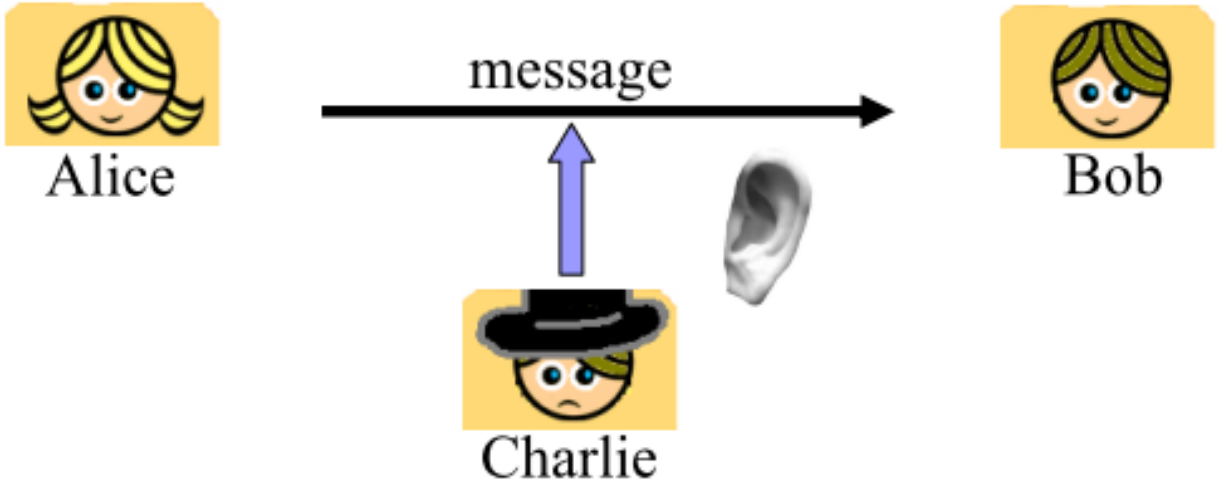
Chapitre 3

# INTÉGRITÉ ET AUTHENTIFICATION

# Introduction

- **Services souhaités par la cryptographie**

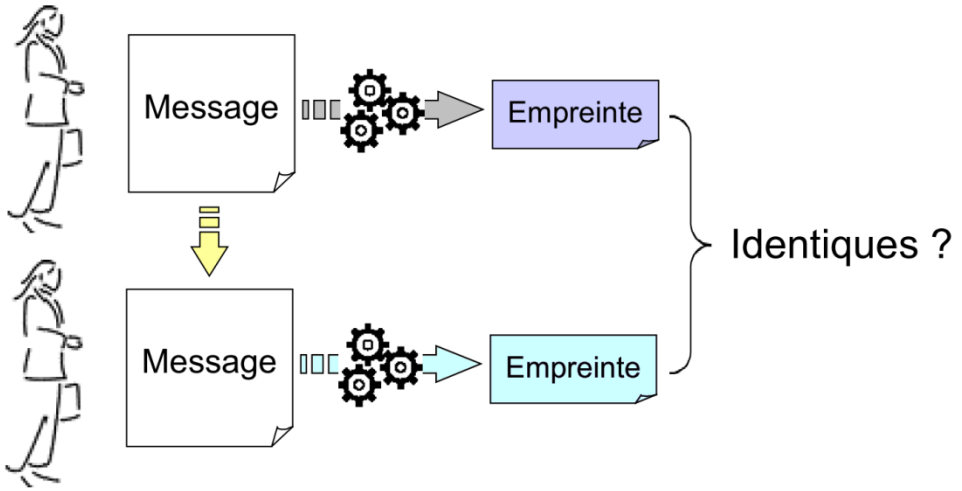



- Confidentialité :
  - Rendre le message secret entre deux tiers
- Authentification :
  - Le message émane t-il de l'expéditeur anoncé ?
- Intégrité :
  - Le message a-t-il été modifié durant le transfert ?

# Vérification d'intégrité

- **Définition**
  - Garantir l'intégrité d'un message, c'est vérifier que le message envoyé n'a pas été altéré (intentionnelement ou de manière fortuite) durant la transmission.

- **Principe**
  - Le destinataire calcule le condensé (ou empreinte) du message reçu et le compare avec le condensé accompagnant le message envoyé par l'expéditeurs.




  - Le destinataire et l'expéditeur utilise la même fonction de hachage.
  - Si les deux condensés sont différents, alors le message a été falsifié durant la communication

# Authentification

- **Définition**
  - Action de s'assurer de l'identité d'une entité donnée ou de l'origine d'une communication ou d'un fichier.

- **Remarques**
  - Authentification de l’origine des données et intégrité sont inséparables
  - Authenticité = authentification + intégrité
  - « authentification » souvent utilisé pour désigner en fait l’authenticité

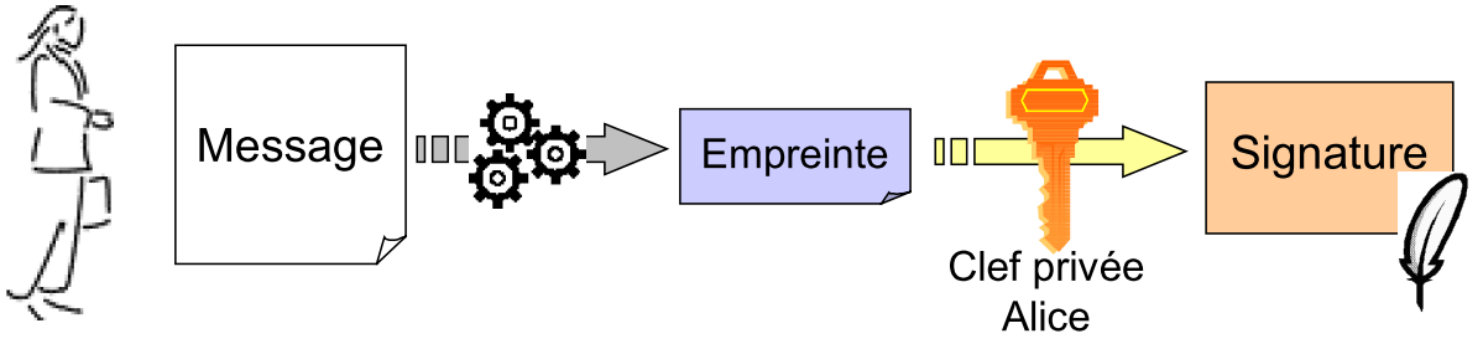
# Signature

- **Définition**
  - La **signature électronique** (ou *signature numérique*) est un procédé permettant de :
    - garantir l'**authenticité** de l'expéditeur,
    - de vérifier l'**intégrité** du message reçu.
  - Elle assure également une fonction de non-répudiation qui permet d'assurer que l'expéditeur a bien envoyé le message.

# Signature

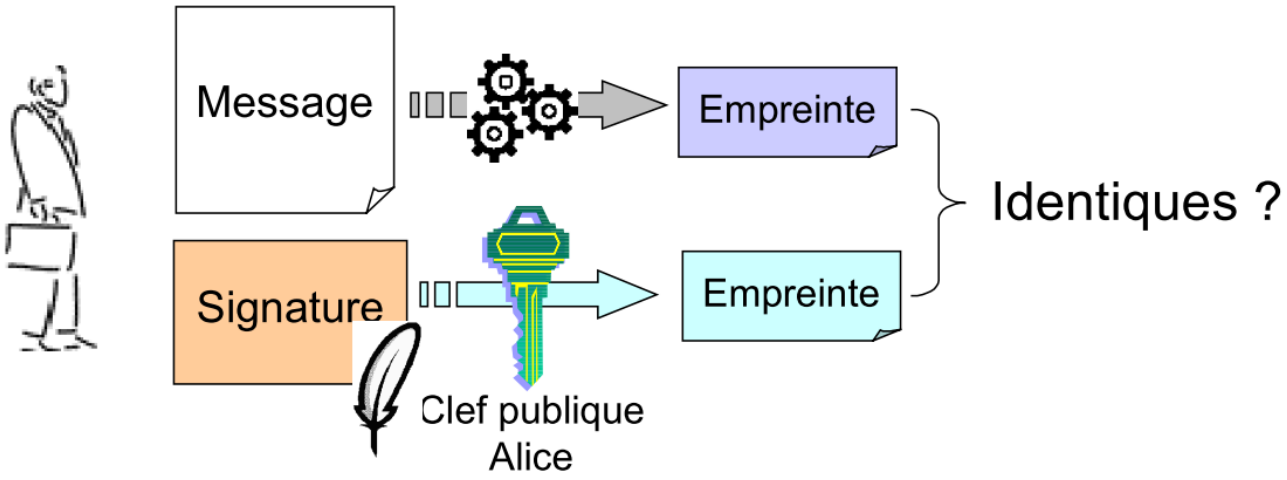
- **Principe de la signature**
  1. La **signature électronique** utilise une fonction de hachage permettant d'obtenir un condensé (appelé **empreinte**) d'un texte, c'est-à-dire une suite de caractères assez courte représentant le texte qu'il condense.

  2. Ensuite, l'empreinte est chiffrée avec la clé privée de l'émetteur : on obtient alors la **signature électronique**.



# Signature

- **Principe de la vérification**
  - Pour vérifier que la correspondance entre un message et une signature donnée, il suffit de vérifier l'égalité entre :
    1. L'empreinte du message, en utilisant la fonction de hachage qui a été utilisée pour calculer la signature,
    2. Et l'empreinte issue du déchiffrement de la signature avec la clé publique de l'émetteur de la signature.



# Signature

- **Propriétés d'une signature**
  1. Elle ne peut être contrefaite
  2. Elle n'est pas réutilisable
  3. Un message signé est inaltérable
  4. La signature ne peut être reniée

- **Remarques**
  - Sur un support électronique :
    - La signature doit dépendre du message,
    - sinon il y a risque de copie et/ou de réemploi

  - **Signer ≠ chiffrer !**

# Signature

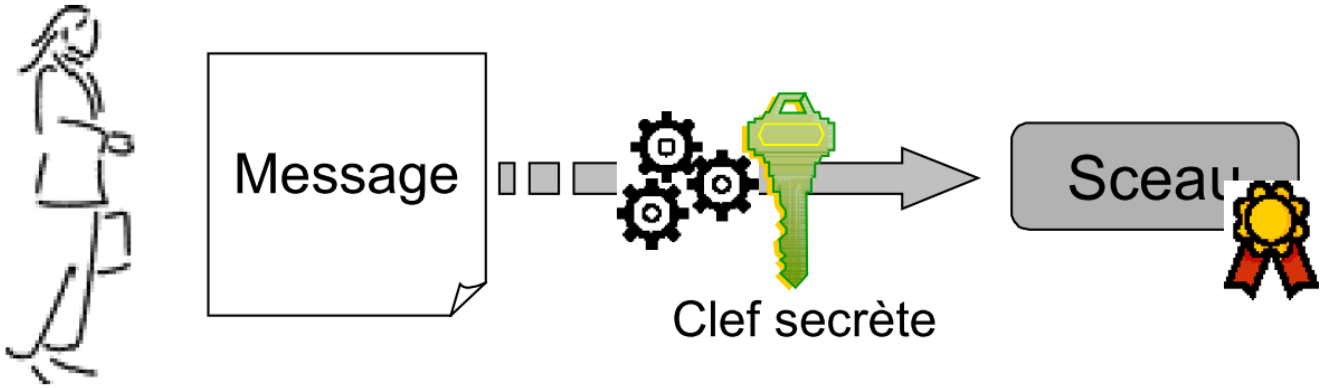
- **Algorithmes les plus couramment utilisés**
  - **DSA (Digital Signature Algorithm) :**
    - Algorithme de signature standardisé par le NIST aux Etats-Unis
    - Fonction de hachage : SHA-1 (empreinte sur 160 bits)
    - Chiffrement asymétrique : ElGamal
  - **RSA :**
    - Norme de fait
    - Fonction de hachage : MD5 (empreinte sur 128 bits) ou SHA-1
    - Chiffrement asymétrique : RSA

# Scellement

- **Définition**
  - **Le scellement des données** permet de garantir que le condensé utilisé pour vérifier l'intégrité des données a bien été envoyé par le bon expéditeur.
  - Pour cela, on utilise un chiffrement asymétrique pour chiffrer le condensé : le condensé est alors **signé**. Le condensé ainsi signé est appelé **sceau** ou **code d'authentification de message (*Message Authentification Code*, MAC)**.
- **Remarque**
  - Tout comme la signature électronique, le scellement fournit les services suivants :
    - L'**authentification** de l'origine des données,
    - L'intégrité des données.
  - Cependant, le scellement ne fournit pas la non-répudiation.

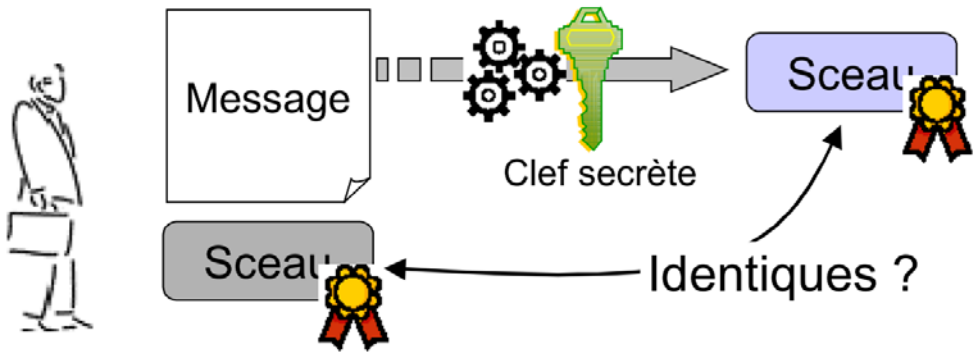
# Scellement

- **Principe du scellement**
  - La création du sceau se fait en deux étapes :
    1. Utilisation d'une fonction de hachage permettant d'obtenir un condensé du message
    2. Obtention du sceau par chiffrement du condensé à l'aide de la clé secrète (ou privée) de l'émetteur du message



# Scellement

- **Principe de vérification**
  - A la réception du message, il suffit au destinataire de :
    - Déchiffrer le sceau avec la clé publique de l'expéditeur,
    - Puis de comparer ce condensé avec celui obtenu à l'aide de la fonction de hachage (la même que celle utilisée par l'expéditeur) appliqué au message



# Certification

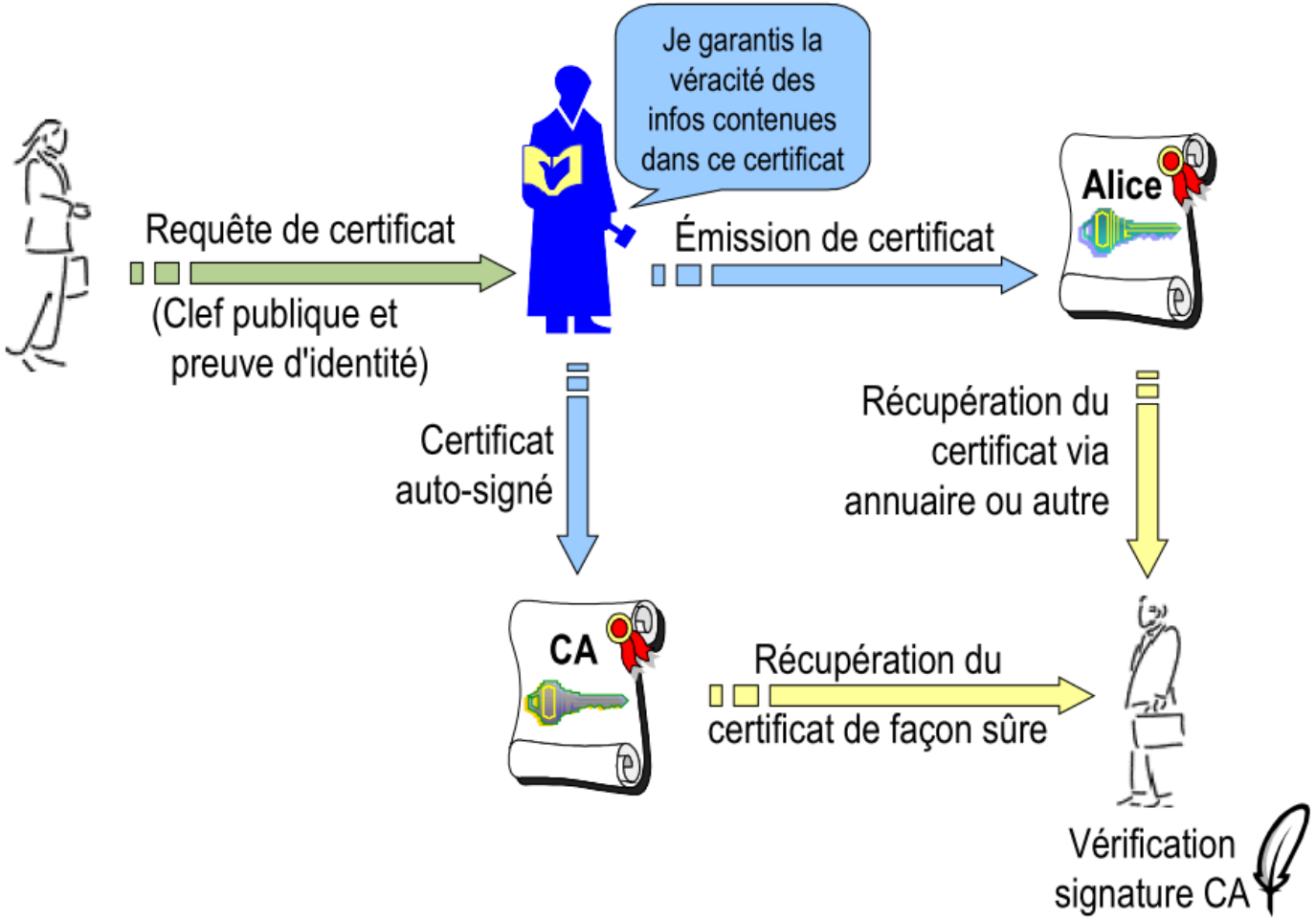
- **Définition**
  - Un **certificat** permet d'associer une clé publique à une entité, au moyen de la signature d'une autorité de confiance (appelé autorité de certification, souvent notée CA pour **Certification Authority**), afin d'en assurer la validité :
    - Nom du propriétaire de la clef
    - Dates de validité
    - Type d'utilisation autorisée
    - …

# Certification

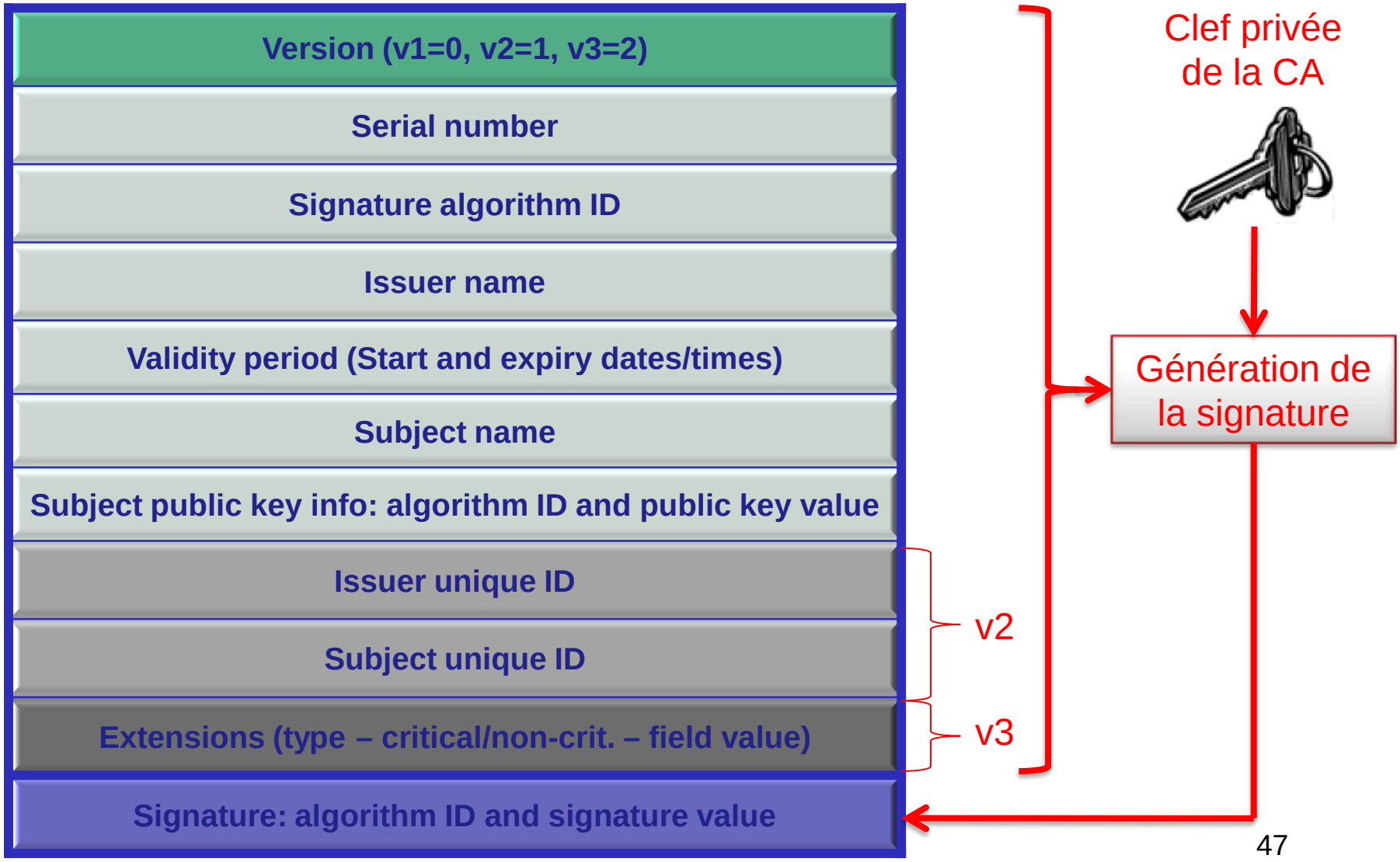
- **Emission et vérification des certificats**
  - Les certificats sont émis par une autorité de certification
    - ***Certificate Authority – CA***
    - Garantit l'exactitude des données (identification du propriétaire de la clef)
    - Certificats vérifiables au moyen de la clef publique de la CA (seule clef à stocker de façon sûre)
    - Format actuel du certificat : X.509v3, profil PKIX
  - Listes de révocation
    - ***Certificate Revocation LIST – CRL***
    - Permettre de révoquer des certificats avant leur expiration normale

# Certification

- **Emission et vérification des certificats**



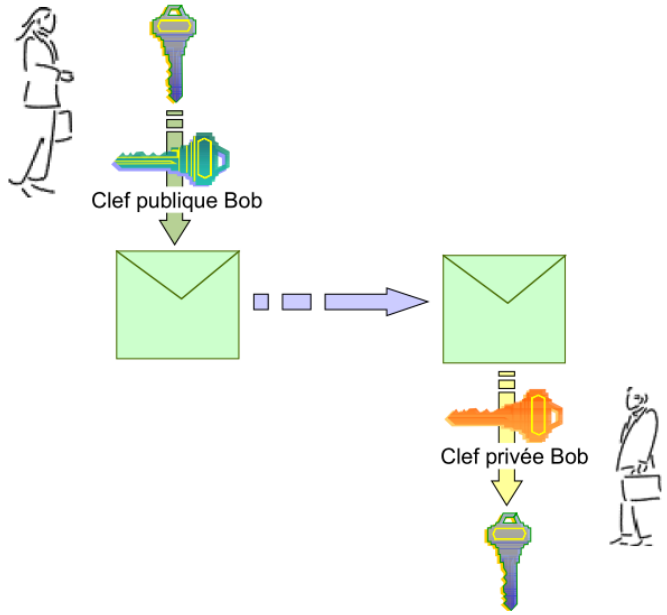
# Certificats X.509v3



# Clefs de session

- **Définition**
  - Un **clé de session** est une clé secrète utilisée pour chiffrer chaque jeu de données dans un système de transaction ou de communication.
  - Une clé de session différentes est utilisée pour chaque session de communication.

- **Principe**
  - On crypte une clé de session avec la clé publique du destinataire,
  - Le décryptage est fait avec sa clé privée,
  - Les échanges sont ensuite cryptés et décryptés à l'aide de la clé de session.

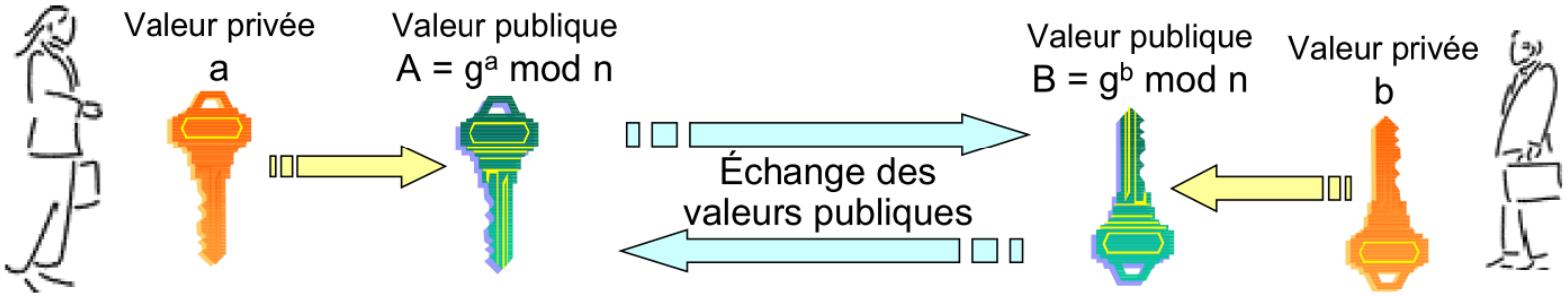


# Clefs de session

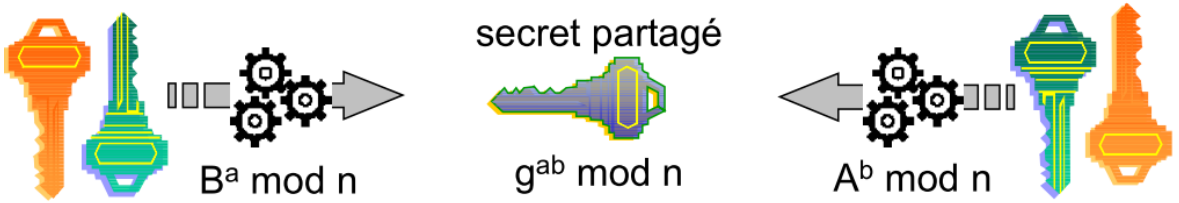
- **Intérêt**
  - Permet d'étendre l'authentification à l'ensemble de la comminucation, sans à avoir besoin d'envoyer la clé pour chaque message.
- **Problème**
  - L'échange de clefs doit être authentifié pour éviter les attaques
- **Solution**
  - Utilisation d'une protocole d'authentification mutuelle avec échange de clefs tout-en-un
- **Types d'échange de clefs**
  - Transport
    - Exemple : transport RSA (utilisé par SSL)
  - Génération
    - Exemple : Diffie-Hellman

# Clefs de session

- **Principe du protocole Diffie-Hellman**
  1. Echange de valeurs publiques




  2. Génération d'un secret partagé




  3. Un espion ne peut reconstituer le secret partagé à partir des valeurs publiques**.**

# Clefs de session

- **Propriétés du protocole Diffie-Hellman**
  - Problème : sensible à l'attaque de l'intercepteur
    - L'attaquant evoir sa valeur publique à la place de deux tiers en communication : il partage donc un secret avec chaque tiers
  - Solution : authentifier les valeurs publiques
    - Exemple : utilisation de certificats
  - Résultat : protocole Diffie-Hellman authentifié

- **Propriété de Perfect Forward Secrecy (PFS)**
  - La découverte du secret à long terme ne compromet pas les clefs de session
  - Car le secret à long terme n'intervient pas dans la génération ou la protection en confidentialité des clefs