4DPER

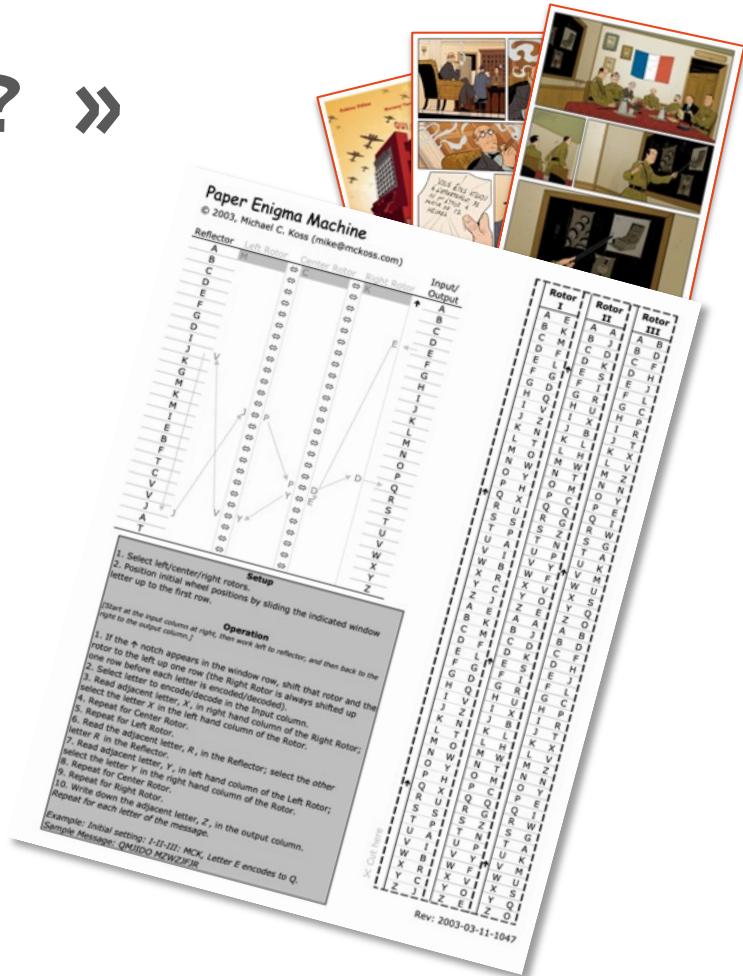
# « Qui a cassé Enigma ? »

**1932.** Dans la salle de bains d'un hôtel bruxellois, un espion français photographie les premiers documents décrivant une nouvelle machine à coder *a priori* inviolable : Enigma. Une machine que s'apprêtent à adopter les services secrets allemands. Quelques mois plus tard, avec l'aide des Français, un groupe de mathématiciens polonais entreprend de percer à jour le fonctionnement complexe de la machine.

**1940.** Après la défaite française face aux nazis, les Français et les Polonais transmettent leurs trouvailles aux Britanniques. À Bletchley Park se déploie une gigantesque entreprise de décodage dont va dépendre l'issue de la guerre.

**1942.** Sous le nez des Allemands, à Vichy même, Français et Polonais continuent leurs efforts de décodage. La Gestapo est à leurs trousses et le MI-6 a pour priorité absolue de les exfiltrer. Pendant ce temps dans l'Atlantique, les U-Boote allemands mènent une traque dévastatrice contre les navires américains qui ravitaillent la Grande-Bretagne en armes, vivres et marchandises. Si on ne réussit pas très vite à décoder les messages de la Marine allemande, le Royaume-Uni ne tiendra pas.

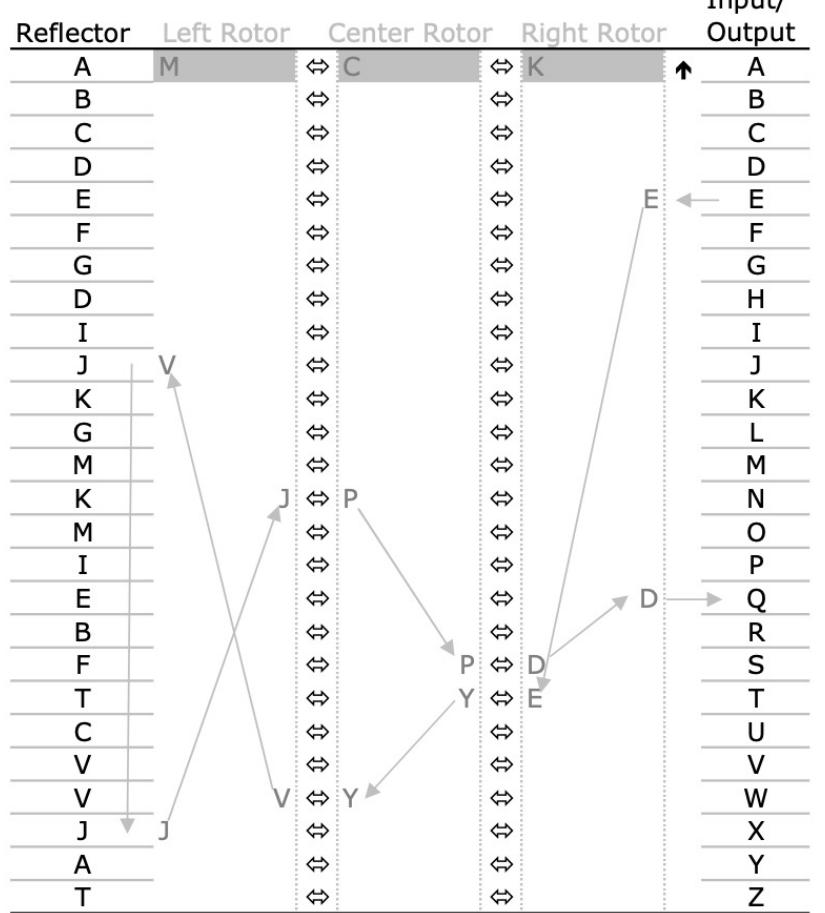
À Bletchley Park, un des cerveaux les plus brillants de l'histoire scientifique, **Alan Turing**, va apporter une contribution décisive...





# Paper Enigma Machine

© 2003, Michael C. Koss (mike@mckoss.com)



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

# Very Short Crypto Story



3000 ans de crypto. **symétrique**

*recettes militaro-diplomatiques  
de confusion et de diffusion*

100 ans de crypto. **moderne**

*de Kerckhoffs ...  
au crypto-système incassable*

50 ans de crypto. **asymétrique**

*LA véritable révolution*



DH 1976



RSA 1977

20 ans de crypto. **quantique**

*révolution ? (ou pas)*



[https://fr.wikipedia.org/wiki/Histoire\\_de\\_la\\_cryptographie](https://fr.wikipedia.org/wiki/Histoire_de_la_cryptographie)

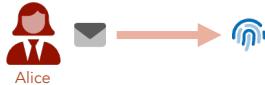
Les dates clefs de la cryptographie

# Hachage

Hachage cryptographique

SHA

Secure  
Hash  
Algorithm



SHA



empreinte cryptographique

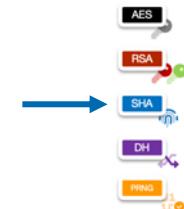
Une fonction de hachage cryptographique est une fonction sans clé qui permet de transformer une donnée de taille arbitraire en une chaîne de bits de taille  $h$  fixe, appelée haché.

R3

## Fonctions de hachage

Les fonctions de hachage de la famille SHA-2 [FIPS180] et de la famille SHA-3 [FIPS202] dont la taille de sortie est supérieure ou égale à 256 bits sont recommandées.

Primitive	Taille de paramètre	R/O	Notes
SHA-2 [FIPS180, ISO10118-3]	$h = 256$ bits (SHA-256)	R	
	$h = 384$ bits (SHA-384)	R	
	$h = 512$ bits (SHA-512)	R	
	$h = 256$ bits (SHA-512/256)	R	
SHA-3 [FIPS202]	$h = 256$ bits	R	
	$h = 384$ bits	R	
	$h = 512$ bits	R	



1. BLAKE
2. RIPEMD
3. WHIRLPOOL

# Hachage

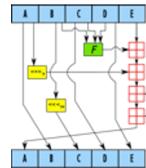
Hachage cryptographique

SHA



Secure  
Hash  
Algorithm

```
SHA1-compress(H, M) {
    (a0, b0, c0, d0, e0) = H // parsing H as five 32-bit big endian words
    (a, b, c, d, e) = SHA1-blockcipher(a0, b0, c0, d0, e0, M)
    return (a + a0, b + b0, c + c0, d + d0, e + e0)
}
```



```
SHA1-blockcipher(a, b, c, d, e, M) {
    W = expand(M)
    for i = 0 to 79 {
        new = (a <<< 5) + f(i, b, c, d) + e + K[i] + W[i]
        (a, b, c, d, e) = (new, a, b >>> 2, c, d)
    }
    return (a, b, c, d, e)
}
```

```
expand(M) {
    // the 512-bit M is seen as an array of sixteen 32-bit words
    W = empty array of eighty 32-bit words
    for i = 0 to 79 {
        if i < 16 then W[i] = M[i]
        else
            W[i] = (W[i - 3] ⊕ W[i - 8] ⊕ W[i - 14] ⊕ W[i - 16]) <<< 1
    }
    return W
}
```

```
f(i, b, c, d) {
    if i < 20 then return ((b & c) ⊕ (~b & d))
    if i < 40 then return (b ⊕ c ⊕ d)
    if i < 60 then return ((b & c) ⊕ (b & d) ⊕ (c & d))
    if i < 80 then return (b ⊕ c ⊕ d)
}
```

b1e9feb2d6015f3fa4bfac79788cb21f03560984

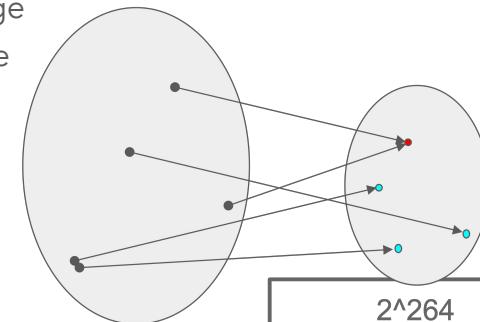
SHA1



- Fonctions à sens unique  
d'un espace infini vers un espace fini de :  
128, 160, 224, 256 ou 512 bits

## Résistantes aux attaques

- 1<sup>ère</sup> pré-image
- 2<sup>de</sup> pré-image
- collisions



2<sup>264</sup>  
2<sup>(80\*3,3)</sup>  
10<sup>80</sup>  
nb. atomes univers

# Hachage

Hachage cryptographique

SHA



Secure  
Hash  
Algorithm

SHA256

f9dcc621fc8ac6a172c40fd3ffcbfcf20fa400e3b216d3777362ee7c22965ea

SHA256("Guess #0") =  
1101000101011001000101011010001  
0111100101100000011100100000111  
11101100101001100101110111101100  
0001001110100011101111000101011  
10000001001111000011001100100111  
10001101011011101001000110000101  
1110101111101011000101011011000  
100011101000000110000110100000000

# Hachage

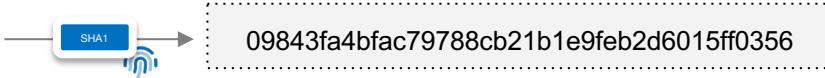
## Hachage cryptographique

SHA



Secure  
Hash  
Algorithm

La cryptographie moderne est une branche de la cryptologie qui utilise des algorithmes mathématiques pour protéger et sécuriser les informations. Elle est devenue essentielle dans le monde numérique pour garantir la confidentialité, l'intégrité et l'authenticité des données échangées. Les principaux objectifs de la cryptographie moderne sont de garantir la confidentialité, l'authenticité et l'intégrité des données. Elle est utilisée dans de nombreux domaines tels que les transactions financières, la communication en ligne, la sécurité des réseaux informatiques, et la protection des données personnelles.



Benoit LEGER-DERVILLE

12 mars 1974

5 rue de la Villa  
65432 Bourg-la-Ville



ba81264abab6e296cc48838ebb9df78db1970267



194a0023224bf8db193346aa8b09cdfc0689f1fd



3a1637db49251af125ebc74662eb6c9e9dda0c16

# Hachage

Hachage cryptographique

SHA



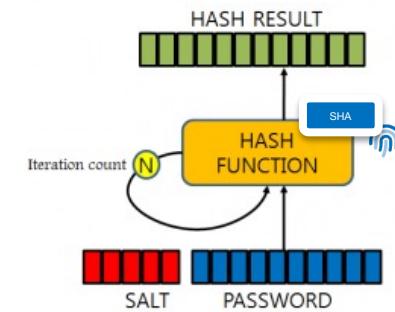
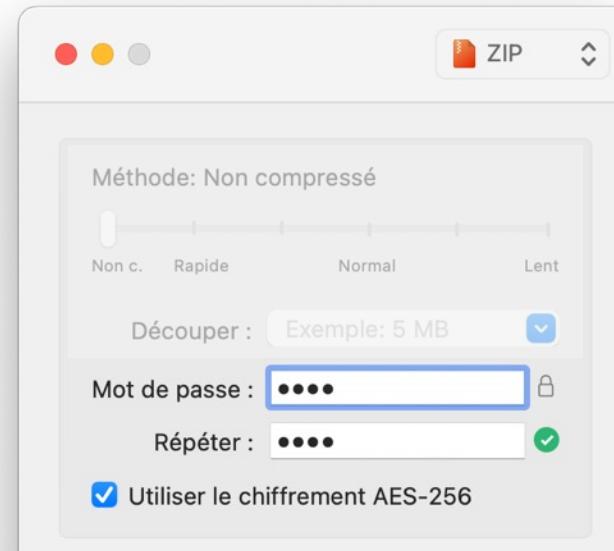
Secure  
Hash  
Algorithm

P@55w0rD/2023

SHA



6967e32dfaafdde85d7d40db50cd906c9200caee



# Hachage

Hachage cryptographique

SHA



Secure  
Hash  
Algorithm



les carottes  
sont cuites

c257a561c0af1d7c174c07daae51a10e28863226

SHA1



les carottes  
sont cuites



les carottes  
sont cuites  
19/06/23-14:20



01c48eed586a9b0f15903efb4bf56e12f2ff0deb

SHA1



les carottes  
sont cuites  
19/06/23-14:20



153591

TOTP - HMAC

BNP Paribas

Pour vous authentifier et accéder à vos  
comptes, entrez le code 600121 sur la  
page de connexion, si vous êtes bien à  
l'origine de l'opération

# Hachage

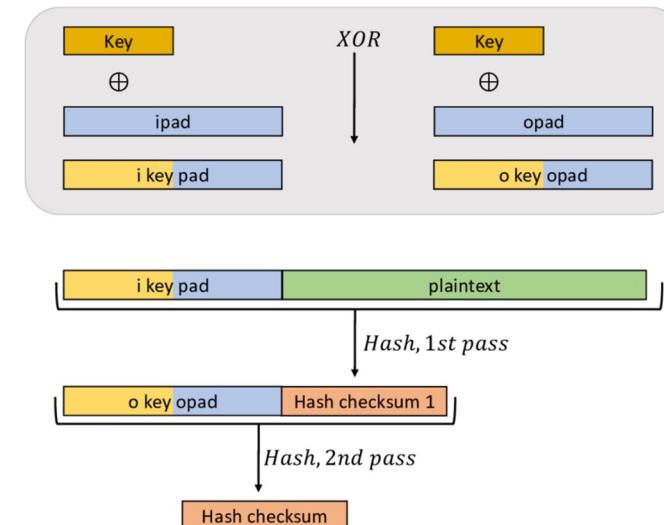
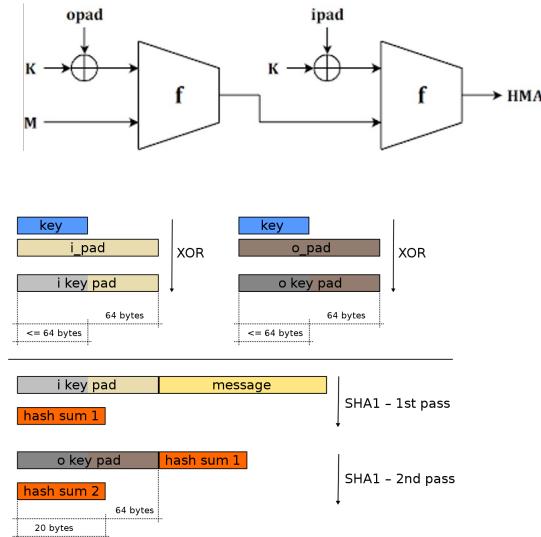
Hachage cryptographique

SHA



Secure  
Hash  
Algorithm

$$\text{HMAC}_K(m) = h\left((K \oplus opad) \parallel h\left((K \oplus ipad) \parallel m\right)\right)$$



## Construction du HMAC

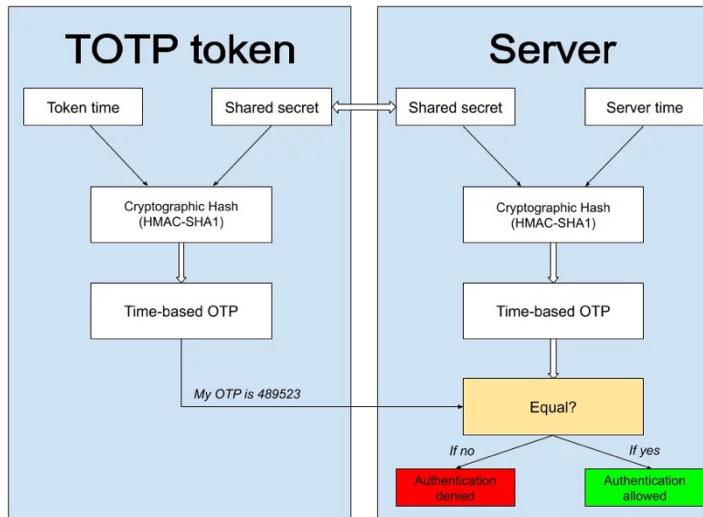
## Construction Hachage Crypto: TOTP

Hachage cryptographique

SHA



Secure  
Hash  
Algorithm



### 5.4. Example of HOTP Computation for Digit = 6

The following code example describes the extraction of a dynamic binary code given that hmac\_result is a byte array with the HMAC-SHA-1 result:

```

int offset  = hmac_result[19] & 0xf ;
int bin_code = (hmac_result[offset] & 0x7f) << 24
| (hmac_result[offset+1] & 0xff) << 16
| (hmac_result[offset+2] & 0xff) << 8
| (hmac_result[offset+3] & 0xff) ;
  
```

SHA-1 HMAC Bytes (Example)

Byte Number
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19
Byte Value
1f 86 98 69 0e 02 ca 16 61 85 50 ef 7f 19 da 8e 94 5b 55 5a

M'Raihi, et al.

Informational

[Page 7]

RFC 4226

HOTP Algorithm

December 2005

- \* The last byte (byte 19) has the hex value 0x5a.
- \* The value of the lower 4 bits is 0xa (the offset value).
- \* The offset value is byte 10 (0xa).
- \* The value of the 4 bytes starting at byte 10 is 0x50ef7f19, which is the dynamic binary code DBC1.
- \* The MSB of DBC1 is 0x50 so DBC2 = DBC1 = 0x50ef7f19 .
- \* HOTP = DBC2 modulo 10^6 = 872921.

We treat the dynamic binary code as a 31-bit, unsigned, big-endian integer; the first byte is masked with a 0x7f.

We then take this number modulo 1,000,000 ( $10^6$ ) to generate the 6-digit HOTP value 872921 decimal.

# Hachage

Hachage cryptographique

SHA



Secure  
Hash  
Algorithm

## Ce que n'est pas un HMAC

	Hash	MAC	Digital signature
Integrity	Yes	Yes	Yes
Authentication	No	Yes	Yes
Non-repudiation	No	No	Yes
Kind of keys	None	Symmetric	Asymmetric

**Intégrité** : Le destinataire peut-il être sûr que le message n'a pas été modifié accidentellement ?

**Authentification** : Le destinataire peut-il être sûr que le message provient de l'expéditeur ?

**Non-répudiation** : Si le destinataire transmet le message et la preuve à une tierce partie, cette dernière peut-elle être certaine que le message provient de l'expéditeur ?

Pour les signatures un vérificateur doit être sûr que la clé de vérification appartient réellement au signataire.

Pour les MAC, un destinataire doit être sûr que la clé symétrique partagée n'a été partagée qu'avec l'expéditeur.

# Hachage

Hachage cryptographique

SHA

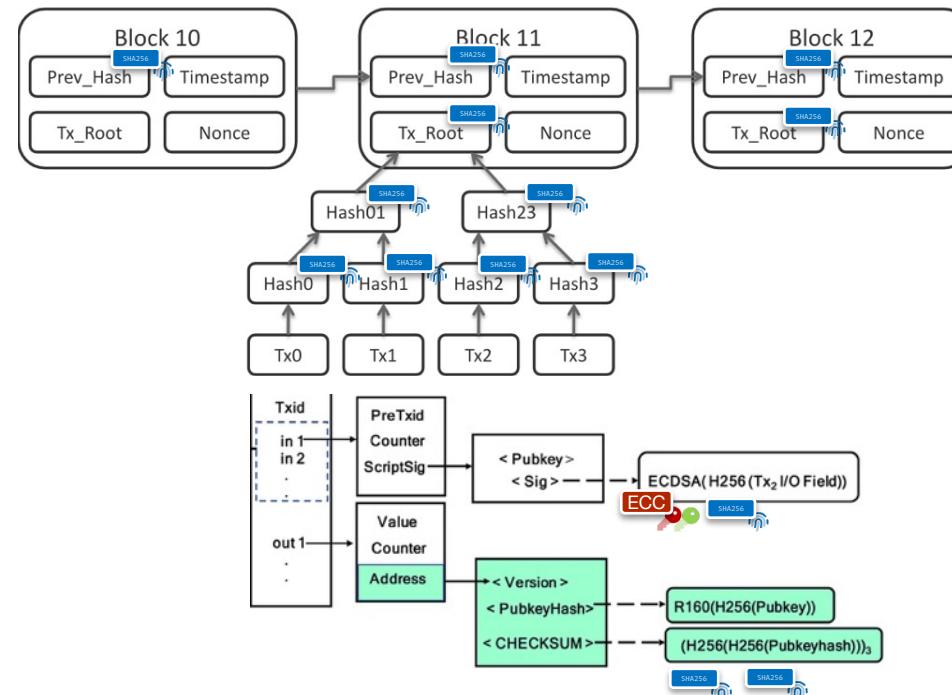


Secure  
Hash  
Algorithm

SHA256

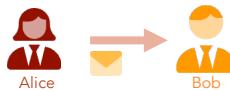


f9dcc621fcb8ac6a172c40fd3ffcbfcf20fa400e3b216d3777362ee7c22965ea

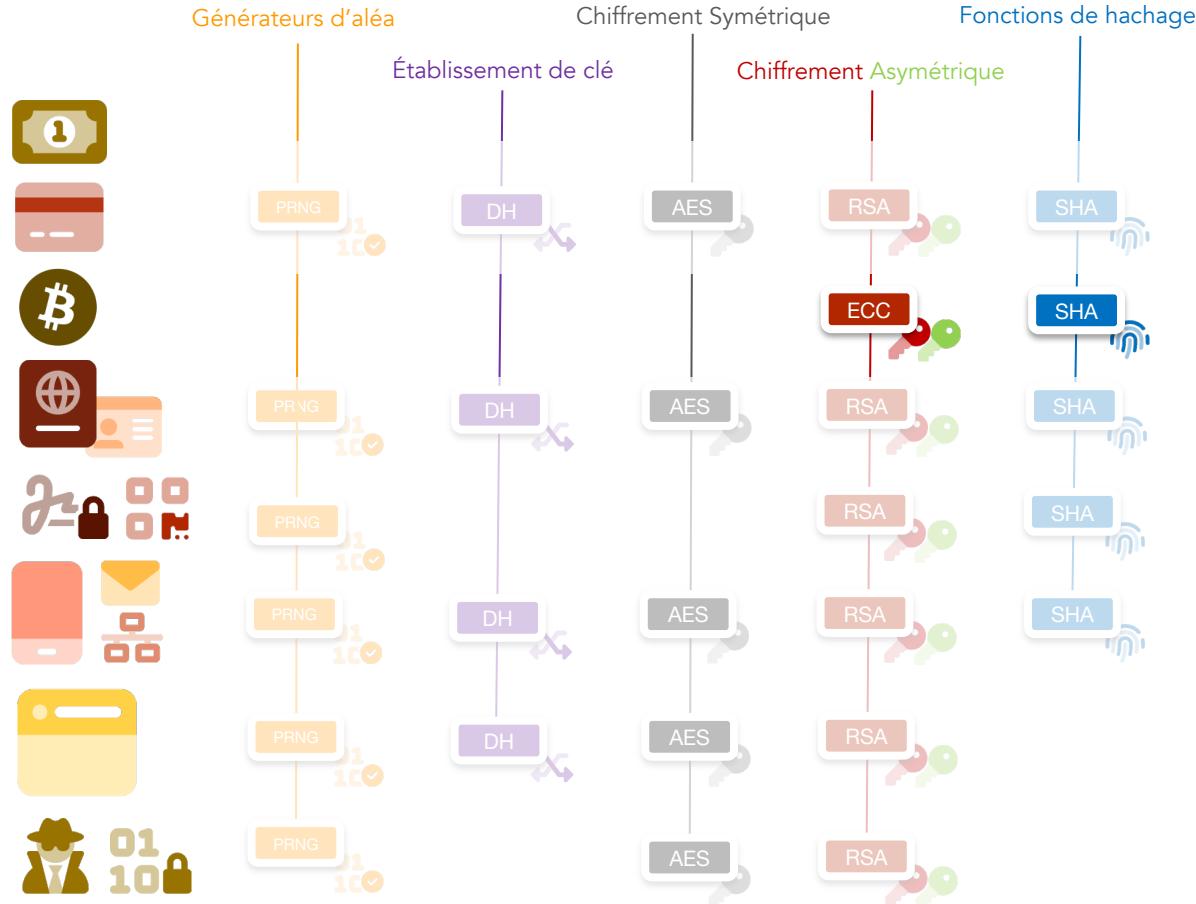


# Cryptographie

La cryptographie ne se limite plus aujourd'hui à assurer la **confidentialité** des secrets



1. Confidentialité
2. Intégrité
3. Authenticité
  
4. Disponibilité
5. Originalité
6. Non-répudiation
  
7. Traçabilité
8. Preuve à divulgation nulle (zero-knowledge)



# Génération d'Aléa

Pseudo Aléatoire

PRNG



Pseudo  
Random  
Number  
Generator



S



1. Confidentialité
2. Intégrité
3. Authenticité
4. Disponibilité
5. Originalité
6. Non-répudiation
7. Traçabilité

91690410bec9

graine

PRNG



798

aléa

Une source d'aléa est un processus probabiliste duquel on peut extraire une séquence de bits aléatoires. Il est difficile de s'assurer de la qualité des sorties d'une source d'aléa.

Deux approches : 1. Réaliser des tests statistiques sur les sorties de la source ;  
2. Modéliser le processus stochastique de la source.

R23

## Générateur d'aléa déterministe

HMAC-DRBG, Hash-DRBG et CTR-DRBG [FIPS197, ISO18033-3] sont recommandés.

Schéma	R/O	Notes
HMAC-DRBG [SP800-90A, ISO18031]	R	
Hash-DRBG [SP800-90A, ISO18031]	R	
CTR-DRBG [SP800-90A, ISO18031]	R	

AES

RSA

SHA

DH

PRNG



1. Test de primalité : Miller-Rabin
2. RSA :  $|p - q| \geq 2^{n/2 - 100}$

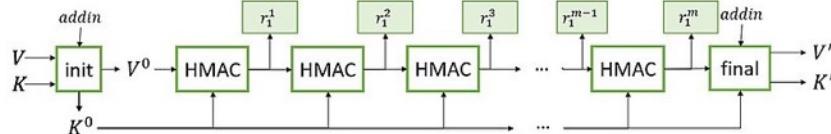
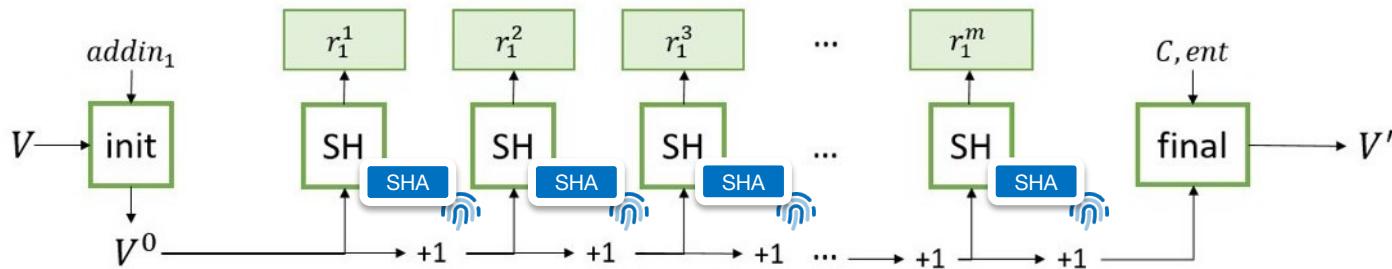
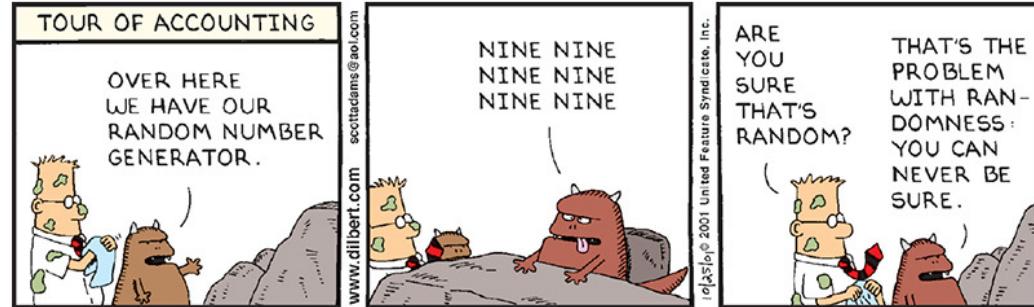
# Génération d'Aléa

Pseudo Aléatoire

PRNG



Pseudo  
Random  
Number  
Generator



# Symétrique

## Chiffrement Symétrique

AES



Advanced  
Encryption  
Standard



1. Confidentialité
  - Très rapide
  - Gros volume
2. Intégrité
3. Authenticité
4. Disponibilité
5. Originalité
6. Non répudiation
7. Traçabilité

Hello !

M

AES



K

C

Un mécanisme de chiffrement symétrique permet, à l'aide d'une clé secrète K, de transformer un message clair M en un message chiffré C

R1

### Algorithmes de chiffrement par bloc

AES [FIPS197, ISO18033-3] est recommandé pour ses trois tailles de clés (128, 192 et 256 bits).

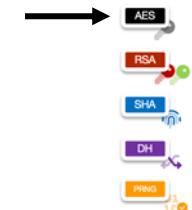
Primitive	Taille de clé	Taille de bloc	R/O	Notes
AES [FIPS197, ISO18033-3]	k = 128 bits	n = 128 bits	R	
	k = 192 bits		R	
	k = 256 bits		R	
Triple-DES [SP800-67, ISO18033-3]	k = 112 bits	n = 64 bits	O	3.1.a, 3.1.b
	k = 168 bits		O	3.1.b

R2

### Algorithmes de chiffrement par flot

L'algorithme de chiffrement par flot ChaCha20 [RFC8439] est recommandé.

Primitive	R/O	Notes
ChaCha20 [RFC8439]	R	



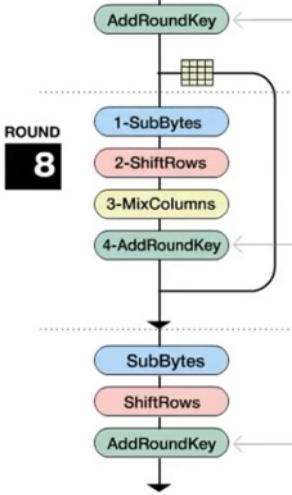
1. AES
2. Blowfish
3. DES
4. 3DES
5. IDEA
6. RC4
7. RC5
8. Serpent
9. Twofish

# Symétrique

Chiffrement Symétrique

AES

Advanced  
Encryption  
Standard



## Symétrique

Chiffrement Symétrique

AES



Advanced  
Encryption  
Standard

# Qui écoute la planète ?

Les USA



AES



Rijndael

Joan Daemen et Vincent Rijmen

Les Chinois



SHA-3



Keccak

Les Belges



## Etablissement de clé

Partage de secret

DH

Diffie  
Hellman



26, 47, 10

DH



Un mécanisme d'établissement de clé permet à deux entités (ou plus) de se mettre d'accord sur une valeur de clé secrète.

1. Confidentialité
- Rapide
  - vuln. MitM

2. Intégrité

3. Authenticité

4. Disponibilité

5. Originalité

6. Non répudiation

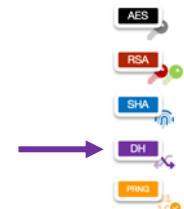
7. Traçabilité

R22

### Mécanismes d'établissement de clé

Les mécanismes d'établissement de clé DH et EC-DH sont recommandés.

Primitive	Mécanisme	R/O	Notes
FF-DLOG	DH [SP800-56A, ISO11770-3]	R	6.4.a
EC-DLOG	EC-DH [SP800-56A, ISO11770-3]	R	6.4.a



1. ECDH

Partage de secret

DH

Diffie  
Hellman

## Discrete Log Problem

$$2^n = 16$$

$$2^n \bmod 17 = 16$$

Alice

$$a, g, p$$

$$A = g^a \bmod p$$

$$K = B^a \bmod p$$

$$3 \bmod 17 \equiv$$

$$K = A^b \bmod p = (g^a \bmod p)^b \bmod p =$$

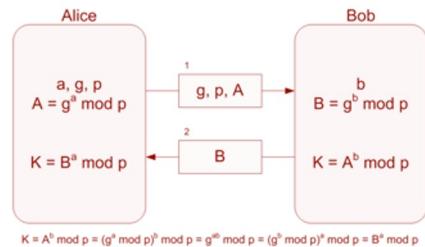
# Etablissement de clé

Partage de secret

DH



Diffie  
Hellman



## 1. Confidentialité

## 2. Attaque MitM Man in the Middle

## 3. Intégrité

## 4. Authenticité

## 5. Disponibilité

## 6. Originalité

## 7. Non-répudiation

## 8. Traçabilité

G:	11124136727432308918812736531716408346878332998901256519320663274861933580804712151 00797237775771467149894260123512660136402158406998369481252844452718796		
N:	77818714124031252720819944202374984988485179601906368139945936248931661607804455138 87821412313009361625936687452586174696909303141691170346568788390764347		
Next Bob and Alice will generate two random numbers (X and Y), calculate an X value and a Y value, respectively:			
Bob's X Value	39	Alice's Y value	86
Bob's A value	156437054090510643708213487397359650 653706600103085237076814872023455692 501938943883924390306994339326651518 339656294271164634334019120166300414 0138530961	Alice's B value	867562183713756921586080163515215184 994404228948268998120566392949664779 371069184674731003959263884633988486 672884899488683959430851633340784964 621708228
A=G <sup>x</sup> mod N		B=G <sup>y</sup> mod N	
and Bob will send his A value to Alice, and Alice will send her B value to Bob, and they now re-calculate the values to generate the <b>same shared key</b> :			
Bob's Key	8444217310575129445600473794887733095 7898956649193468344758061625930880277 3327317656211099035659946951701937245 2044286993289357313115568251376348740 50285	Alice's Key	8444217310575129445600473794887733095 7898956649193468344758061625930880277 3327317656211099035659946951701937245 2044286993289357313115568251376348740 50285
Key=B <sup>x</sup> mod N		Key=A <sup>y</sup> mod N	

# Etablissement de clé

Partage de secret

DH

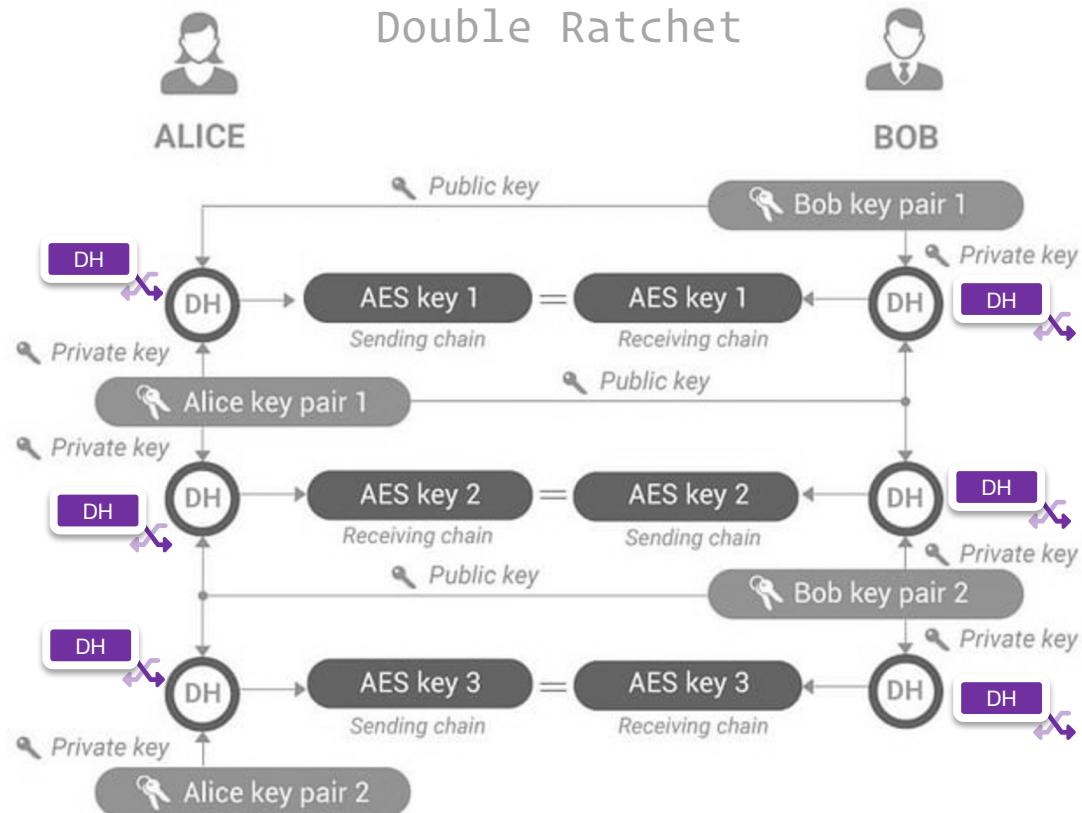


Diffie  
Hellman



- **Symmetric end-to-end encryption.** Message decrypted on all recipients' ends with the same exchange.
- **Forward secrecy.** Unique ephemeral keys are compromised, all your other messages remain therefore safe.
- **Independent key renewal.** The algorithm does get new keys. It uses key derivation functions
- **Plausible deniability.** If a message gets intercepted, who has sent it.
- **No lost or out-of-order messages.** Each message has a header. This way, if a message gets lost or out of order,

## Double Ratchet



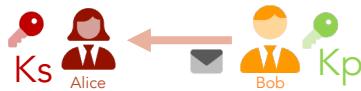
# Asymétrique

Chiffrement Asymétrique

RSA



Rivest  
Shamir  
Adleman



## 1. Confidentialité

- Lent
- Petit volume

## 2. Intégrité

## 3. Authenticité

## 4. Disponibilité

## 5. Originalité

## 6. Non-répudiation

## 7. Traçabilité

le secret

RSA

M



C

RSA



M

L'opération publique de chiffrement transforme à l'aide de la clé publique **Kp** un message clair **M** en un message chiffré **C**. L'opération privée de déchiffrement permet de recalculer **M** à partir de **C** et de la clé privée **Ks**. Le chiffrement asymétrique permet donc à toute personne ayant accès à la clé publique de chiffrer des messages à l'intention du détenteur de la clé privée.

R18

## Paramètres de courbes elliptiques pour le DLOG

Les paramètres de courbes elliptiques P256r1, P384r1 et P512r1 de la fa pool, les paramètres de courbes elliptiques P-256, P-384 et P-521 du paramètres de courbes Curve25519 et Curve448 sont recommandés.

Familles de courbes	Courbes	R/O	Notes
Brainpool [RFC5639]	BrainpoolP256r1	R	
	BrainpoolP384r1	R	5.3.a
	BrainpoolP512r1	R	
NIST [FIPS186] (voir Annexe D.1.2)	NIST P-256	R	
	NIST P-384	R	5.3.a
	NIST P-521	R	
IETF [RFC7748]	Curve25519	R	
	Curve448	R	5.3.b

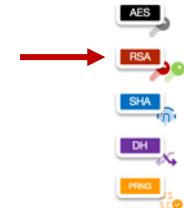
R16

## Dimensionnement du schéma asymétrique RSA

On recommande d'utiliser des modules RSA d'au moins 3072 bits publics  $e$  de taille strictement supérieure à  $2^{16}$ .

Primitive	Taille des paramètres	R/O	Notes
RSA	$n \geq 3072, \log_2(e) > 16$	R	
	$n \geq 2048, \log_2(e) > 16$	O	

1. ECC
2. Post Quantique
  - FALCON
  - SPHINCS+
  - CRYSTALS-KYBER
  - CRYSTALS-DILITHIUM



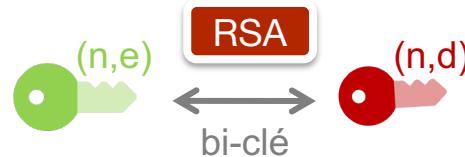
# Asymétrique

Chiffrement Asymétrique

RSA



Rivest  
Shamir  
Adleman



$$C \equiv M^e \pmod{n}$$

$$M \equiv C^d \pmod{n}$$

p et q premiers

$$n=p \cdot q$$

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

e premier avec  $\phi(n)$

d inverse de e modulo  $\phi(n)$

- Comme  $c = m^e \pmod{n}$ ,  $c^d \pmod{n} = m^{ed} \pmod{n}$
- Comme  $ed = 1 \pmod{(p-1)(q-1)}$  il existe un entier k tel que  $ed = 1 + k(p-1)(q-1)$
- Par conséquent  $c^d \pmod{n} = m^{1+k(p-1)(q-1)} \pmod{n}$
- Or (théorème de Fermat)  $m^{(p-1)} \pmod{p} = 1$  si m n'est pas multiple de p. Par élévation à la puissance  $k(q-1)$  puis multiplication par m on obtient :  $m^{1+k(p-1)(q-1)} \pmod{p} = m$  égalité qui reste vraie (2 membres=0) si m est multiple de p
- Par symétrie  $m^{1+k(p-1)(q-1)} \pmod{q} = m$  donc  $m^{1+k(p-1)(q-1)} - m$  est divisible par p et q donc par pq (p et q premiers et différents)
- Par conséquent  $c^d \pmod{n} = m^{1+k(p-1)(q-1)} \pmod{pq} = m$

# Asymétrique

Chiffrement Asymétrique

RSA



Rivest  
Shamir  
Adleman

$$C \equiv M^e \pmod{n}$$

$$M \equiv C^d \pmod{n}$$

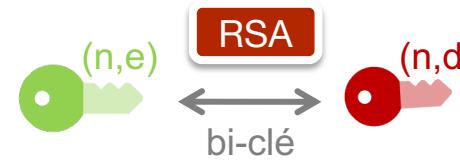
Message=hello

p=1780731609485571264810668272791995899914711193175875151378804  
074228378407969900390762278303079206056838884626265923868792218  
367632557891163061154753567325103171346019724100725958290999956  
638624959629388394207924641815440071623570127595625788276627675  
29379905026780616246336287030939352827543039555774118880861

q=1659131989902429311320647733356628360786681694637180433412560  
894004052610620886878487919454167845518159615723611150571927731  
077520725426964037514349809386962699556220313541603209708580761  
255180830815503806133227993555854257708408662876270644910978887  
84189682166960030796126554428589177031865285035399557919043

e=65537

d=7541588446120496966253234620344712333501069795303049916555478  
181024739612748768714849307670020087520498050753853969336769842  
721963177053759904513002332911786096655412117391051494984460634  
179886021618010916184586203664729878176106218580539243017832573  
064752374135639173722301872742910212929520185645517211756292260  
442337408227686415915940057868098542605732388150060822257263017  
768153073048470043906529305219251168471687810973059378400133633  
644318414348198698711374492042237109200383333030466757157771841  
088217670969580298385949540902819878485019509485570883603413467  
23410869909782189658972296699532605527832607563513



p et q premiers  
n=p.q  
 $\phi(n) = (p - 1)(q - 1)$   
e premier avec  $\phi(n)$   
d inverse de e modulo  $\phi(n)$

N=2954468778727951519381542455099117772733932758531747159909715903  
818628489453739346434808277662182293208684035240167304823272057321  
422288076172288054479598539813596184732724354963397672285147033236  
886721151377782449815143774439760136571897904056785781223890706818  
861764171429839404340406627783874748342767092341923727807298159971  
873503674156291633101481940976644619125965267514073300546813320042  
580729928081378915771201586619482771747482544538290338600642244942  
010670574478156402877487966789398331122247930143951606695548203913  
045284760438130510844259880483102317018508493449083312299070937455  
1035113290128717200136023

Public key (e,n)   Private key (d,n)

cipher=273743492279956537274854241317285273489687949926990383595577  
818303469365389369383595210177750765405360365710781579824329830305  
899759170691700224655572029292953466922904049974236228138933712837  
677798511174101407776261740120993841843494109034252357715708754150  
858729358036738829777908947650774051399543308967832986991616410972  
346874156044067581889268723764995402779650371468342794857706752356  
242882149636133999260072349301958518578960262847313752426012359065  
189237744326344835963663284896710050292993953459275970081116400819  
064990504268034858744518958991288065553666631228513430001665872030  
55372566737379510918409870253

Decipher=hello

# Les 5 primitives cryptographiques

Chiffrement Symétrique



Chiffrement Asymétrique



Fonctions de hachage



Établissement de clé

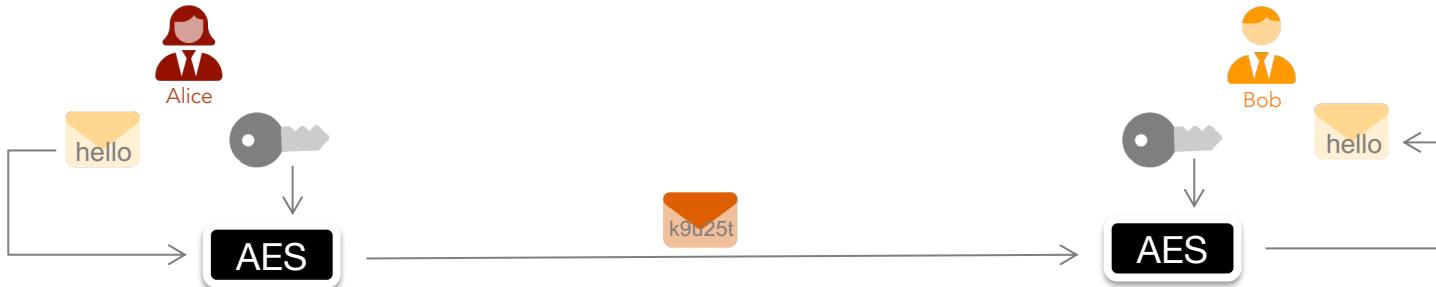


Générateurs d'aléa



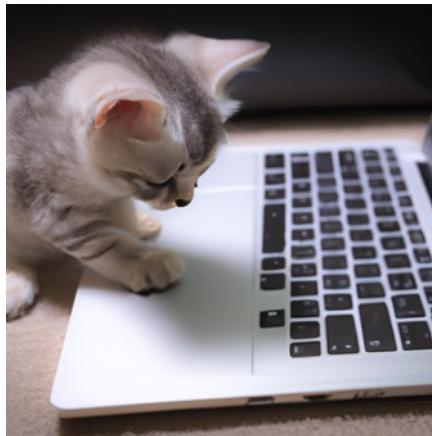
# chiffrement symétrique

AES



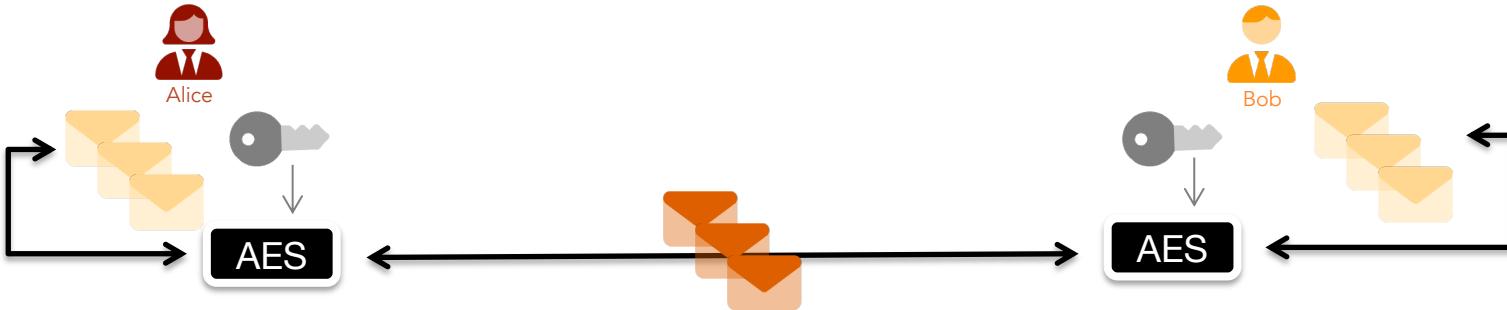
# Chiffrement symétrique

AES  
•



# Chiffrement symétrique

AES

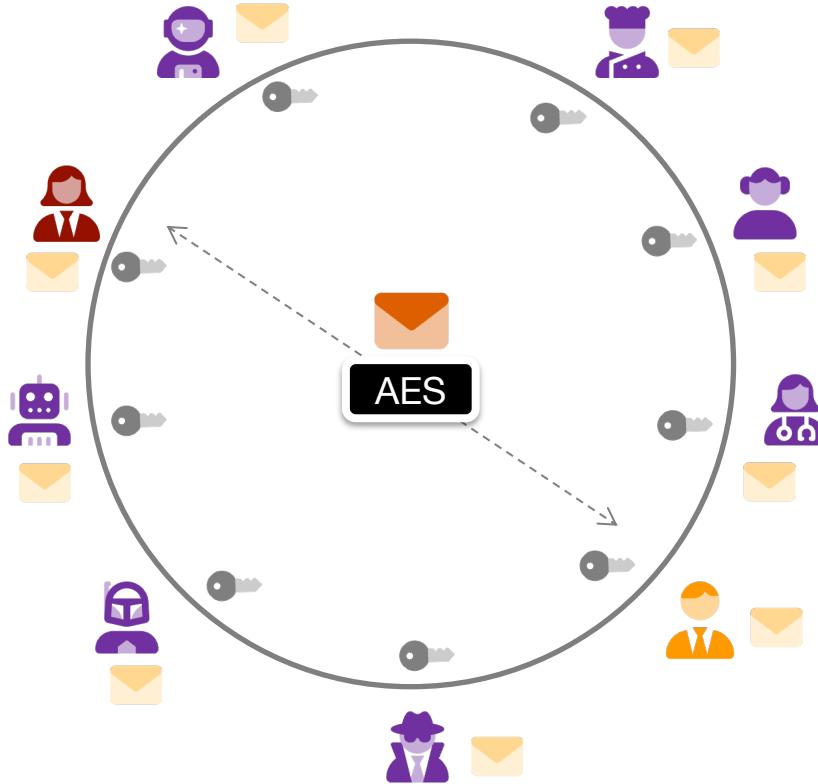


- rapide et optimisé
- chiffrement de larges volumes
- usage simple et direct (dans les 2 sens : A<=>B)
- d'où vient cette unique clé symétrique ?
- établissement de la clé
- partage/échange de la clé

# chiffrement symétrique

AES

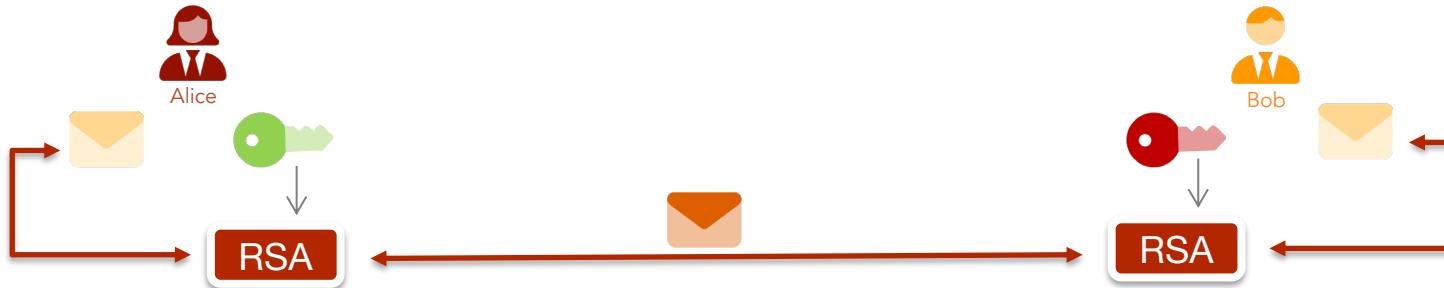
le partage et le maintien confidentiel de la clé est une difficulté



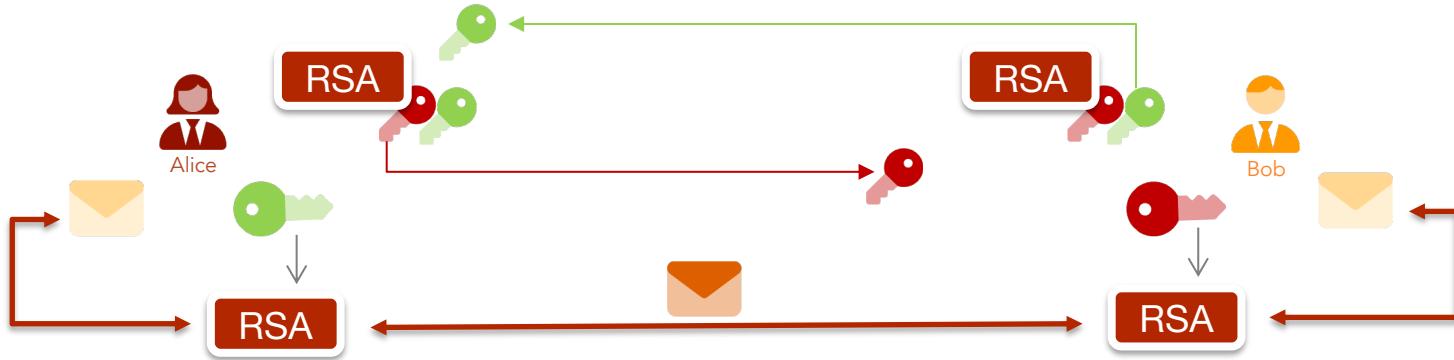
# Chiffrement asymétrique



bi-clé  
issu du  
même  
algorithme

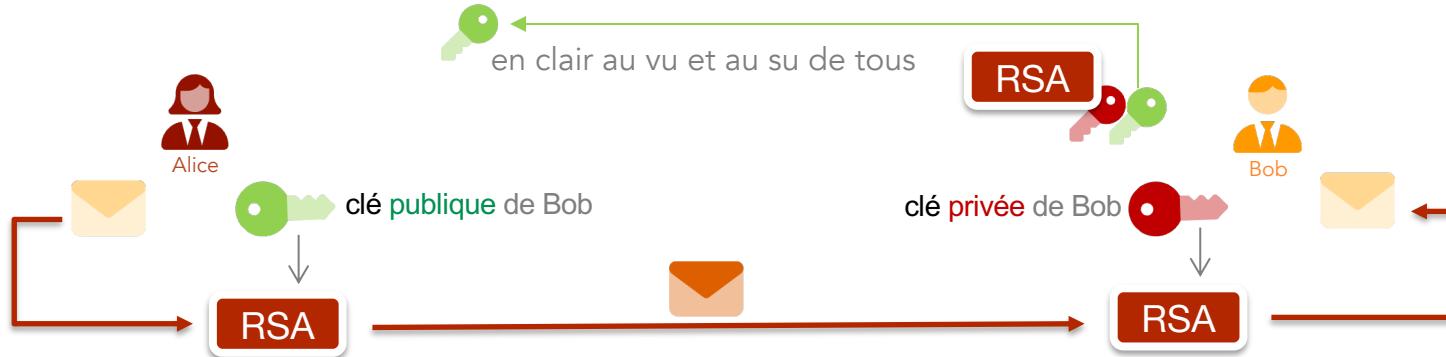


# Chiffrement asymétrique



- génération du bi-clé par celui qui déchiffre
- une des bi-clés est nommée **clé privée**
- l'autre bi-clé est nommée **clé publique**
  
- la **clé publique** est envoyée en claire en vu et au su de tous

# Chiffrement asymétrique



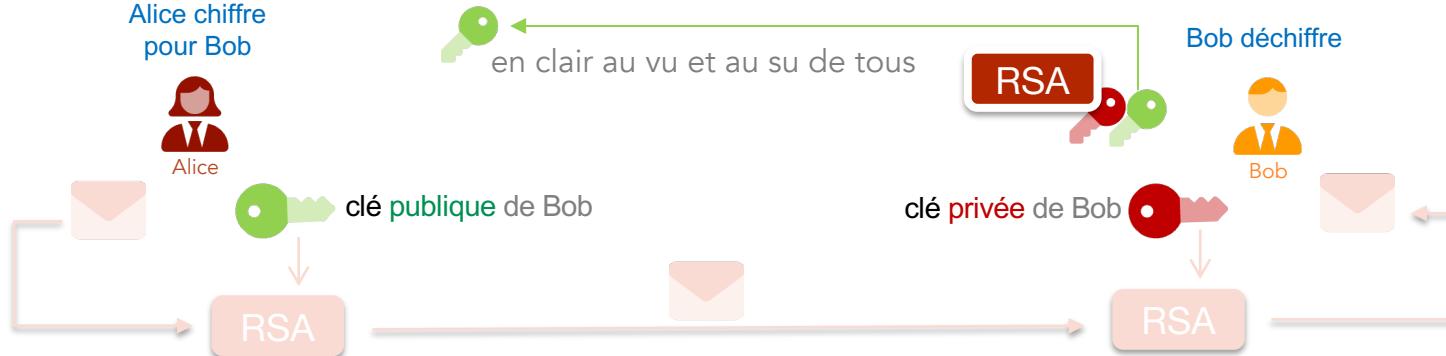
- génération du bi-clé par celui qui déchiffre
- une des bi-clé est nommée **clé privée**
- l'autre bi-clé est nommée **clé publique**
  
- la **clé publique** est envoyée en clair au vu et au su de tous
- seule la **clé publique** permet de chiffrer
- seule la **clé privée** permet de déchiffrer

# Chiffrement asymétrique

RSA



Alice chiffre  
pour Bob



Alice déchiffre

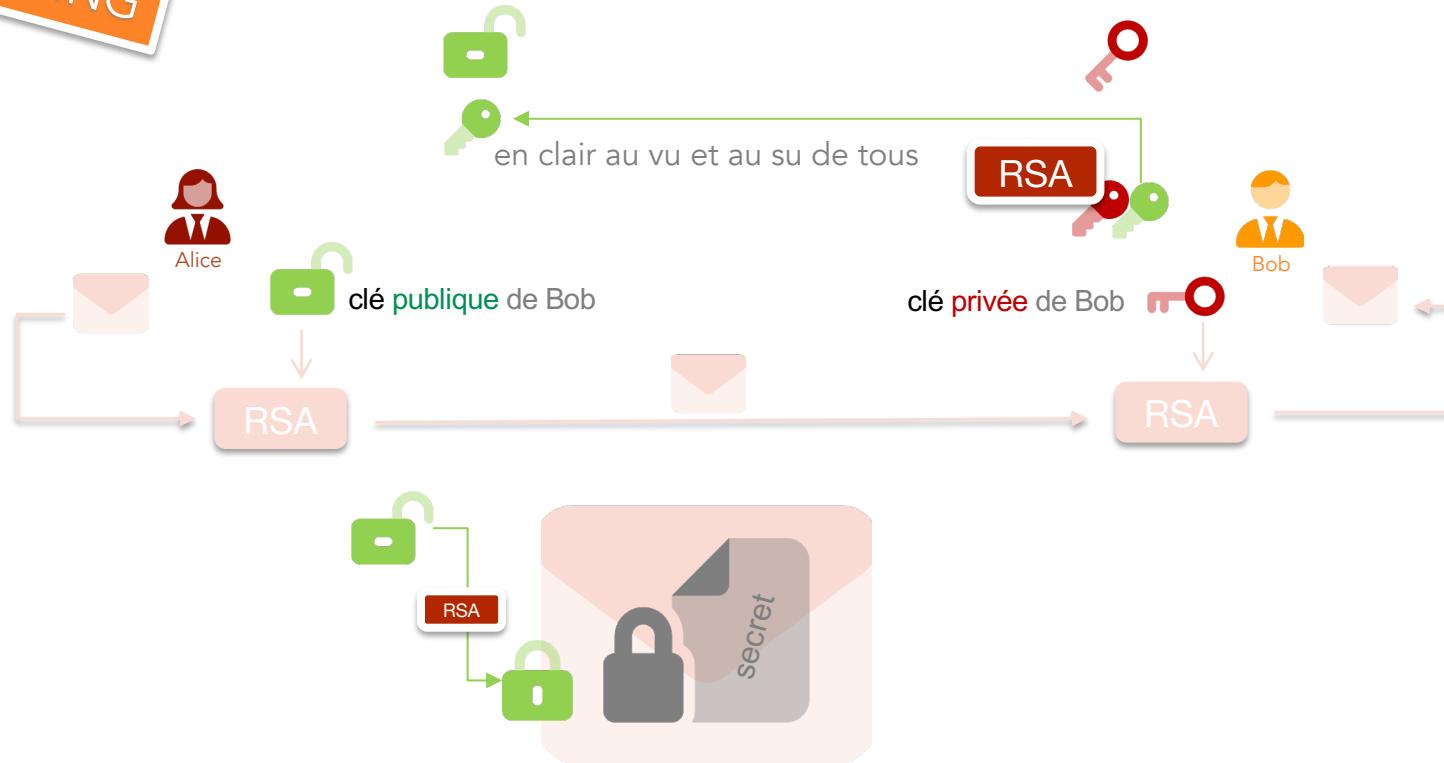


# Analogie Chiffrement asymétrique

RSA

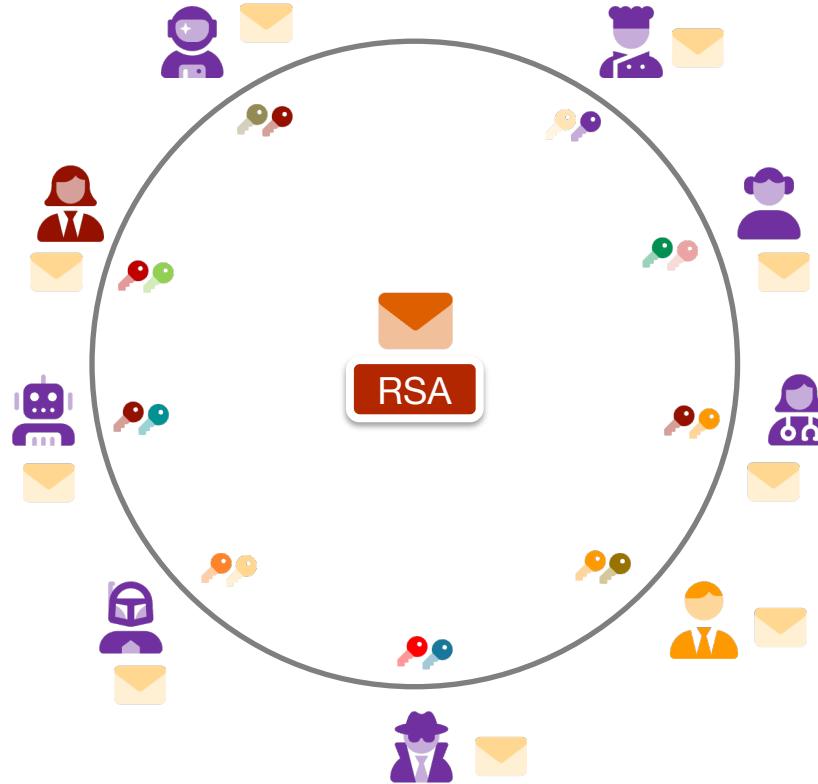


WARNING



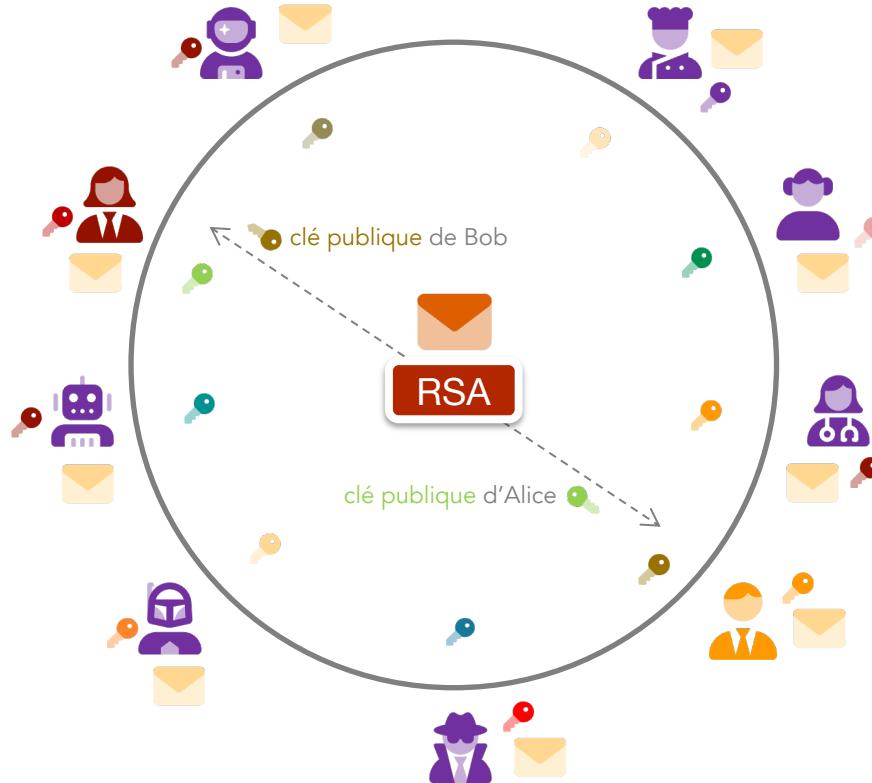
# Chiffrement asymétrique

des pairs de clé avec des clés publiques partagées et diffusées



# Chiffrement asymétrique

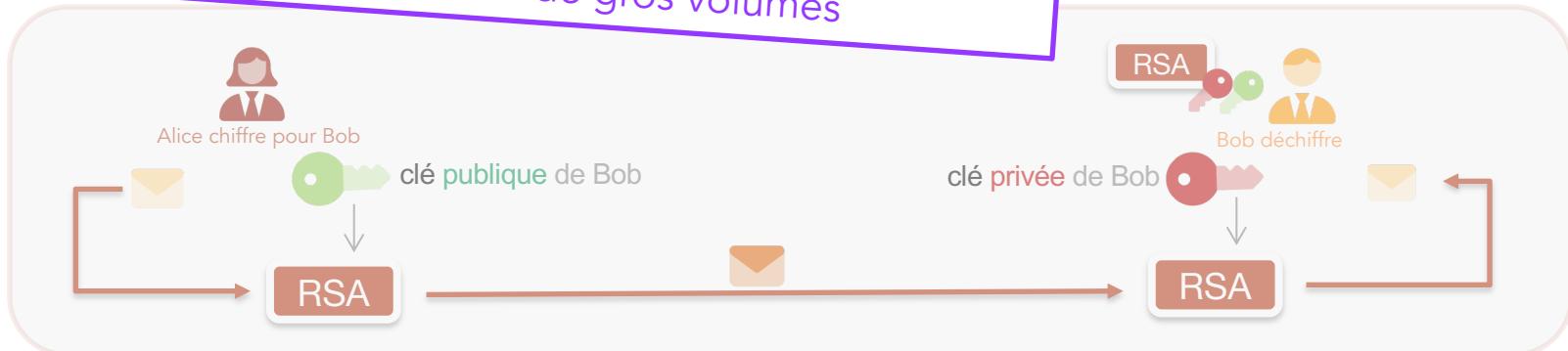
des pairs de clé avec des clés publiques partagées et diffusées



# Chiffrement asymétrique



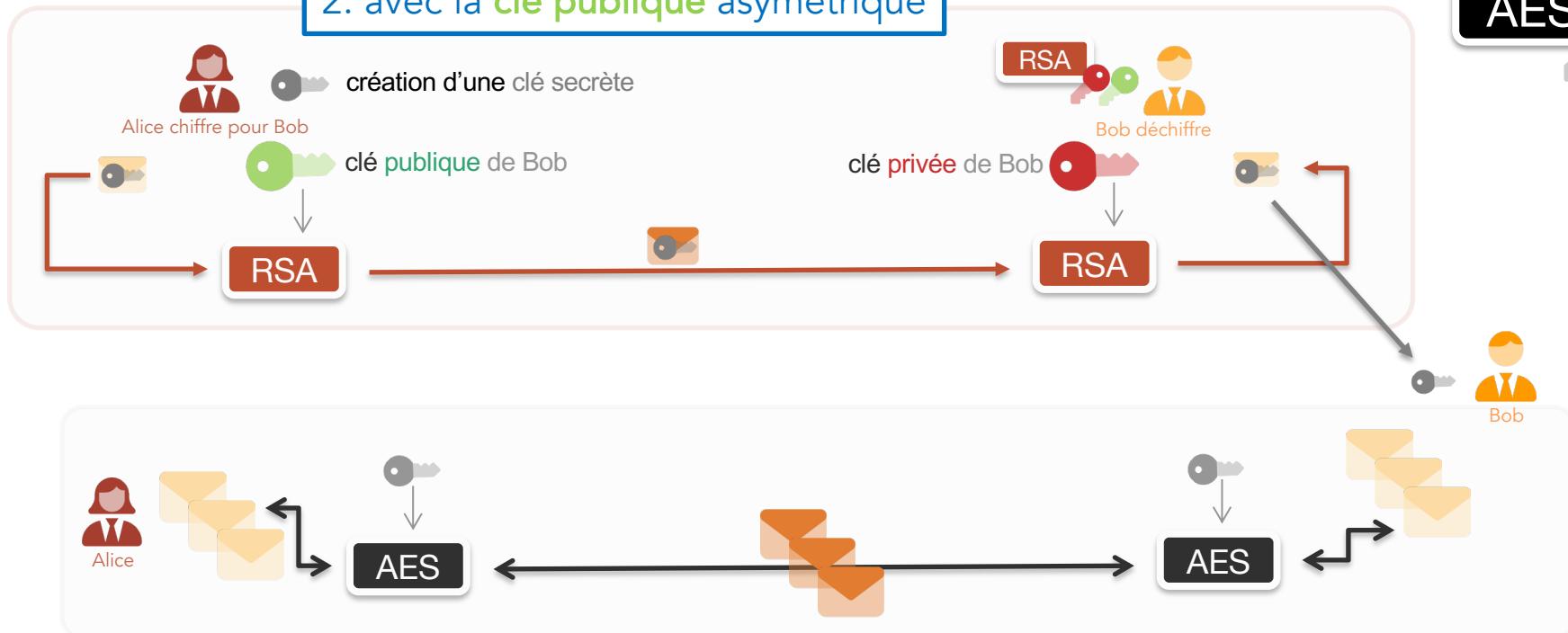
- 1. Beaucoup plus lent que la crypto. symétrique
- 2. Difficile de chiffrer de gros volumes



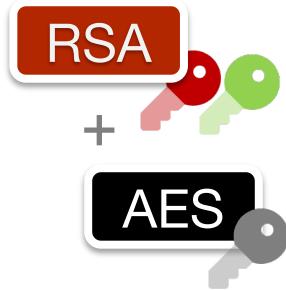
# Chiffrement réel



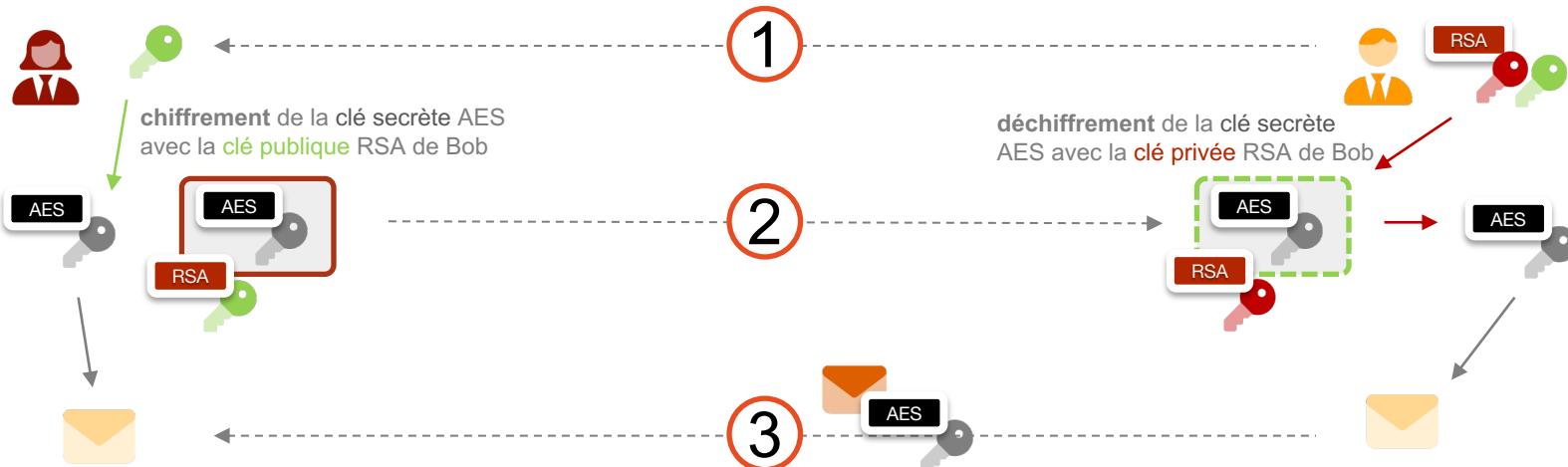
1. Chiffrement de la clé symétrique
2. avec la clé publique asymétrique



# Chiffrement robuste



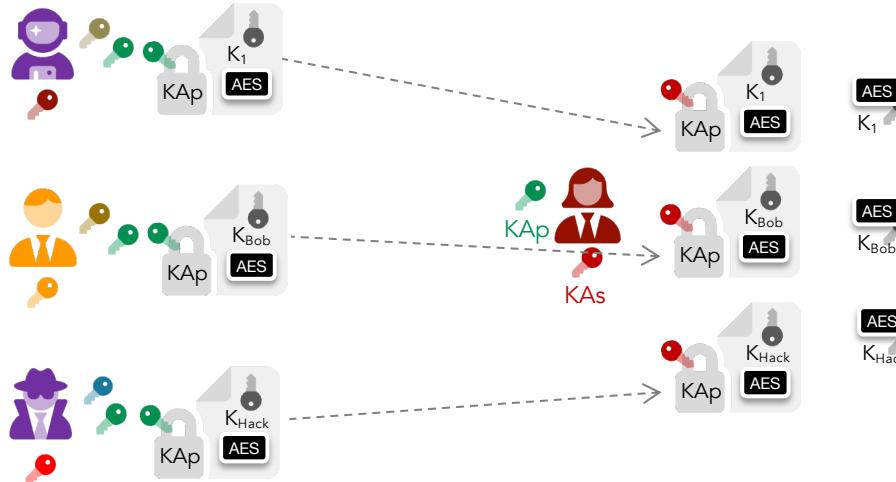
1. Chiffrement de la clé symétrique
2. avec la clé publique asymétrique





# Usage: chiffrement pour la Confidentialité

clé publique d'Alice



Type	Nom
pub	Loïc PERRY
pub	
sec/pub	Benoit LEGER
mult	stelau
mult	elsi

59 clés sur 59 listées



# Usage: chiffrement/Déchiffrement pour confidentialité

clé publique KAp d'Alice



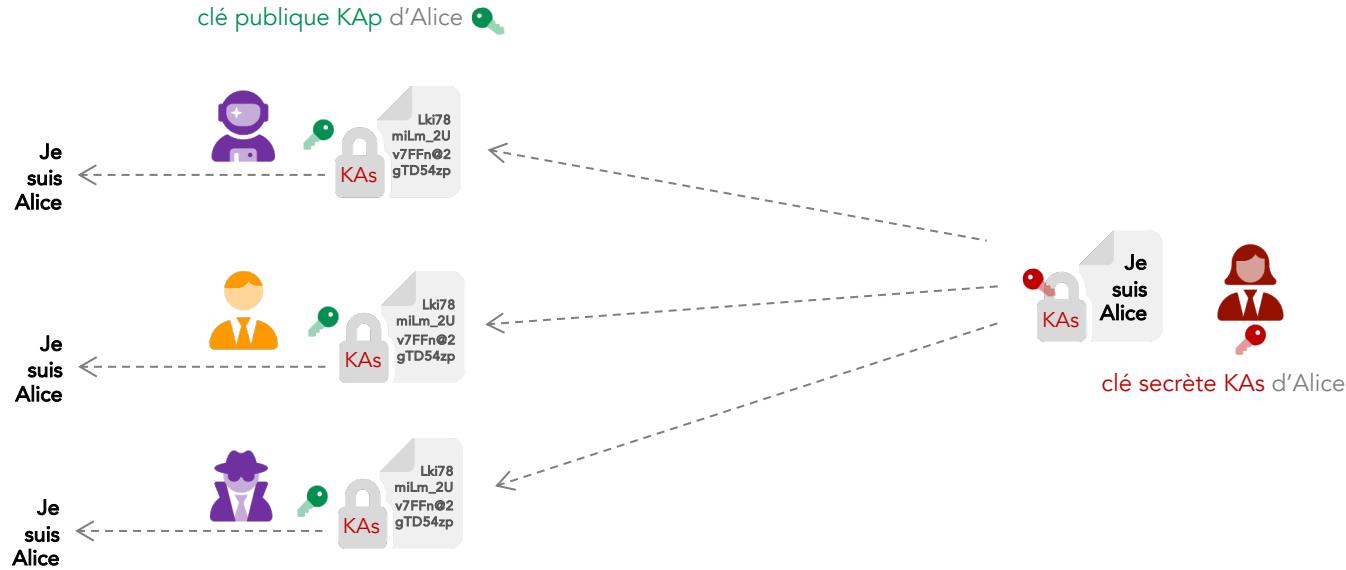
clé secrète KAs d'Alice



Je t'aime -Bob  
Je t'aime -Bob

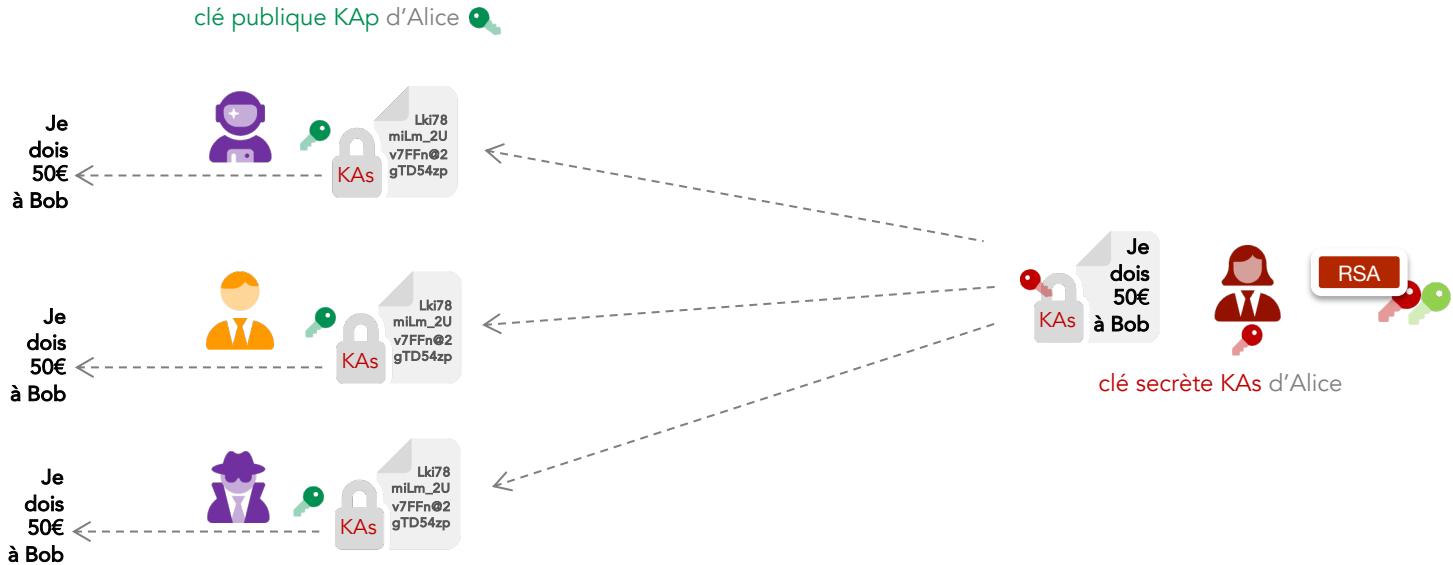


# Usage: ? / ? pour ?



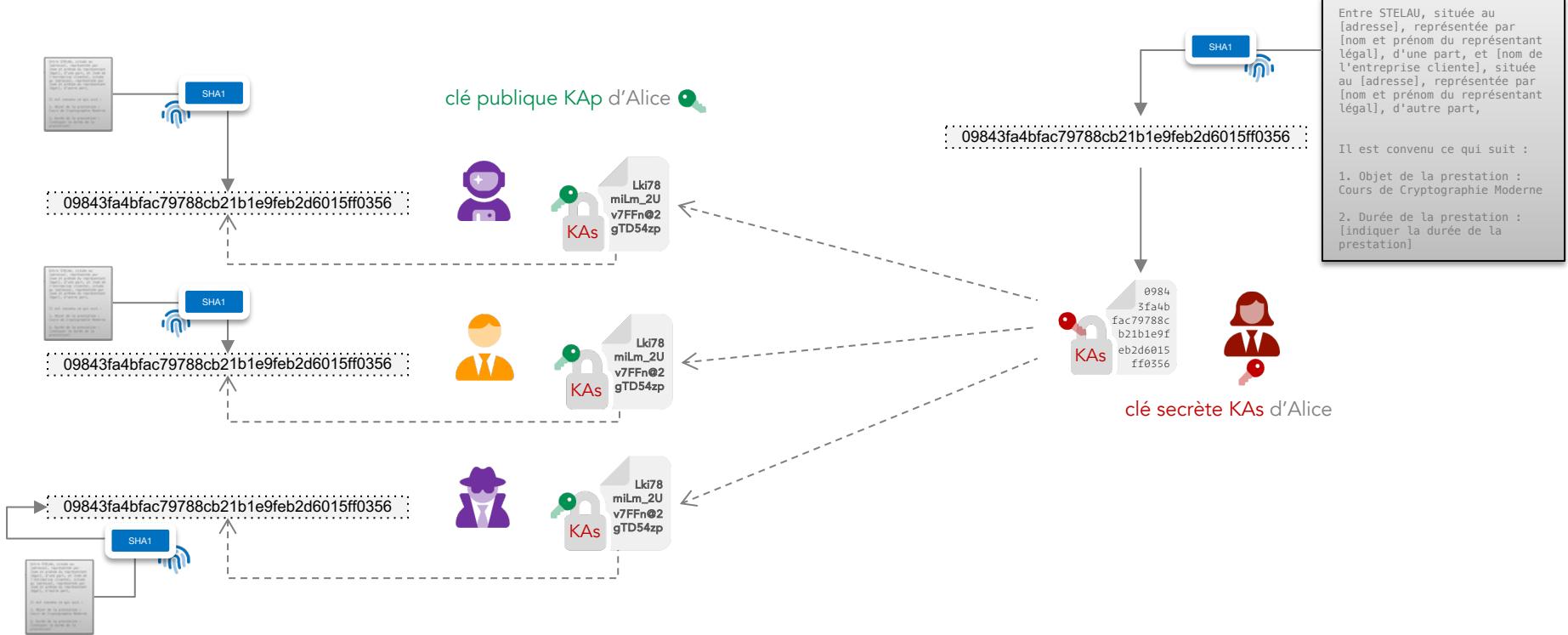


# Usage: signature / vérification pour non-répudiation





# Usage : signature / vérification pour non-répudiation



# Deux usages différents



clé privée

clé publique

chiffrement

déchiffrer

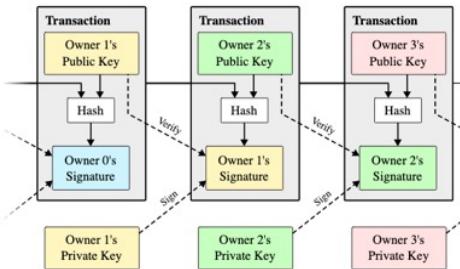
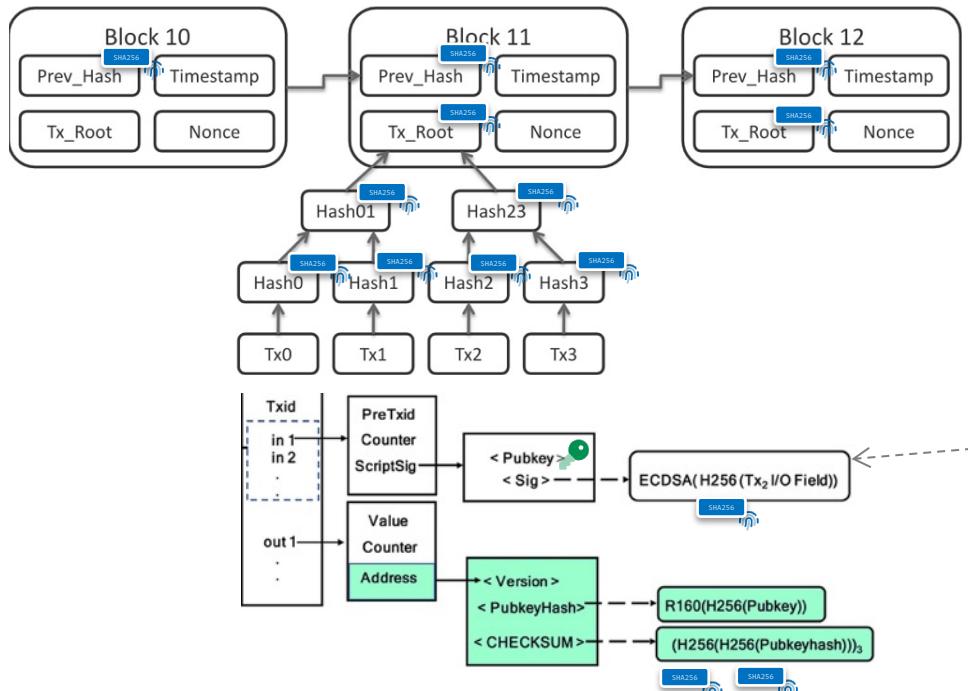
chiffrer

signature

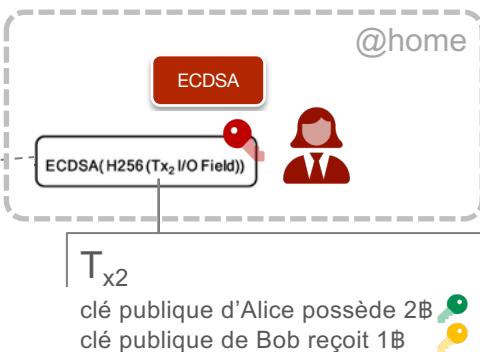
signer

vérifier

# Bitcoin : hashes et signature

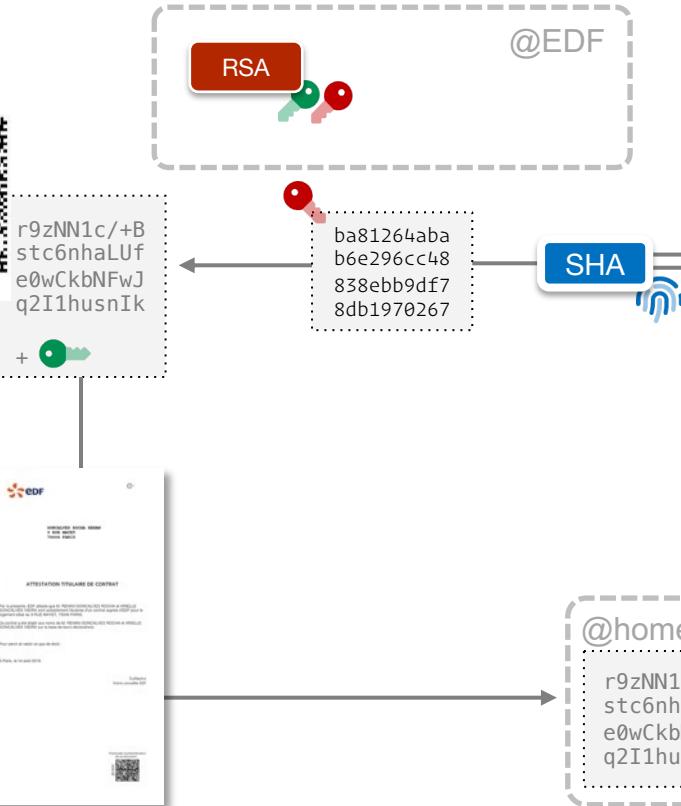


La clé privée de l'expéditeur est utilisée pour signer le hachage SHA-256 de toutes les données de la transaction, y compris les entrées, les sorties et les montants.



Un Tx (ou transaction) en Bitcoin est une opération dans laquelle une ou plusieurs entrées de Bitcoin sont utilisées pour créer une ou plusieurs sorties de Bitcoin. Les entrées de Bitcoin sont des sorties de transactions antérieures qui ont été envoyées à l'adresse Bitcoin du destinataire et qui sont maintenant disponibles pour être dépensées. Les sorties de Bitcoin sont des montants de Bitcoin qui sont envoyés à une adresse Bitcoin.

# CEV : Cachet Electronique Visible



**NOUS CONTACTER**

N° client : Identifiant Internet

**Par Internet et Mobile**  
edf.fr  
sur Smartphone et Tablette  
Télécharger l'app mobile EDF&MOI

**Par téléphone**  
Du lundi au vendredi de 8h et jusqu'à 21h  
09 39 32 15 15 (appel gratuit, prix appel)

**Par courrier**  
EDF SERVICE CLIENT  
TSA 20012  
75197 Paris Cedex 01

**Nos boutiques**  
Retrouvez la boutique la plus proche de chez vous sur [boutiques.edf.com](#)

**Lieu de consommation**

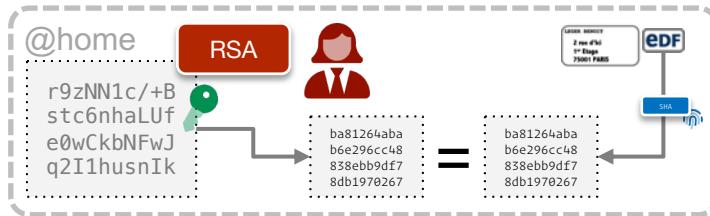
M. LEGER BENOIT  
Votre contrat

## ATTESTATION TITULAIRE DE CONTRAT

Par la présente, EDF atteste que M. BENOIT LEGER est actuellement titulaire auprès d'EDF pour le logement situé au

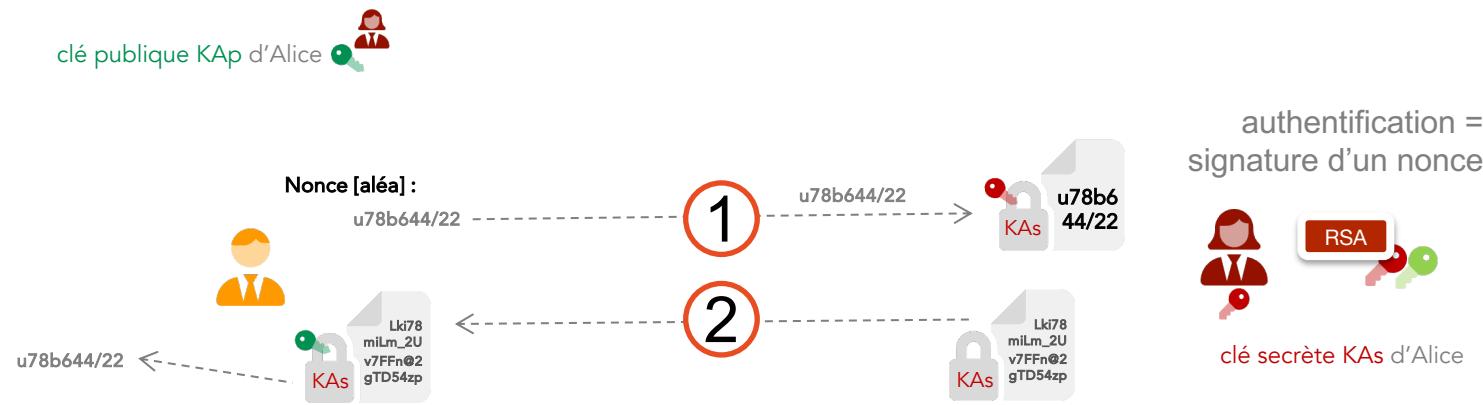
Ce contrat a été établi au nom de M. BENOIT LEGER sur la base de ses décl

Pour servir et valoir ce que de droit.



# Usage: Authentification

RSA



~ 100% des authentifications  
sont établies sur une signature

# Trois usages différents



	clé privée	clé publique
chiffrement	déchiffrer	chiffrer
signature	signer	vérifier
authentification	signer	vérifier

# Les 5 primitives cryptographiques

Chiffrement Symétrique



Chiffrement Asymétrique



Résout la difficulté de  
l'échange de clé  
+  
Permet l'usage  
du principe de **Signature**  
et d'**Authentification**

Fonctions de hachage



Établissement de clé



Générateurs d'aléa



S/MIME (Secure/Multipurpose Internet Mail Extensions)  
e-Passport (Passport électronique)  
Smartphones modernes (sécurisation des données et communications)  
HTTPS (SSL/TLS)  
IPsec (Internet Protocol Security)  
PGP (Pretty Good Privacy) / GPG (GNU Privacy Guard)  
TOTP (Time-Based One-Time Password)  
Bitcoin et autres cryptomonnaies  
SSH (Secure Shell)  
TLS (Transport Layer Security)  
IKE (Internet Key Exchange)  
HSM (Hardware Security Modules)  
Kerberos (protocole d'authentification réseau)  
OTR (Off-the-Record Messaging)  
Signal Protocol (messagerie chiffrée)  
Tor (The Onion Router)  
Wi-Fi Protected Access (WPA2 et WPA3)  
YubiKey (dispositif d'authentification)  
Zoom (chiffrement de communications vidéo)  
Apple Pay / Google Pay (paiements sécurisés)  
Z RTP (Zimmermann Real-Time Transport Protocol)  
OpenSSL (bibliothèque de cryptographie)  
VPNs utilisant OpenVPN ou WireGuard  
ECC (Elliptic Curve Cryptography) utilisé dans divers protocoles et produits  
Let's Encrypt (autorité de certification utilisant ACME)  
RDP (Remote Desktop Protocol) sécurisé  
LUKS (Linux Unified Key Setup) pour le chiffrement de disque  
Signal (application de messagerie)  
WhatsApp (chiffrement de bout en bout pour la messagerie)  
Telegram (mode "Secret Chat" pour la messagerie sécurisée)  
X.509 certificats (pour SSL/TLS)  
TrueCrypt et VeraCrypt (chiffrement de disque)  
RSA SecureID (tokens d'authentification)  
LDAP (LDAP over SSL)  
DKIM (DomainKeys Identified Mail)  
DMARC (Domain-based Message Authentication, Reporting & Conformance)  
SPF (Sender Policy Framework)  
SCTP (Stream Control Transmission Protocol) avec DTLS (Datagram Transport Layer Security)  
SCADA systems (pour sécuriser les systèmes de contrôle industriel)  
EMV (Europay, MasterCard et Visa) pour la sécurité des transactions par carte  
Microsoft BitLocker (chiffrement de disque)  
Apple FileVault (chiffrement de disque)  
Docker Content Trust (signature et vérification d'image)  
Secure Boot et Trusted Platform Module (TPM) dans les systèmes modernes  
HCE (Host Card Emulation) pour paiements mobiles  
1Password, LastPass (gestionnaires de mots de passe)  
ProtonMail (service de courriel sécurisé)  
Brave (navigateur avec des fonctionnalités de sécurité améliorées)

cclé  
EFAIL (vulnérabilités affectant certains clients de messagerie avec S/MIME et PGP)  
HomeKit (sécurité pour les appareils de maison intelligente)  
QUIC (Quick UDP Internet Connections) avec cryptage intégré  
Azure Key Vault (gestion des clés dans le cloud)  
AWS KMS (Amazon Web Services Key Management Service)  
GPG (GNU Privacy Guard) pour le chiffrement des fichiers et des emails  
RSA SecurID Access (solution d'authentification multifactorielle)  
GlobalPlatform (sécurisation des transactions pour les cartes à puce et les dispositifs mobiles)  
MEGA (stockage en nuage avec chiffrement de bout en bout)  
Square (transactions de paiement sécurisées pour les commerçants)  
SSL/TLS pour FTPS (File Transfer Protocol Secure)  
TLS pour SMTP (Simple Mail Transfer Protocol) avec STARTTLS  
Stripe (traitement de paiement en ligne sécurisé)  
Java Card pour les applications sur cartes à puce  
Azure Active Directory (gestion des identités et des accès)  
OAuth 1.0a (protocole d'autorisation)  
Keepass (générationnaire de mots de passe open-source)  
Samsung Knox (sécurité pour les appareils mobiles)  
Android Keystore System (sécurité pour les clés de chiffrement sur les appareils Android)  
Cisco AnyConnect Secure Mobility Client (VPN)  
Fortinet FortiGate (solutions de sécurité réseau et de pare-feu)  
s2n (Signal to Noise), bibliothèque de cryptographie implémentée par AWS  
Keybase (sécurité des équipes et collaboration)  
OpenPGP (cryptage des emails et des fichiers)  
Secure Enclave (sécurisation des données sur les appareils Apple)  
Thales nShield HSMs (Hardware Security Modules pour la protection des clés)  
CyberArk (protection des priviléges et gestion sécurisée des sessions)  
Silent Circle (communications chiffrées)  
Wickr (messagerie chiffrée et appels sécurisés)  
Dust (messagerie sécurisée avec auto-destruction des messages)  
Duo Security (solutions d'autentification à deux facteurs)  
Microsoft Encrypting File System (EFS) pour NTFS  
Ghostery (protection de la vie privée et bloqueur de trackers pour navigateurs)  
EFF's HTTPS Everywhere (extension de navigateur pour forcer le chiffrement HTTPS)  
Mullvad VPN (réseau privé virtuel axé sur la confidentialité)  
NordVPN (VPN avec fonctionnalités de sécurité avancées)  
Let's Encrypt (fournisseur d'autorité de certification pour HTTPS)  
Yubico YubiHSM (module de sécurité matériel pour la protection des clés)  
OpenSSH (implémentation sécurisée du protocole SSH)  
DigiCert (fournisseur d'autorités de certification SSL/TLS)  
Okta (gestion de l'identité et de l'accès)

pfSense (pare-feu open source et routeur)  
I2P (Invisible Internet Project pour une communication sécurisée)  
Lastline (protection avancée contre les menaces)  
Silent Circle's Blackphone (smartphone axé sur la vie privée)  
OpenBlockchain (implémentation Android de PGP)  
Telegram Passport (identification sécurisée pour les services en ligne)  
Hushmail (service email sécurisé)  
VMware Horizon (virtualisation de postes de travail sécurisée)  
Crypto.com (plateforme de paiement et d'échange de cryptomonnaies)  
EncroChat (service de communication crypté, désormais fermé)  
PureVPN (service VPN avec chiffrement sécurisé)  
Line (application de messagerie avec option de chiffrement de bout en bout)  
Signal Private Messenger (messagerie sécurisée et privée)  
Auth0 (plateforme d'authentification et d'autorisation)  
Keycloak (gestion de l'identité et des accès avec support SSO)  
Cloudflare Access (sécurisation des applications internes)  
AxCrypt (logiciel de chiffrement de fichiers pour Windows)  
Citrix Access Gateway (accès sécurisé aux réseaux d'entreprise)  
Windows Hello (authentification biométrique et sécurisée)  
Virtru (chiffrement des emails et des fichiers)  
Clevis and Tang (chiffrement des disques déverrouillable en réseau)  
SpiderOak One (service de stockage et de sauvegarde sécurisé)  
Silent Phone (service d'appels sécurisés)  
Keybase Teams (collaboration sécurisée pour les équipes)  
ESET Secure Authentication (authentification multifactorielle)  
Avast SecureLine VPN (réseau privé virtuel offert par Avast)  
Stunnel (logiciel pour sécuriser les connexions client-serveur)  
Barracuda Networks (solutions de sécurité, pare-feu et systèmes anti-spam)  
Cisco Duo (solution d'autentification à deux facteurs)  
Sophos UTM (Unified Threat Management pour la sécurité réseau)  
Kaspersky Secure Connection (VPN du fournisseur d'antivirus Kaspersky)  
WatchGuard (solutions de sécurité réseau et de pare-feu)  
Steganos Privacy Suite (ensemble d'outils pour la protection de la vie privée)  
Knox de Samsung (sécurisation des téléphones et tablettes Android)  
Boxcryptor (chiffrement pour le stockage cloud)  
Veeam (sécurisation des sauvegardes de données et de la restauration)  
Ledger Nano S (portefeuille matériel pour cryptomonnaies)  
Trezor (autre portefeuille matériel pour cryptomonnaies)  
Knox de Samsung (solution de sécurité pour dispositifs mobiles de Samsung)  
3CX (système de téléphonie IP qui sécurise les communications VoIP)  
FortiClient (client VPN de Fortinet)  
Zix (solutions de chiffrement des emails pour les entreprises)  
CryptoWall (malware de type ransomware utilisant le chiffrement, mentionné ici à titre informatif)

# Où utilise-t-on cela ?

# Crypto asymétrique : attention

## Echange sécurisé de secret

### Clarification

#### ► Key exchange :



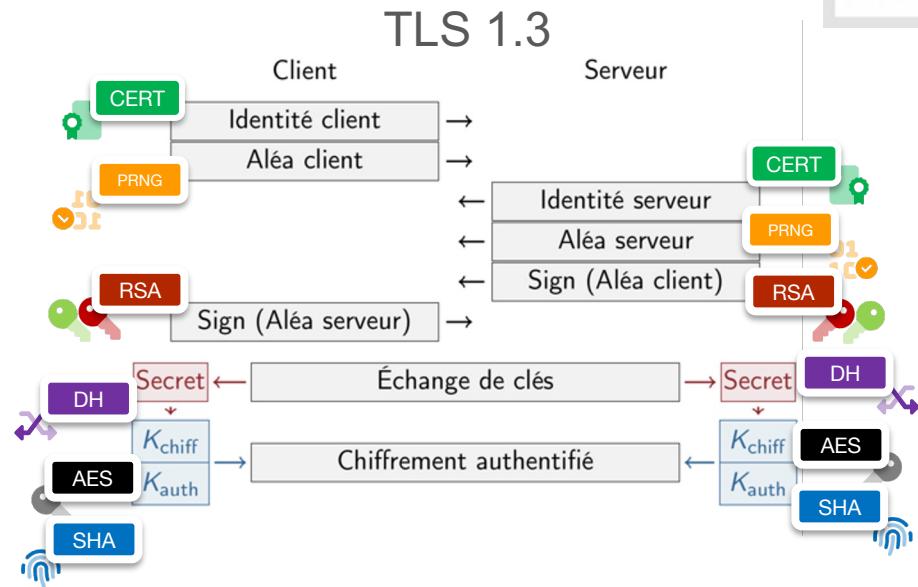
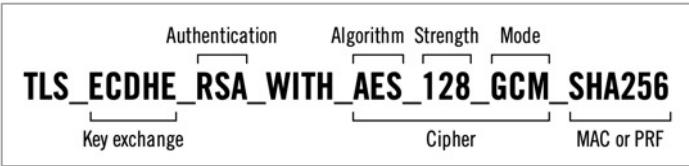
- Sender generates a key and encrypts it using receiver's public key
- Receiver does not participate in key generation. Only sender.
- RSA is typically used for key exchange.

#### ► Key agreement :



- Sender and receiver work together to generate a key.
- This is what DH provides.

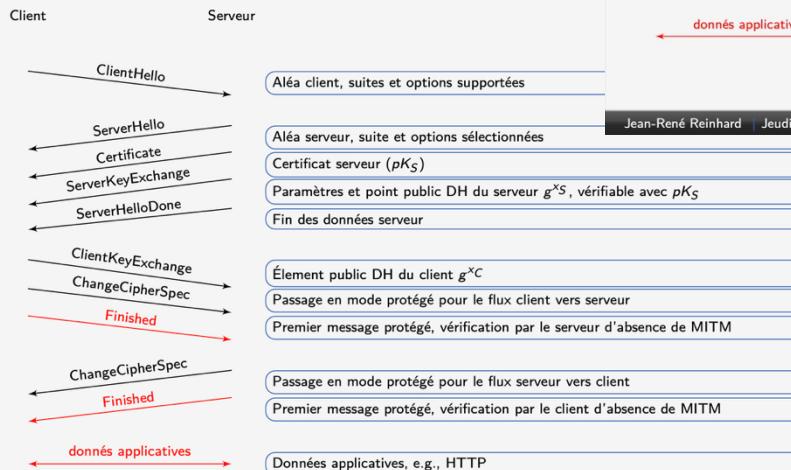
# Assemblage Crypto



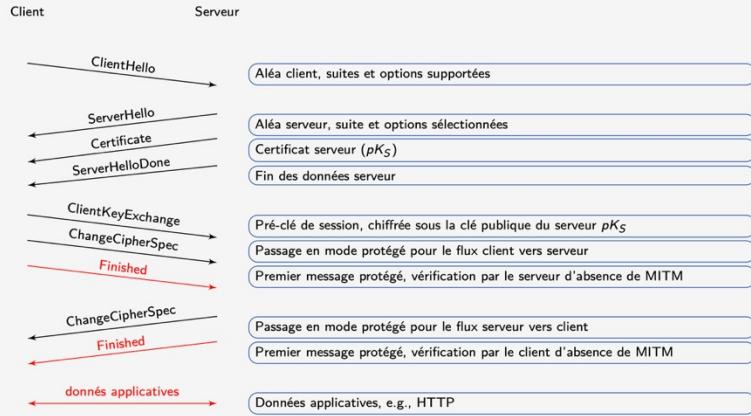
Cipher Suite Name	Auth	KX	Cipher	MAC	PRF
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	RSA	ECDHE	AES-128-GCM	-	SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	ECDSA	ECDHE	AES-256-GCM	-	SHA384
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA	RSA	DHE	3DES-EDE-CBC	SHA1	Protocol
TLS_RSA_WITH_AES_128_CBC_SHA	RSA	RSA	AES-128-CBC	SHA1	Protocol
TLS_ECDHE_ECDSA_WITH_AES_128_CCM	ECDSA	ECDHE	AES-128-CCM	-	SHA256

# Assemblage Crypto

## SSL/TLS : établissement de clé DHE



## SSL/TLS : établissement de clé RSA



Jean-René Reinhard | Jeudi 14 avril 2016

15 / 29

# Certificat électronique



Ce n'est pas une primitive cryptographique

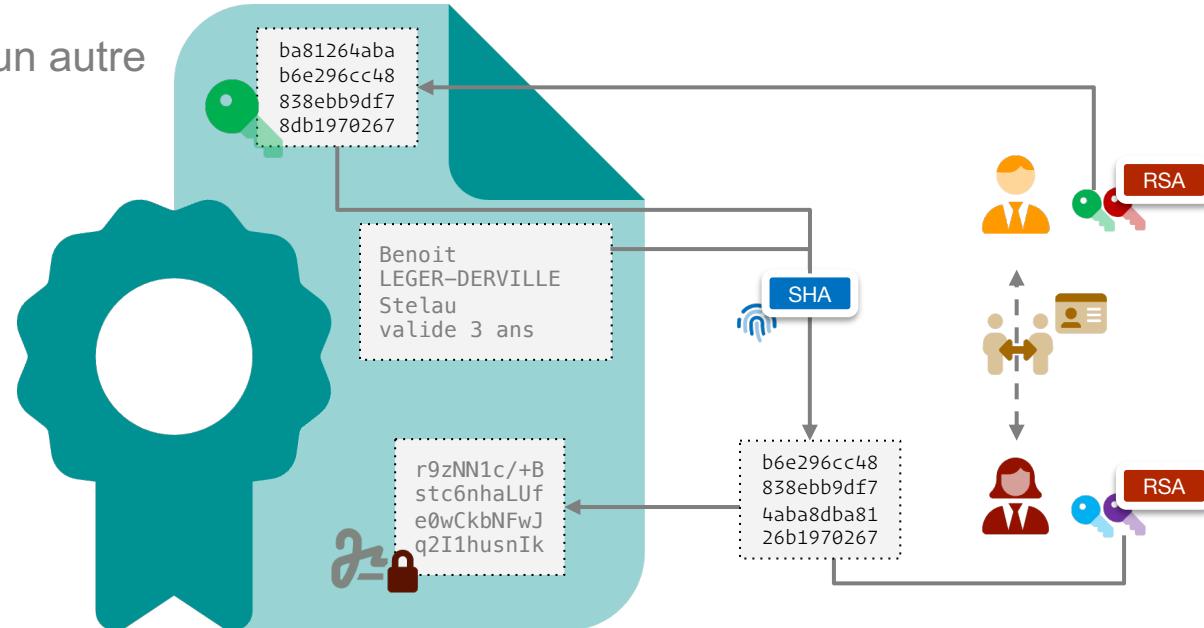
C'est un assemblage

C'est un fichier comme un autre

1. identité

2. clé publique

3. signature  
de l'identité  
par un tiers



# Very Short Crypto Story

3000 ans de crypto. **symétrique**

*recettes militaro-diplomatiques  
de confusion et de diffusion*

**Confusion et Diffusion**  
*« tant bien que mal »*  
de César à Enigma

100 ans de crypto. **moderne**

*de Kerckhoffs ...  
au crypto-système incassable*



1. Principes de Kerckhoffs - 1883
2. One Time Pad - 1917

50 ans de crypto. **asymétrique**

*LA véritable révolution*



Résout la difficulté de  
l'échange de clé  
+

Permet l'usage du  
principe de **Signature**

20 ans de crypto. **quantique**

*révolution ? (ou pas)*

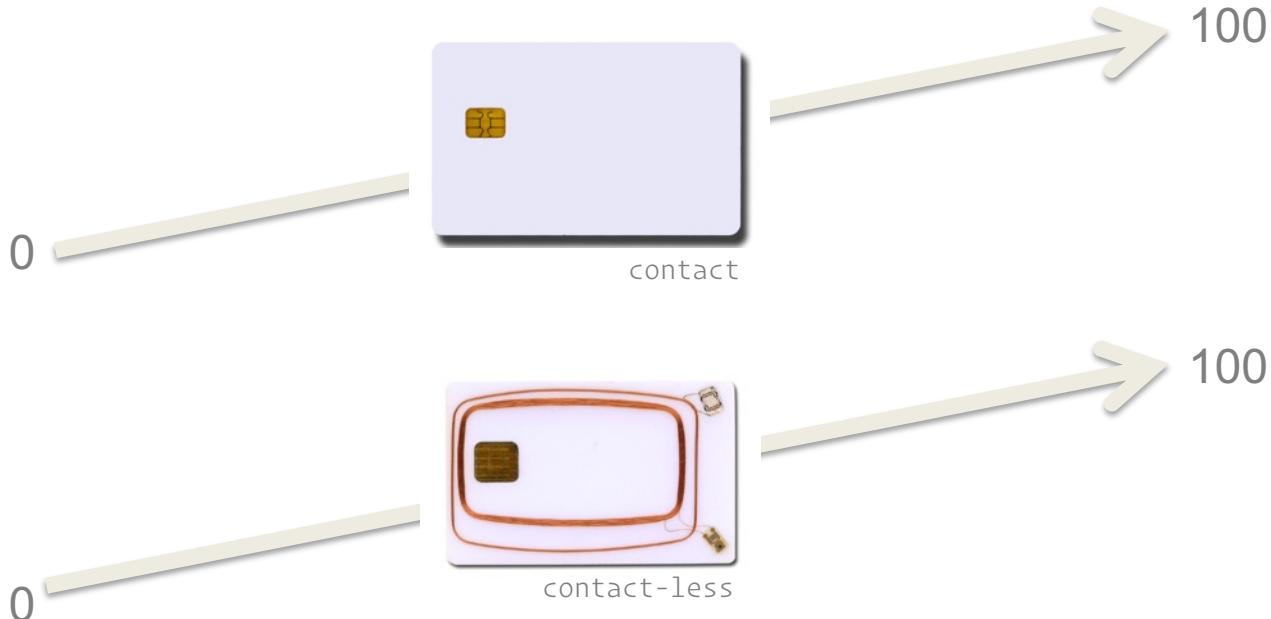


# Les vraies difficultés de la cryptographie moderne

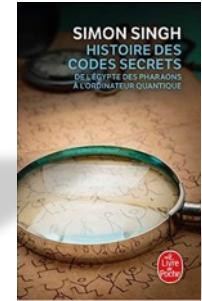
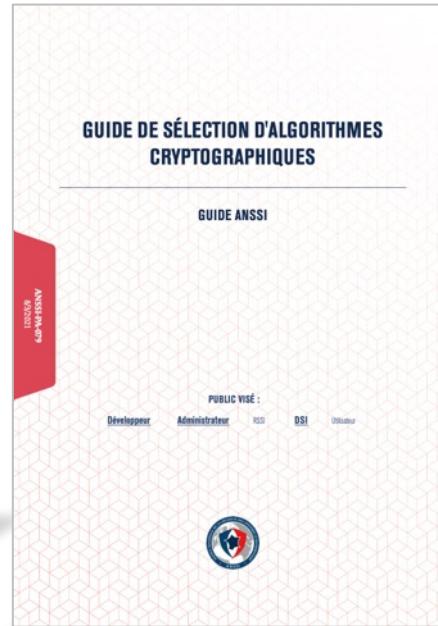
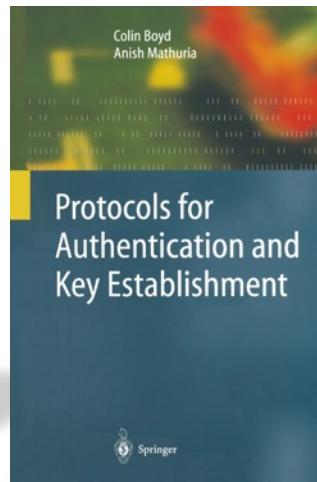
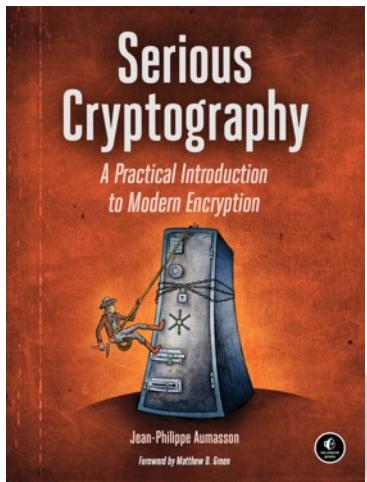
1. THEORIE : Cryptologie
  - failles théoriques = **mathématiques**
  - cryptologue est un métier
  
2. CODE : Implémentations
  - erreurs/failles/vuln = **informatique**
  - « Do not implement cryptography yourself ! »
  
3. UX : Usages
  - mauvais usages = ignorance/pusillanimité
  - bons usages = **formation**



# Attention : « C'est (pas) sécurisé ! » ne veut rien dire

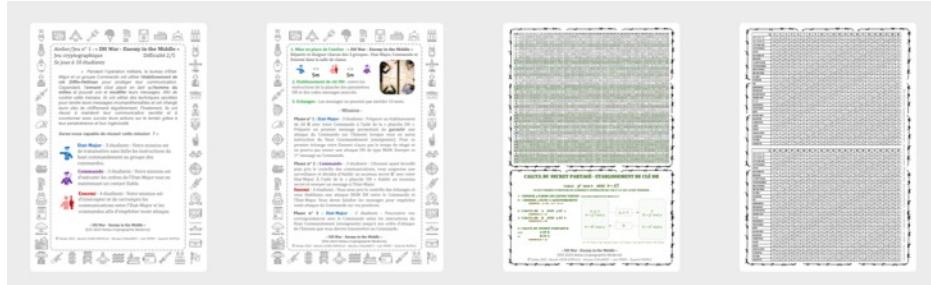


# Fin Ouvrages Cryptographiques



# Ateliers/jeux Cryptographie Moderne

## Atelier 1 : DH War- Ennemy in the Middle



4 groupes  
de 10 étudiants

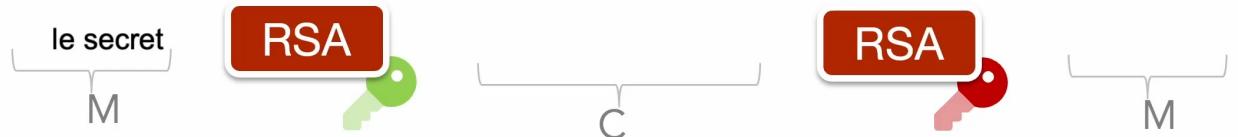
4 groupes  
de 5 étudiants

## Atelier 2 : Wannacry Nightmare Puzzle



Hello !

AES



Hello !

SHA



26, 47, 10

DH



91690410bec9  
graine

PRNG



798

aléa



7e0950bb938539162d268b379595  
44efb87b718950b4f721dd5c945f7  
d12fc4efac9d9b5f0c81bbc1555c3d7  
6610ef3080a354e60b625fc50a23  
a6bffd13ec024239ddc047706c9a23  
11fc38e37161e87501236542732797  
2469b3985721cc0fea3b04047a9c5  
b559e3471a736f5e4c7b473b2e86b1  
b21dd8a829828d f8d6

## Rappels

hash crypto – RSA - certificats – IGC/PKI



# Hash

## fonctions de hachage cryptographiques



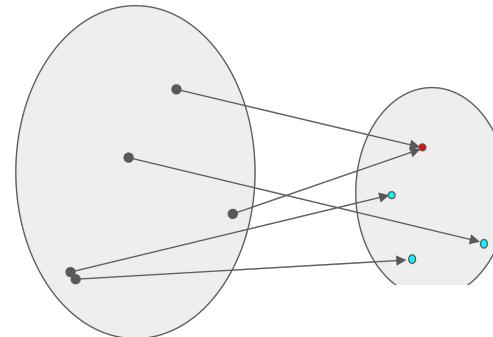
- Fonctions à sens unique
  - d'un espace infini vers un espace fini
  - 128, 160, 224, 256 ou 512 bits
- Résistantes aux attaques
  - 1<sup>ère</sup> pré-image
  - 2<sup>de</sup> pré-image
  - collision

# Hash

## fonctions de hachage cryptographiques



- Fonctions à sens unique
  - d'un espace infini vers un espace fini
  - 128, 160, 224, 256 ou 512 bits
- Résistantes aux attaques
  - 1<sup>ère</sup> pré-image
  - 2<sup>de</sup> pré-image
  - collision

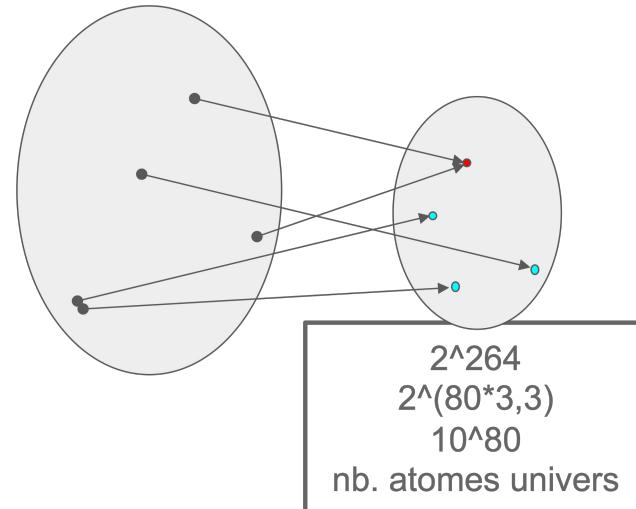


# Hash

## fonctions de hachage cryptographiques



- Fonctions à sens unique
  - d'un espace infini vers un espace fini
  - 128, 160, 224, 256 ou 512 bits
- Résistantes aux attaques
  - 1<sup>ère</sup> pré-image
  - 2<sup>de</sup> pré-image
  - collision



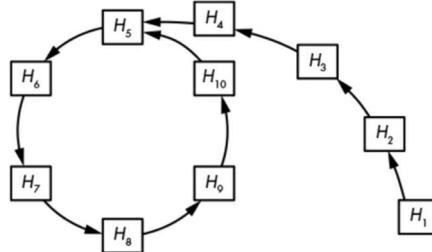
# Hash

## fonctions de hachage cryptographiques



- Résistantes aux attaques

- 1<sup>ère</sup> pré-image
- 2<sup>de</sup> pré-image
- collision



1. il est très difficile de trouver le contenu du message à partir de son condensat
2. à partir d'un message donné et de son condensat (et de la fonction de hachage), il est très difficile de générer un autre message qui donne le même condensat
3. il est très difficile de trouver deux messages aléatoires qui donnent un même condensat (résistance aux collisions)



# Hash

## fonctions de hachage cryptographiques

- Résistance aux préimages

Attaque : avec  $x$  donné, trouver  $m$  tel que  $H(m) = x$

- Résistance aux secondes préimages

Attaque : avec  $m_1$  donné, trouver  $m_2$  tel que  $H(m_1) = H(m_2)$

- Résistance aux collisions

Attaque : trouver  $m_1$  et  $m_2$  tel que  $H(m_1) = H(m_2)$

# Hash

5baaa61e4c9b93f3f0682250b6f8331b7ee68fd8



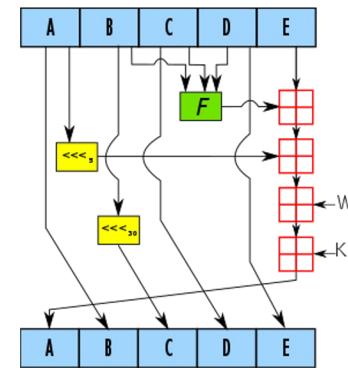
## fonctions de hachage cryptographiques

```
SHA1-compress(H, M) {
    (a0, b0, c0, d0, e0) = H // parsing H as five 32-bit big endian words
    (a, b, c, d, e) = SHA1-blockcipher(a0, b0, c0, d0, e0, M)
    return (a + a0, b + b0, c + c0, d + d0, e + e0)
}

SHA1-blockcipher(a, b, c, d, e, M) {
    W = expand(M)
    for i = 0 to 79 {
        new = (a <<< 5) + f(i, b, c, d) + e + K[i] + W[i]
        (a, b, c, d, e) = (new, a, b >>> 2, c, d)
    }
    return (a, b, c, d, e)
}

expand(M) {
    // the 512-bit M is seen as an array of sixteen 32-bit words
    W = empty array of eighty 32-bit words
    for i = 0 to 15 {
        if i < 16 then W[i] = M[i]
        else
            W[i] = (W[i - 3] ⊕ W[i - 8] ⊕ W[i - 14] ⊕ W[i - 16]) <<< 1
    }
    return W
}

f(i, b, c, d) {
    if i < 20 then return ((b & c) ⊕ (~b & d))
    if i < 40 then return (b ⊕ c ⊕ d)
    if i < 60 then return ((b & c) ⊕ (b & d) ⊕ (c & d))
    if i < 80 then return (b ⊕ c ⊕ d)
}
```



```
275     x[6] = byte(s >> 8)
276     x[7] = byte(s)
277 }
278 func putUInt32(x []byte, s uint32) {
279     _ = x[3]
280     x[0] = byte(s >> 24)
281     x[1] = byte(s >> 16)
282     x[2] = byte(s >> 8)
283     x[3] = byte(s)
284 }
285 }
```

Source : Serious Cryptography  
Copyright © 2018 by Jean-Philippe Aumasson

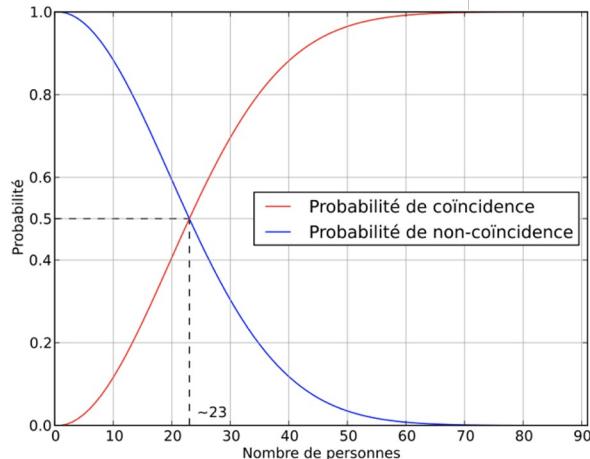
# Hash

## le paradoxe des anniversaires



Soit  $E$  un ensemble fini. La probabilité  $p(n)$  que, parmi  $n$  éléments de  $E$ , chaque élément étant tiré uniformément dans tout l'ensemble  $E$ , deux éléments au moins soient identiques vaut :

$$\bar{p}(n) = \frac{365!}{(365 - n)!} \cdot \frac{1}{365^n}$$



$n$	$p(n)$
5	2,71 %
10	11,69 %
15	25,29 %
20	41,14 %
23	50,73 %
25	56,87 %
30	70,63 %
40	89,12 %
50	97,04 %
60	99,41 %

# Hash

## le paradoxe des anniversaires



Si une fonction de hachage a une sortie de  $n$  bits (**n grand**) alors l'ensemble d'arrivée possède  $2^n$  éléments et il faut **environ**  $2^{(n/2)}$  hachés d'éléments distincts pour produire une collision avec 50 % de chance.

$$n(p) \approx \sqrt{2 \cdot |E| \ln\left(\frac{1}{1-p}\right)}$$

$$p(n) = 1 - \frac{|E|!}{(|E|-n)!} \cdot \frac{1}{|E|^n}$$

```
for j in $(seq 1 100); do
    for i in $(seq 1 █); do
        empreinte=$(head -c 142 /dev/urandom | shasum5.18 -b)
        Bits8=$(echo $empreinte | head -c 2)
        echo $Bits8
        # echo $empreinte " " $(echo $empreinte | head -c2)
        # echo $(echo $empreinte | head -c 2)
    done | sort | uniq -c | grep -v '1 ' | head -n 1
    # echo $j
done | wc -l
```

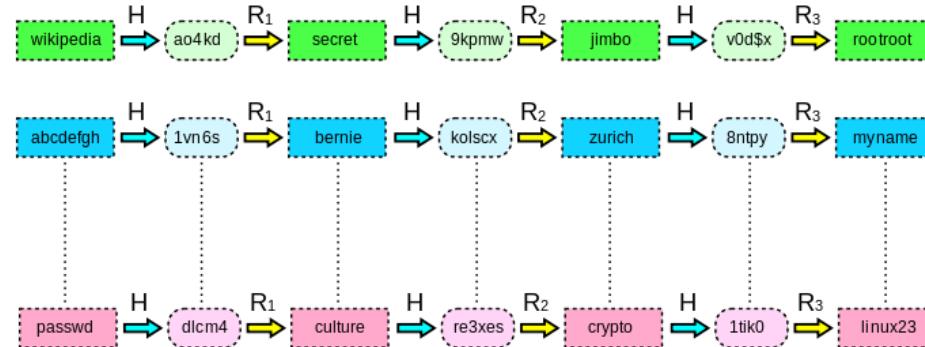
4b5171fcc7dcb79851a0471bf65bc012	4b
581bff7d931ac867fb7e1ed5c2d303c7	58
643504d90e1236b6f63feffe33ca4fe4	64
7a940a030cfb822565abd2d93f30369c	7a
7defce06508bc9e3a227e1267bc91d3b	7d
80eef2c533d0efc1d20d3b498d7aec8a	80
a0286479a5d60986fcac43d3d863b21e	a0
ab2d4b8df2e6ade91fc39bdbe1a6a22a	ab
c617bdee475a7221864ec7535aa46b9f	c6
f12f395b4c4a3c3d0d43b6224e875aeb	f1
fb7c125c99ee3f897a23b95081b18f9a	fb

# Hash

## compromis temps-mémoire



- Rainbow Tables
  - génération longue délicate



```
empreinte = h(mot_de_passe + sel)
```

# Rappel : Signature numérique



## Signature manuscrite

- atteste de l'approbation du contenu d'un document par le signataire
- vérifiable à l'aide d'une signature de référence
- difficile à imiter sur un autre document (**forge**)
- non-répudiable : le signataire ne peut nier avoir signé le document
- transférable : Bob peut convaincre un juge qu'Alice a bien signé un document portant sa signature

## Signature numérique *on souhaite conserver les mêmes propriétés*

- approbation
- vérifiable
- non forgeable
- non répudiable
- transférable

# Signature numérique

## Crypto Asymétrique



- Fonctions à sens unique
  - à trappe (RSA)
  - ou pas (DSA, ECDSA)
- Bi-clés
  - une privée, connue du seul signataire
  - une publique, connue de tous

# Signature

## Cryptosystème RSA



$$C \equiv M^e \pmod{n}$$

$$M \equiv C^d \pmod{n}$$



p et q premiers

$$n=p \cdot q$$

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

e premier avec  $\varphi(n)$

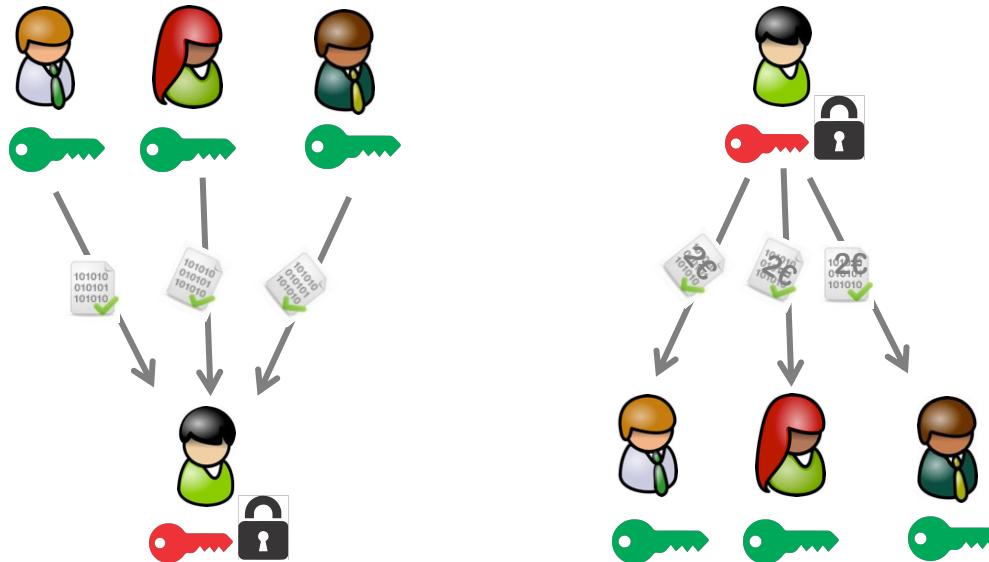
d inverse de e modulo  $\varphi(n)$

- Comme  $c=m^e \pmod{n}$ ,  $c^d \pmod{n} = m^{ed} \pmod{n}$
- Comme  $ed = 1 \pmod{(p-1)(q-1)}$  il existe un entier k tel que  $ed = 1 + k(p-1)(q-1)$
- Par conséquent  $c^d \pmod{n} = m^{1+k(p-1)(q-1)} \pmod{n}$
- Or (théorème de Fermat)  $m^{(p-1)} \pmod{p} = 1$  si m n'est pas multiple de p. Par élévation à la puissance  $k(q-1)$  puis multiplication par m on obtient :  $m^{1+k(p-1)(q-1)} \pmod{p} = m$  égalité qui reste vraie (2 membres=0) si m est multiple de p
- Par symétrie  $m^{1+k(p-1)(q-1)} \pmod{q} = m$  donc  $m^{1+k(p-1)(q-1)} - m$  est divisible par p et q donc par pq (p et q premiers et différents)
- Par conséquent  $c^d \pmod{n} = m^{1+k(p-1)(q-1)} \pmod{pq} = m$

# Crypto asymétrique

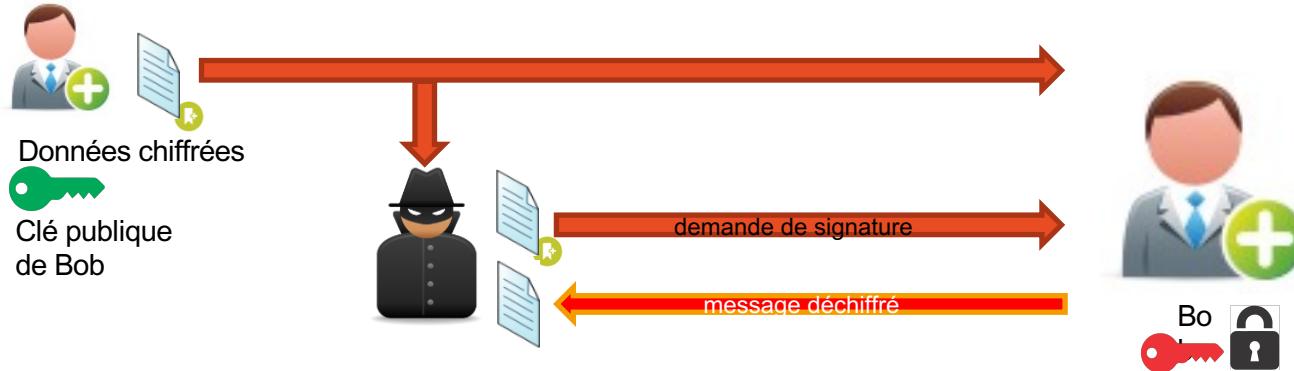


Usage : chiffrement vs signature



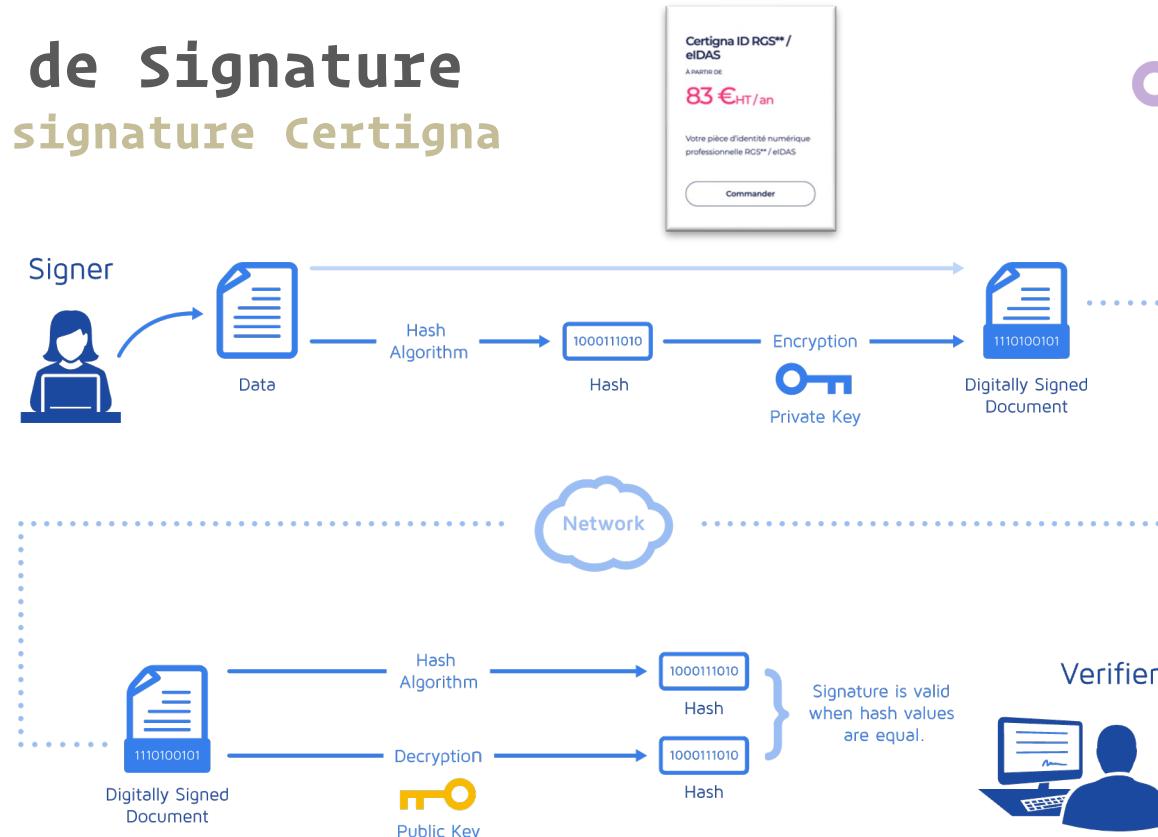
# Rappels

## Certificat – Séparation des Key Usage

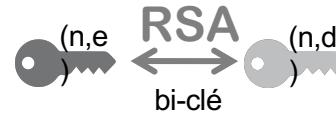


Cryptographie : utilisation de sa clé privée  
déchiffrement = signature

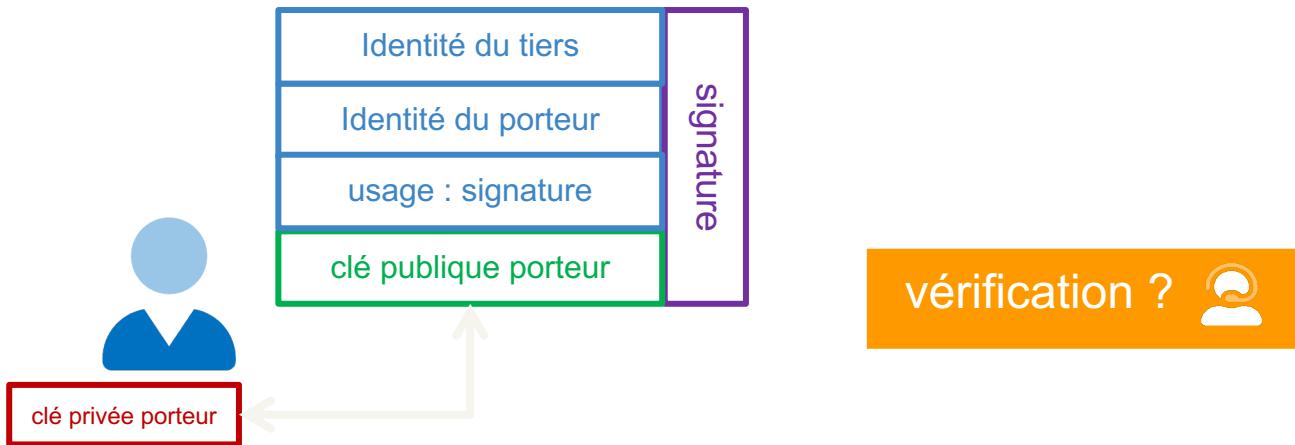
# Exemple de Signature bi-clé de signature Certigna



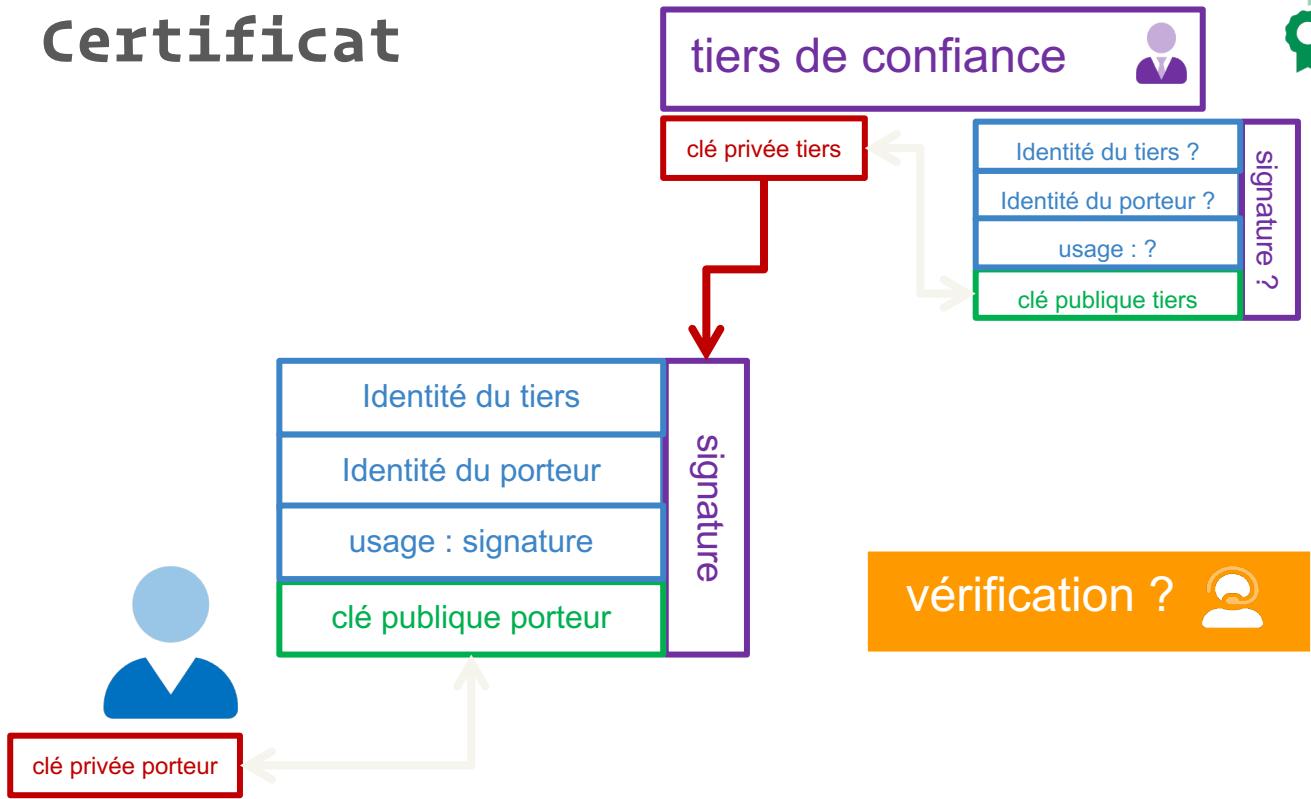
# Certificat



# Certificat

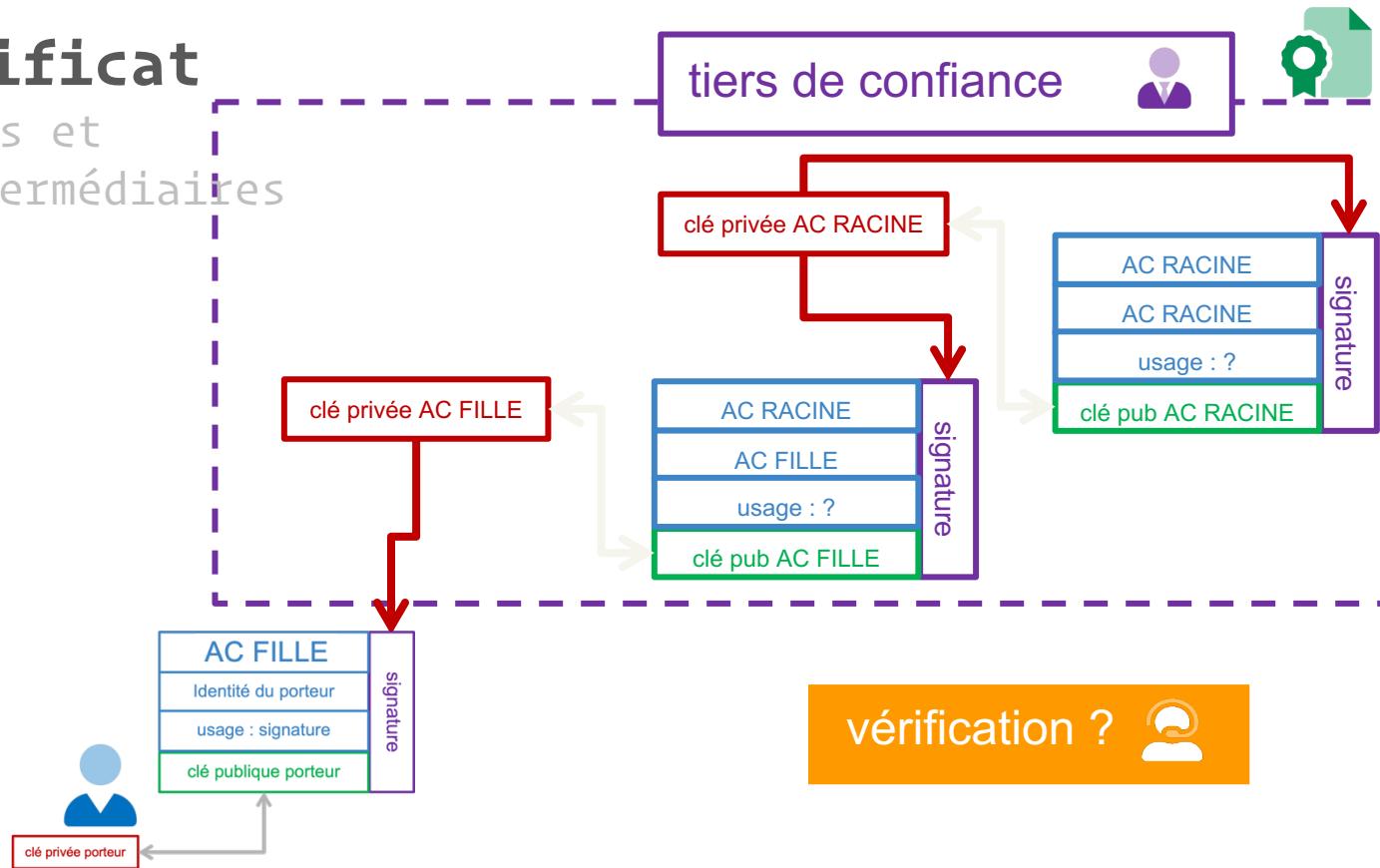


# Certificat



# Certificat

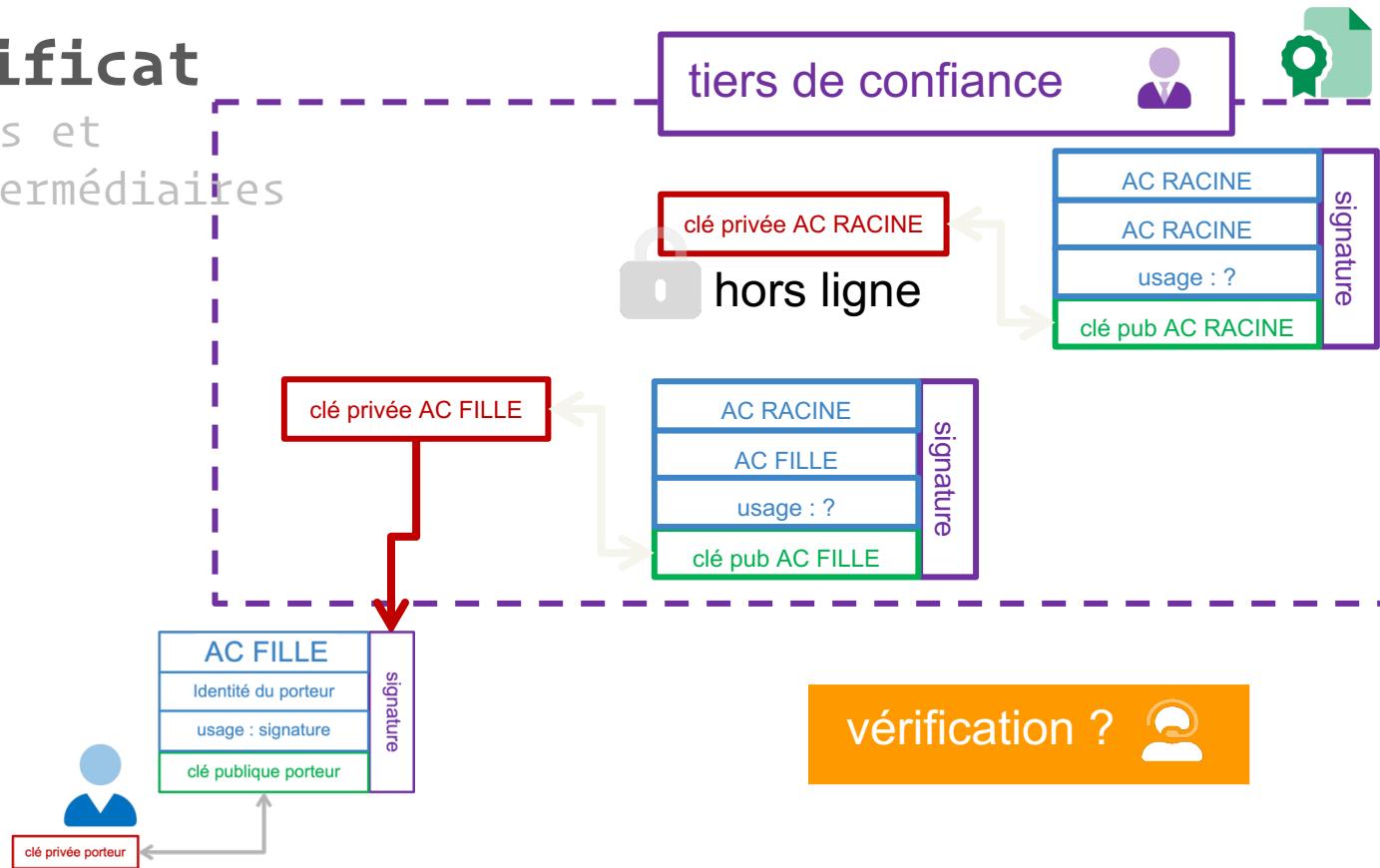
Racines et  
AC Intermédiaires



vérification ?

# Certificat

Racines et  
AC Intermédiaires



# Certificat

## Gabarits des certificats X509 v3



```
$ openssl x509 -in pierre_dupond.crt -noout -text  
Certificate:
```

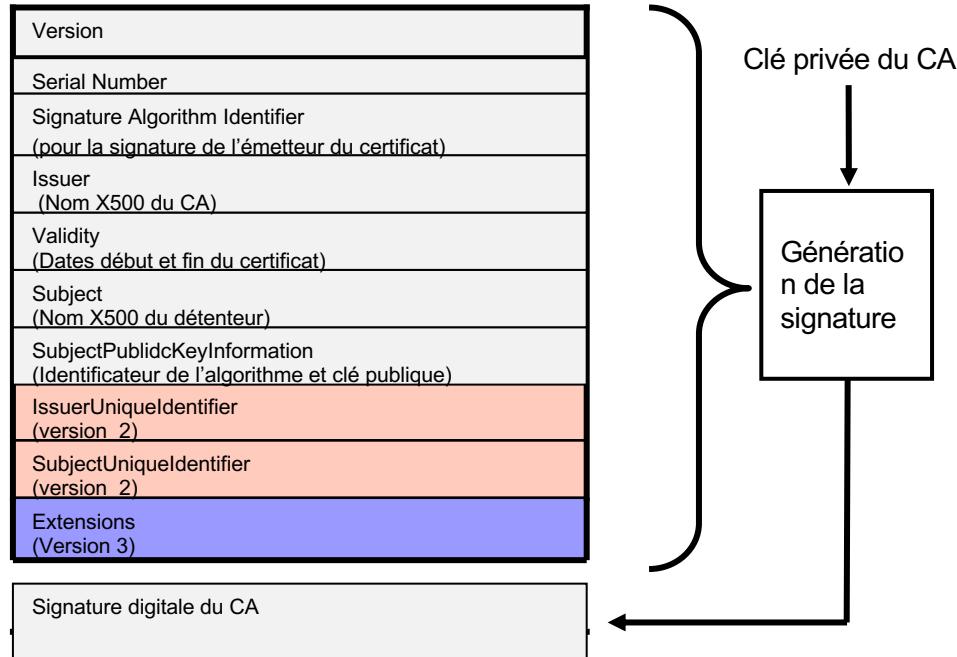
```
Data:  
    Version: 3 (0x2)  
    Serial Number: 1 (0x1)  
    Signature Algorithm: sha256WithRSAEncryption  
    Issuer: C=FR, O=EPITA, OU=0002 12345678912345,  
            CN=Cours SigElec  
    Validity  
        Not Before: Nov 24 17:48:27 2009 GMT  
        Not After : Nov 23 17:48:27 2014 GMT  
    Subject: CN=Pierre Dupond  
    Subject Public Key Info:
```

```
        Public Key Algorithm: rsaEncryption  
        Public-Key: (2048 bit)  
        Modulus:  
            00:d1:ea:2a:f8:b1:c6:86:fc:2c:0c:ed:c1:d4:0d:  
            49:9c:bb:2b:3d:ce:58:84:ae:30:59:86:18:05:2b:  
            f8:83:d6:bf:c0:ee:0d:5f:cb:1c:a0:9b:73:2:cea:  
            67:9b:f6:62:d4:07:33:a5:c4:60:3a:0f:73:85:44:  
            98:75:c3:1d:6c:9e:fe:03:99:38:88:12:56:d8:eb:  
            67:05:43:ae:3:09:38:cc:9e:14:d5:a9:62:88:15:  
            18:27:f8:8b:5d:ef:ac:cf:fb:ab:04:9b:eb:b4:  
            27:0c:ie:7:74:7:7c:f9:46:6a:af:c1:7a:92:93:67:  
            b5:3e:7a:c1:c7:27:a4:47:7b:0a:97:4c:49:c8:51:  
            de:91:ce:c3:28:21:b3:d5:d2:d8:bd:38:96:e0:98:  
            b4:ae:7f:72:56:a6:70:b3:71:fc:f7:e4:bd:6e:aa:  
            ed:21:6a:b5:f2:b0:e2:94:54:44:0e:a6:80:30:af:  
            15:9e:61:ae:47:cd:a9:cf:e8:7d:c7:09:fe:98:lc:  
            22:a3:db:38:be:5b:66:dc:c3:52:74:9a:c8:89:de:  
            44:3c:40:59:aa:0f:00:a0:09:8c:b3:f5:37:b4:76:  
            4e:43:d1:99:24:3e:b5:6c:69:c4:1f:eb:b6:6e:2f:  
            1d:5d:fb:66:f7:77:d4:16:ff:1b:al:83:9a:ba:e6:  
            1b:79
```

```
Exponent: 65537 (0x10001)  
X509v3 extensions:  
    X509v3 Basic Constraints: critical  
        CA:FALSE  
    X509v3 Authority Key Identifier:  
  
keyid:0C:88:C2:D1:10:E6:72:D0:7C:63:30:4A:E8:8D:3D:D6:9D:FB:BD  
:9C  
    DirName:/C=FR/O=EPITA/OU=0002  
12345678912345/CN=Cours SigElec  
    serial:8A:5D:41:8A:CA:49:B3:39  
  
X509v3 Subject Key Identifier:  
  
76:97:32:8F:65:62:33:8A:EA:8E:E3:C4:E5:2A:85:73:7E:7A:78:93  
X509v3 Key Usage:  
    Non Repudiation  
Signature Algorithm: sha256WithRSAEncryption  
    1c:80:dc:93:50:24:04:5:dd:c9:6f:95:3d:78:4c:0f:5c:8e:  
    79:ef:d9:f8:32:35:3f:f3:da:2f:ae:35:4d:c0:1b:17:f0:6a:  
    3b:31:14:26:46:a3:61:ed:c4:dd:77:98:86:93:2d:65:78:e3:  
    6d:21:70:23:b0:d3:ce:e7:88:6d:83:ea:85:d6:d8:cf:77:54:  
    6f:78:ee:9a:e9:db:4c:cd:3f:1f:20:b5:2f:bd:43:cd:22:fc:  
    41:fd:52:ab:4b:a4:16:57:61:95:52:8b:9b:e2:69:c2:b8:ec:  
    8f:da:2e:5b:ed:f4:d3:0:a:23:4e:07:ff:db:7:25:dd:38:12:  
    30:d6:3c:9f:9e:e5:bc:99:8f:bc:df:ba:b0:d9:a0:82:05:a2:  
    2b:b6:39:2c:7e:20:4b:ib6:a7:bl:ae:ce:cf:06:ab:62:c9:b0:  
    98:62:0d:94:b5:b9:d1:62:01:a4:4f:56:63:c1:89:67:e4:f8:  
    85:2d:c7:6a:5f:b2:a1:3c:61:2a:b2:6c:2b:92:f3:d6:62:ac:  
    69:84:3d:73:ef:ce:da:0b:a6:92:1d:2d:b5:60:04:59:b2:51:  
    9b:5e:69:24:f5:91:29:b4:06:e2:19:7d:0c:12:b0:87:cc:41:  
    84:36:7b:e1:df:bc:e4:29:9e:2d:ib8:b3:70:74:66:f7:3d:a6:  
    50:6a:0b:4c
```

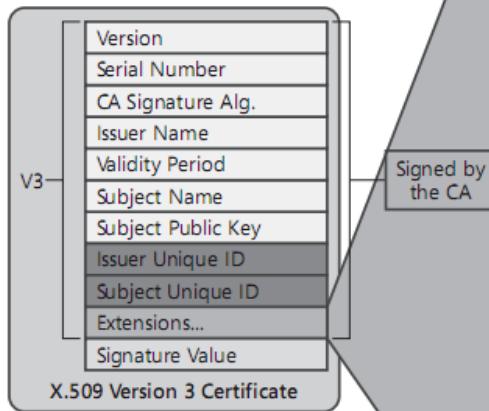
# Certificat

## Gabarits des certificats X509 v3



# Certificat

## Gabarits des certificats X509 v3



x.509v3 Standard Extensions	
Type	Critical
AuthorityKeyIdentifier	No
SubjectKeyIdentifier	No
KeyUsage	Should Be
PrivateKeyUsagePeriod	No
CertificatePolicies	No
PolicyMappings	No
SubjectAlternativeName	See RFC 3280
IssuerAlternativeName	See RFC 3280
SubjectDirAttribute	No
BasicConstraints	Yes
NameConstraints	Yes
PolicyConstraints	Maybe
ExtendedKeyUsage	Maybe
ApplicationPolicies	No
AuthorityInfoAccess	No
CRLDistributionPoint	No

# Certificat

## Key Usages (RFC 5280)



```
KeyUsage ::= BIT STRING {
    digitalSignature           (0),
    nonRepudiation             (1), -- recent editions of X.509 have
                                    -- renamed this bit to contentCommitment
    keyEncipherment            (2),
    dataEncipherment           (3),
    keyAgreement               (4),
    keyCertSign                (5),
    cRLSign                    (6),
    encipherOnly                (7),
    decipherOnly                (8) }

id-kp-serverAuth          OBJECT IDENTIFIER ::= { id-kp 1 }
id-kp-clientAuth           OBJECT IDENTIFIER ::= { id-kp 2 }
id-kp-codeSigning           OBJECT IDENTIFIER ::= { id-kp 3 }
id-kp-emailProtection       OBJECT IDENTIFIER ::= { id-kp 4 }
id-kp-timeStamping          OBJECT IDENTIFIER ::= { id-kp 8 }
id-kp-OCSPSigning           OBJECT IDENTIFIER ::= { id-kp 9 }
```

# Certificat

## Gabarits des certificats X509 v3



Nom de l'émetteur \_\_\_\_\_

Organisation DIRECTION GENERALE DES IMPOTS

Nom AC SERVICES INDIVIDUELS IAS1 C

Numéro de série 02 06 45 B3 F5 63 DE 72 FB B1 15 CC 68 10 48 7A

Version 3

Algorithme de signature SHA-1 avec chiffrement RSA ( 1 2 840 113549 1 1 5 )

Paramètres aucun

Non valide avant samedi 7 février 2009 18:09:51 HEC

Non valide après mardi 7 février 2012 18:09:51 HEC

Infos de clé publique \_\_\_\_\_

Algorithme Chiffrement RSA ( 1 2 840 113549 1 1 1 )

Paramètres aucun

Clé publique 256 octets : C9 73 CB 76 B8 8A DF E6 ... ⓘ

Exposant 65537

Dimension de la clé 2048 bits

Utilisation de la clé Vérification

Signature 128 octets : 06 63 F3 08 9C E4 6B D7 ... ⓘ

Authentification (Vérification de signature)

Extension Utilisation de la clé ( 2 5 29 15 )

Critique NON

Utilisation Signature numérique, Non répudiation

Signature électronique

Extension Contraintes élémentaires ( 2 5 29 19 )

Critique OUI

Autorité de certification NON

# IGC - PKI

## IGC - Demande de certificat



# IGC - PKI

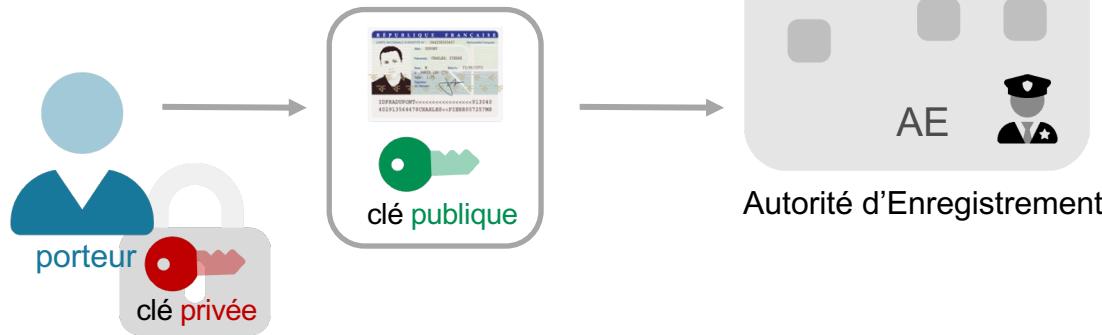
## IGC - Demande de certificat



génération du bi-clés

# IGC - PKI

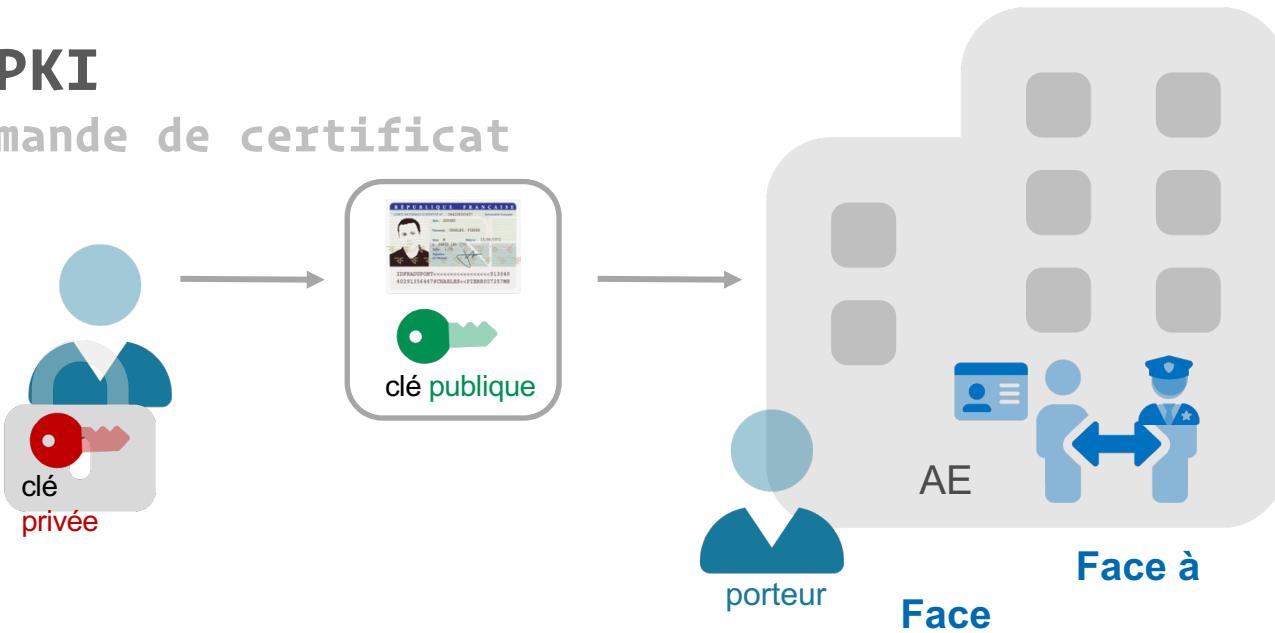
## IGC - Demande de certificat



1. Sécurisation de la clé privée
2. Envoi de la clé publique et des informations d'identité

# IGC - PKI

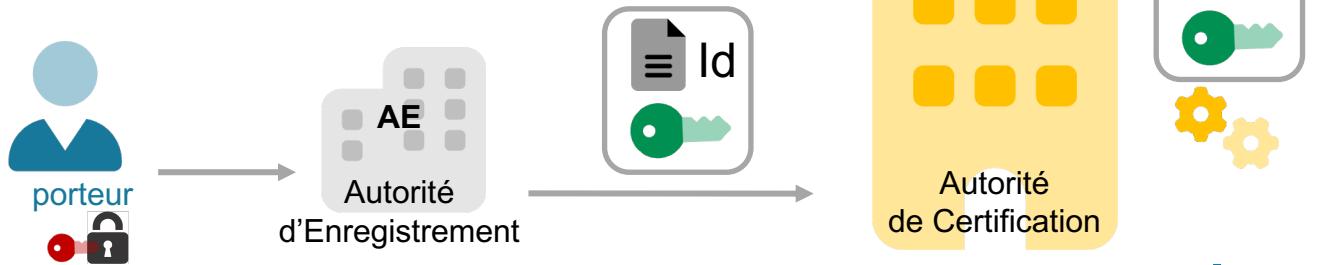
## IGC - Demande de certificat



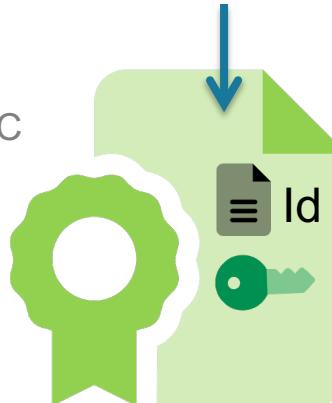
3. Vérification des informations d'identité du porteur par l'Autorité d'Enregistrement

# IGC - PKI

## IGC - Demande de certificat

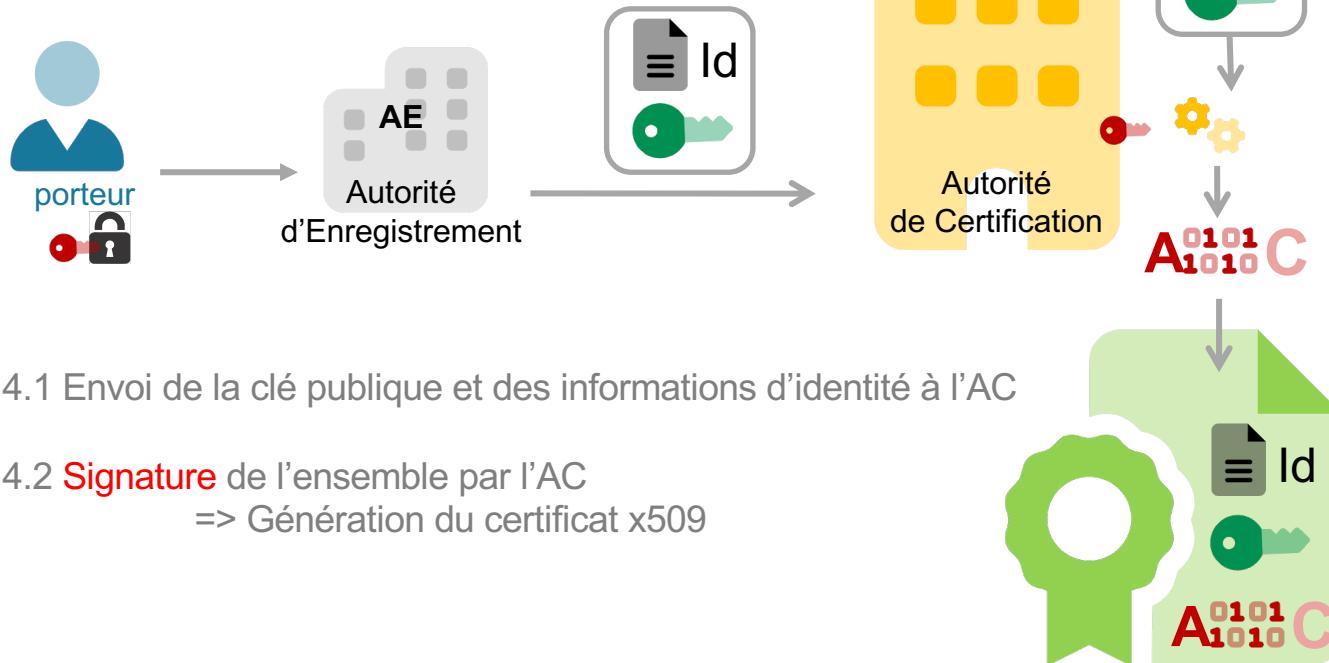


4.1 Envoi de la clé publique et des informations d'identité à l'AC



# IGC - PKI

## IGC - Demande de certificat

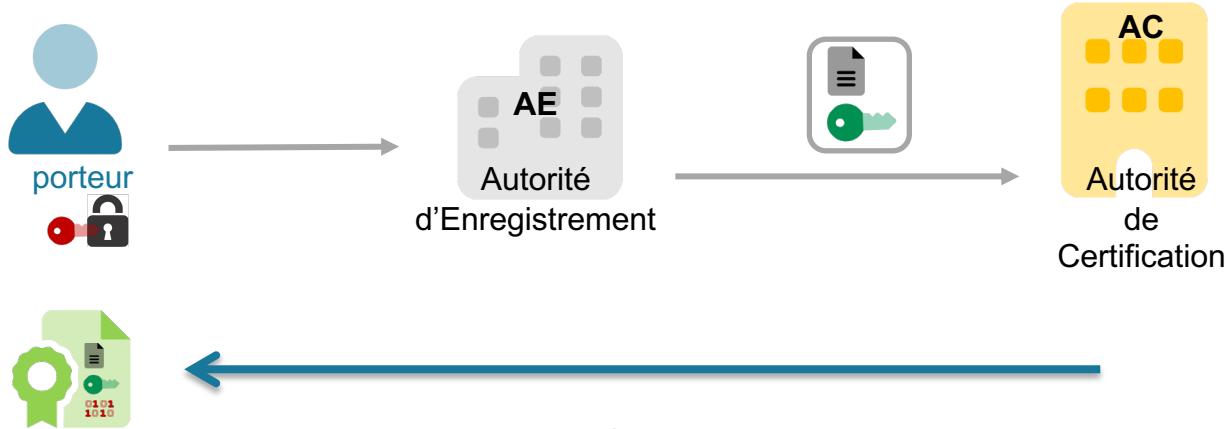


4.1 Envoi de la clé publique et des informations d'identité à l'AC

4.2 **Signature** de l'ensemble par l'AC  
=> Génération du certificat x509

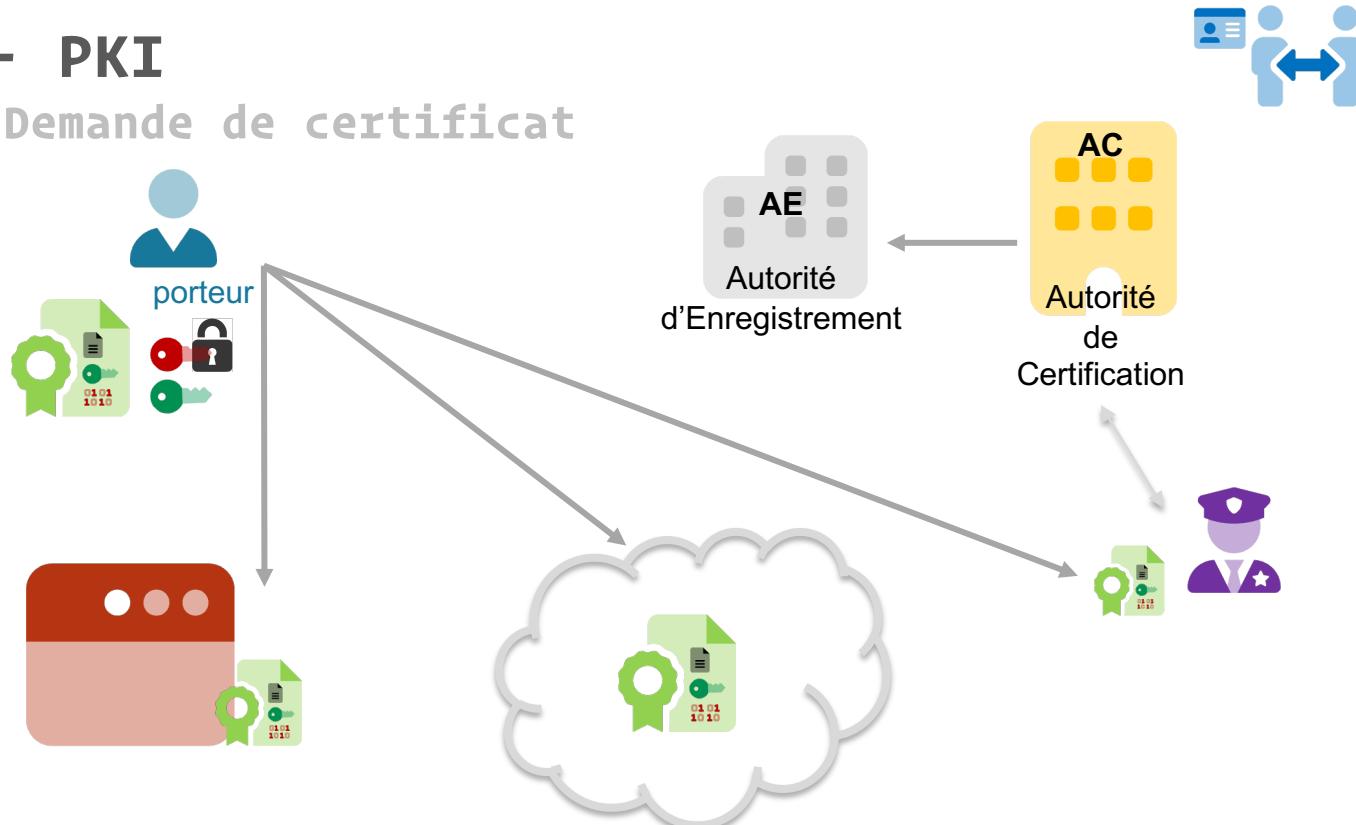
# IGC - PKI

## IGC - Demande de certificat



# IGC - PKI

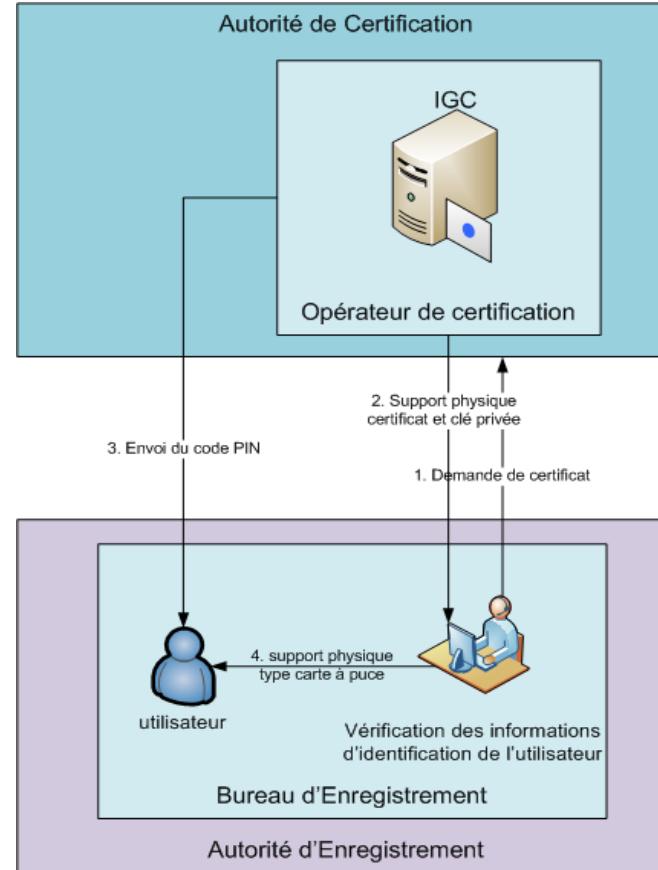
## IGC - Demande de certificat



## 6. Utilisation des éléments

# IGC - PKI

AC - OC - AE



# Question ?

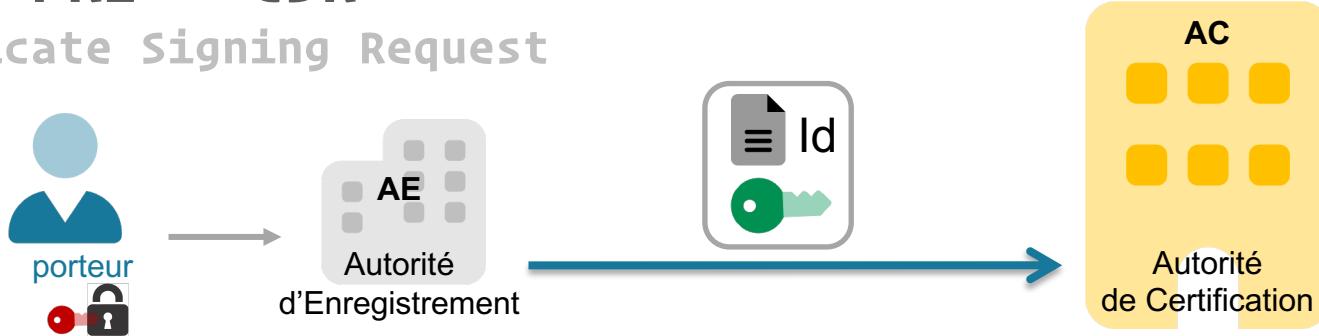


Une Autorité de Certification émettant des certificats qualifiés doit :

- Garantir que les clés de signature privées de l'AC stockées par le matériel cryptographique sont détruites lorsque que le dispositif n'est plus utilisé
- Vérifier par des moyens appropriés conformes au droit national l'identité de la personne à qui est délivré un certificat qualifié
- Conserver les informations du porteur aussi longtemps que nécessaire pour faire la preuve de la certification en justice
- Toutes ces réponses

# IGC - PKI - CSR

## Certificate Signing Request

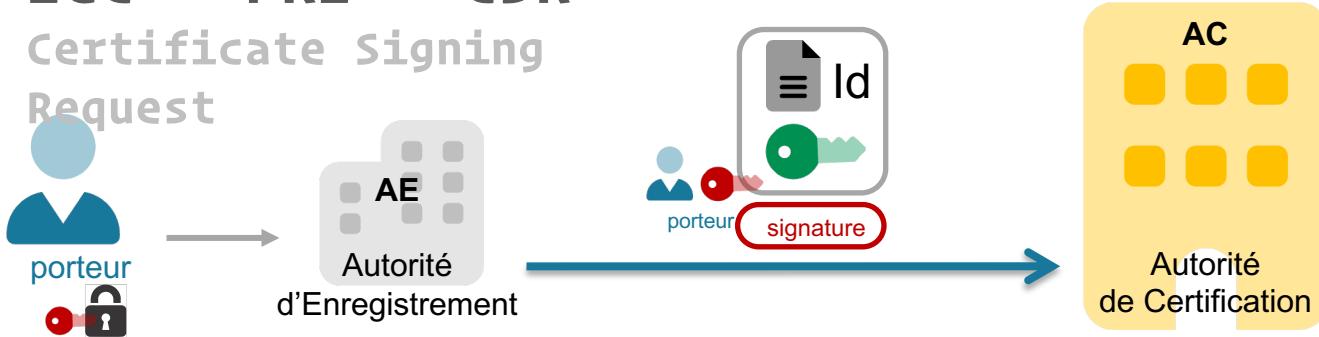


Comment assurer la preuve de **possession** de la **clé privée** du porteur ?

- cela permet d'assurer le principe de **non répudiation** de la signature
- très simple avec une seule requête

# IGC - PKI - CSR

## Certificate Signing Request



Comment assurer la preuve de **possession** de la **clé privée** du porteur ?

=> CSR est la spécification **PKCS#10 v1.7 - RFC 2986**

# **IGC - PKI -**

## **Horodatage**

### **Question**

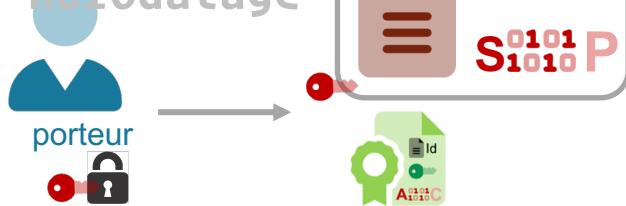
41. En cas de compromission de sa clé privée, peut-on répudier une signature électronique qui n'a pas d'horodatage et n'a pas encore été vérifiée ?

- A. Oui, dans tous les cas
- B. Oui, uniquement pour les signatures créées après la révocation de la clé privée
- C. Non, jamais
- D. Non, sauf si le format de signature est XAdES-EPES

# IGC - PKI -

## Horodatage

### Horodatage



# IGC - PKI

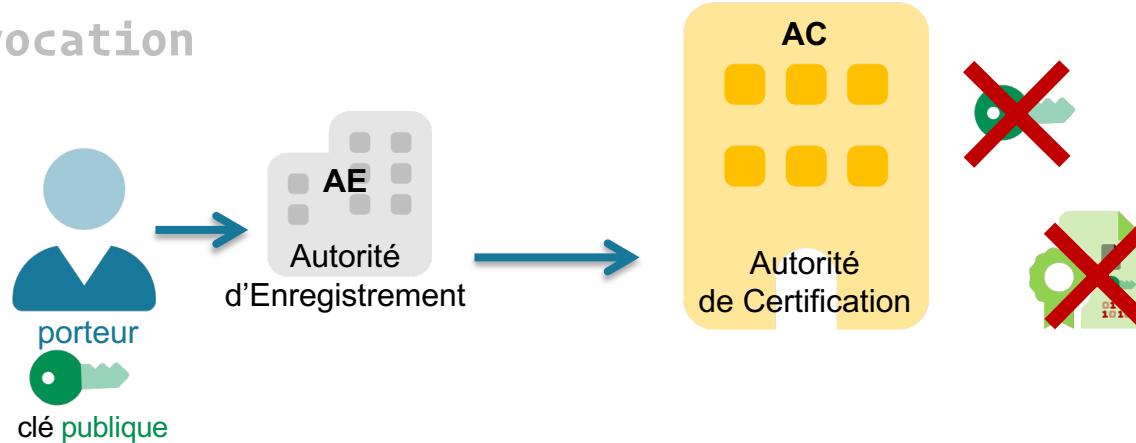
## IGC - Révocation



- Porteur demandeur d'une révocation
- Compromission et/ou perte de la clé privée

# IGC - PKI

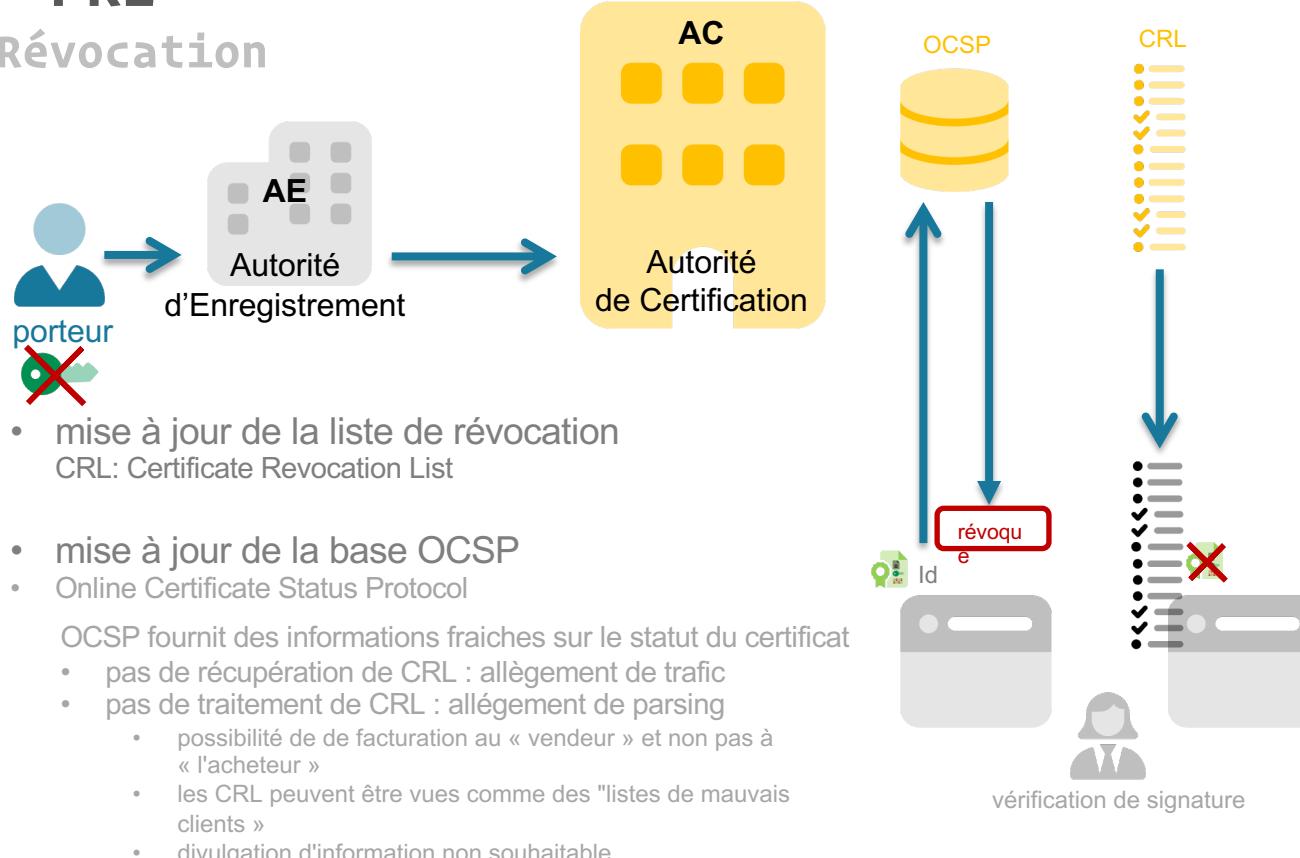
## IGC - Révocation



- Demande de révocation par le porteur
- Validation de la demande de révocation par l'AE
- Révocation de la clé publique par l'Autorité de Certification

# IGC - PKI

## IGC - Révocation



# La signature électronique sécurisée #4



F96DE8C227A259C87EE1DA2AED  
57C93FE5DA36ED4EC87EF2C63A  
AE5B9A7EFFD673BE4ACF7BE892  
3CAB1ECE7AF2DCF7AE29A3DA44  
F235A24C963FF0DF3CA3599A70  
E5DA36BF1ECE77F8DC34BE129A  
6CF4D126BF5B9A7CFEDF3EB850  
D37CF0C63AA2509A76FF9227A5  
5B9A6FE3D720A850D97AB1DD35  
ED5FCE6BF0D138A84CF8DC34BE  
129F8DC34B

# La signature électronique sécurisée sécurisée = avancée et/ou qualifiée



F96DE8C227A259C87EE1DA2AED  
57C93FE5DA36ED4EC87EF2C63A  
AE5B9A7EFFD673BE4ACF7BE892  
3CAB1ECE7AF2DCF7AE29A3DA44  
F235A24C963FF0DF3CA3599A70  
E5DA36BF1ECE77F8DC34BE129A  
6CF4D126BF5B9A7CFEDF3EB850  
D37CF0C63AA2509A76FF9227A5  
5B9A6FE3D720A850D97AB1DD35  
ED5FCE6BF0D138A84CF8DC34BE  
129F8DC34B

# Complexité

- la signature
  - Compréhension facile
  - Mise en œuvre facile





- la signature électronique
  - Compréhension difficile
  - Mise en œuvre délicate

7e0950bb938539162d268b379595  
44efbf87b718950bf4721dd5c94f5f7  
d12fc4efac9d9b5fc081bbc1555c3d7  
6610ef3080a354e60b625f5c50a23  
a6bfd13ec024239ddc0b47706c9a23  
11fc38e37161e87501236542732797  
2469b3985721cc0fea3b04047a9c5  
b559e3471a736f5e4c7b473b2e86b1  
b21dd8a829828d f8d6



- la signature électronique sécurisée
  - Compréhension difficile
  - Mise en œuvre très difficile

7e0950bb938539162d268b379595  
44efbf87b718950bf4721dd5c94f5f7  
d12fc4efac9d9b5fc081bbc1555c3d7  
6610ef3080a354e60b625f5c50a23  
a6bfd13ec024239ddc0b47706c9a23  
11fc38e37161e87501236542732797  
2469b3985721cc0fea3b04047a9c5  
b559e3471a736f5e4c7b473b2e86b1  
b21dd8a829828d f8d6



# Statut légal d'une signature électronique

## Décret n° 2017-1416 du 28 septembre 2017 relatif à la signature électronique

Le Premier ministre,

Sur le rapport de la garde des sceaux, ministre de la justice,

Vu le **règlement (UE) n° 910/2014 du Parlement européen (eIDAS)** et du Conseil du 23 juillet 2014 sur l'identification électronique et les services de confiance pour les transactions électroniques au sein du marché intérieur et abrogeant la directive 1999/93/CE ;

Vu l'article 1367 du code civil dans sa rédaction issue de l'article 4 de l'ordonnance n° 2016-131 du 10 février 2016 portant réforme du droit des contrats, du régime général et de la preuve des obligations ;

Le Conseil d'Etat (section de l'intérieur) entendu,

Décrète :

### Article 1

La fiabilité d'un procédé de signature électronique est présumée, jusqu'à preuve du contraire, lorsque ce procédé met en œuvre une signature électronique **qualifiée**.

Est une signature électronique **qualifiée** une signature électronique **avancée**, conforme à l'article 26 du règlement susvisé et créée à l'aide d'un dispositif de création de signature électronique **qualifié** répondant aux exigences de l'article 29 dudit règlement, qui repose sur un certificat **qualifié** de signature électronique répondant aux exigences de l'article 28 de ce règlement.

<https://www.legifrance.gouv.fr/loda/id/JORFTEXT000035676246>

# Règlement eIDAS

## Périmètre

Le Règlement « eIDAS » n° 910/2014 du 23 juillet 2014 a pour ambition d'accroître la confiance dans les transactions électroniques au sein du marché intérieur. Il établit un socle commun pour les interactions électroniques sécurisées entre les citoyens, les entreprises et les autorités publiques.

Le règlement eIDAS concerne principalement les organismes du secteur public et les prestataires de services de confiance établis sur le territoire de l'Union européenne. Il instaure un cadre européen en matière d'identification électronique et de services de confiance, afin de faciliter l'émergence du marché unique numérique. Il couvre notamment le sujet de la signature électronique, et abroge la directive 1999/93/CE. L'ANSSI est l'un des organismes nationaux chargés de la mise en œuvre de ce règlement.



[https://www.ssi.gouv.fr/entreprise/reglementation/  
confiance-numerique/le-reglement-eidas/](https://www.ssi.gouv.fr/entreprise/reglementation/confiance-numerique/le-reglement-eidas/)

# Règlement RGS

## Pour les Autorités Administratives

- RGS 2.0

« Règles auxquelles les systèmes d'information mis en place par les **autorités administratives** doivent se conformer pour assurer la sécurité des informations échangées, et notamment leur confidentialité et leur intégrité, ainsi que la disponibilité et l'intégrité de ces systèmes et l'identification de leurs utilisateurs ».



# Contexte légal

## RGS

- RGS 2.0

### Documents applicables concernant l'utilisation de certificats électroniques

 <b>RGS A1</b> PDF - 453.6 ko Règles relatives à la mise en oeuvre des fonctions de sécurité basées sur l'emploi de certificats électroniques, version 3.0	 <b>RGS A2</b> PDF - 1.3 Mo Politique de Certification Type « certificats électroniques de personne », version 3.0	 <b>RGS A3</b> PDF - 1.1 Mo Politique de Certification Type « services applicatifs », version 3.0	 <b>RGS A4</b> PDF - 458.8 ko Profils de certificats, CRL, OCSP et algorithmes cryptographiques, version 3.0	 <b>RGS A5</b> PDF - 740 ko Politique d'Horodatage Type, version 3.0
--	--	---	--	--

# Normalisation Européenne

CEN : Comité Européen de Normalisation – cen.eu  
ETSI : European Telecommunications Standards Institute -  
etsi.org

CWA : CEN Workshop Agreement  
TS : Technical Specification  
EN : European standard

- ETSI
  - EN 319 411-1 – AC non qualifiée
  - EN 319 411-2 – AC qualifiée
  - EN 319 122 – CAdES (CMS)
  - EN 319 132 – XAdES (XML)
  - EN 319 142 – PAdES (ISO-32000 / PDF)
- CEN
  - CWA 14167 : Trustworthy systems / PP des HSM
  - CWA 14169 : PP SSCD
  - CWA 14170 : Application de création de Signature électronique
  - CWA 14171 : Application de vérification de Signature électronique

# Entités et vocabulaire

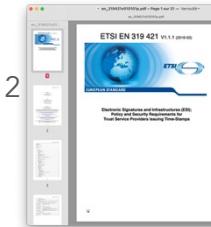
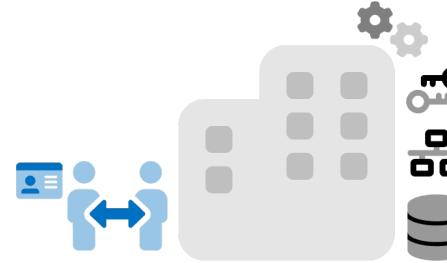
## les termes et leurs synonymes

- AE et AC
  - PSCe – PSCo – TSP
- HSM – SSCD – QSCD – Carte à puce

On entend par **SSCD** [Secure-Signature-Creation Device] ou **QSCD** [Qualified-Signature-Creation Device] un Dispositif Sécurisé de Création de Signature. Un SSCD correspond à une « carte à puce » contenant un crypto-système hardware sécurisé, ou encore à un **HSM** (Hardware Security Module), ou encore un « **Secure Element** » ou **TPM** (Trusted Platform Module) dans un smartphone ou sur une carte mère d'ordinateur.



- eIDAS - RGS 2.0
  - eIDAS : Délivrance de certificats qualifiés - Audit ETSI 319 401 / 411-1&2 / 412
  - RGS : Délivrance de certificats qualifiés - Audit RGS – Annexes A2 / A3 / etc.



# EU Trusted List

## 26 PSCE - PSCo - TSP en France - Services diverses

Trusted List France Trust service providers	
Currently active trust service providers	
Agence Nationale des Titres Sécurisés	<a href="#">QCert for ESig</a> <a href="#">QCert for ESeal</a> AR24 <a href="#">QeIDOS</a>
Caisse des dépôts et consignations	<a href="#">QCert for ESig</a> CEGEDIM SA <a href="#">QCert for ESig</a> <a href="#">QCert for ESeal</a>
CertEurope	<a href="#">QCert for ESig</a> <a href="#">QCert for ESeal</a> QWAC
Certigna	<a href="#">QCert for ESig</a> <a href="#">QCert for ESeal</a> QWAC <a href="#">QTimestamp</a>
Certinomis	<a href="#">QCert for ESig</a> <a href="#">QCert for ESeal</a> QWAC <a href="#">QTimestamp</a>
ChamberSign France	<a href="#">QCert for ESig</a> <a href="#">QCert for ESeal</a>
CLEARBUS	<a href="#">QTimestamp</a> <a href="#">QeIDOS</a> Conseil Supérieur du Notariat <a href="#">QCert for ESig</a> <a href="#">QTimestamp</a>
Cryptolog International	<a href="#">QCert for ESig</a> <a href="#">QCert for ESeal</a> QVal for QESig QPtes for QESig QVal for QESeal QPtes for QESeal <a href="#">QTimestamp</a>
DARVA	<a href="#">QTimestamp</a> <a href="#">QeIDOS</a>
Docaposte ARKHINEO	<a href="#">QVal for QESig</a> QPtes for QESig QVal for QESeal QPtes for QESeal
DOCUMENT CHANNEL	<a href="#">QeIDOS</a>
Docusign France	<a href="#">QCert for ESig</a> <a href="#">QCert for ESeal</a> <a href="#">QTimestamp</a>
Equisign	<a href="#">QeIDOS</a>
Gendarmerie Nationale	<a href="#">QCert for ESig</a> <a href="#">QCert for ESeal</a>
Imprimerie Nationale	<a href="#">QCert for ESig</a>
Le Groupe La Poste	<a href="#">QTimestamp</a> <a href="#">QeIDOS</a>
Lex Persona	<a href="#">QTimestamp</a>
Ministère de l'Intérieur	<a href="#">QCert for ESig</a> <a href="#">QTimestamp</a>
Ministère de la Justice	<a href="#">QCert for ESig</a>
Ministères économiques et financiers	<a href="#">QCert for ESig</a>
TESSI DOCUMENTS SERVICES	<a href="#">QeIDOS</a>
VIALINK	<a href="#">QCert for ESig</a> <a href="#">QCert for ESeal</a>
Worldline France	<a href="#">QTimestamp</a>
Yousign	<a href="#">QCert for ESig</a> <a href="#">QCert for ESeal</a> <a href="#">QTimestamp</a>

# Audit PSCo/TSP eIDAS - RGS



# eIDAS

## Evaluation de la conformité

The image displays a collection of ETSI EN (European Standard) documents and a snippet from a computer security standard, likely related to eIDAS evaluation.

**Top Row:**

- en\_319401v02301p.pdf – Page 1 sur 23 (ETSI EN 319 401 V2.3.1 (2021-05))
- en\_319 411-2 v222.pdf – Page 1 sur 31 (ETSI EN 319 411-2 V2.2.2 (2018-04))

**Middle Row:**

- en\_319 411-1 v122.pdf – Page 1 sur 52 (ETSI EN 319 411-1 V1.2.2 (2018-04))
- en\_319421v01010p.pdf – Page 1 sur 31 – Verrouillé (ETSI EN 319 421 V1.1.1 (2016-03))

**Bottom Left:**

en 319 411-1 v122.pdf – Page 40 sur 52

40

### 6.5.5 Computer security controls

**OVRL-6.5.5-01:** The requirements REQ-7.4-01, REQ-7.4-02, REQ-7.4-03, and REQ-7.4-04 shall apply.

**NOTE:** Requirements for the trustworthy systems can be ensured by applying the requirements of CEN TS 419 261 [1.9] or to a suitable protection profile such as ISO/IEC 15408 [1].

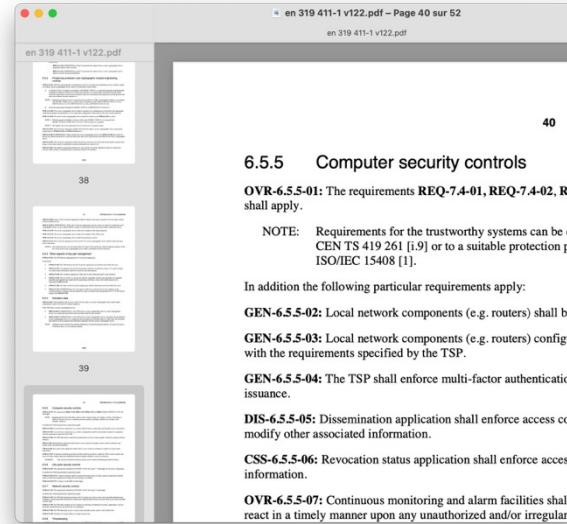
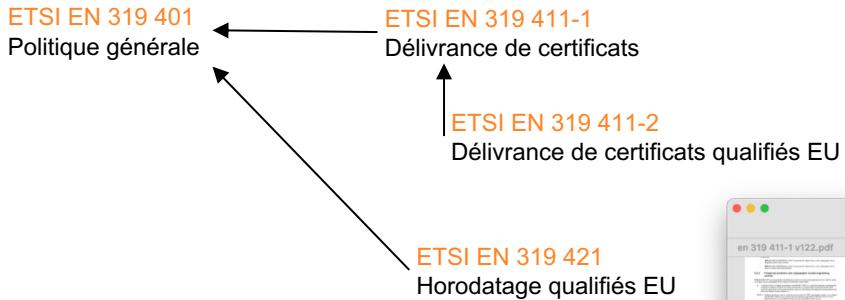
In addition the following particular requirements apply:

- GEN-6.5.5-02:** Local network components (e.g. routers) shall be kept up-to-date.
- GEN-6.5.5-03:** Local network components (e.g. routers) configured by the TSP shall be kept up-to-date according to the requirements specified by the TSP.
- GEN-6.5.5-04:** The TSP shall enforce multi-factor authentication for certificate issuance.
- DIS-6.5.5-05:** Dissemination application shall enforce access control and shall not modify other associated information.
- CSS-6.5.5-06:** Revocation status application shall enforce access control and shall not modify other associated information.

**OVRL-6.5.5-07:** Continuous monitoring and alarm facilities shall be available and shall react in a timely manner upon any unauthorized and/or irregular activity.

**Bottom Right:**

Services de délivrance des certificats qualifiés de signature électronique, de cachet électronique et



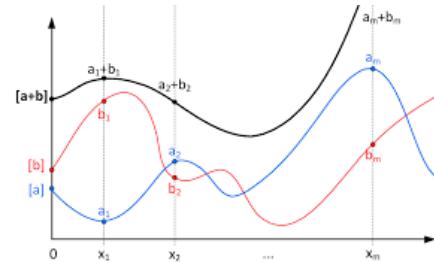
# KC : Key Ceremony



# Audit des parts de secrets



Shamir's Secret Sharing ( $m$ -of- $n$ )



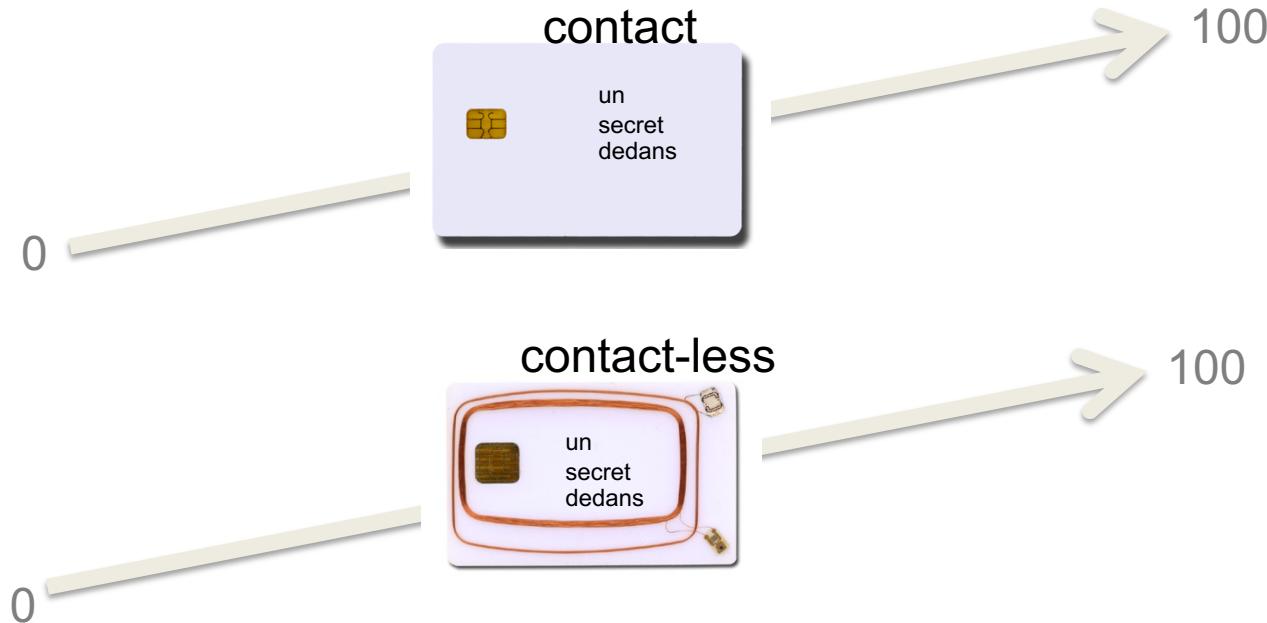
# le dernier sanctuaire des petits secrets et autres clés privées ?



MultiApp ID IAS ECC Combi complies with the following international and European standards:

Java Card 2.2.1  
Global Platform 2.1.1  
ISO 7816 parts 1, 2, 3, 4, 5, 6, 8 & 9  
ISO14443 type-A and type B  
CEN TS 15480 part 1 and 2  
E-SingK EN 14890 part 1 and 2  
ICAO EAC V1.11  
ICAO Doc 9303 Sixth Edition  
ICAO Machine Readable Travel Document ?  
RF Protocol and Application Test Standard for e-Passport.  
Pre-loaded applets in ROM  
IAS ECC applet  
ICAO applet  
One time password applet  
Mifare emulation upon request.  
Security  
MultiApp ID IAS ECC Combi includes multiple hardware and software countermeasure against various public & non-public attacks as:  
Side channel attacks (SPA, DPA, Timing attacks etc)  
Invasive attacks  
Advanced fault attacks.

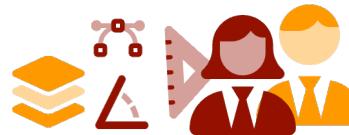
# Attention : « c'est (pas) sécurisé » ne veut rien dire



# Voilà ... c'est fini !

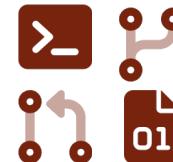
## Ingénieur(e)s

- scientifiques
- design (capacité à concevoir)



## Code

- préhistoire
- tout à inventer



## Stage

- pas très grave
- management



1<sup>ère</sup> page du rapport le 1<sup>er</sup> jour + répétitions



Stelau  
Hack different.

# Entités et vocabulaire

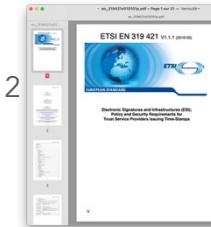
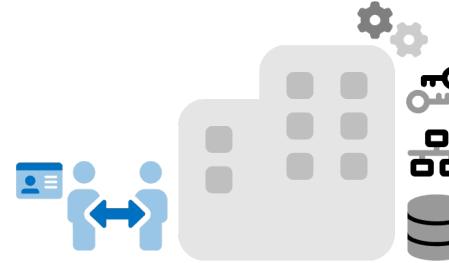
## les termes et leurs synonymes

- AE et AC
  - PSCe – PSCo – TSP
- HSM – SSCD – QSCD – Carte à puce

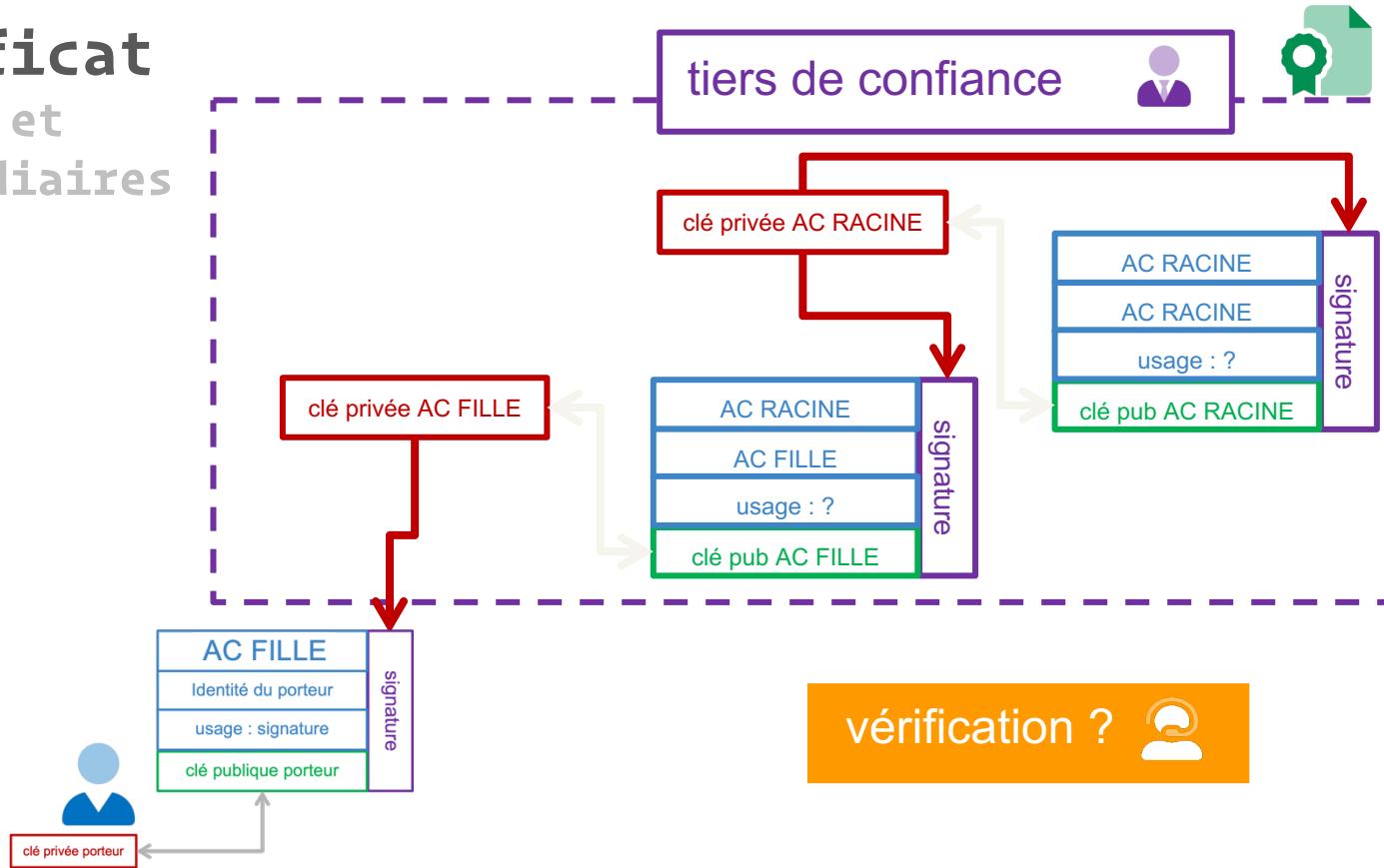
On entend par **SSCD** [Secure-Signature-Creation Device] ou **QSCD** [Qualified-Signature-Creation Device] un Dispositif Sécurisé de Création de Signature. Un SSCD correspond à une « carte à puce » contenant un crypto-système hardware sécurisé, ou encore à un **HSM** (Hardware Security Module), ou encore un « **Secure Element** » ou **TPM** (Trusted Platform Module) dans un smartphone ou sur une carte mère d'ordinateur.



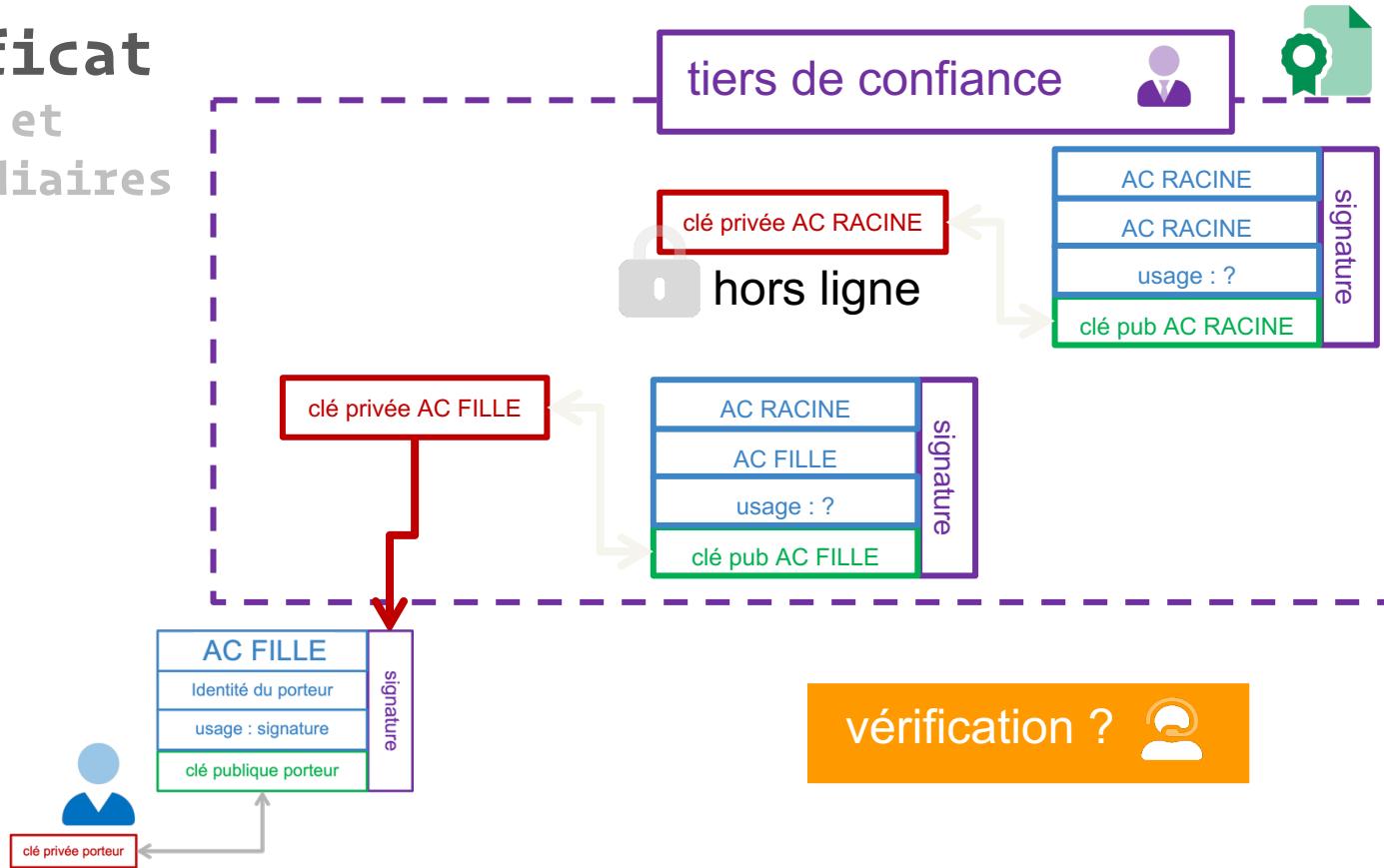
- eIDAS - RGS 2.0
  - eIDAS : Délivrance de certificats qualifiés - Audit ETSI 319 401 / 411-1&2 / 412
  - RGS : Délivrance de certificats qualifiés - Audit RGS – Annexes A2 / A3 / etc.



# Certificat Racines et Intermédiaires

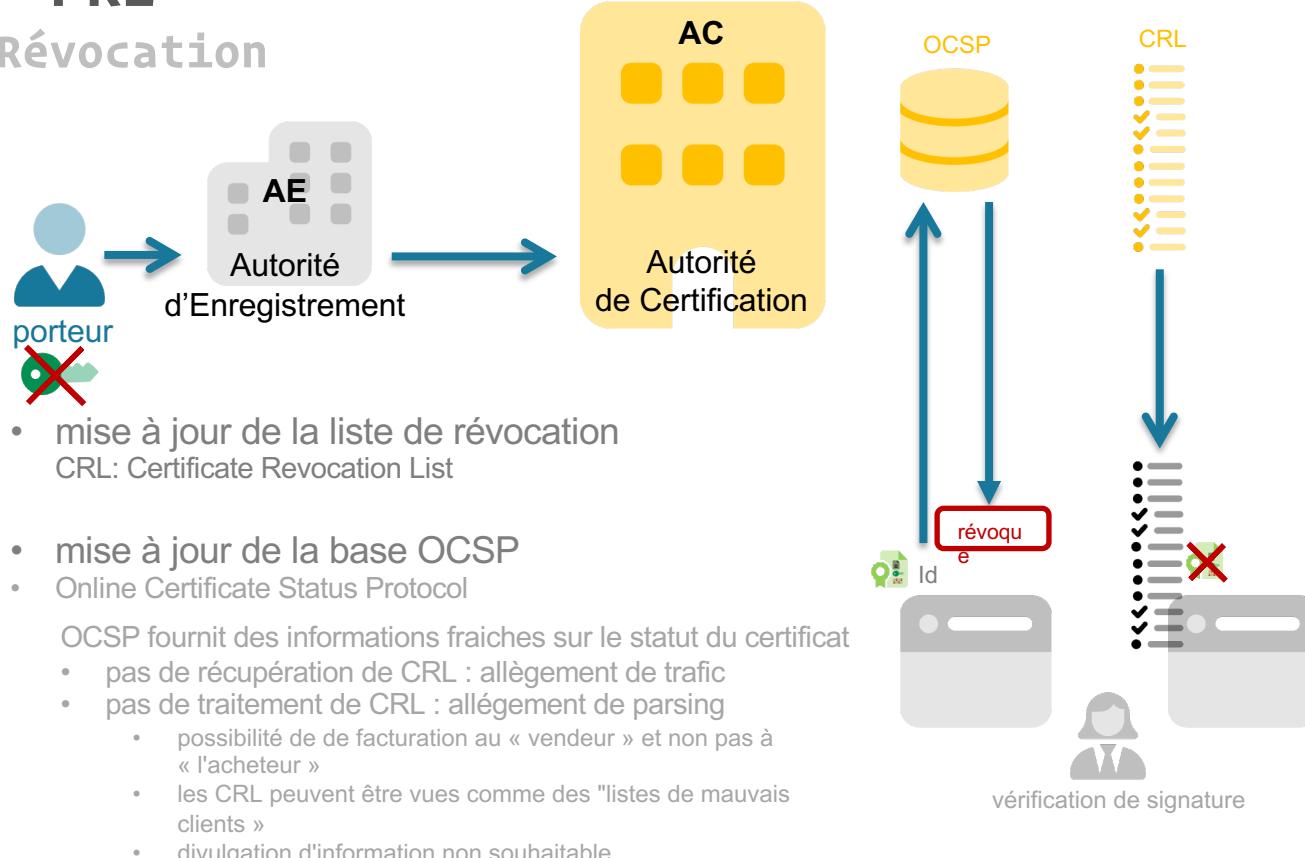


# Certificat Racines et Intermédiaires



# IGC - PKI

## IGC - Révocation



# Règlement eIDAS

## signature

Article 25

### Effets juridiques des signatures électroniques

1. L'effet juridique et la recevabilité d'une signature électronique comme preuve en justice ne peuvent être refusés au seul motif que cette signature se présente sous une forme électronique ou qu'elle ne satisfait pas aux exigences de la signature électronique qualifiée.
2. L'effet juridique d'une signature électronique qualifiée est équivalent à celui d'une signature manuscrite.
3. Une signature électronique qualifiée qui repose sur un certificat qualifié délivré dans un État membre est reconnue en tant que signature électronique qualifiée dans tous les autres États membres.

Article 26

### Exigences relatives à une signature électronique avancée

Une signature électronique avancée satisfait aux exigences suivantes:

- a) être liée au signataire de manière univoque;
- b) permettre d'identifier le signataire;
- c) avoir été créée à l'aide de données de création de signature électronique que le signataire peut, avec un niveau de confiance élevé, utiliser sous son contrôle exclusif; et
- d) être liée aux données associées à cette signature de telle sorte que toute modification ultérieure des données soit détectable.

# Règlement eIDAS à distance

## Article 24

### Exigences applicables aux prestataires de services de confiance qualifiés

1. Lorsqu'un prestataire de services de confiance qualifié délivre un certificat qualifié pour un service de confiance, il vérifie, par des moyens appropriés et conformément au droit national, l'identité et, le cas échéant, tous les attributs spécifiques de la personne physique ou morale à laquelle il délivre le certificat qualifié.

Les informations visées au premier alinéa sont vérifiées par le prestataire de services de confiance qualifié directement ou en ayant recours à un tiers conformément au droit national:

- a) par la présence en personne de la personne physique ou du représentant autorisé de la personne morale; ou
- b) à distance, à l'aide de moyens d'identification électronique pour lesquels, avant la délivrance du certificat qualifié, la personne physique ou un représentant autorisé de la personne morale s'est présenté en personne et qui satisfont aux exigences énoncées à l'article 8 en ce qui concerne les niveaux de garantie substantiel et élevé; ou
- c) au moyen d'un certificat de signature électronique qualifié ou d'un cachet électronique qualifié délivré conformément au point a) ou b); ou
- d) à l'aide d'autres méthodes d'identification reconnues au niveau national qui fournissent une garantie équivalente en termes de fiabilité à la présence en personne. La garantie équivalente est confirmée par un organisme d'évaluation de la conformité.

# Règlement eIDAS

## remote signing

- (52) La **création de signatures électroniques à distance**, système dans lequel l'environnement de création de signatures électroniques est géré par un prestataire de services de confiance au nom du signataire, est appelée à se développer en raison de ses multiples avantages économiques. Toutefois, afin que ces signatures électroniques reçoivent la même reconnaissance juridique que les signatures électroniques créées avec un environnement entièrement géré par l'utilisateur, les prestataires offrant des services de signature électronique à distance devraient appliquer des procédures de sécurité spécifiques en matière de gestion et d'administration et utiliser des systèmes et des produits fiables, notamment des canaux de communication électronique sécurisés, afin de garantir que l'environnement de création de signatures électroniques est fiable et qu'il est utilisé sous le contrôle exclusif du signataire. Dans le cas de la création d'une signature électronique qualifiée à l'aide d'un dispositif de création de signature électronique à distance, les exigences applicables aux prestataires de services de confiance qualifiés énoncées dans le présent règlement devraient s'appliquer.

# Examen ESLI

## QCM

- pas de point négatif
- 40 min

## Conception

- pertinence - granularité - cohérence - clarté
- 50-60 minutes

