

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



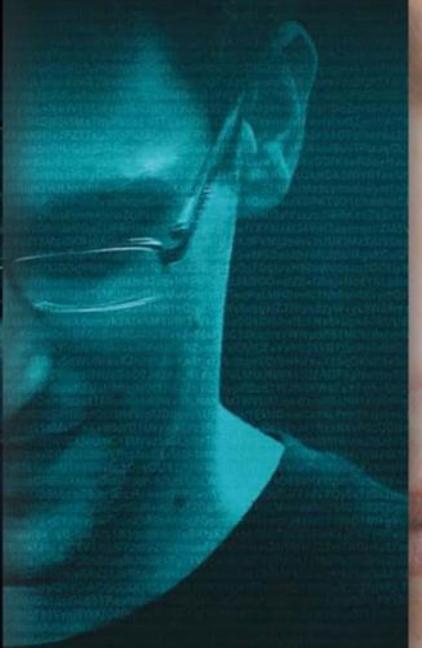
"A WHITE KNUCKLE THRILLER.
Clear, urgent, and positively terrifying at times."
-Peter Debruge, Variety

ZERO DAYS

© 2014 CIRQUE PICTURES and PARTICIPANT MEDIA, Inc. in association with ZINNITATIVE DOCUMENTARY FILMS & GLOBAL PRODUCE / OSCAR® winning "ZERO DAYS" by ANDY GREENE, ANTHONY HERRERA and BRETT WILLEY. ALSO: JEFF SORRELL, DAVID VERNERIAN, SARAH DUNBLAD, ALEXANDER SODERBERGH and REX SWINNEY. PRODUCED BY ANDREW SODERBERGH and REX SWINNEY. WRITTEN BY ANDREW SODERBERGH and REX SWINNEY.

© 2014 CIRQUE PICTURES and PARTICIPANT MEDIA, Inc. in association with ZINNITATIVE DOCUMENTARY FILMS & GLOBAL PRODUCE / OSCAR® winning "ZERO DAYS" by ANDY GREENE, ANTHONY HERRERA and BRETT WILLEY. ALSO: JEFF SORRELL, DAVID VERNERIAN, SARAH DUNBLAD, ALEXANDER SODERBERGH and REX SWINNEY. PRODUCED BY ANDREW SODERBERGH and REX SWINNEY. WRITTEN BY ANDREW SODERBERGH and REX SWINNEY.

COMING JULY 8



AUX OSCARS®
HOMMAGE
AU FILM DOCUMENTAIRE

PAR
A POITRAS

PROJETEUR EXÉCUTIF:
SODERBERGH

ENFOUR

© 2014 CIRQUE PICTURES and PARTICIPANT MEDIA, Inc. in association with ZINNITATIVE DOCUMENTARY FILMS & GLOBAL PRODUCE / OSCAR® winning "ZERO DAYS" by ANDY GREENE, ANTHONY HERRERA and BRETT WILLEY. ALSO: JEFF SORRELL, DAVID VERNERIAN, SARAH DUNBLAD, ALEXANDER SODERBERGH and REX SWINNEY. PRODUCED BY ANDREW SODERBERGH and REX SWINNEY. WRITTEN BY ANDREW SODERBERGH and REX SWINNEY.



ONCE YOU'RE IN, THERE'S NO WAY
TEHRA

La signature électronique sécurisée



F96DE8C227A259C87EE1DA2AED57C93
FE5DA36ED4EC87EF2C63AAE5B9A7EFF
D673BE4ACF7BE8923CAB1ECE7AF2DCF
7AE29A3DA44F235A24C963FF0DF3CA3
599A70E5DA36BF1ECE77F8DC34BE129
A6CF4D126BF5B9A7CFEDF3EB850D37C
F0C63AA2509A76FF9227A55B9A6FE3D
720A850D97AB1DD35ED5FCE6BF0D138
A84CF8DC34BE129F8DC34B

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



Les 5 primitives cryptographiques

Chiffrement Symétrique



Chiffrement Asymétrique



Fonctions de hachage



Établissement de clé

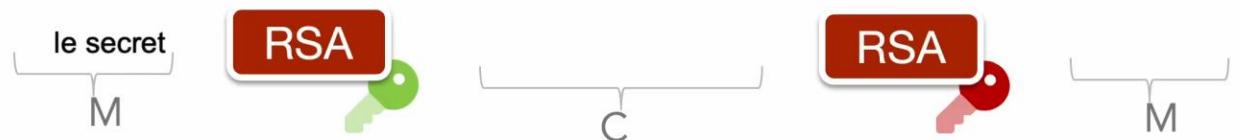


Générateurs d'aléa



Hello !

AES



Hello !

SHA



26, 47, 10

DH



91690410bec9
graine

PRNG

11
10 ✓

798

aléa

Confusion et Diffusion

Substitution et Permutation

Cryptographie Symétrique => Confusion et Diffusion

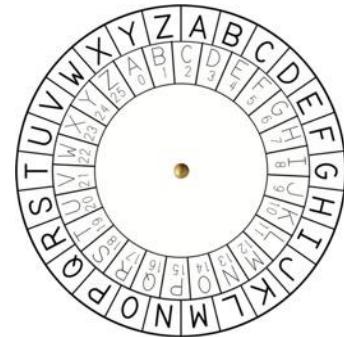
Confusion par Substitution

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



Hachage

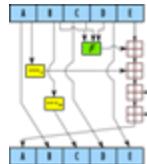
Hachage cryptographique

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



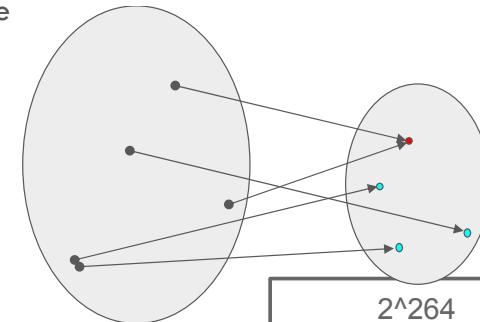
- Fonctions à sens unique

d'un espace infini vers un espace fini
de :

128, 160, 224, 256 ou 512 bits

- Résistantes aux attaques

- 1^{ère} pré-image
- 2^{de} pré-image
- collisions



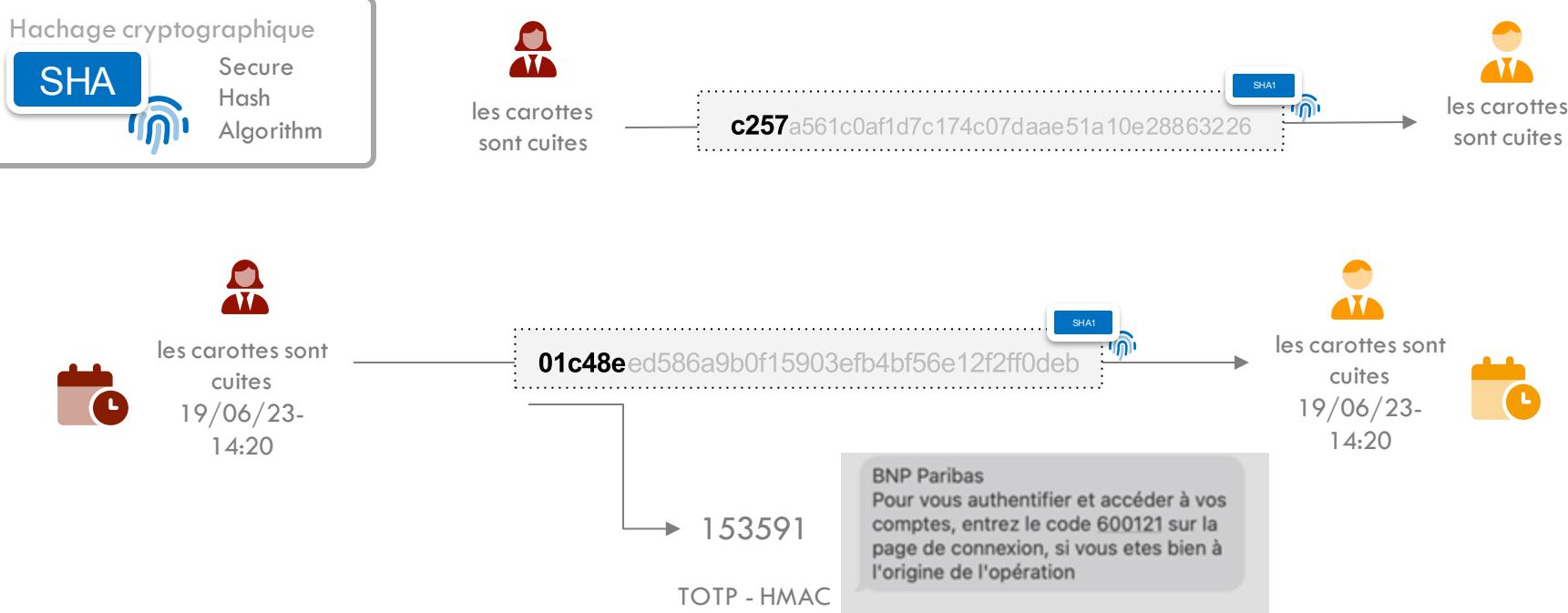
2^{264}

$2^{(80*3,3)}$

10^{80}

nb. atomes univers

Hachage



Hachage

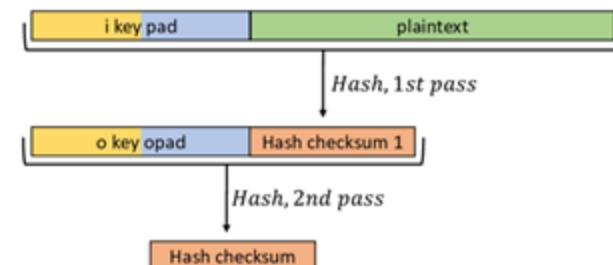
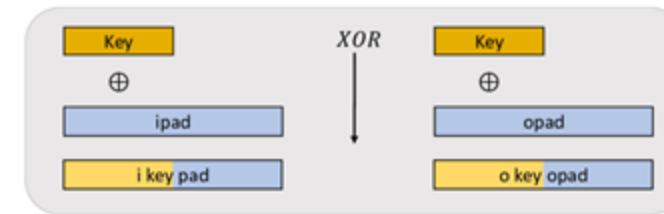
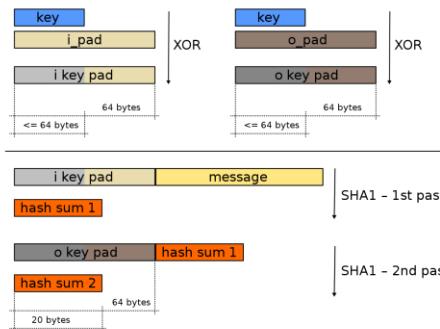
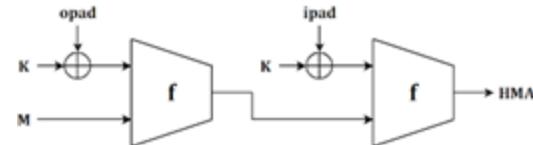
Hachage cryptographique

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

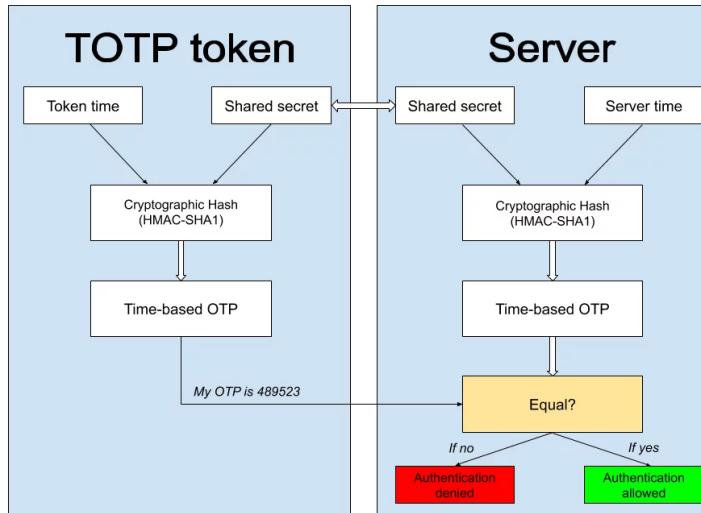
Hachage cryptographique

SHA



Secure
Hash
Algorithm

Construction Hachage Crypto: TOTP



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.

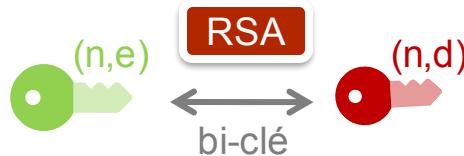
Asymétrique

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

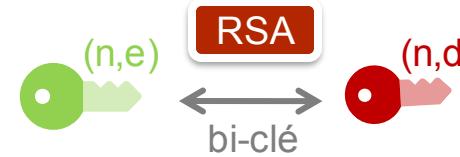
RSA



Rivest
Shamir
Adleman

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

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



Message=hello

p=1780731609485571264810668272791995899914711193175875151378804
074228378407969900390762278303079206056838884626265923868792218
367632557891163061154753567325103171346019724100725958290999956
638624959629388394207924641815440071623570127595625788276627675
2937990502678061624633628703093952827543039555774118880861

q=1659131989902429311320647733356628360786681694637180433412560
89400405261062088687848791945167845518159615723611150571927731
077520725426964037514349809386962699556220313541603209708580761
255180830815503806133227993555854257708408662876270644910978887
84189682166960030796126554428589177031865285035399557919043

e=65537

d=7541588446120496966253234620344712333501069795303049916555478
181024739612748768714849307670020087520498050753853969336769842
721963177053759904513002332911786096655412117391051494984460634
179886021618010916184586203664729878176106218580539243017832573
064752374135639173722301872742910212929520185645517211756292260
442337408227686415915940057868098542605732388150060822257263017
768153073048470043906529305219251168471687810973059378400133633
644318414348198698711374492042237109200383333030466757157771841
088217670969580298385949540902819878485019509485570883603413467
234108699097821896589722966999532605527832607563513

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=295446877827951519381542455099117772733932758531747159909715903
818628489453739346434808277662182293208684035240167304823272057321
422288076172288054479598539813596184732724354963397672285147033236
886721151377782449815143774439760136571897904056785781223890706818
861764171429839404340406627783874748342767092341923727807298159971
873503674156291633101481940976644619125965267514073300546813320042
580729928081378915771201586619482771747482544538290338600642244942
010670574478156402877487966789398331122247930143951606695548203913
045284760438130510844259880483102317018508493449083312299070937455
1035113290128717200136023

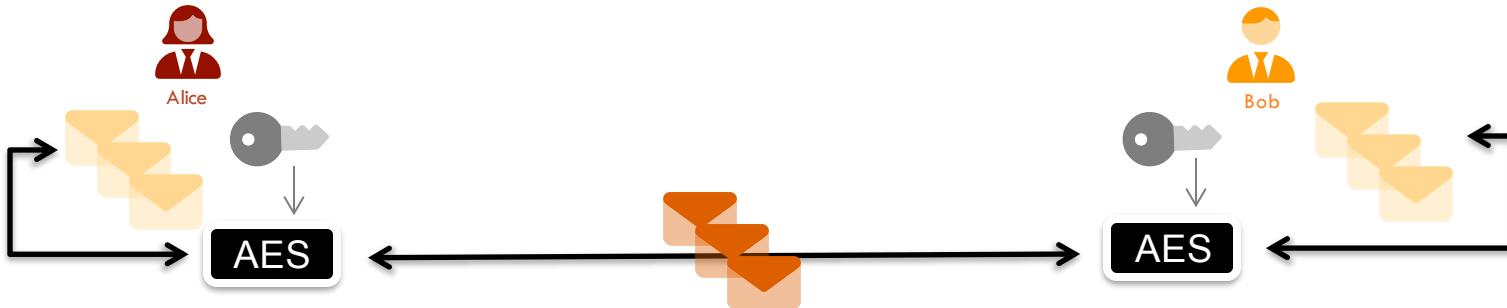
Public key (e,n) Private key (d,n)

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

Decipher=hello

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 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**

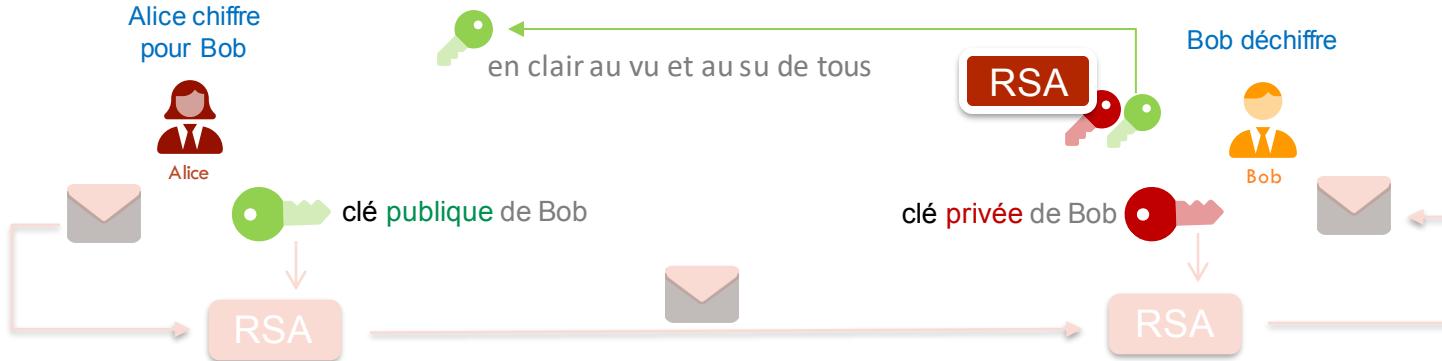
- la **clé publique** est envoyée en clair au vu et au su de tous
- seule la **clé publique** permet de chiffrer
- seule la **clé privée** permet de déchiffrer

Chiffrement asymétrique

RSA



une paire de bi-clé ☺



Chiffrement réel

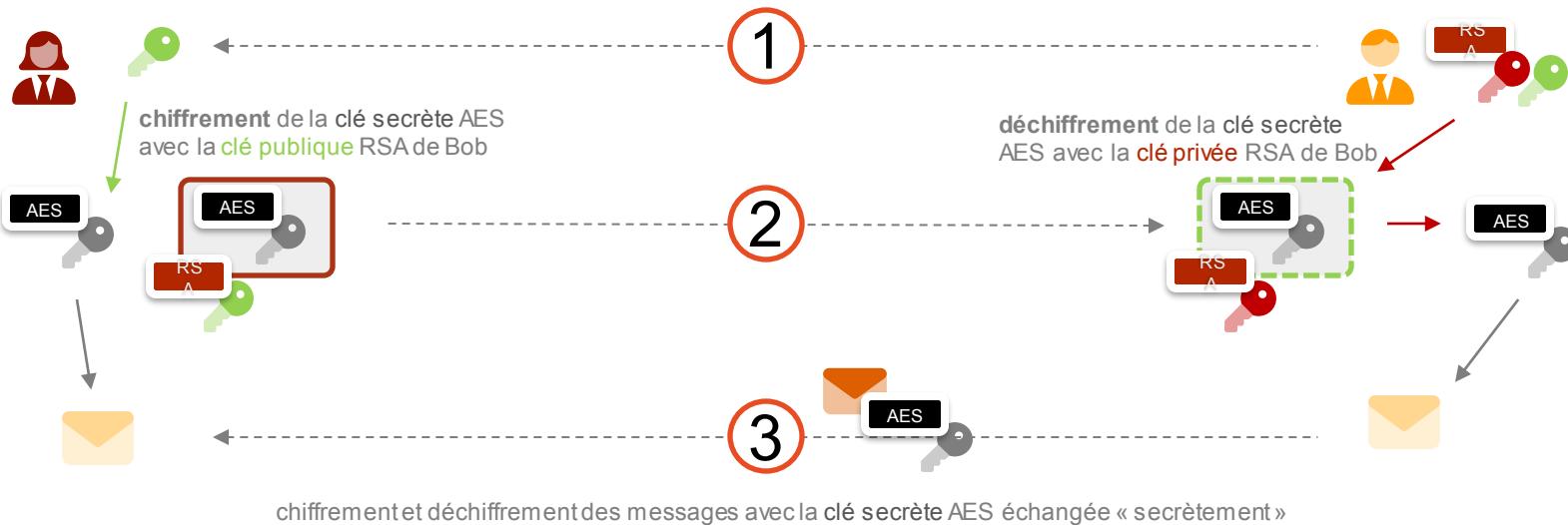
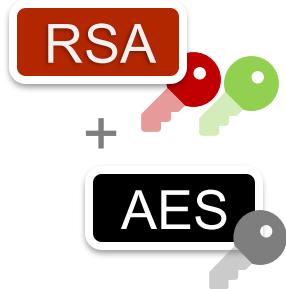


1. Chiffrement de la clé symétrique
2. avec la clé publique asymétrique



Chiffrement robuste

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

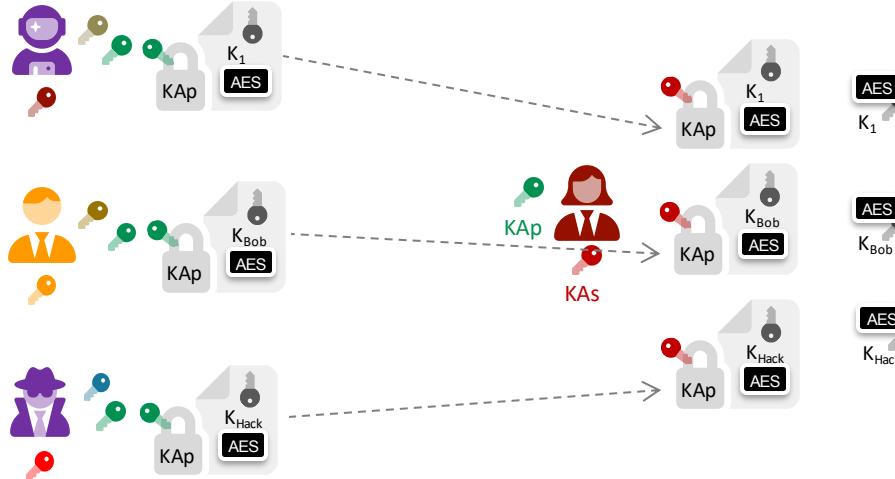


Usage: chiffrement pour la Confidentialité

RSA



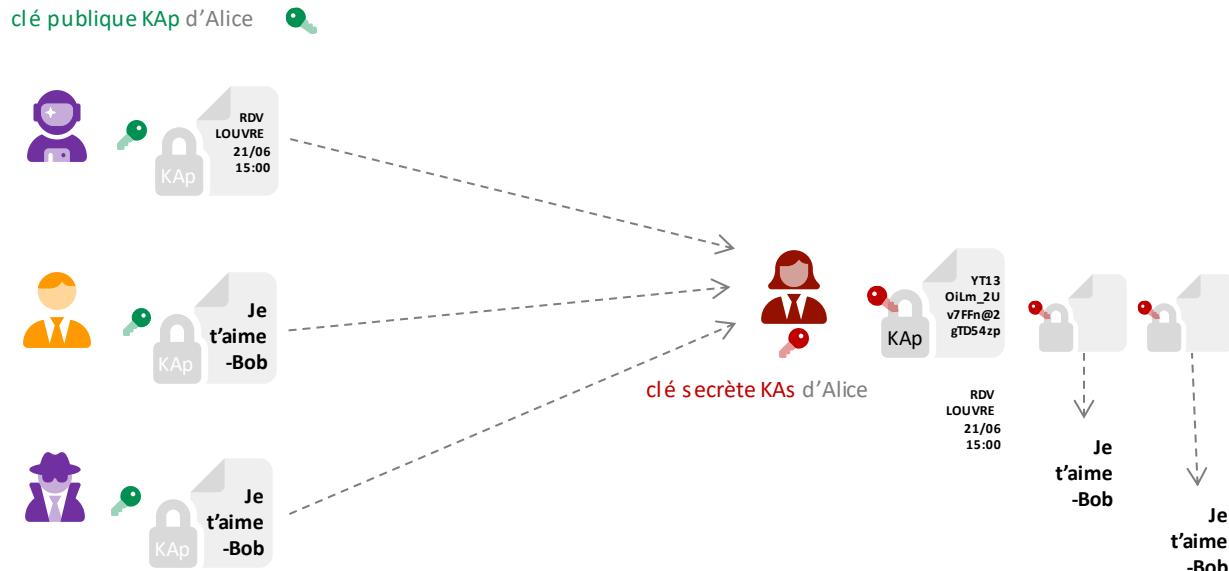
clé publique d'Alice



Type	Nom
pub	Loïc PERRY
pub	
sec/pub	Benoit LEGER
pub	etatin

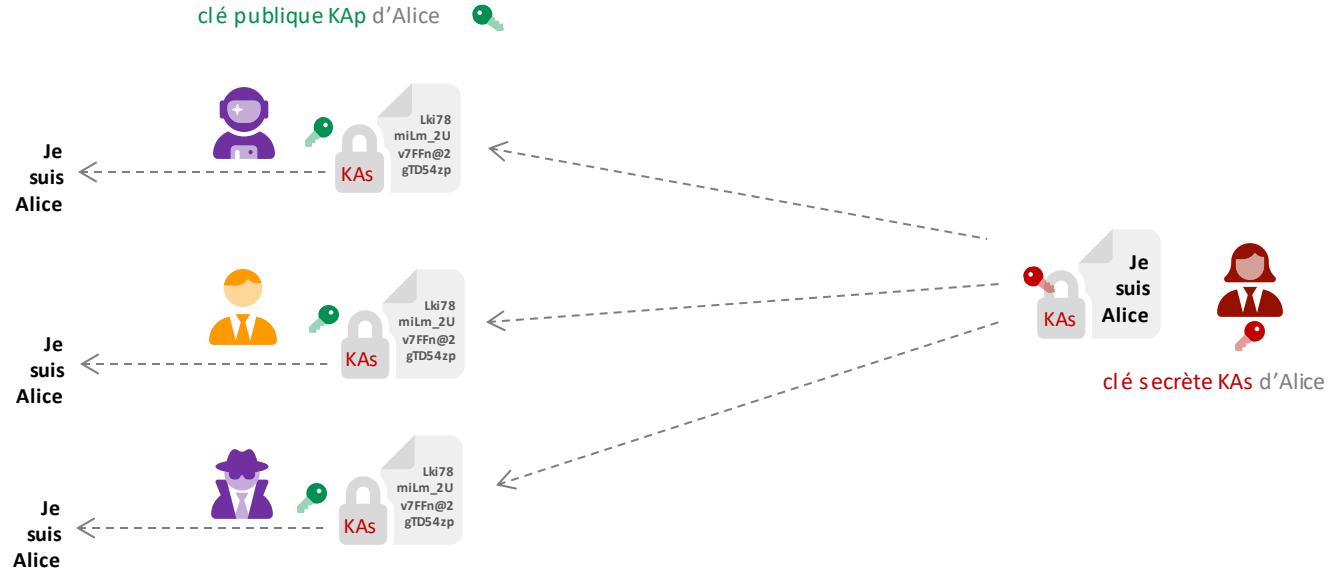


Usage: Chiffrement/Déchiffrement pour confidentialité



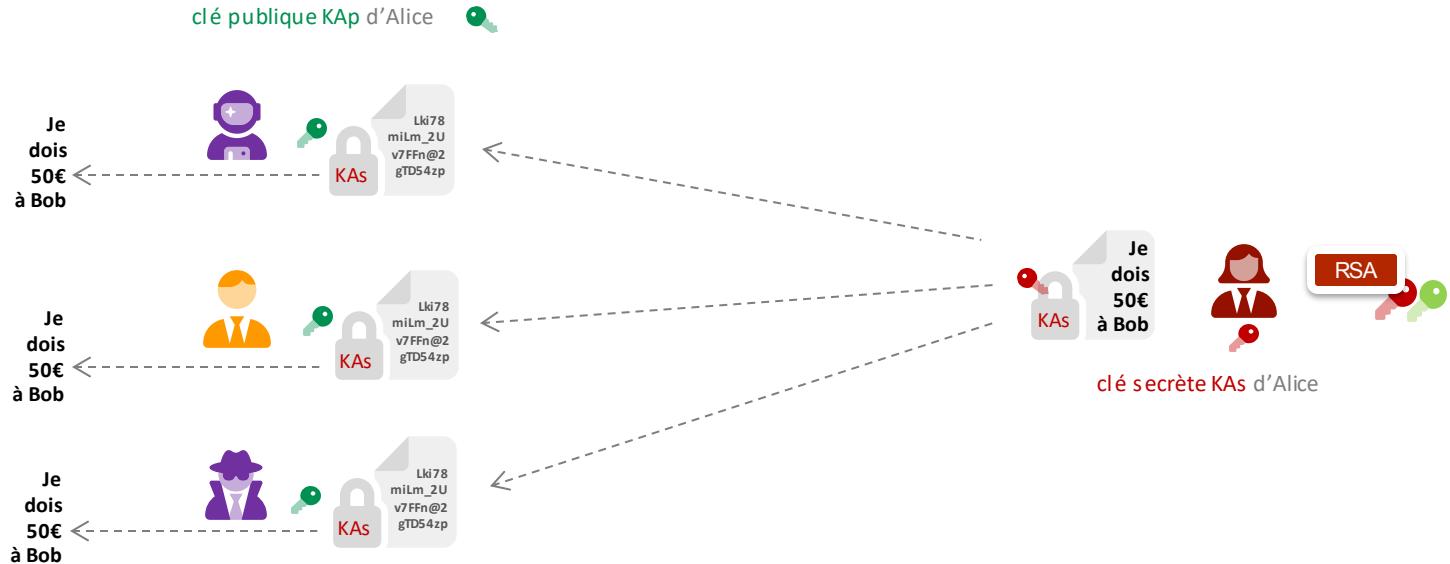


Usage: ? / ? pour ?



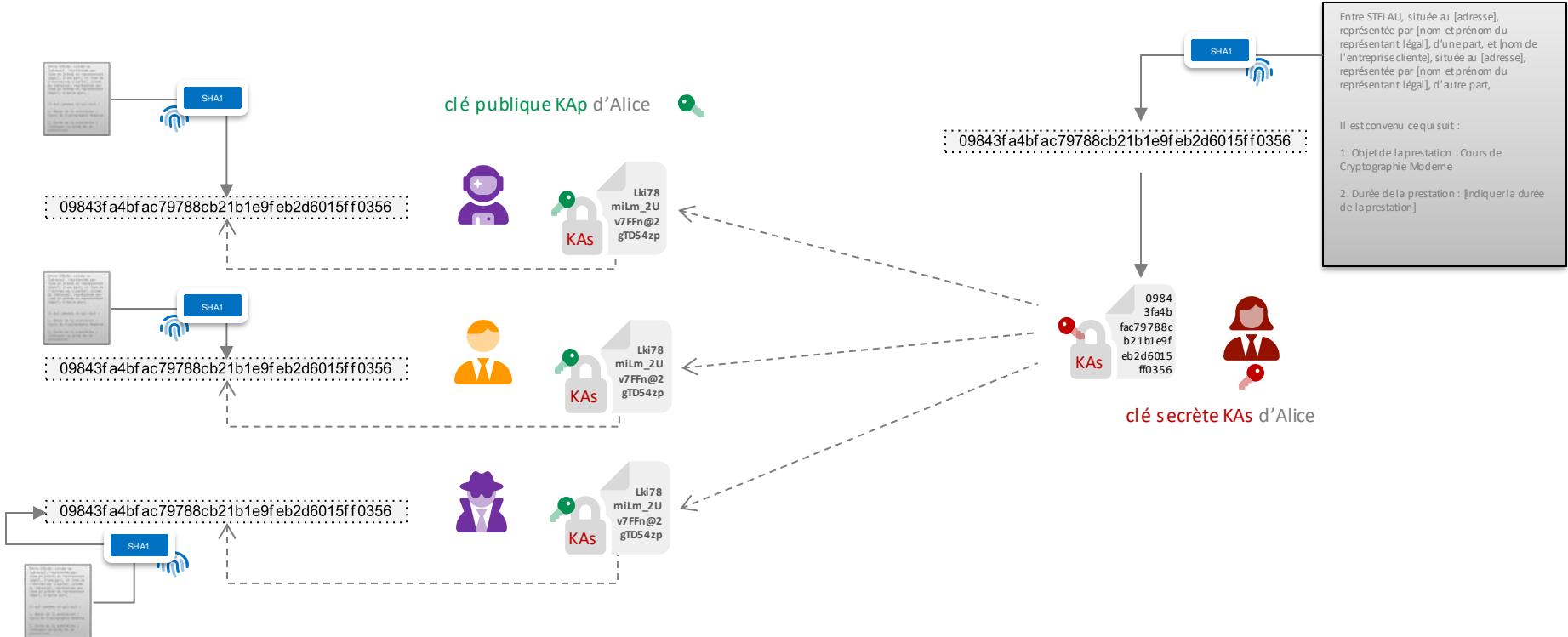


Usage: Signature / Vérification pour non-répudiation





Usage: Signature / Vérification pour non-répudiation



Deux usages différents



clé privée

clé publique

chiffrement

déchiffrer

chiffrer

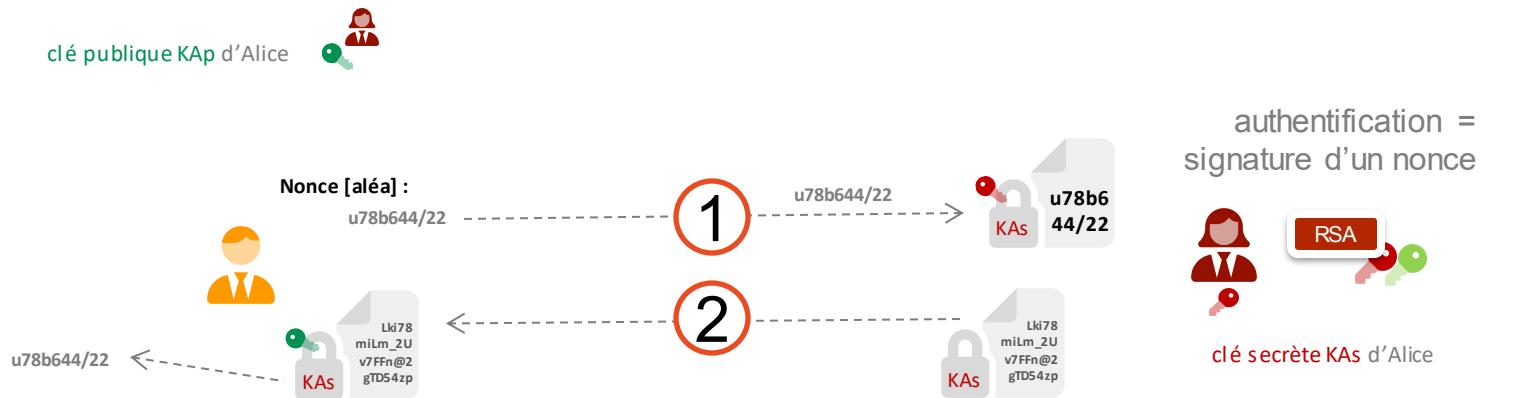
signature

signer

vérifier

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



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.

RSA

► Key agreement :

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

DH

Certificat électronique

Ce n'est pas une primitive cryptographique

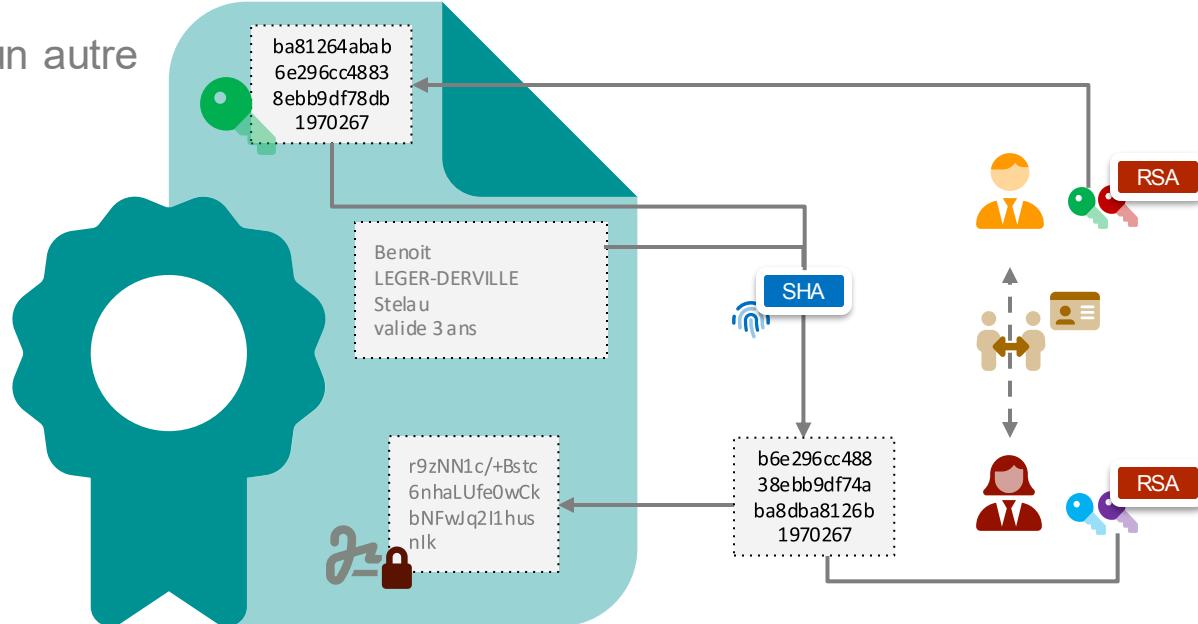
C'est un assemblage

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*

100 ans de crypto. **moderne**

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



50 ans de crypto. **asymétrique**

LA véritable révolution



20 ans de crypto. **quantique**

révolution ? (ou pas)



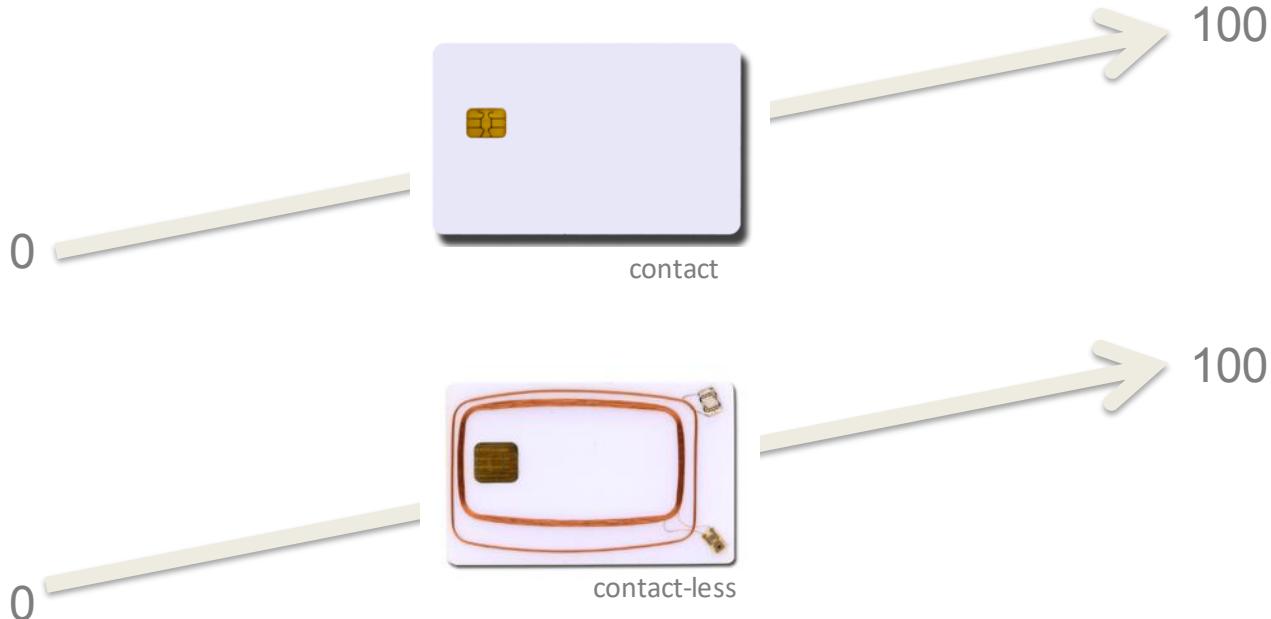
Confusion et Diffusion
« tant bien que mal »
de César à Enigma

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

Résout la difficulté de
l'échange de clé
+

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

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





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 $\varphi(n)$

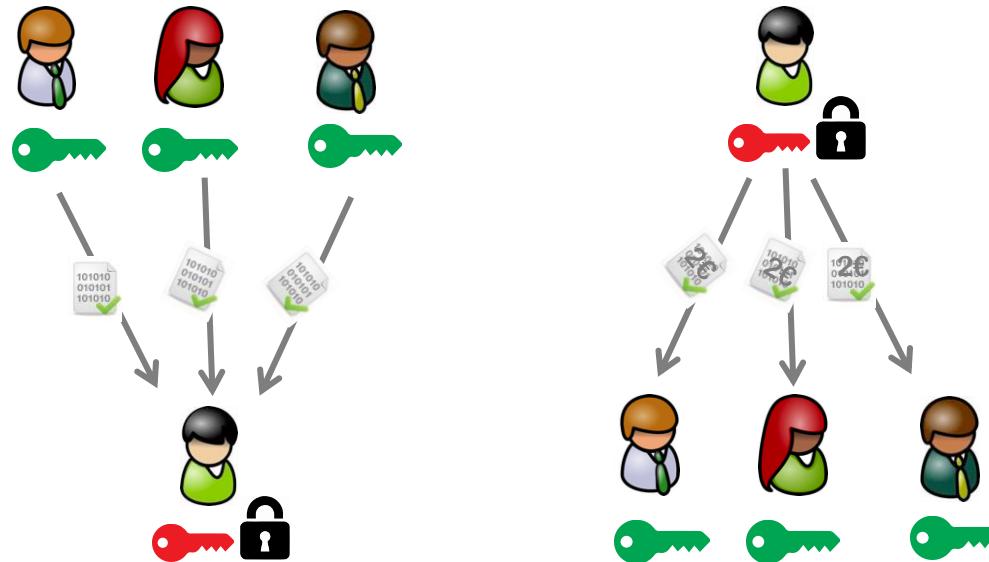
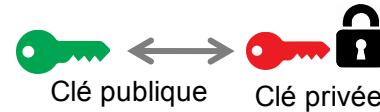
d inverse de e modulo $\varphi(n)$

- Comme $c \equiv m^e \pmod{n}$, $c^d \pmod{n} = m^{ed} \pmod{n}$
- Comme $ed \equiv 1 \pmod{(p-1)(q-1)}$ il existe un entier k tel que $ed \equiv 1 + k(p-1)(q-1)$
- 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



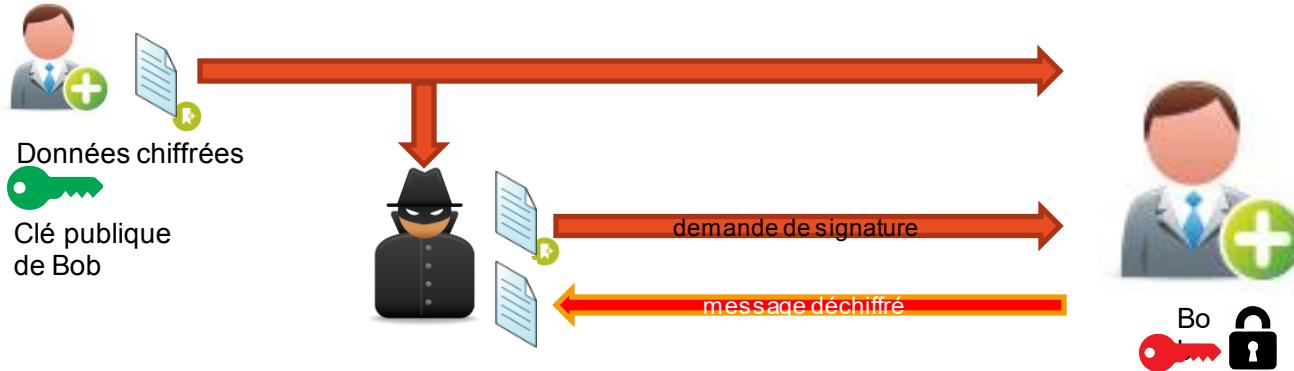
Usage : chiffrement vs signature



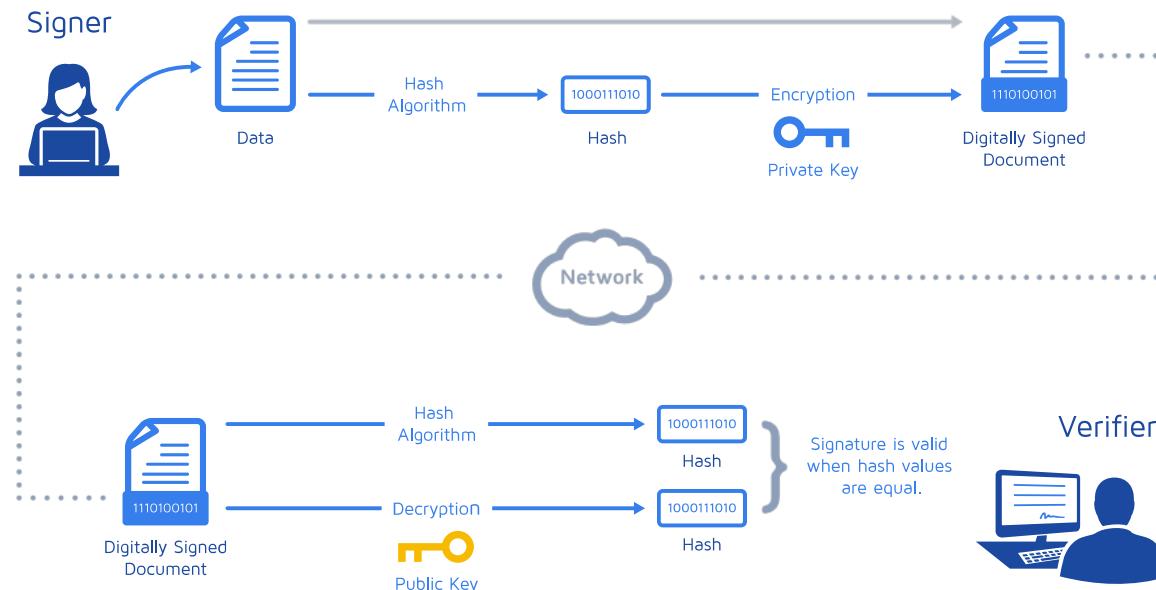
Rappels



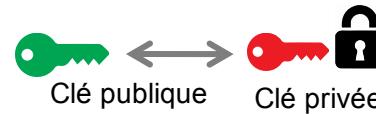
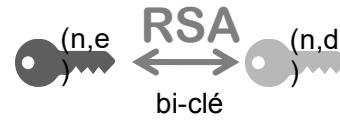
Certificat – Séparation des Key Usage



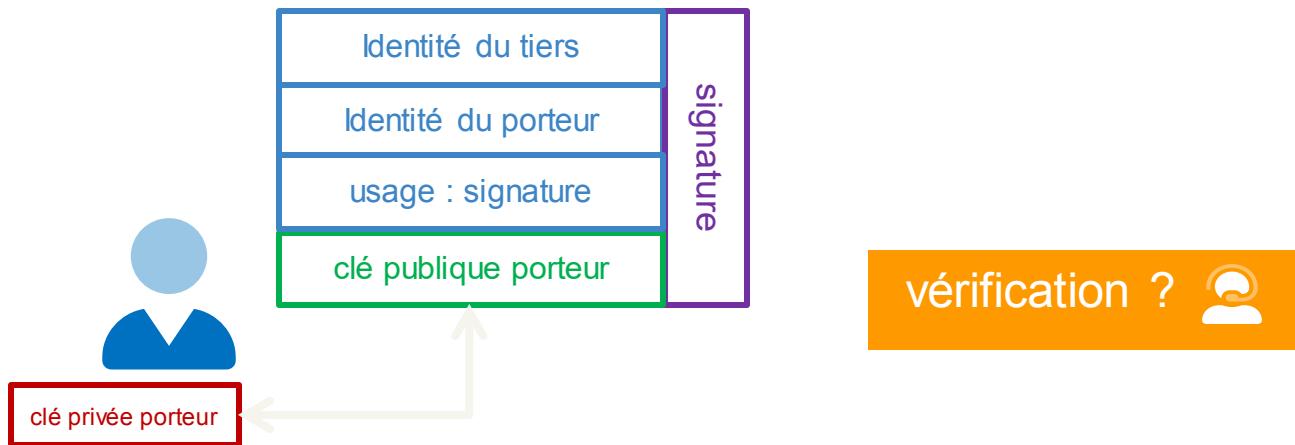
Exemple de Signature bi-clé de signature Certigna



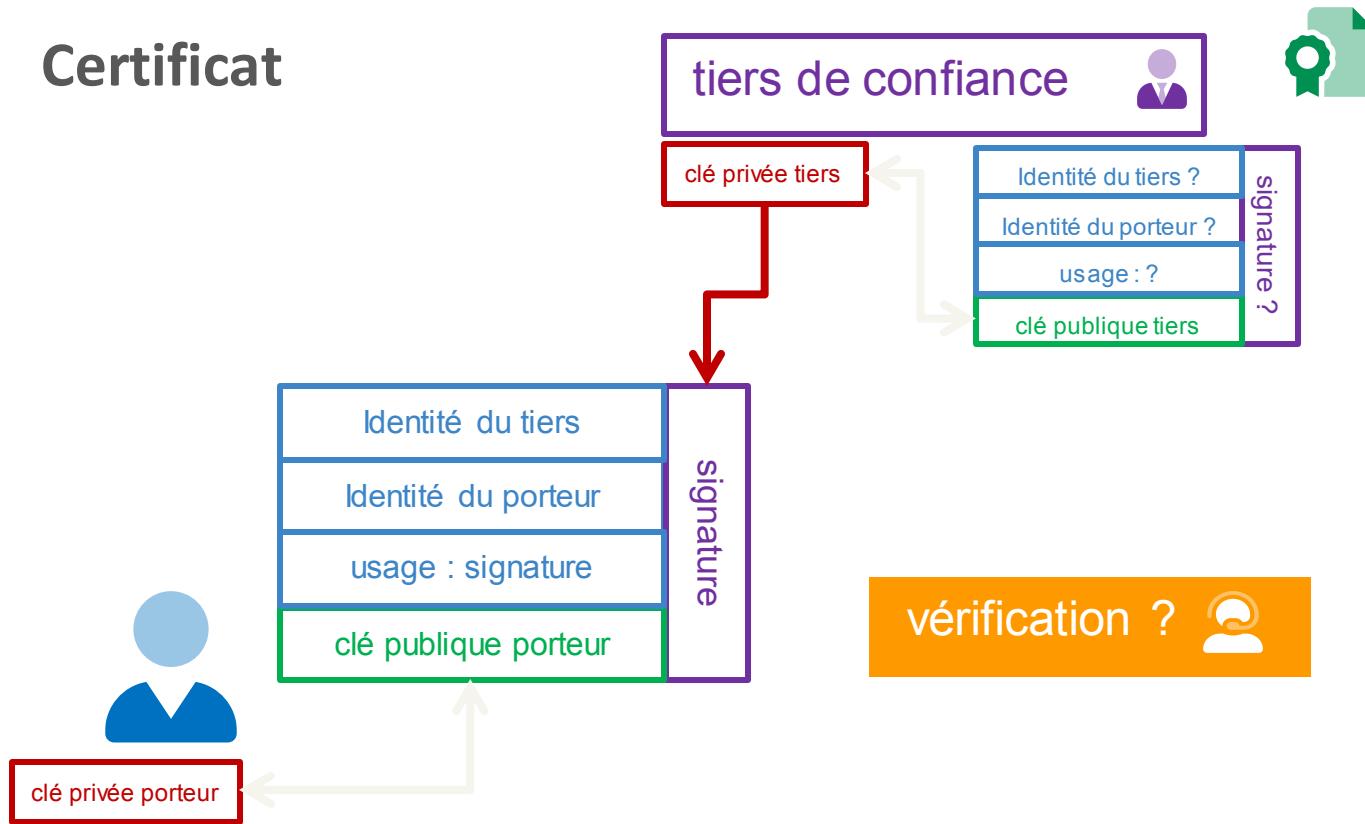
Certificat



Certificat

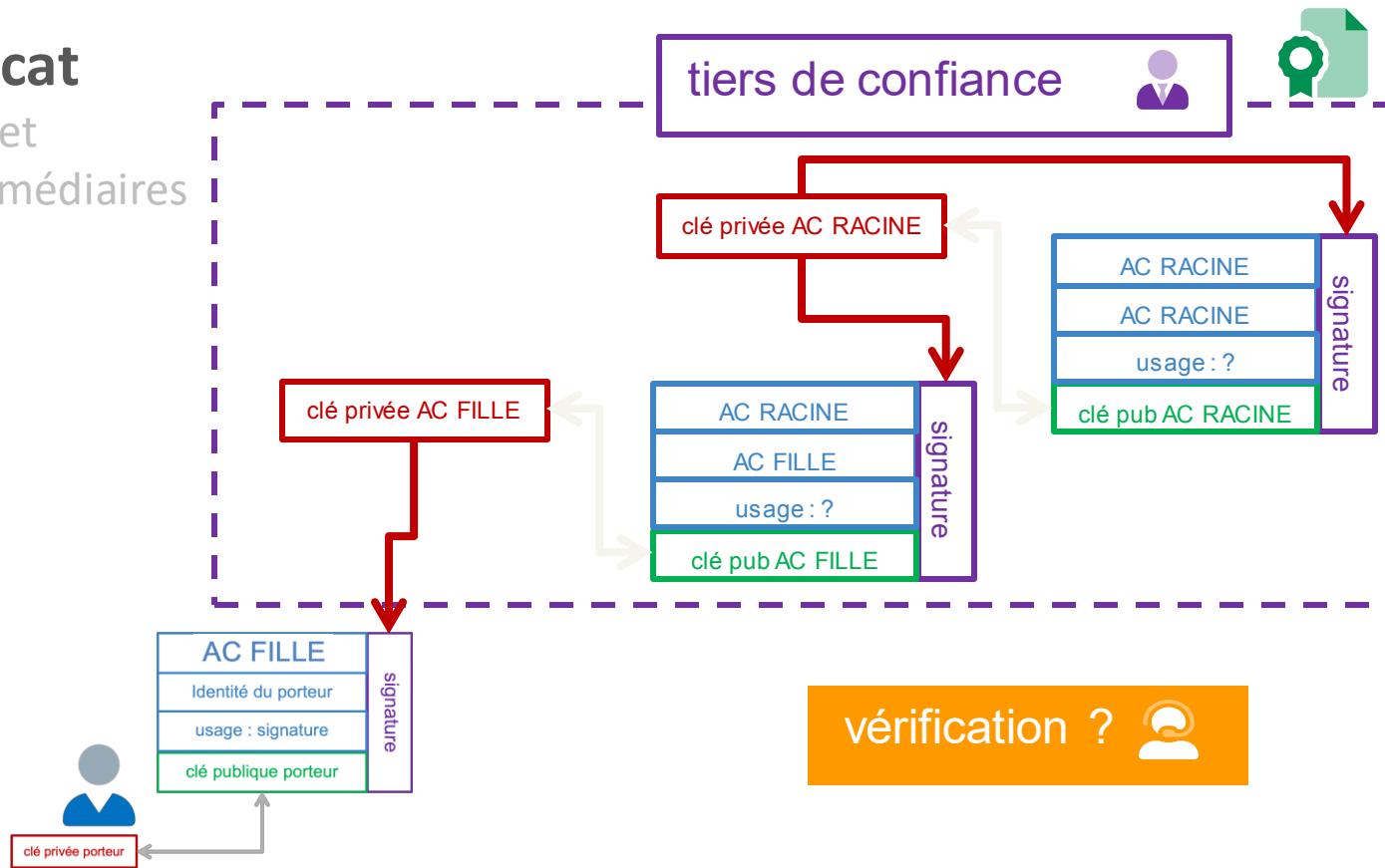


Certificat



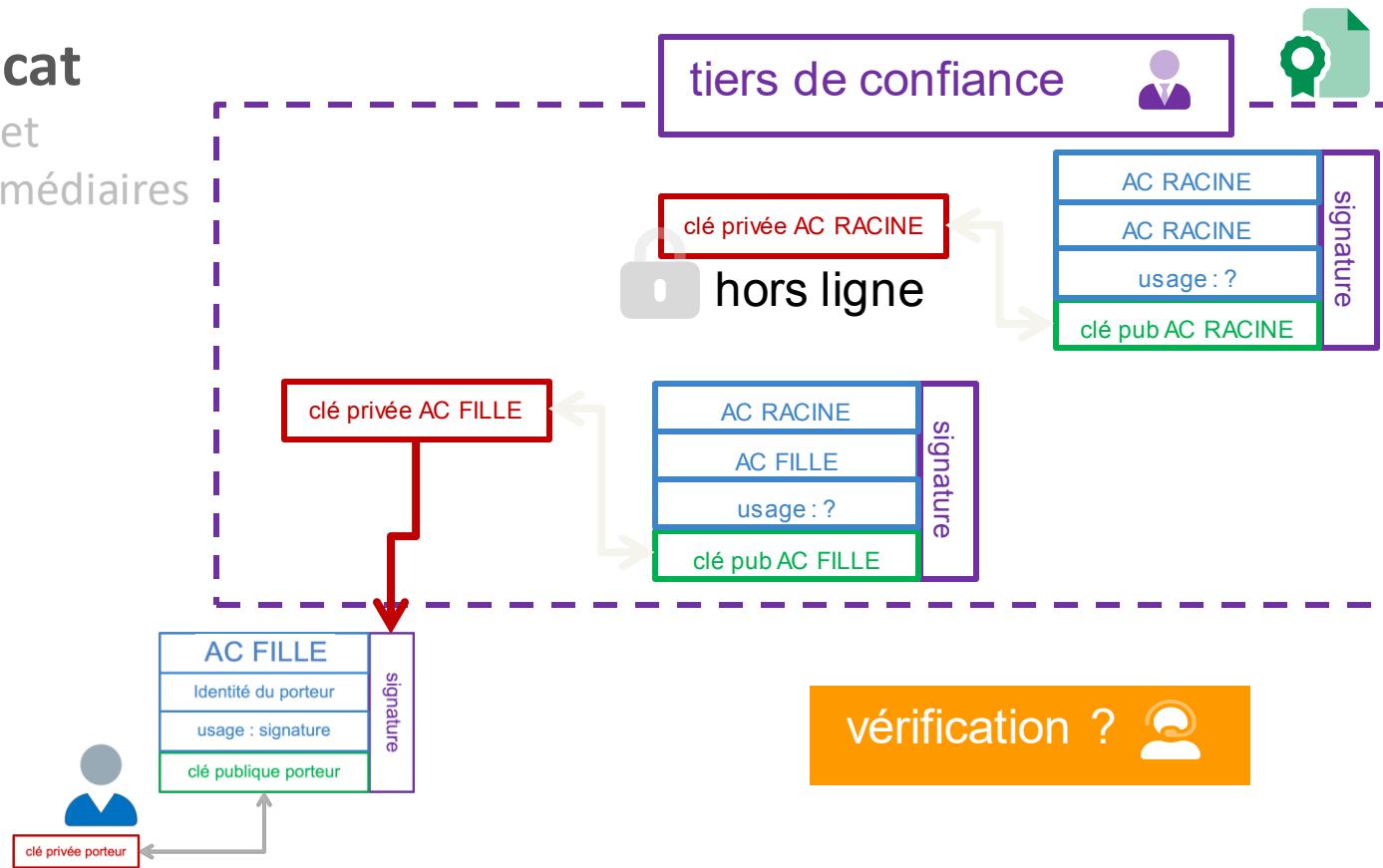
Certificat

Racines et
AC Intermédiaires



Certificat

Racines et
AC Intermédiaires



Certificat

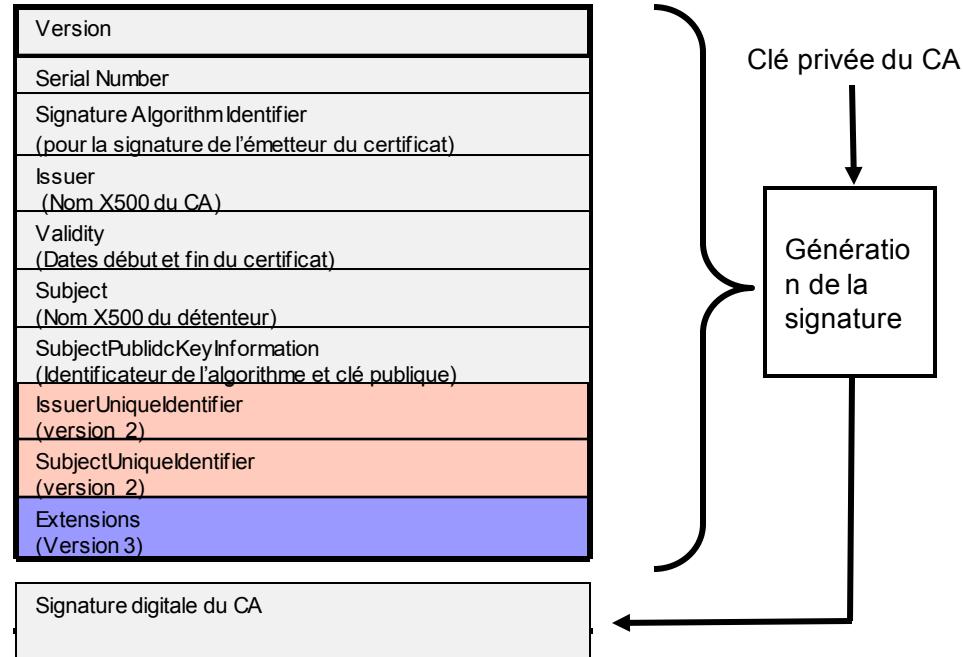
Gabarits des certificats X509 v3



```
$ 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=OOO2 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:73:2c:ea:
            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:c3:09:38:cc:9e:14:d5:a9:62:88:15:
            18:27:f8:8b:5d:ef:ac:cf:db:fb:ab:04:9b:eb:b4:
            27:0c:67:74:a7:7c: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:bo:e2:94:54:44:0e:a6:80:30:af:
            15:9e:61:ae:47:cd:a9:cf:e8:7d:c7:09:fe:98:1c:
            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:a1:83:9a: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=OOO2
      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:93:50:24:04:5a: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:7:88:6d: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:d3:0a:23:4e:07:ff:db:e7:25: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:b6:a7:b1:ae:ce:cf:06:ab:62:c9:b0:
  98:62:0d:94:b5:b9:d1:62:01:a4:4f:56:63:c1:89:67:a4: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:b8:b3:70:74:66:f7:3d:a6:
  50:6a:0b:4c
```

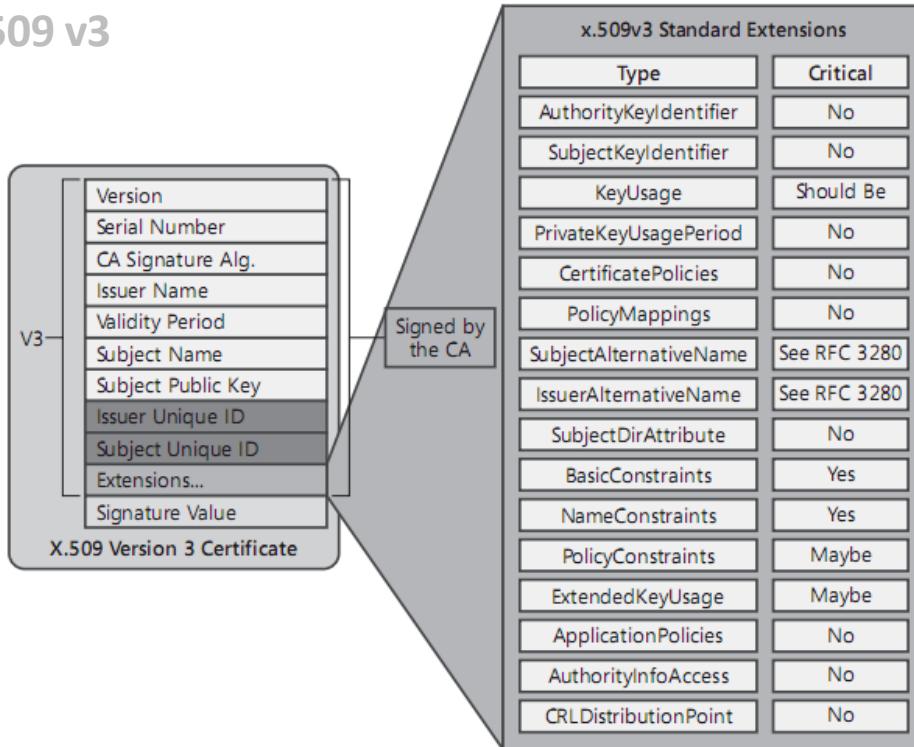
Certificat

Gabarits des certificats X509 v3



Certificat

Gabarits des certificats X509 v3



IGC - PKI

IGC – Demande de certificat



IGC - PKI



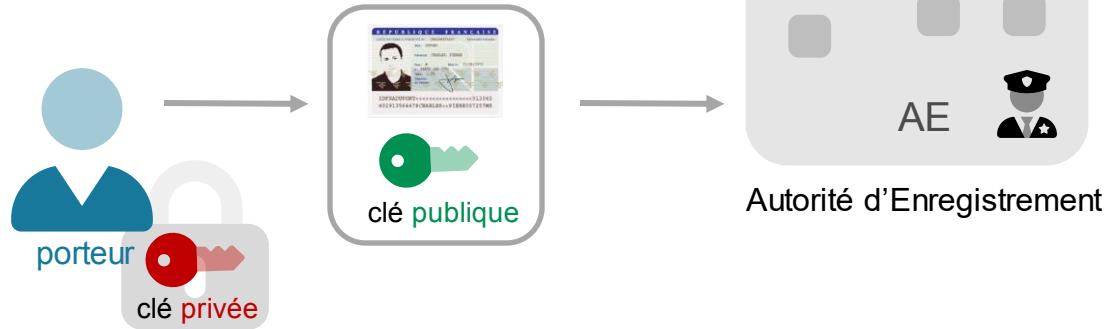
IGC – Demande de certificat



génération du bi-clés

IGC - PKI

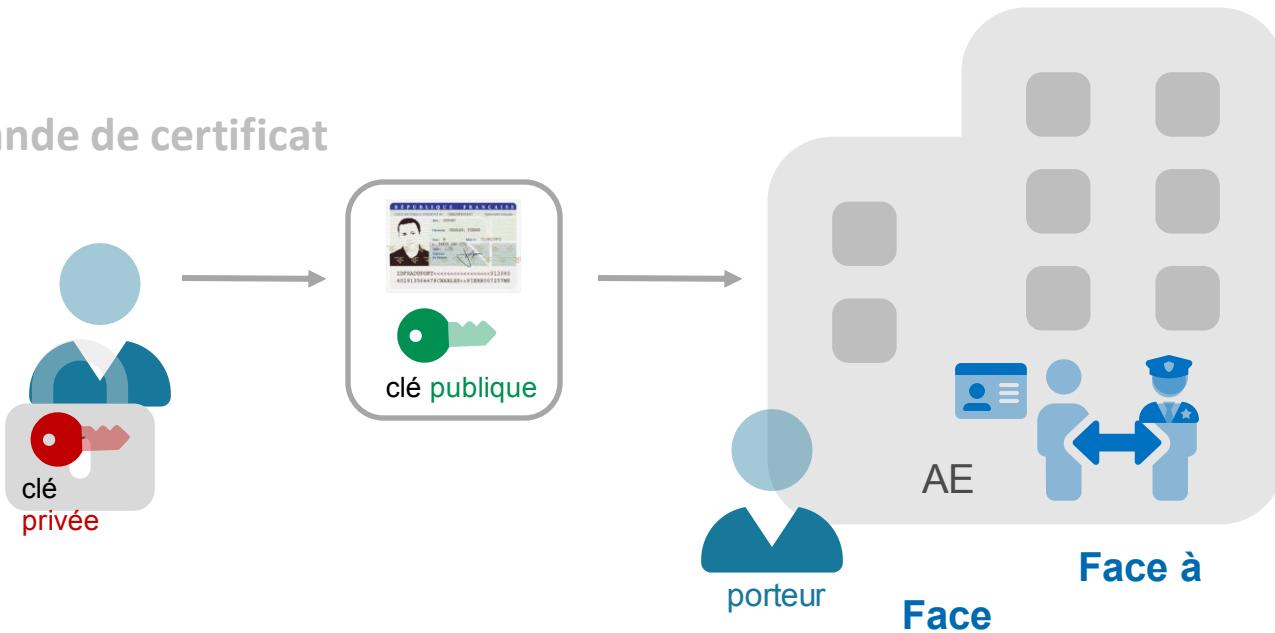
IGC – Demande de certificat



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

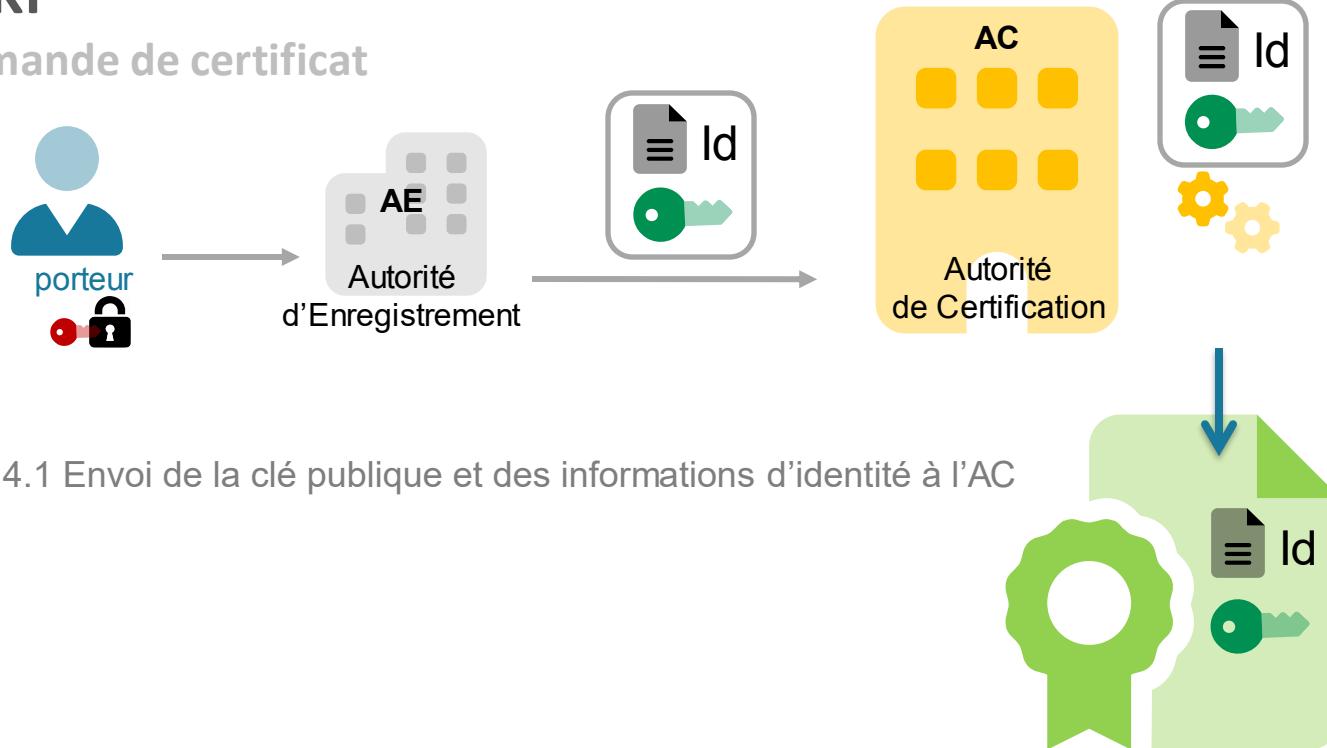
IGC - PKI

IGC – Demande de certificat



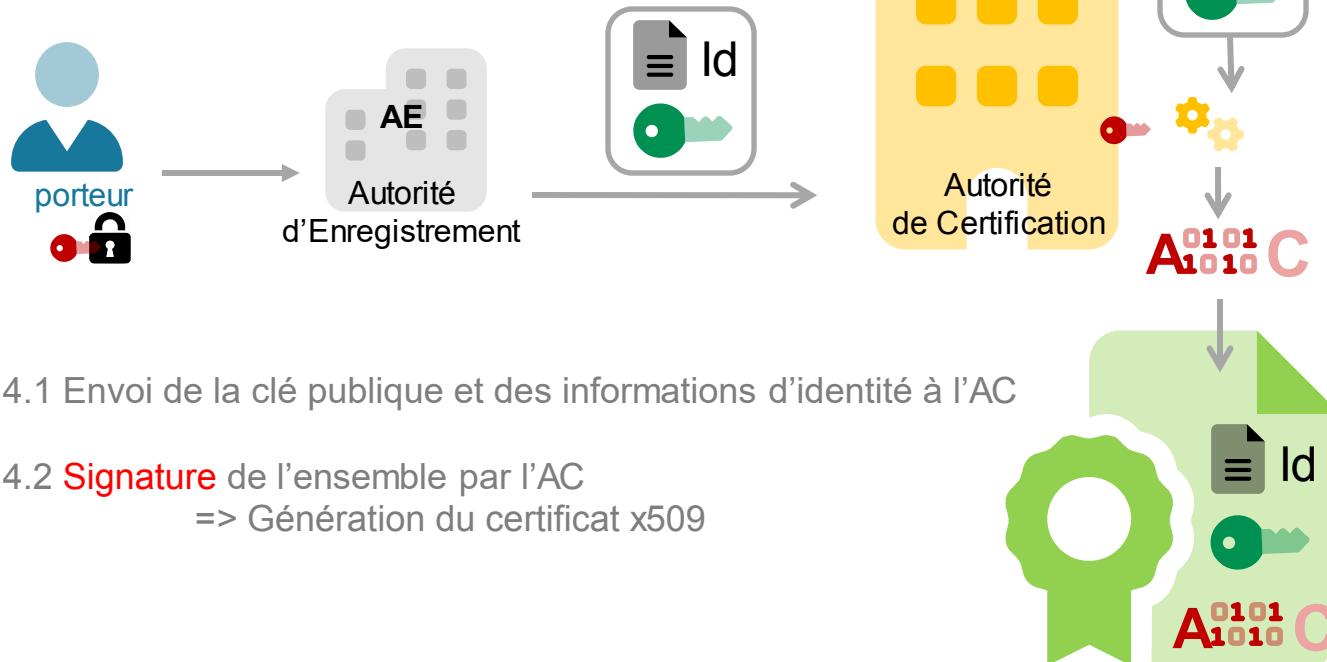
IGC - PKI

IGC – Demande de certificat



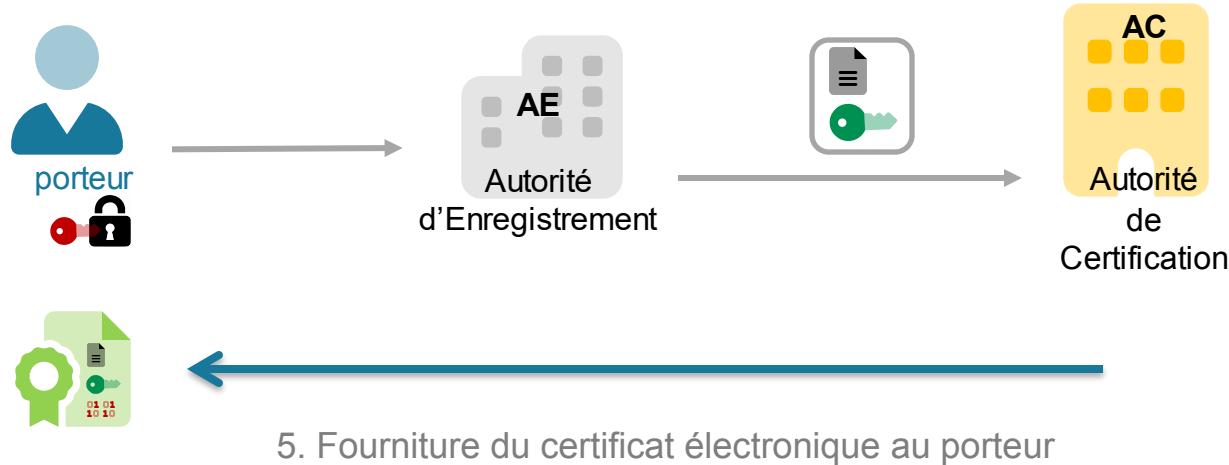
IGC - PKI

IGC – Demande de certificat



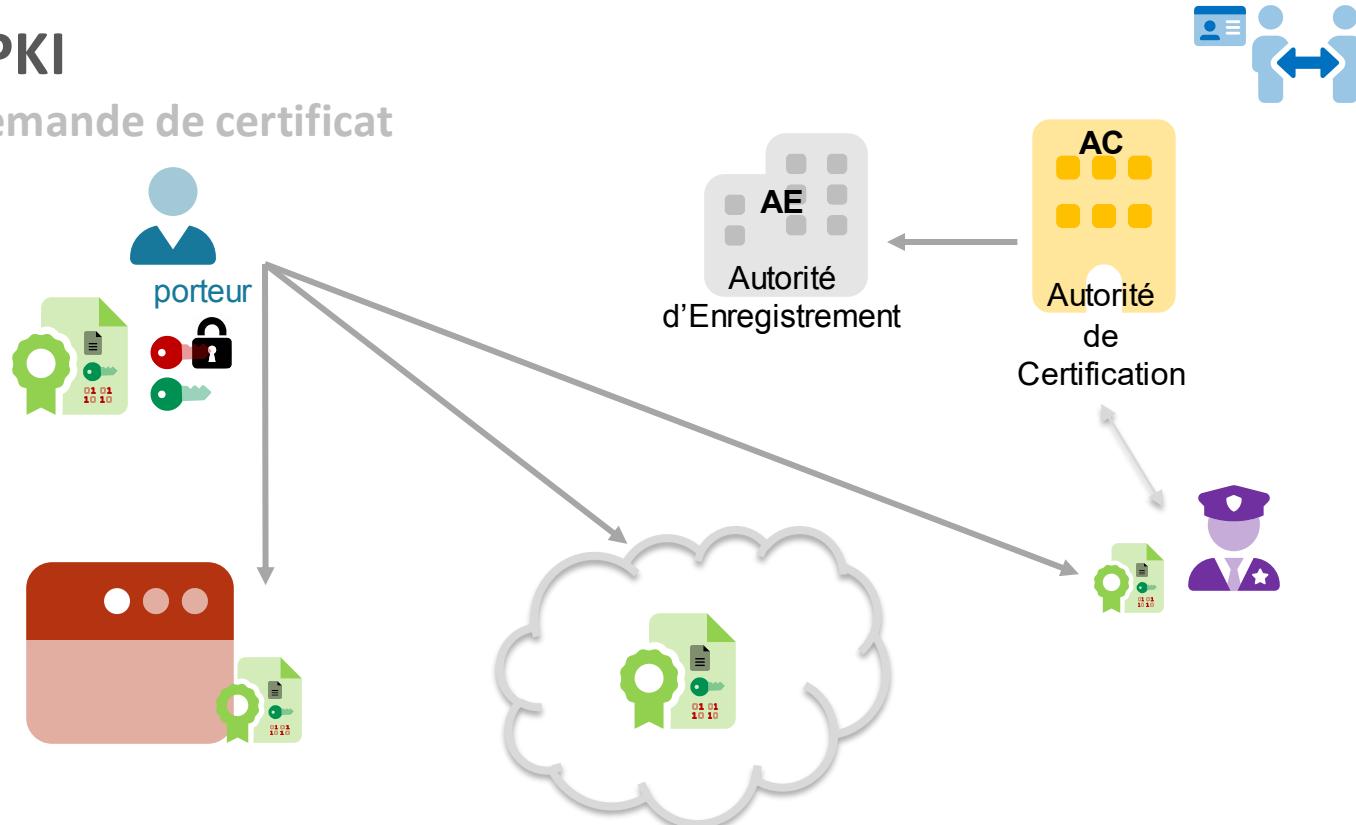
IGC - PKI

IGC – Demande de certificat



IGC - PKI

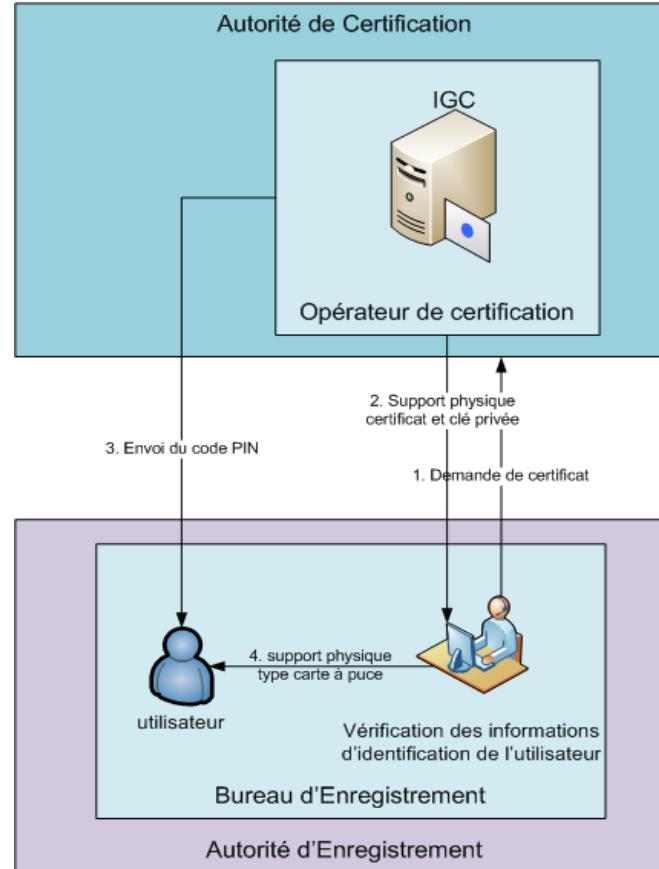
IGC – Demande de certificat



6. Utilisation des éléments

IGC - PKI

AC - OC - AE





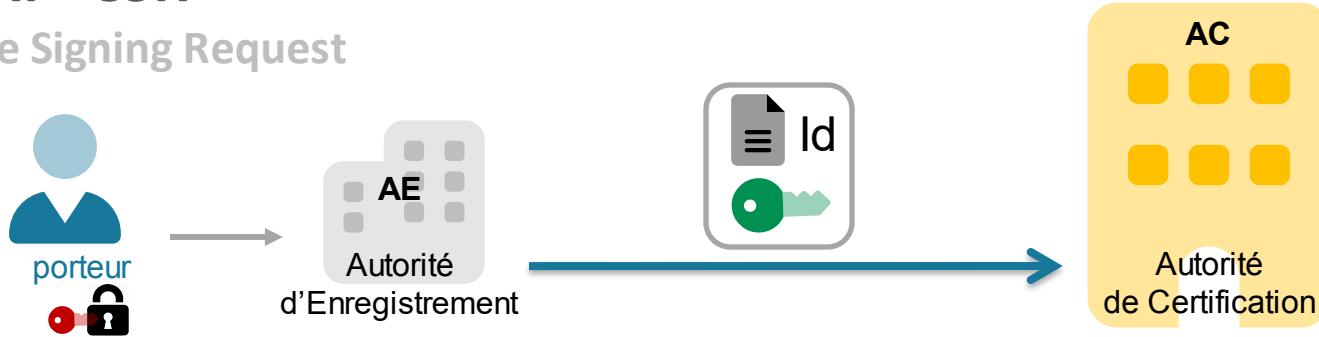
Question ?

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

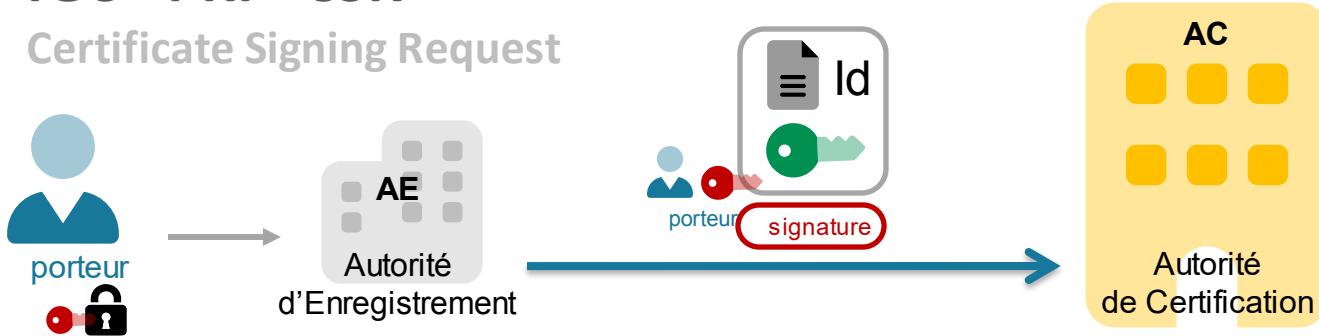


Comment assurer la preuve de **possession** de la **clé privée** du porteur ?

- 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

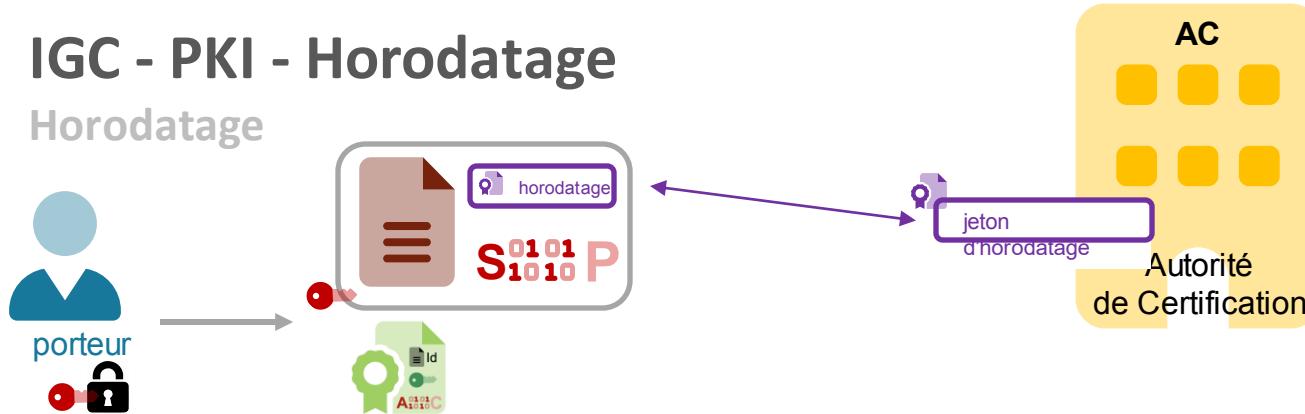
Question

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



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

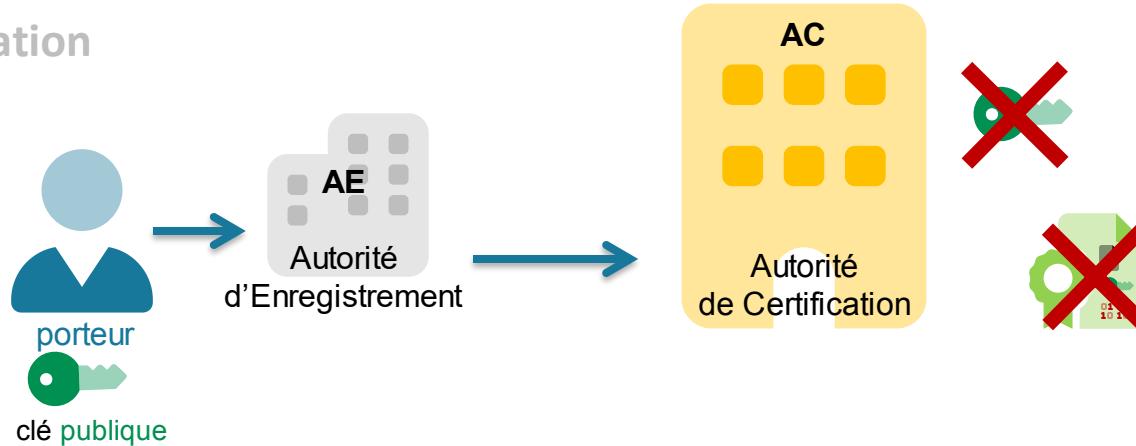
IGC – Révocation



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

IGC - PKI

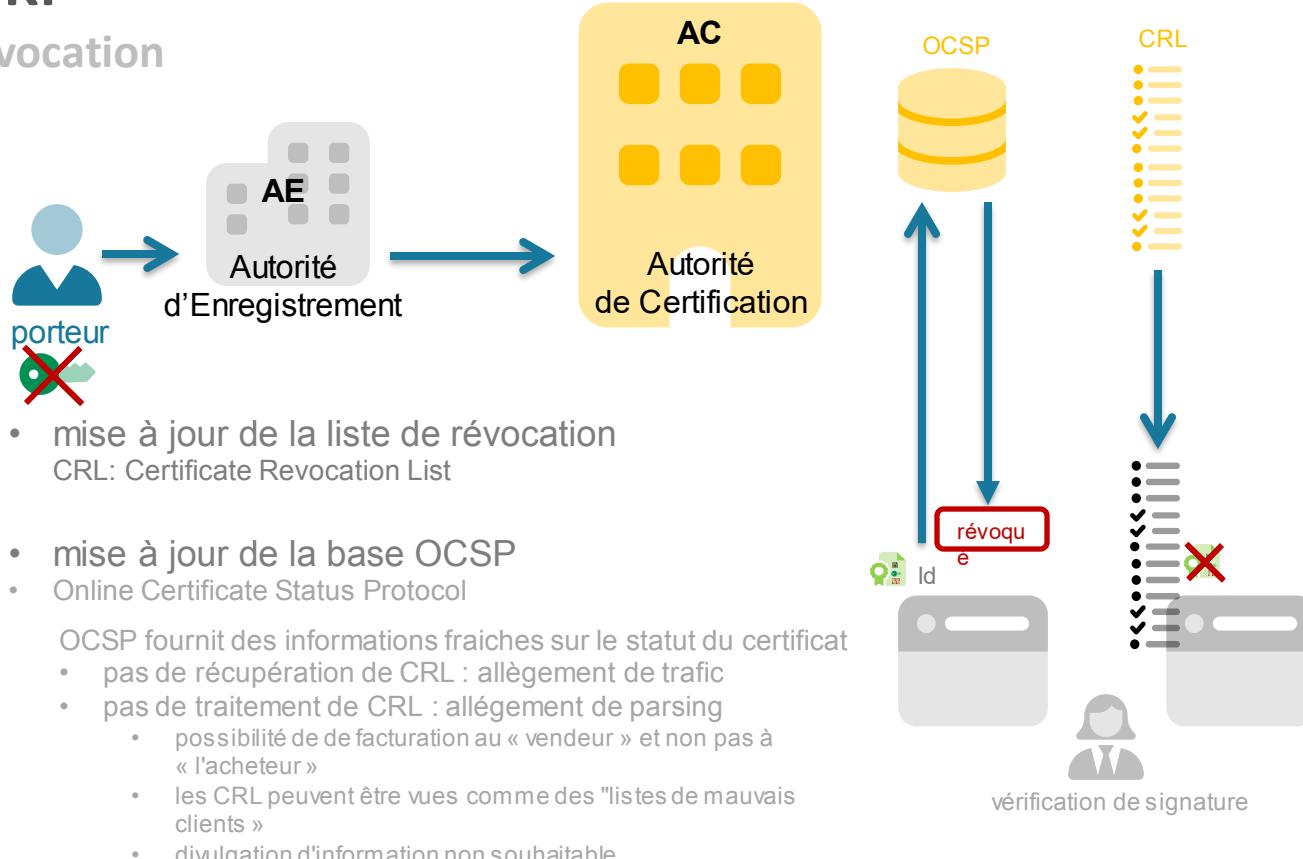
IGC – Révocation



- 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

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
44efb87b718950bf4721dd5c94f5f7
d12fc4efac9d9b50c81bbc1555c3d7
6610ef3080a354e60b625f5c50a23
a6bfd13ec024239ddcb47706c9a23
11fc38e37161e87501236542732797
2469b3985721cc0fea3b04047a9c5
b559e3471a736f5e4c7b473b2e86b1
b21dd8a829828d f8d6



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

7e0950bb938539162d268b379595
44efb87b718950bf4721dd5c94f5f7
d12fc4efac9d9b50c81bbc1555c3d7
6610ef3080a354e60b625f5c50a23
a6bfd13ec024239ddcb47706c9a23
11fc38e37161e87501236542732797
2469b3985721cc0fea3b04047a9c5
b559e3471a736f5e4c7b473b2e86b1
b21dd8a829828d f8d6



Entités et vocabulaire

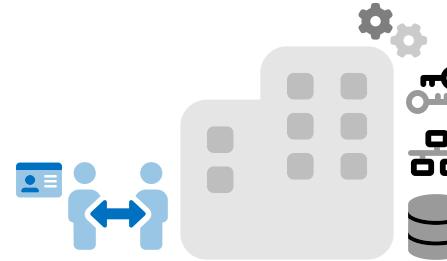
les termes et leurs synonymes

- 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.

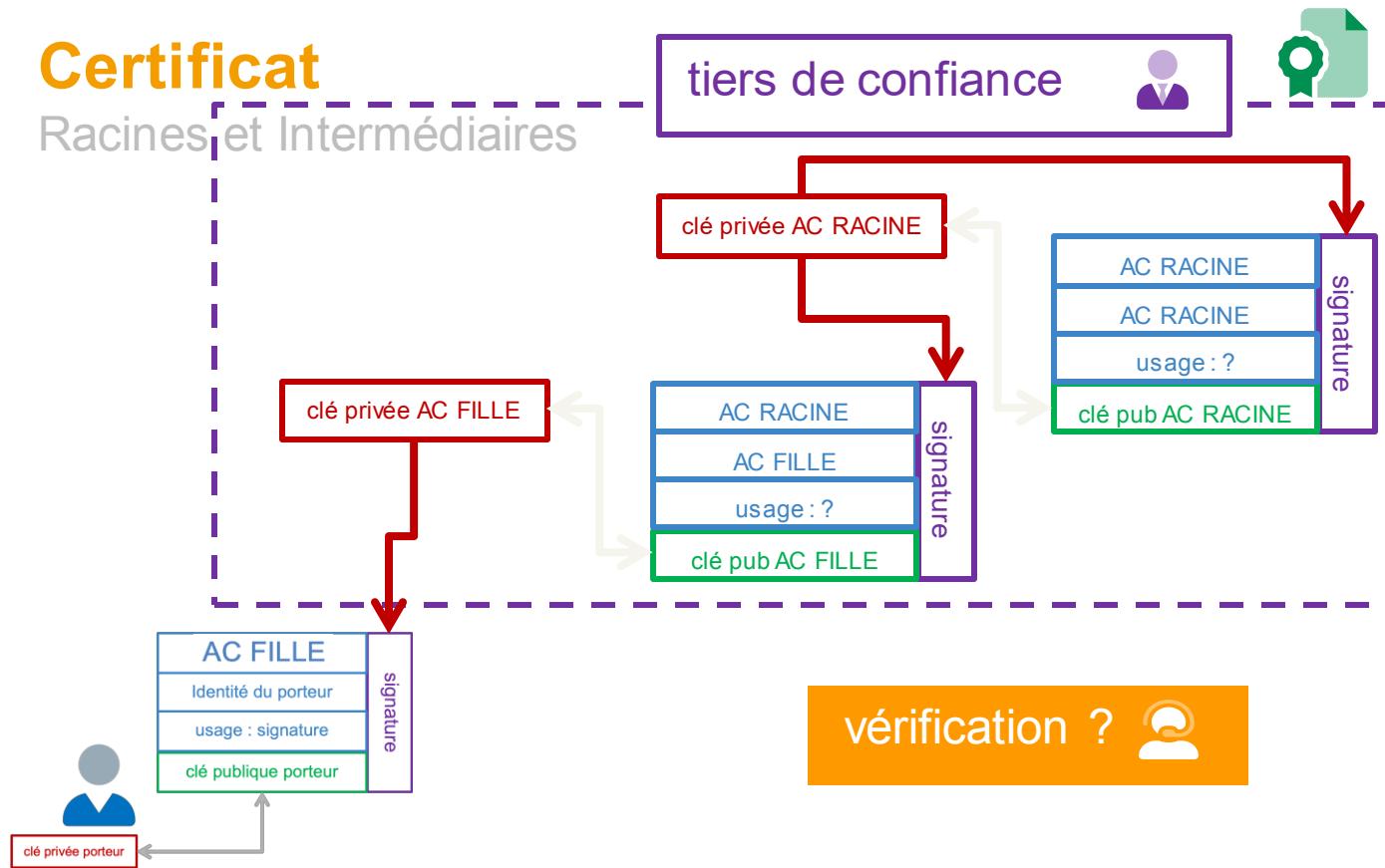


- eIDAS - RGS 2.0
 - eIDAS : Délivrance de certificats qualifiés - Audit ETSI 319 401 / 411-1&2 / 412
 - RGS : Délivrance de certificats qualifiés - Audit RGS – Annexes A2 / A3 / etc.



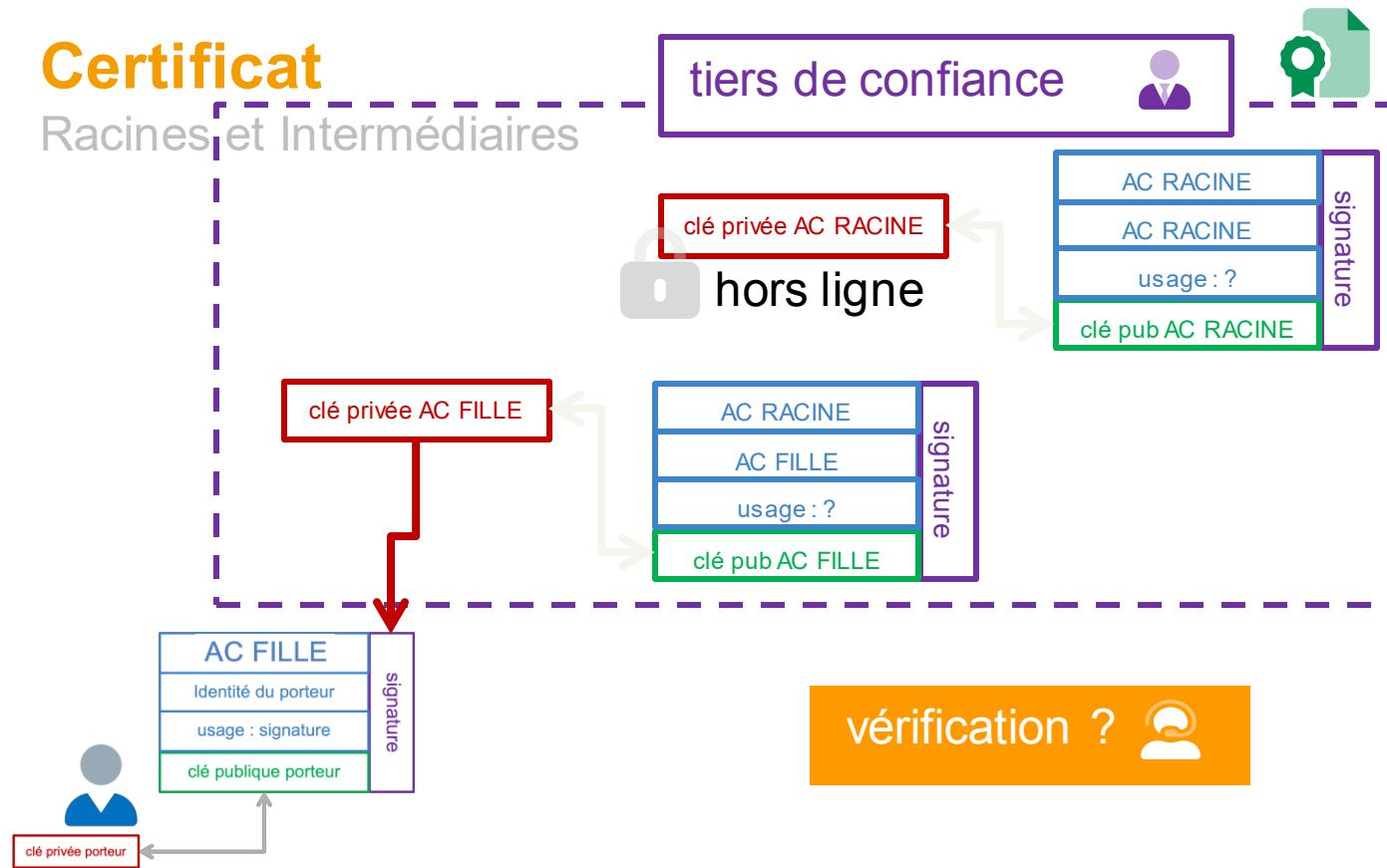
Certificat

Racines et Intermédiaires



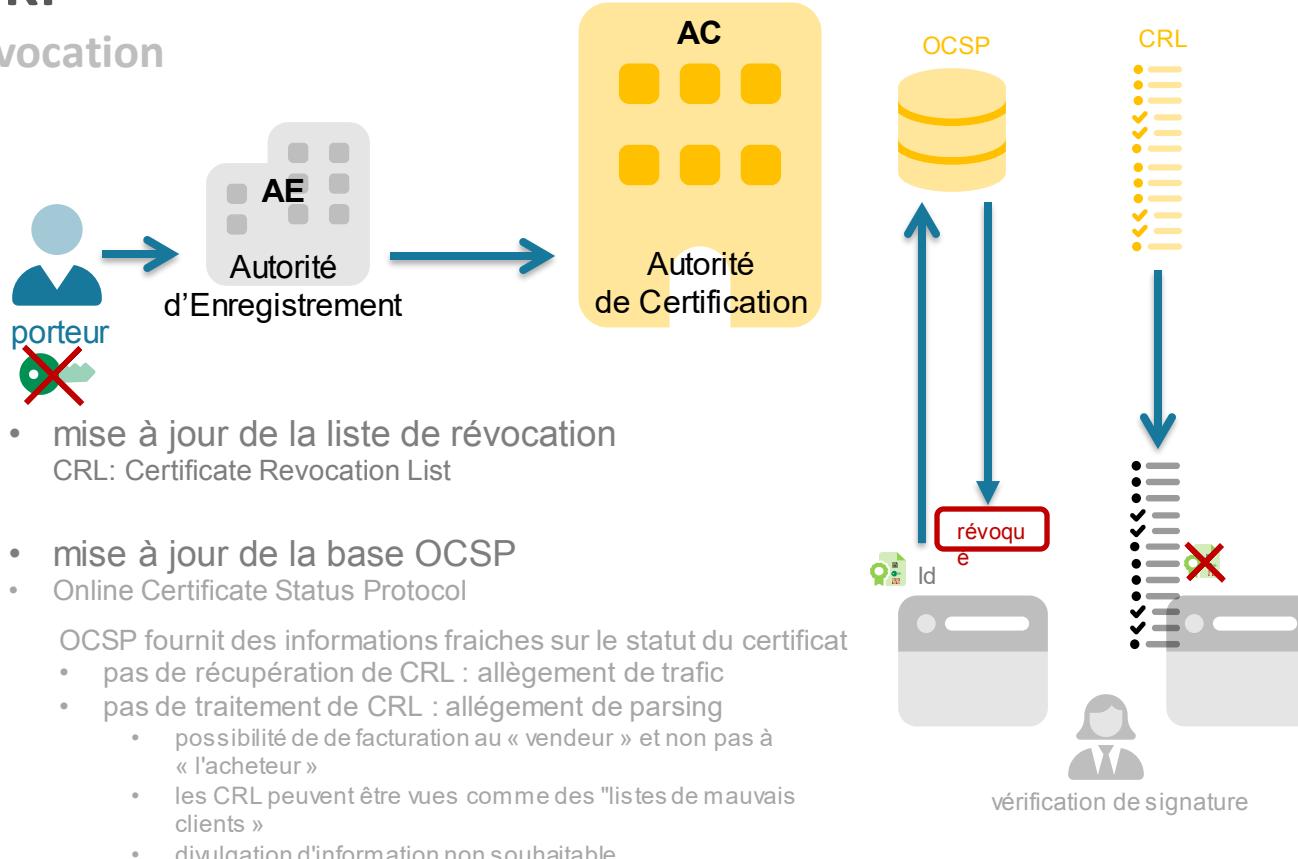
Certificat

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 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.