Cryptographie
Chiffrement

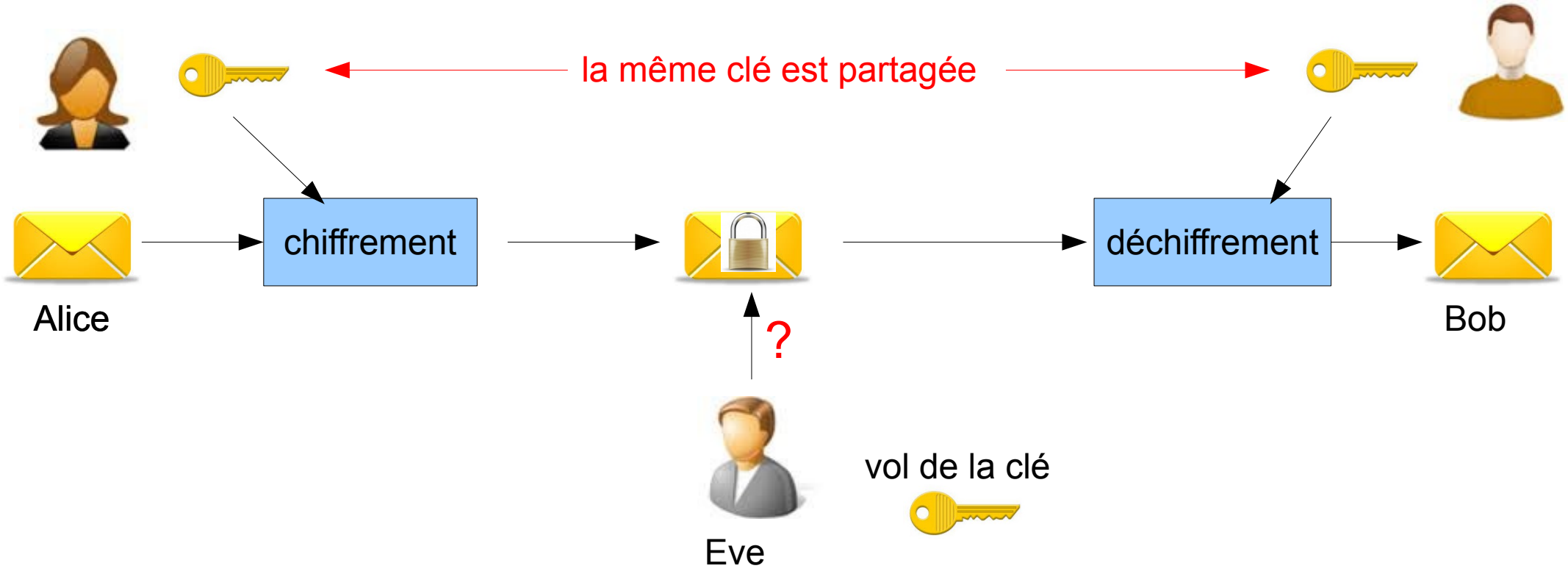
V 2.0

Chiffrement symétrique

- Chiffrement par substitution
  - remplacement dans un message une ou plusieurs entités par une ou plusieurs autres
    - chiffrement de César : décalage de 3 caractères vers la gauche
      - dans "L'odyssée de l'espace" IBM → HAL
    - chiffrement ROT13 : décalage de 13 caractères sur les 26 lettres de l'alphabet
      - deux rotations successives redonnent le texte initial
- Chiffrement par transposition
  - les entités à chiffrer sont ré-arrangées pour les rendre visuellement incompréhensibles

Chiffrement symétrique

- La même clé est utilisée pour le chiffrement et le déchiffrement



Chiffrement symétrique

- Quelques algorithmes de clés symétriques
  - AES (Advances Encryption Standard)
    - blocs de 128 bits chiffrés avec une clé de 128, 192 ou 256 bits
  - DES (Data Encryption Standard)
    - chiffre des blocs de 64 bits avec une clé de 56 bits
  - Triple DES
    - 112/128 bits
  - RC2, RC4, RC5
    - jusqu'à 2048 bits
  - pour une liste plus exhaustive :
    - *http://fr.wikipedia.org/wiki/Catégorie:Algorithme_de_cryptographie_symétrique*

Chiffrement symétrique

- Comparaison des forces relatives des algorithmes de chiffrement

Type	Année	Sécurité	Taille des clés	Performance	Variante
Idea	1992	4 sur 5	128 bits	2 sur 5	FOX
Blowfish	1993	5 sur 5	32 à 448 bits	5 sur 5	TWOFISH
3DES	1979	4 sur 5	56, 112, 168 bits	1 sur 5	
CAST-5	1996	5 sur 5	40 à 128 bits	4 sur 5	CAST-256
AES	1998	5 sur 5	128, 192, 256 bits	4 sur 5	

année : 2012

Chiffrement symétrique

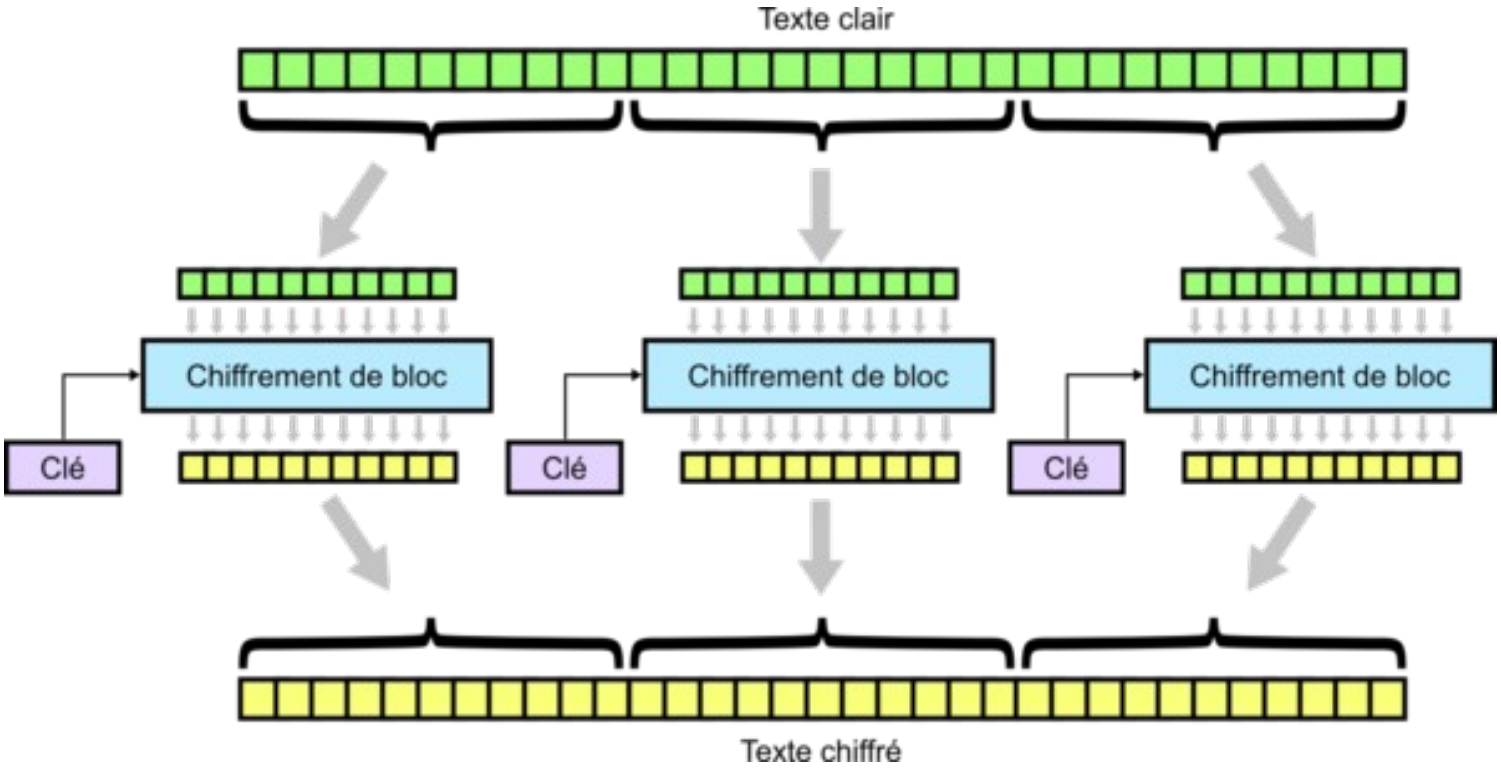
- Les algorithmes de chiffrement chiffrent les messages
  - par flux
    - stream cipher
    - traitement de message de longueur quelconque
  - par blocs
    - block cipher
    - le message est découpé en blocs de taille fixe
      - entre 32 et 512 bits
    - les blocs sont traités les uns après les autres
      - **mode d'opération**

Chiffrement symétrique

- Le chiffrement est constitué
  - d'un algorithme
    - comment un bloc est chiffré
    - AES, DES, RSA, …
  - d'un mode d'opération
    - comment les blocs sont traités entre eux
    - ECB, CBC, CFB, …
  - un padding
    - comment un bloc incomplet, par rapport à la taille du bloc à traité, est complété
      - sans, PKCS5 padding, ...

Chiffrement symétrique

- mode d'opération ECB - dictionnaire de codes



source : Wikipedia

Chiffrement symétrique

- Fonctionnement de ECB
  - chiffrement
    - divisions du text plain en blocs de taille fixe
      - 128 bits pour AES
    - chiffrement de chaque bloc en utilisant la même clé
    - concaténation du texte chiffré
  - le déchiffrement reprend le même processus, mais inversé

Chiffrement symétrique

- ECB
  - chaque bloc est chiffré séparément des autres blocs
  - des informations sur la structure du message à chiffrer peuvent transparaître
    - exemple : le textplain est un fichier Word => en-tête du fichier connue

message à chiffrer

```
000102030405060708090A0B0C0D0E0F
000102030405060708090A0B0C0D0E0F
```

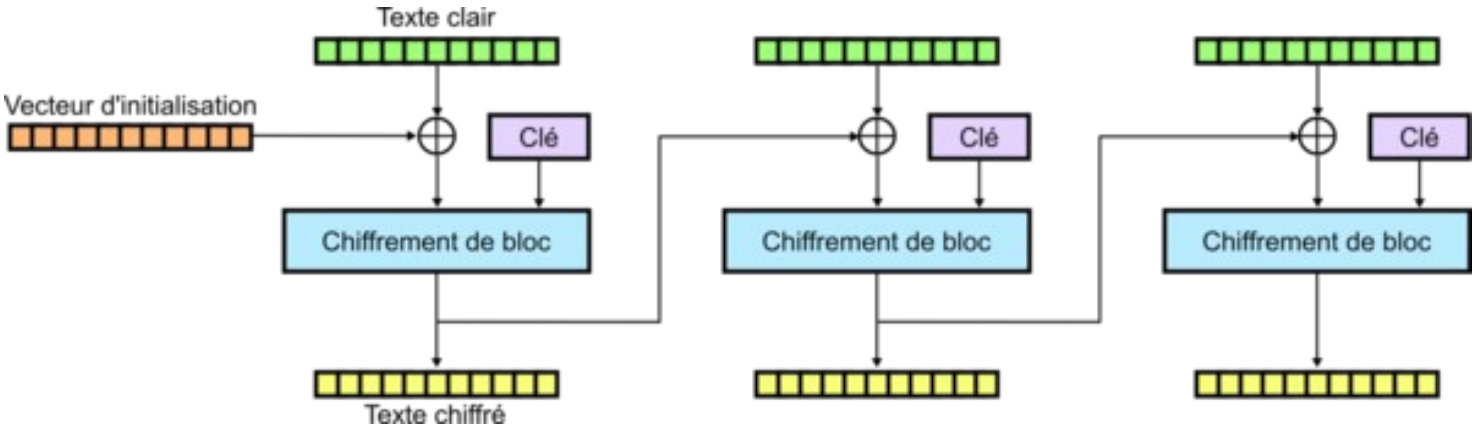
message chiffré

```
1B3E0EA1B1215696DEE00C90682D7352
1B3E0EA1B1215696DEE00C90682D7352
870DD2F19B55221ADB210B9294DB46CC
```

Chiffrement symétrique

- La plupart des modes d'opération ajoutent à chaque bloc le résultat du chiffrement du bloc précédent
- Pour le chiffrement du premier bloc il faut fournir un vecteur d'initialisation (IV - Initialisation Vector)

CBC



source : Wikipedia

Chiffrement symétrique

- Vecteur d'initialisation
  - Alice et Bob doivent s'entendre
    - sur le secret à partager (la clé symétrique)
    - sur le type d'algorithme et le mode d'opération
    - sur le vecteur d'initialisation
      - le vecteur n'a pas besoin d'être secret, il peut être envoyé en clair
      - il doit être différent pour chaque conversation
      - il ne doit pas être prédictible
        - pas de valeur constante, ni nulle
  - création du vecteur
    - utiliser un générateur de nombres aléatoires

Chiffrement symétrique

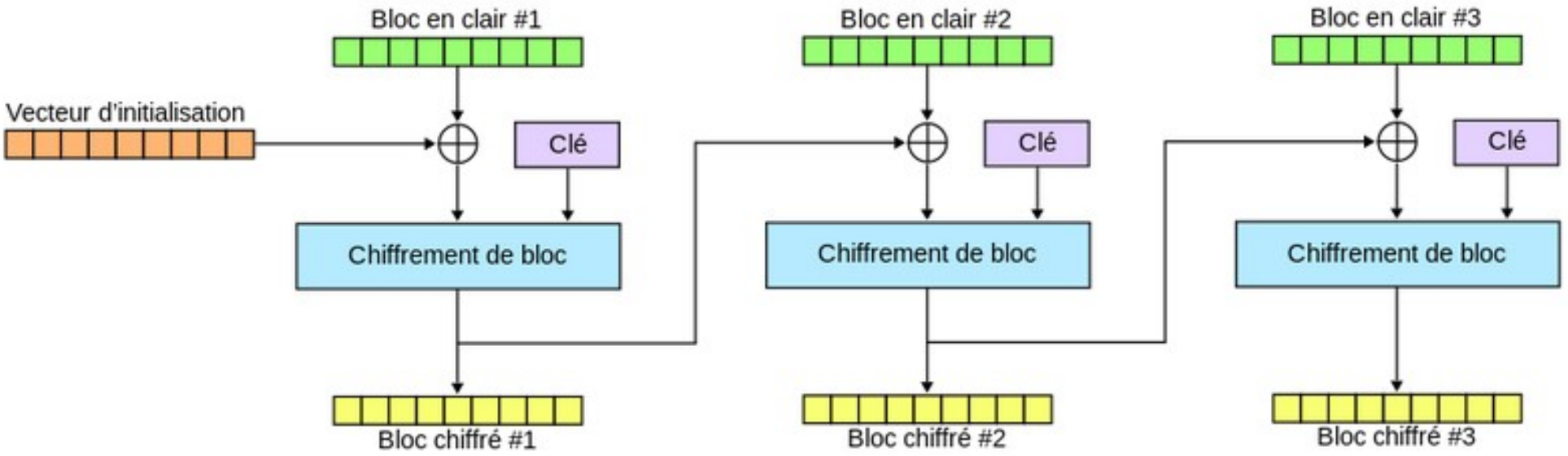
- Modes d'opération
  - ECB : Electronic Code Book
  - CBC : Cipher Block Chaining
  - CFB : Cipher FeedBack
  - OFB : Output FeedBack
  - CTR : CounTeR
  - CTS : Cipher Text Stealing
  - PCBC : Propagating Cipher Block Chaining
  - ...

Chiffrement symétrique

- Modes d'opérations préconisés par le NIST
  - National Institute of Standard and Technology
  - CBC : Cipher Block Chaining
  - CTR : CounTeR mode encryption
  - GCM : Galois Counter Mode

Chiffrement symétrique

- CBC - Cipher Block Chaining



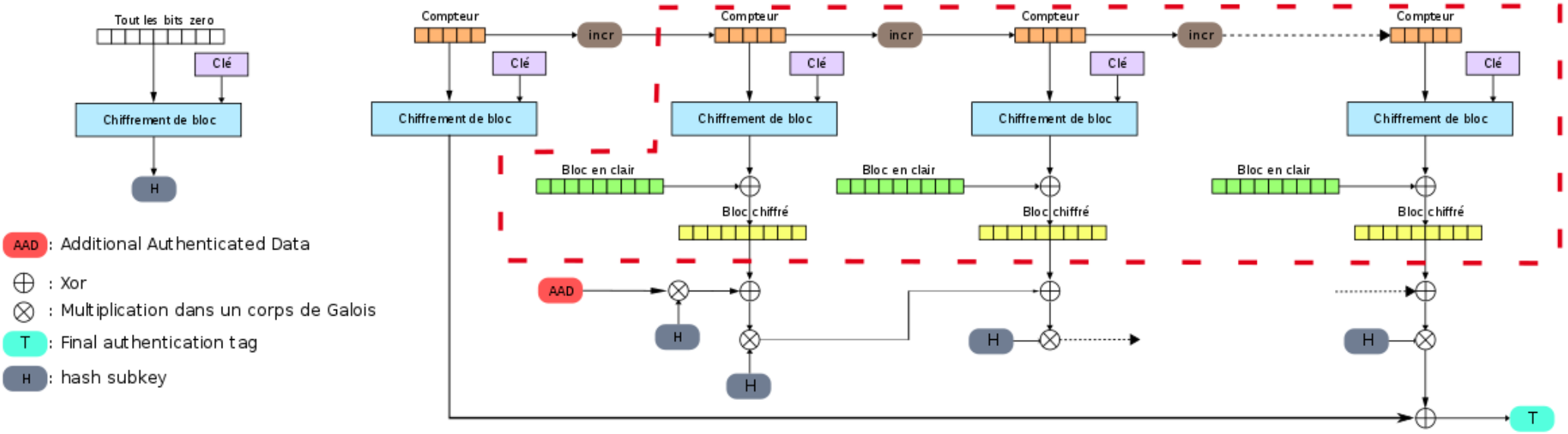
source : Wikipedia

Chiffrement symétrique

- CBC - Cipher Block Chaining
  - chiffrement
    - un vecteur d'initialisation IV aléatoire est généré
      - IV est de la même taille que le bloc à chiffrer et est utilisé sur le premier bloc
    - XOR entre le premier bloc en clair et IV
      - le résultat est ensuite chiffré avec la clé de chiffrement
    - chiffrement des blocs suivants
      - pour chaque bloc en clair, un XOR est appliqué entre le bloc chiffré précédent et le bloc en clair courant
      - le résultat est ensuite chiffré avec la clé de chiffrement
    - le processus est répété pour tous les blocs en clair
    - les blocs chiffrés sont concaténés pour fournir le text chiffré final

Chiffrement symétrique

- GCM - Galois/Counter mode
  - mode de chiffrement par blocs et d'authenticité de message
    - combine à la fois le chiffrement et l'authenticité



source : Wikipedia

Chiffrement symétrique

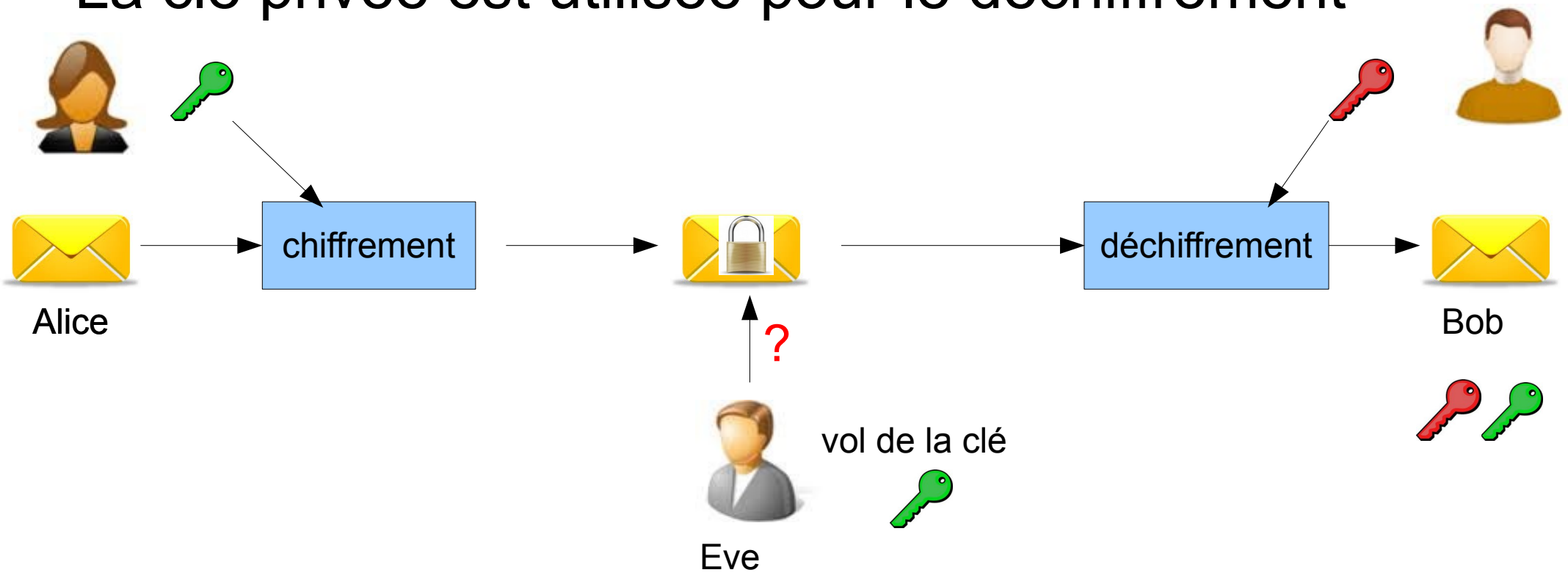
- GCM - Galois/Counter mode
  - chiffrement
    - pour chaque bloc GCM utilise un compteur unique pour générer un nonce (number used once)
    - nonce est combiné avec la clé, ce qui permettra de chiffré le bloc en clair
  - authentification
    - GCM utilise MAC (Message Authentication Code) généré à partir du texte chiffré et est utilisé pour l'intégrité
  - le résultat final est texte chiffré authentifié qui reprend à la fois les données chiffrées et le tag d'authentification généra par MAC

Chiffrement symétrique

- Dans le chiffrement symétrique on à donc
  - l'algorithme de chiffrement (AES)
  - le mode d'opération de chiffrement sur les blocs (ECB, …)
- Dans AES le clé de chiffrement est utilisée pour chiffrer/déchiffrer les données
  - la clé est utilisée avec un mode d'opération pour chiffrer des blocs de données
  - chaque bloc est chiffré/déchiffré en utilisant l'algorithme AES

Chiffrement asymétrique

- La clé publique est diffusée, et est utilisée pour le chiffrement
- La clé privée est utilisée pour le déchiffrement



Chiffrement asymétrique

- La clé publique est connue de tous
  - elle ne permet pas de deviner la clé privée

- Le chiffrement asymétrique repose sur des fonctions mathématiques
  - à sens unique
  - ou sens unique avec trappe

Chiffrement asymétrique

- Fonction à sens unique
  - fonction facilement calculable, mais dont la réciproque est en pratique impossible à calculer par manque de ressource (temps, puissance,...)
    - problème de factorisation
  - soit deux grands nombres premiers X et Y
    - le produit est X * Y est simple
    - mais retrouver à partir du produit les deux nombres X et Y est complexe

Chiffrement asymétrique

- Quelques algorithmes de chiffrement
  - RSA (Rivest - Shamir - Adelman)
    - un des plus populaire
    - utilisé aussi bien pour le chiffrement que pour la signature
    - problème d'authentification de l'envoyeur : comment prouver que c'est bien Alice qui envoie le message
  - GPS (Girault - Poupard - Stern)

Chiffrement asymétrique

- PKCS : Public-Key Cryptography Standards
  - ensemble de spécifications des laboratoires RSA
    - organisme privé (propriétaire de l'algorithme RSA)
  - certaines PKCS ont été retranscrites dans des RFC
  - les plus connues
    - PKCS#10 - standard de requête de certificat
    - PKCS#12 - format de fichier utilisé pour stocker une clé privée et le certificat de clé publique avec protection par mot de passe
  - cf. https://fr.wikipedia.org/wiki/Public_Key_Cryptographic_Standards

Chiffrement symétrique vs asymétrique

- Chiffrement symétrique
  - calculs plus rapides
  - clé de taille plus petite
  - chiffrement des communications (données, voix)
    - important volume de données
- Chiffrement asymétrique
  - chiffrement de messages de petites tailles
    - envoie d'une clé publique
  - signature

Chiffrement hybride

- Chiffrement symétrique
  - rapide mais comment échanger le secret
- Chiffrement asymétrique
  - sûr, mais lent
- Les protocoles PGP ou Diffie-Hellman tentent de résoudre ses problématiques

Diffie-Hellman

- Objectif : créer un secret commun, puis utiliser ce secret pour chiffrer symétriquement les échanges
  - le postulat de base est le suivant
    - étant donné des entiers *p*, *a*, *x* avec *p* premier et $1 \leq a \leq p-1$
      - il est aisé de calculer l'entier $y = a^x \left(mod\ p \right)$
      - si on connaît $y = a^x \left(mod\ p \right)$, *a* et *p*, il est très difficile de retrouver *x*, si *p* est assez grand
        - résolution du logarithme discret, pas d'algorithme efficace
        - fonction à sens unique

Diffie-Hellman

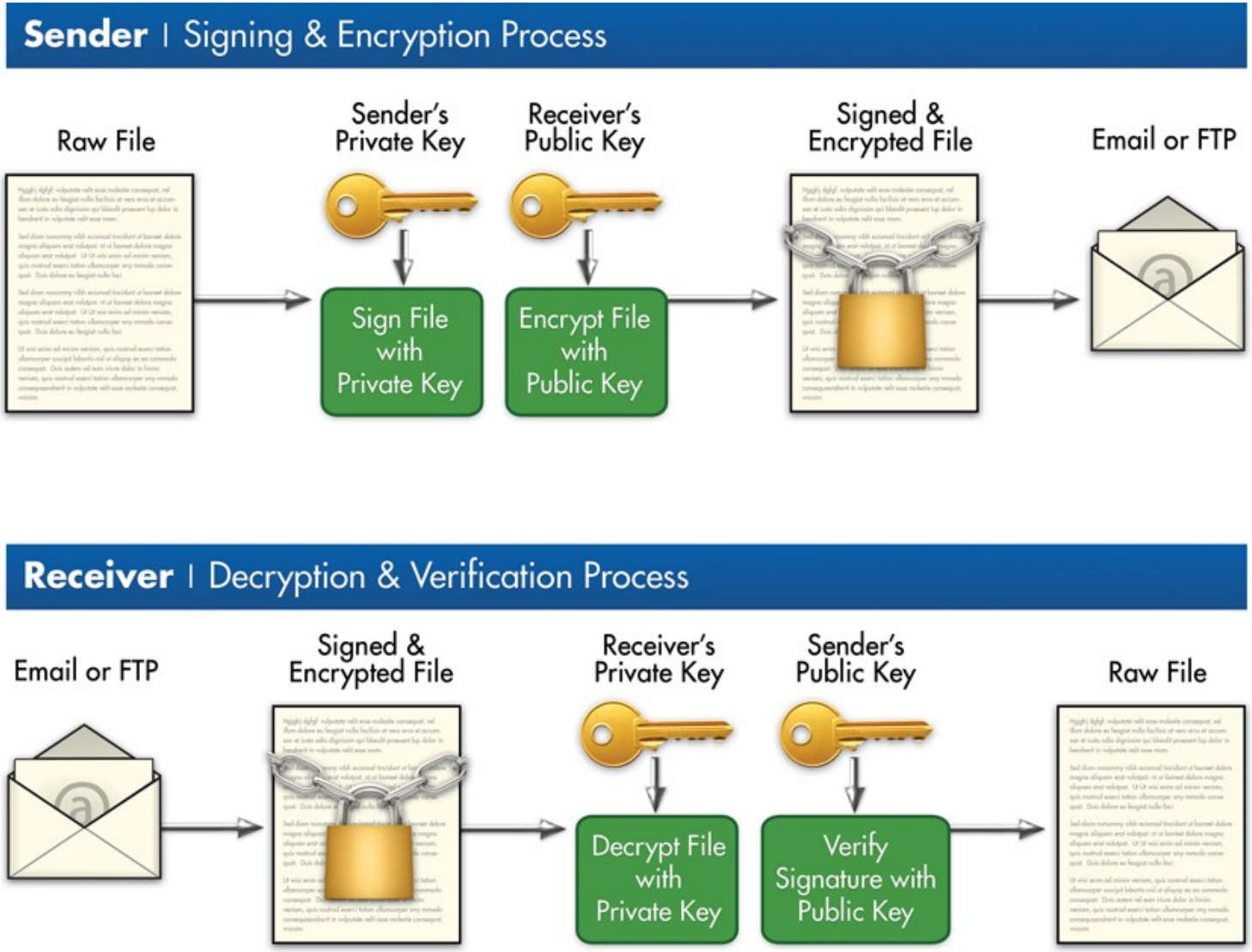
- Étapes d'échange de la clé secrète *K*

	Alice	Bob
Étape 1	Alice et Bob choisissent ensemble un nombre entier ***p*** et entier ***a***	
Étape 2	Alice choisit secrètement **x1**	Bob choisit secrètement **x2**
Étape 3	Alice calcule $y_1 = a^{(x_1)} (mod\ p)$	Bob calcule $y_2 = a^{(x_2)} (mod\ p)$
Étape 4	Alice et Bob s'échangent ***y1*** et ***y2*** (échange non sécurisé)	
Étape 5	Alice calcule $K = y_2^{(x_1)} = (a^{(x_2)})^{(x_1)} = a^{(x_1 x_2)} (mod\ p)$	Bob calcule $K = y_1^{(x_2)} = (a^{(x_1)})^{(x_2)} = a^{(x_1 x_2)} (mod\ p)$

PGP

- Pretty Good Privacy
  - créé par Phil Zimmermann en 1991
  - garantit la confidentialité et l'authentification
    - permet d'authentifier l'expéditeur
    - le message est chiffré avec une clé symétrique
    - la clé symétrique est chiffrée avec la clé privée de l'expéditeur
      - elle sera déchiffrée par le destinataire avec la clé publique de l'expéditeur

PGP



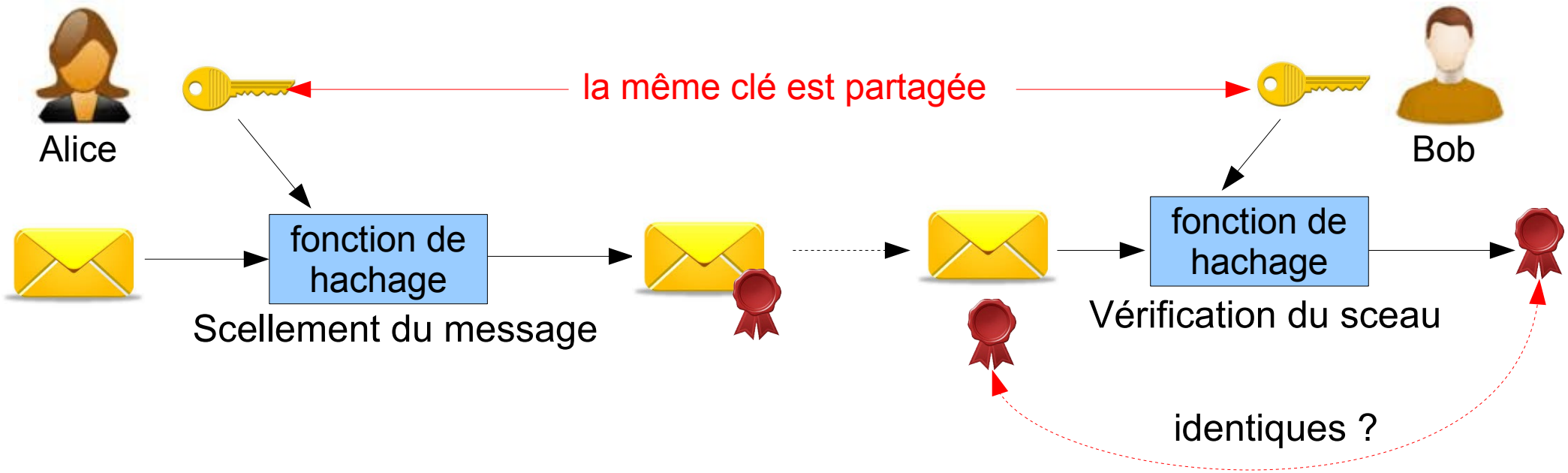
source : http://www.andretimokhin.com/

Intégrité du message

- MAC - Message Authentication Code
  - même idée que la fonction de hachage, mais avec authentification
    - la clé secrète partagée permet d'authentifier l'expéditeur
  - assure l'intégrité du message + authentification
  - MAC = H(key + message)
    - H est un algo de hachage (MD5, SHA1 , …)
    - collisions possibles
      - H(message1) == H(message2)
    - ce qui implique
      - H(key + message1) == H(key + message2)

Intégrité du message

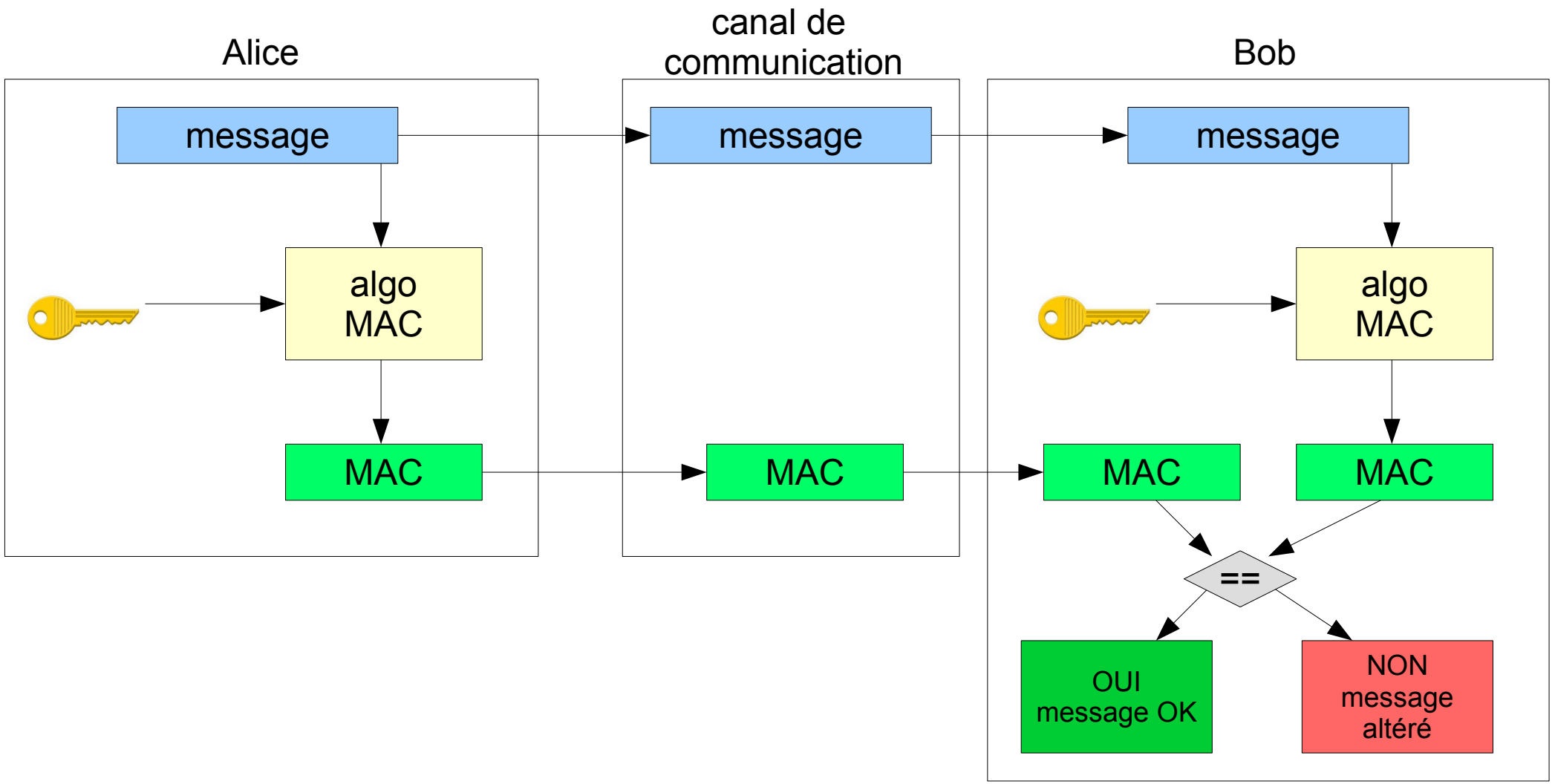
- Scellement - symétrique
  - utilisation d'une fonction de hachage sur un message à l'aide d'une clé secrète
  - résultat de la fonction de hachage = sceau



Intégrité du message

- Signature numérique - asymétrique
  - l'utilisation de la fonction de hachage permet de vérifier l'intégrité
  - il faut s'assurer aussi de l'identité de l'expéditeur
  - l'expéditeur chiffre le hash avec sa clé privée
    - ce qui correspond à la signature
  - le destinataire déchiffre le sceau avec la clé publique de l'expéditeur
    - l'empreinte recalculée est alors comparée à l'empreinte reçue

Intégrité du message



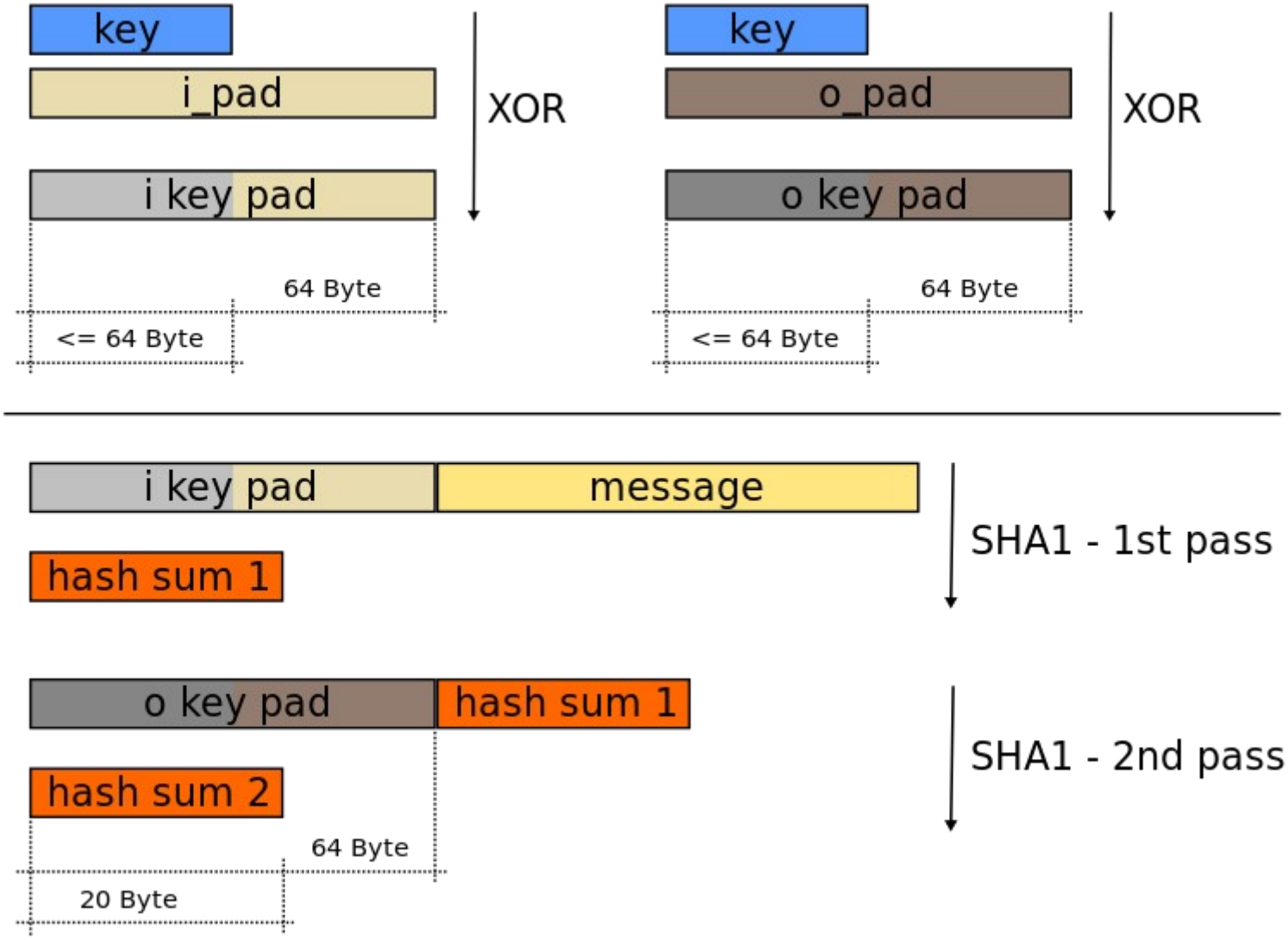
Intégrité du message

- HMAC - keyed-Hash Message Authentication Code
  - un HMAC est calculé en utilisant une fonction de hachage cryptographique (SHA256 par exemple) avec une clé secrère
  - seuls les participants de la conversation connaissent la clé secrète

Intégrité du message

- HMAC - keyed-Hash Message Authentication Code
  - code d'authentification d'empreinte de message avec clé
    - HMAC est MAC spécifique, avec deux passages d'un algo
  - HMAC résout le problème des collisions par la construction suivante
    - $HMAC_k(m) = h((k \oplus opad) \,||\, h((k \oplus ipad) \,||\, m))$
      - opad : répétition du caractère 0x36
      - ipad : répétition du caractère 0x5C
        - opad et ipad sont de la taille d'un bloc
          - si la taille du bloc est de 512 bits => 64 répétitions de l'octet
      - || : concaténation
      - $\oplus$: XOR

Intégrité du message



source : wikimedia

Intégrité du message

- CMAC - Cipher-based MAC
  - code d'authentification de message utilisant un algo de chiffrement au lieu d'un fonction de hachage
  - permet d'assurer l'intégrité et l'authenticité d'un message
  - utilise AES

Ressources

- Cryptographie appliquée
  - auteur : Bruce Schneier
  - éditeur : Thomson Publishing