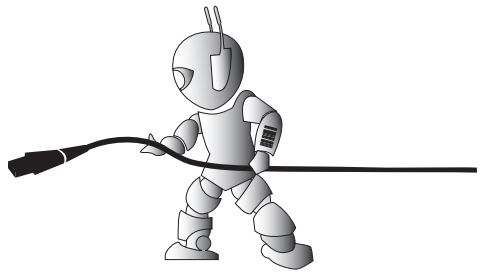
Chiffrement et authentification

Certificats et architecture PKI

Tuyêt Trâm DANG NGOC
<dntt@u-cergy.fr>

Université de Cergy-Pontoise

2012–2013



DÉPARTEMENT
Sciences Informatiques

Plan

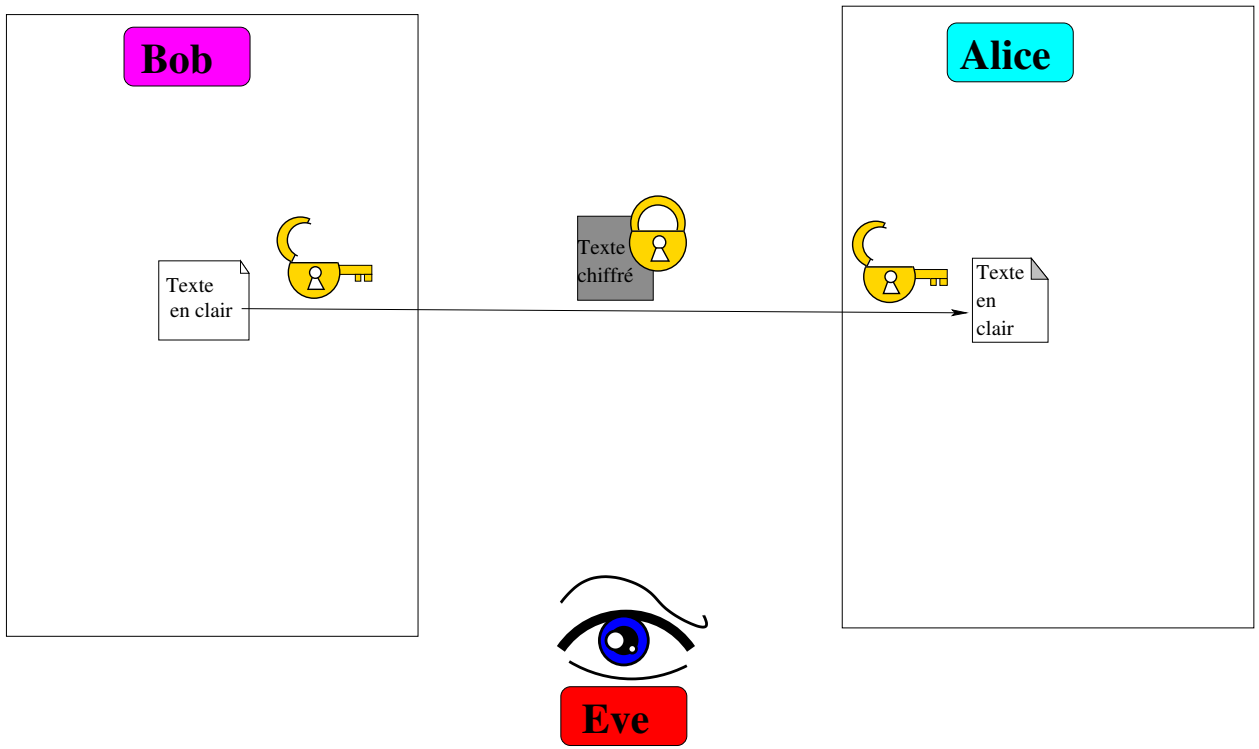
DÉPARTEMENT Sciences Informatiques

DÉPARTEMENT
Sciences Informatiques

Rappel des algorithmes cryptographiques

- Algorithme à clef symétrique
- Algorithme de Diffie-Hellman
- Algorithme à clef asymétrique
  - Confidentialité
  - Signature
- Hachage

Clef partagée entre les correspondants pour chiffrer/déchiffrer un message



DÉPARTEMENT Sciences Informatiques

Rappel des algorithmes cryptographiques

- Algorithme à clef symétrique
- Algorithme de Diffie-Hellman
- Algorithme à clef asymétrique
  - Confidentialité
  - Signature
- Hachage

Clef partagée entre les correspondants pour chiffrer/déchiffrer un message

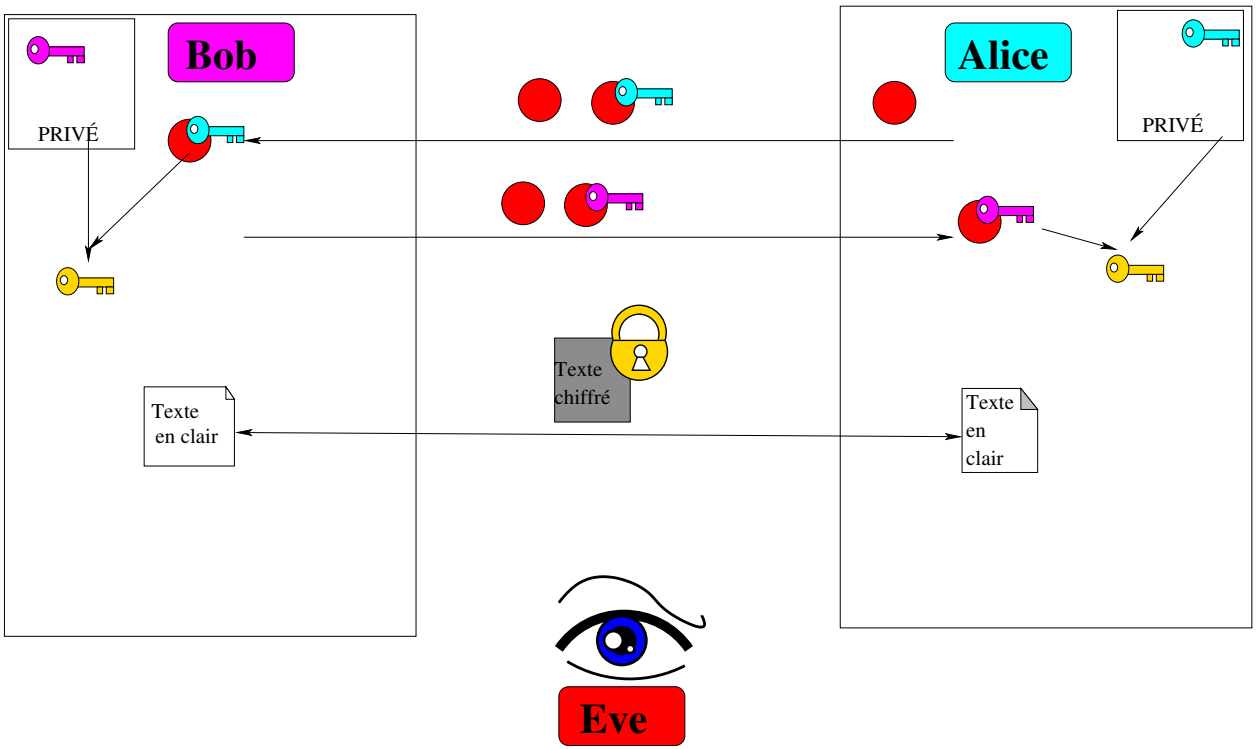
- Confidentialité avec clef partagée
- Rapide
- problème d'échange de la clef
- Chiffrement par bloc : DES, TDES, AES, IDEA, BlowFish, Serpent, TwoFish
- Chiffrement par flot : AR/1, RC4, Py, E0

DÉPARTEMENT Sciences Informatiques

Rappel des algorithmes cryptographiques

- Algorithme à clef symétrique
- Algorithme de Diffie-Hellman
- Algorithme à clef asymétrique
  - Confidentialité
  - Signature
- Hachage

Echange synchrone pour convenir d'une clef partagée



DÉPARTEMENT Sciences Informatiques

Rappel des algorithmes cryptographiques

- Algorithme à clef symétrique
- Algorithme de Diffie-Hellman
- Algorithme à clef asymétrique
  - Confidentialité
  - Signature
- Hachage

Echange synchrone pour convenir d'une clef partagée

- Pas de problèmes d'échange de clef
- nécessite la présence simultanée des deux partis

DÉPARTEMENT Sciences Informatiques

Rappel des algorithmes cryptographiques

- Algorithme à clef symétrique
- Algorithme de Diffie-Hellman
- Algorithme à clef asymétrique
  - Confidentialité
  - Signature
- Hachage

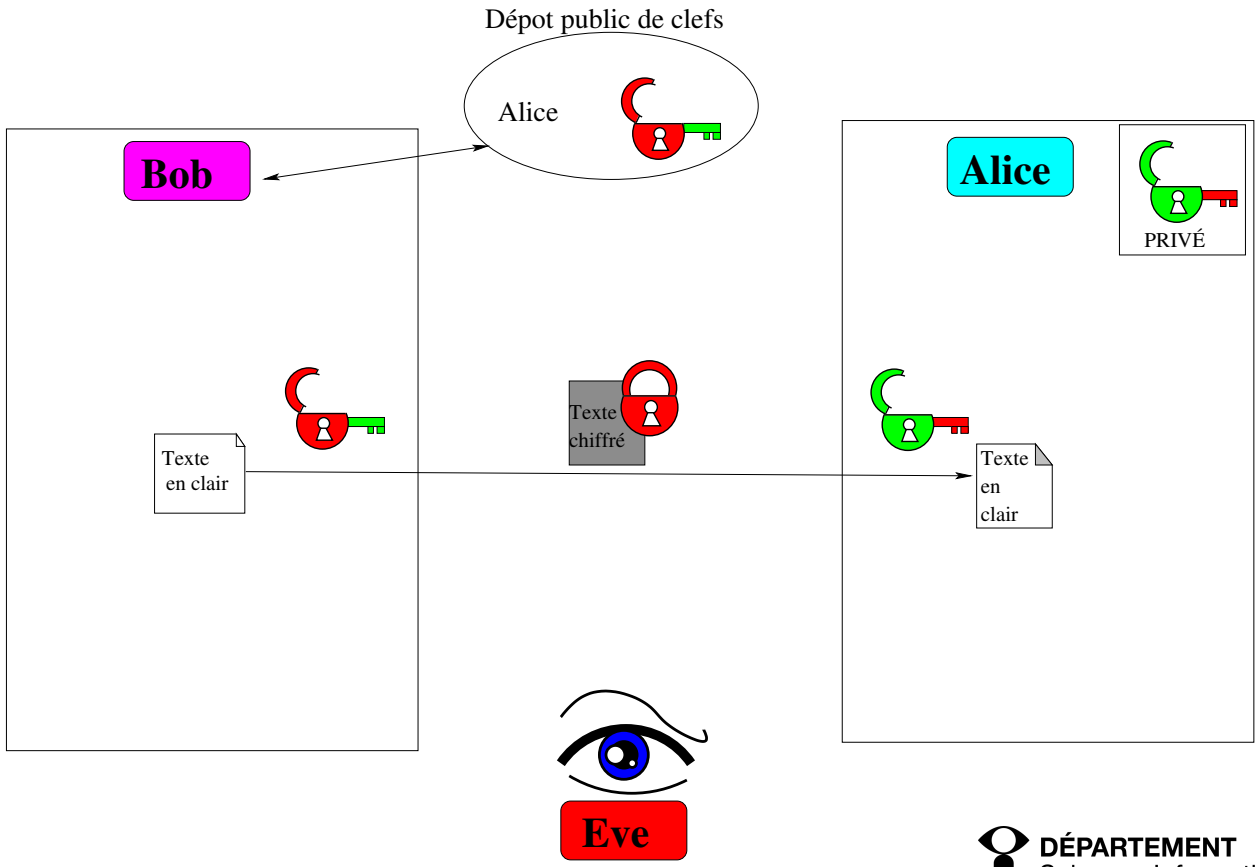
- Une clef publique, une clef privée.
- Un message chiffré avec l'une des clefs ne peut être déchiffré qu'avec l'autre clef.
- Deux utilisations possibles :
  - Confidentialité (chiffrement avec la clef publique, déchiffrement avec la clef privée)
  - Signature (chiffrement avec la clef privée, déchiffrement avec la clef publique)
- RSA, DSA, ElGamal

DÉPARTEMENT Sciences Informatiques

Rappel des algorithmes cryptographiques

- Algorithme à clef symétrique
- Algorithme de Diffie-Hellman
- Algorithme à clef asymétrique
  - Confidentialité
  - Signature
- Hachage

Confidentialité avec clef non partagée asynchrone : chiffrement par clef publique



DÉPARTEMENT Sciences Informatiques

Rappel des algorithmes cryptographiques

- Algorithme à clef symétrique
- Algorithme de Diffie-Hellman
- Algorithme à clef asymétrique
  - Confidentialité
  - Signature
- Hachage

Confidentialité avec clef non partagée asynchrone : chiffrement par clef publique

- Bob chiffre un message avec la clef publique d'Alice

$\Rightarrow$ seule Alice pourra déchiffrer le message

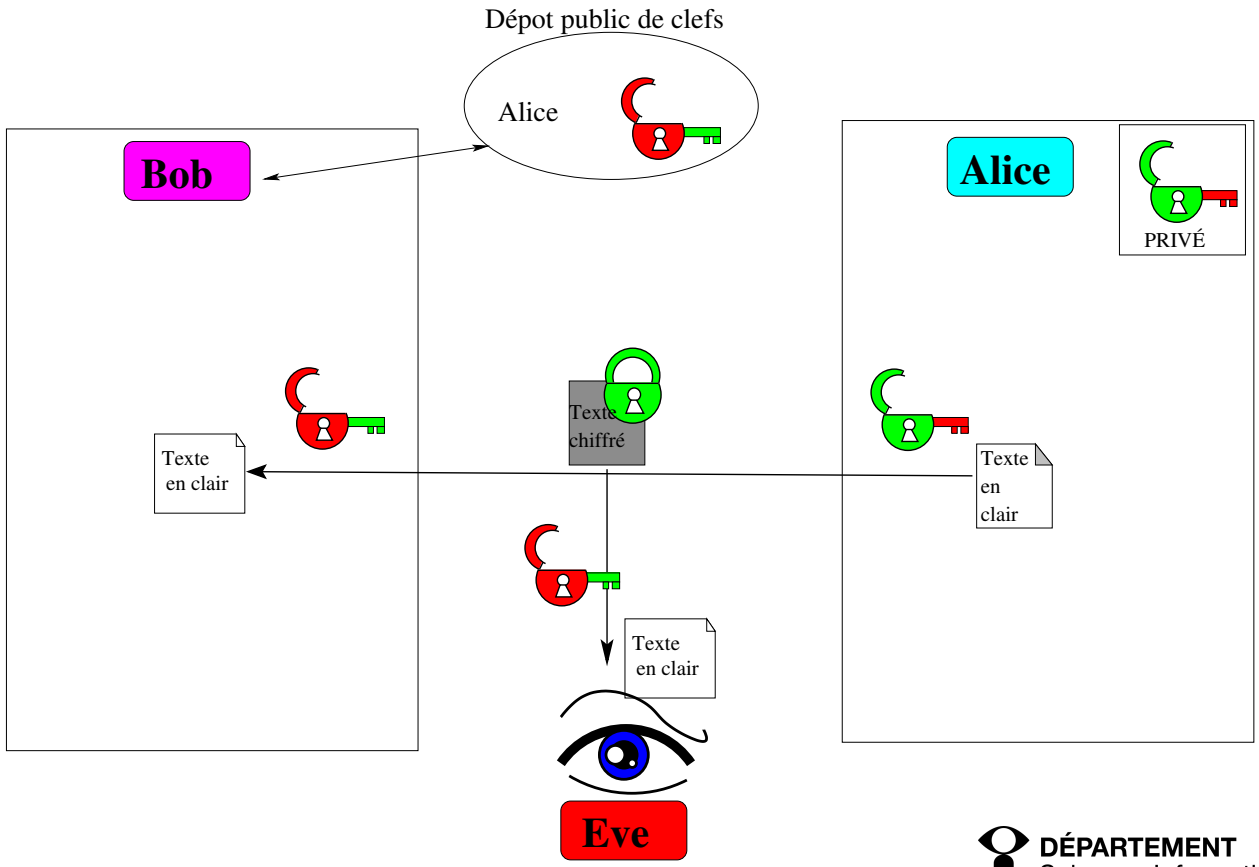
- Pas de problèmes d'échange de clef
- ne nécessite pas la présence simultanée des deux partis
- Lent ! ! !

DÉPARTEMENT Sciences Informatiques

Rappel des algorithmes cryptographiques

- Algorithme à clef symétrique
- Algorithme de Diffie-Hellman
- Algorithme à clef asymétrique
  - Confidentialité
  - Signature
- Hachage

Authentification et non-répudiation : déchiffrement par clef publique



DÉPARTEMENT Sciences Informatiques

Rappel des algorithmes cryptographiques

- Algorithme à clef symétrique
- Algorithme de Diffie-Hellman
- Algorithme à clef asymétrique
  - Confidentialité
  - Signature
- Hachage

Authentification et non-répudiation : déchiffrement par clef publique

- Alice chiffre un message avec sa propre clef privée

$\Rightarrow$ tout le monde (dont Bob et Eve) peut déchiffrer le message ainsi chiffré, et vérifier (si on obtient un texte intelligible) que c'est bien Alice l'auteur du message

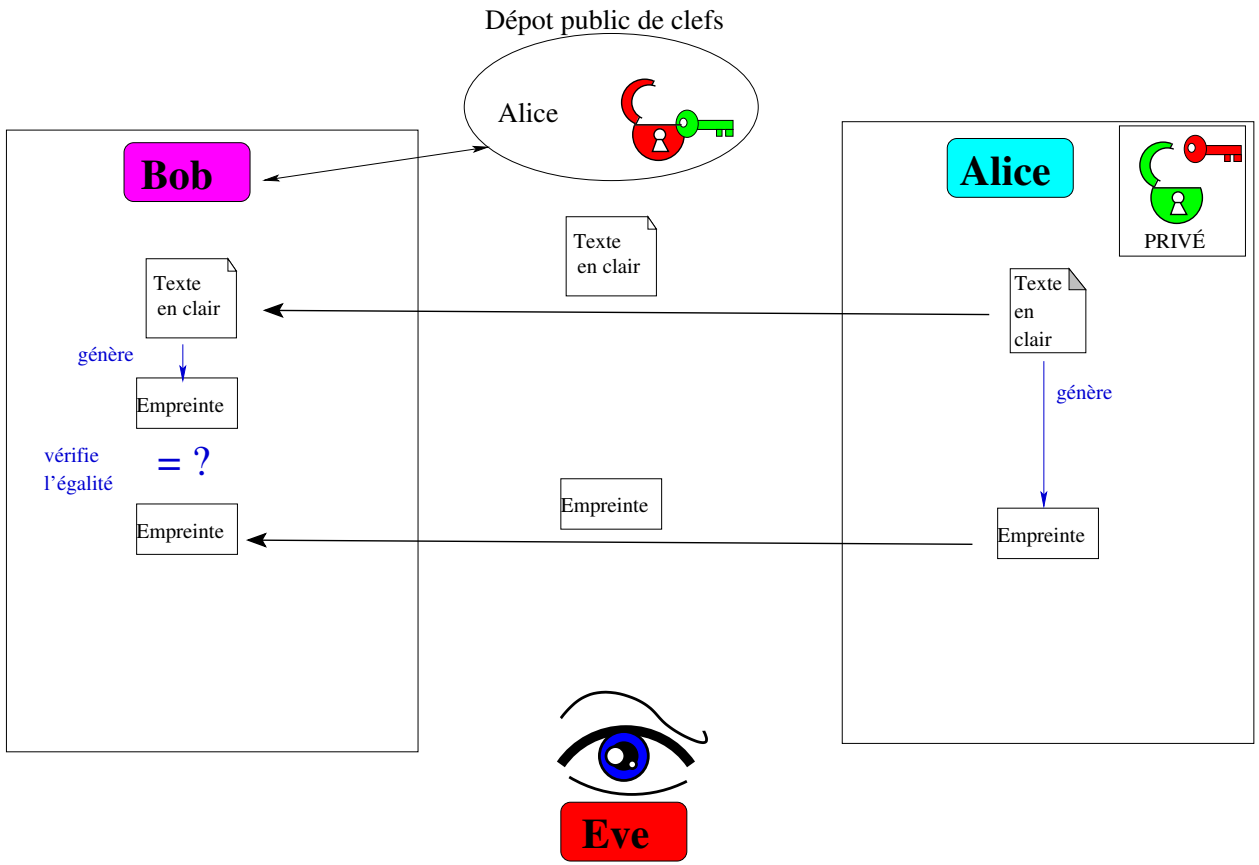
- Garanti l'identité de l'expéditeur
- il faut signer tout le message pour garantir son intégrité $\Rightarrow$ Long !

DÉPARTEMENT Sciences Informatiques

Rappel des algorithmes cryptographiques

Empreinte non-inversible de taille fixe

- Algorithme à clef symétrique
- Algorithme de Diffie-Hellman
- Algorithme à clef asymétrique
  -
    - Confidentialité
    - Signature
- Hachage



DÉPARTEMENT Sciences Informatiques

Rappel des algorithmes cryptographiques

- Algorithme à clef symétrique
- Algorithme de Diffie-Hellman
- Algorithme à clef asymétrique
  - Confidentialité
  - Signature
- Hachage

Empreinte non-inversible de taille fixe

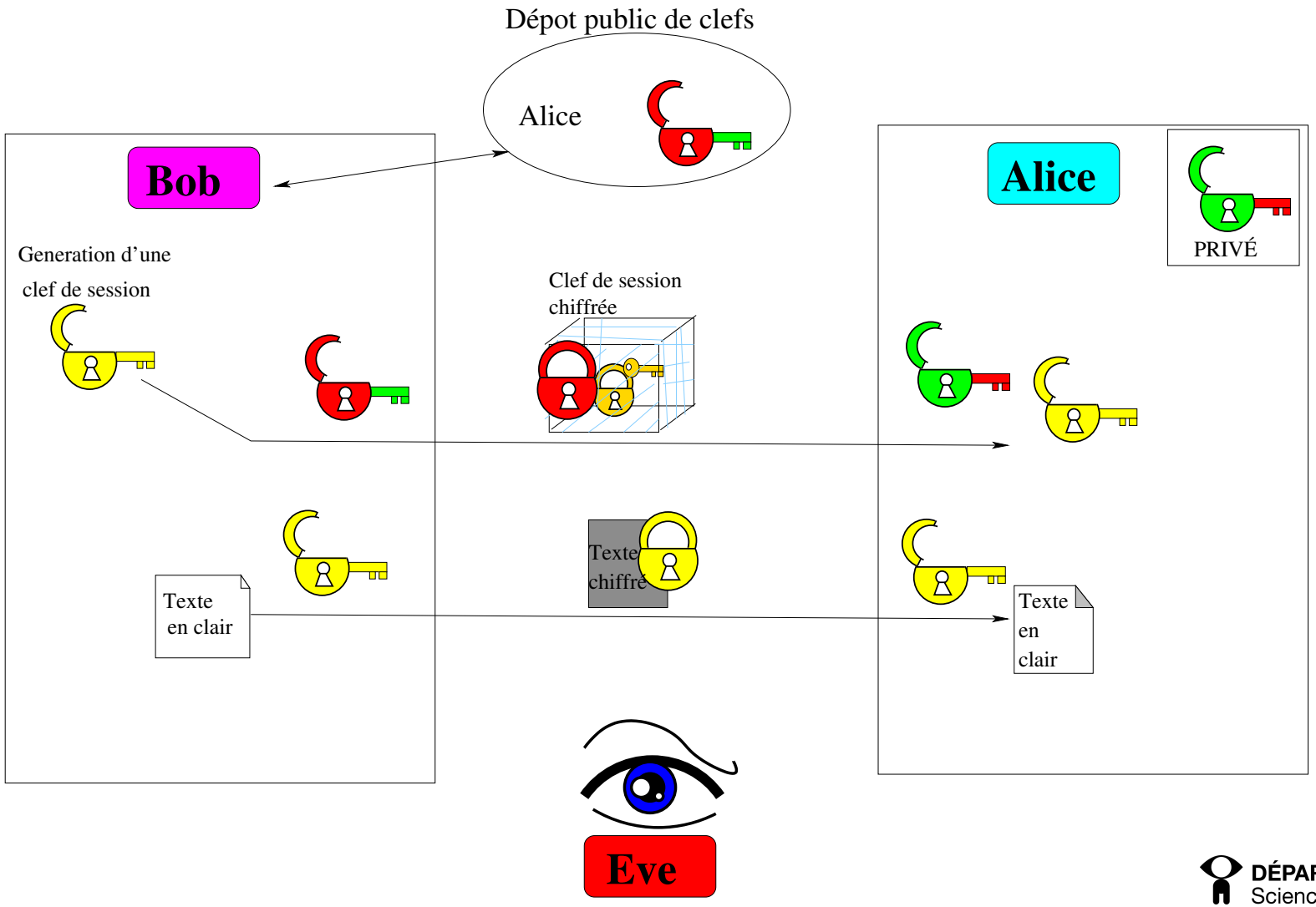
- la fonction de hachage permet de garantir l'intégrité du message
- très rapide à calculer
- ne garanti pas l'identité de l'expéditeur
- MD5, SHA, Whirlpool, RIPEMD, Tiger, Haval

DÉPARTEMENT Sciences Informatiques

Confidentialité avec clef de session transmise par clef publique

- Bob crée une clef de session
- Bob utilise la clef publique d'Alice pour lui transmettre la clef de session de manière sécurisée.
- Seule Alice peut déchiffrer la clef de session avec sa propre clef privée
- Alice et Bob utilisent ensuite cette clef de session partagée entre eux deux seulement pour chiffrer/déchiffrer leurs messages à l'aide d'un algorithme à clef symétrique.

Confidentialité avec clef de session transmise par clef publique

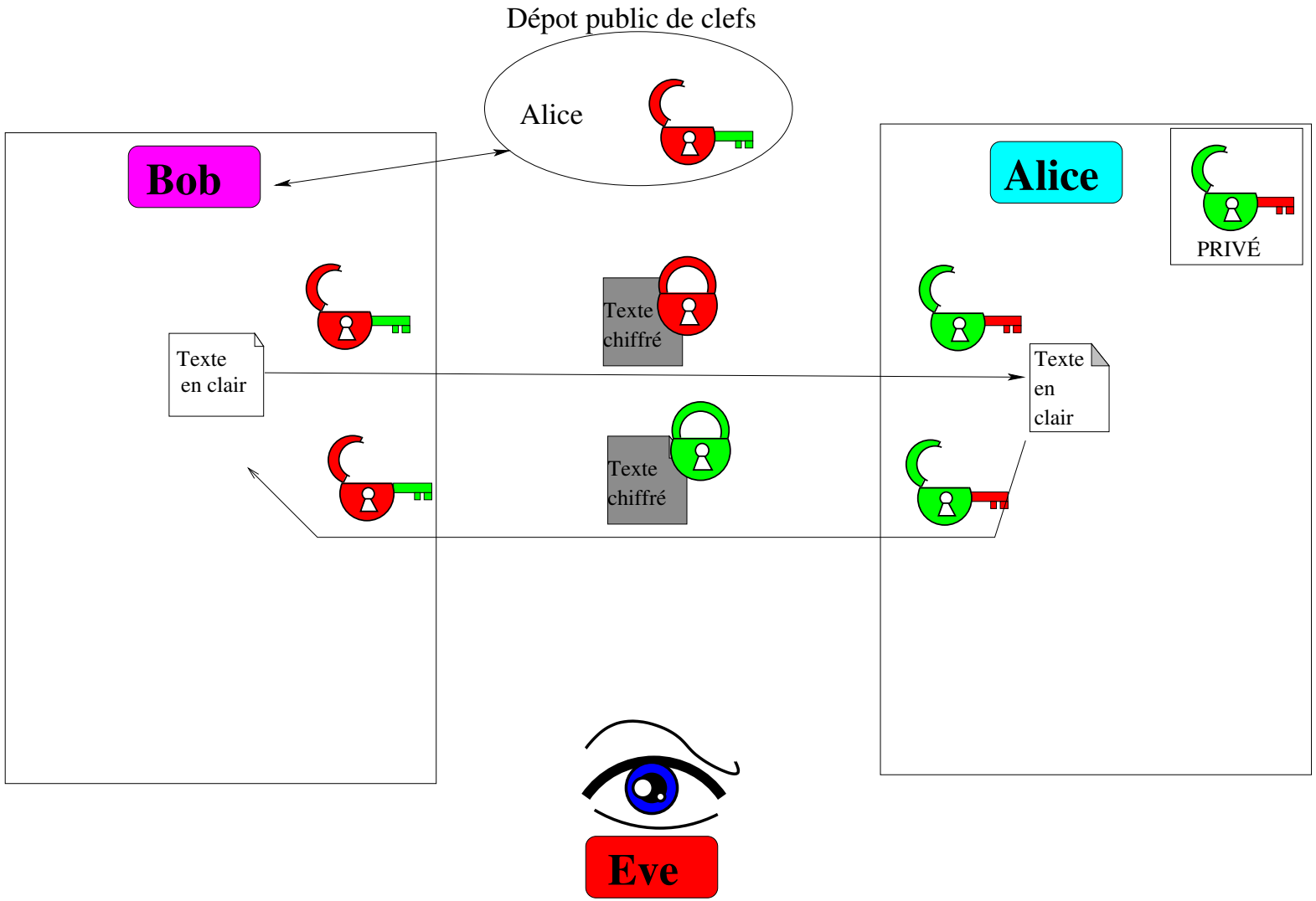


DÉPARTEMENT
Sciences Informatiques

Authentification par challenge

- Bob crée un texte qu'il chiffre avec la clef publique d'Alice et lui envoie.
- Seule Alice peut déchiffrer le texte avec sa propre clef privée.
- Alice renvoie le texte chiffré avec sa propre clef privée.
- Bob déchiffre le texte avec la clef publique d'Alice. S'il arrive à obtenir le même texte que celui qu'il a envoyé au départ, il peut être sûr que c'est bien Alice.

Authentification par challenge



DÉPARTEMENT
Sciences Informatiques

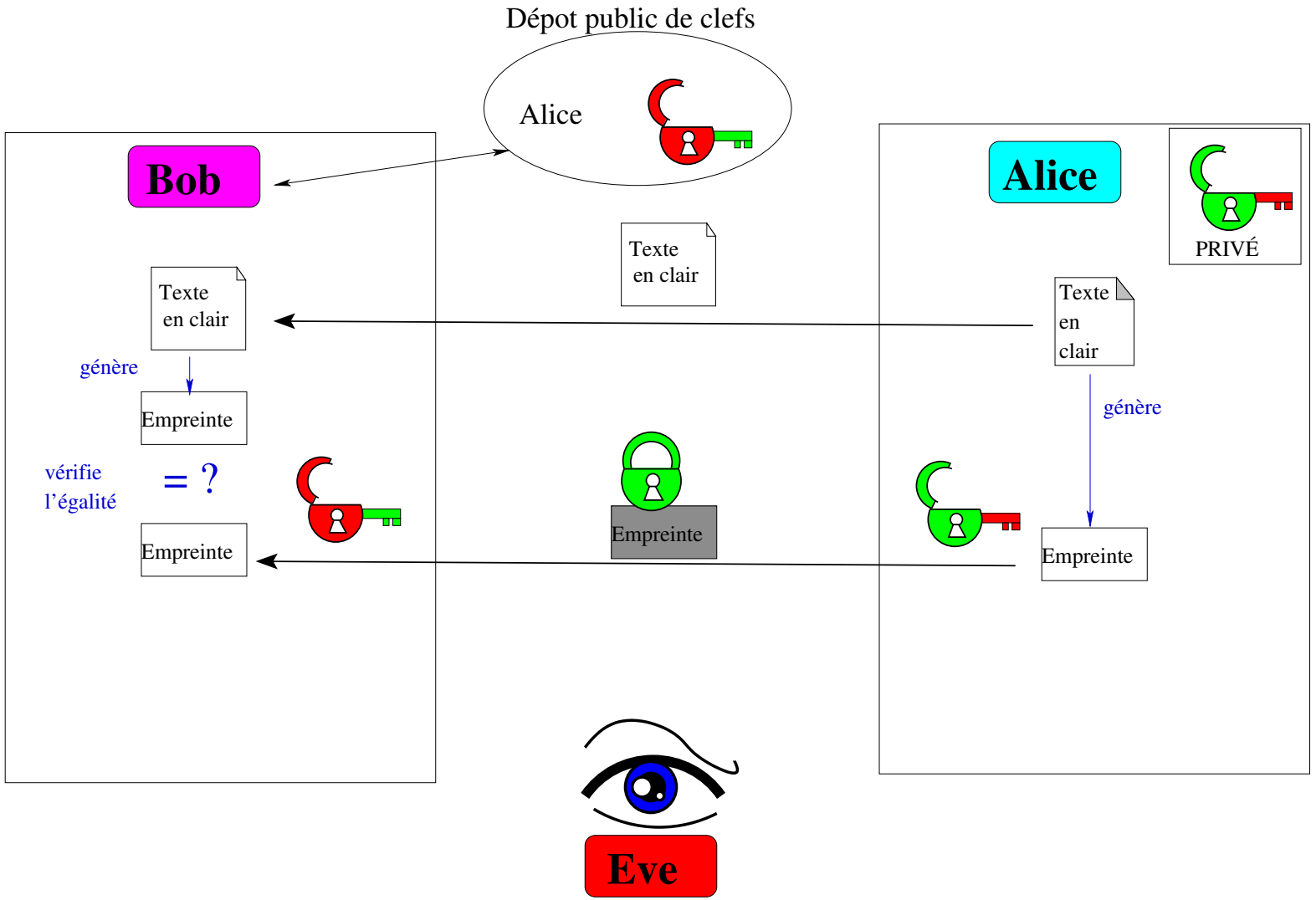
Intégrité et signature par sceau électronique

- Alice crée un texte dont elle calcule l'empreinte.
- Alice chiffre l'empreinte avec sa propre clef privée.
- Alice diffuse le texte en clair et l'empreinte chiffrée.
- Bob voulant calcule l'empreinte à partir du texte en clair.
- Bob déchiffre l'empreinte chiffrée avec la clef publique d'Alice.
- Si Bob tombe sur le même résultat, alors il en déduit que c'est bien Alice et elle seule qui a chiffré le texte et que le texte n'a pas été altéré.

$\Rightarrow$ Procédé très rapide. Garantie l'authentification, l'intégrité et la non-répudation.

DÉPARTEMENT Sciences Informatiques

Intégrité et signature par sceau électronique

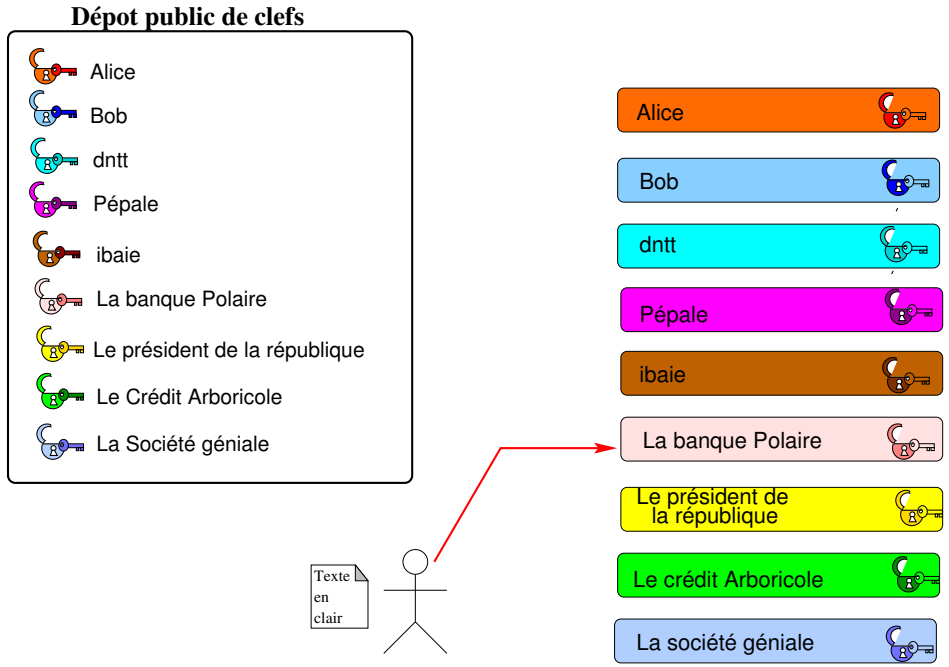


DÉPARTEMENT
Sciences Informatiques

Problèmes de la distribution de clefs publiques

Comment être **certain** que la clef publique récupérée est bien celle de l'entité (la personne, l'entreprise) avec laquelle on veut communiquer ?

- Chiffrement asymétrique = basé sur la distribution de clés publiques (Annuaire, serveur de clefs)
- rien ne garantit que la clé est bien celle de l'utilisateur a qui elle est sensé être associée
- tout repose sur la confiance dans la provenance de la clef publique



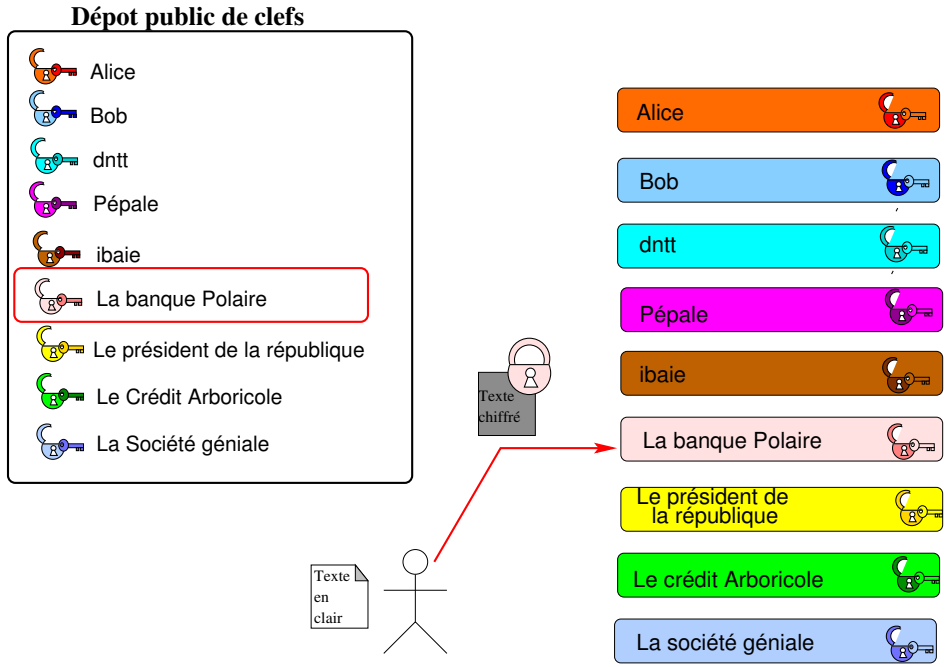
Peut-on faire confiance à l'organisme qui distribue les clefs ?

DEPARTEMENT Sciences Informatiques

Problèmes de la distribution de clefs publiques

Comment être **certain** que la clef publique récupérée est bien celle de l'entité (la personne, l'entreprise) avec laquelle on veut communiquer ?

- Chiffrement asymétrique = basé sur la distribution de clés publiques (Annuaire, serveur de clefs)
- rien ne garantit que la clé est bien celle de l'utilisateur a qui elle est sensé être associée
- tout repose sur la confiance dans la provenance de la clef publique



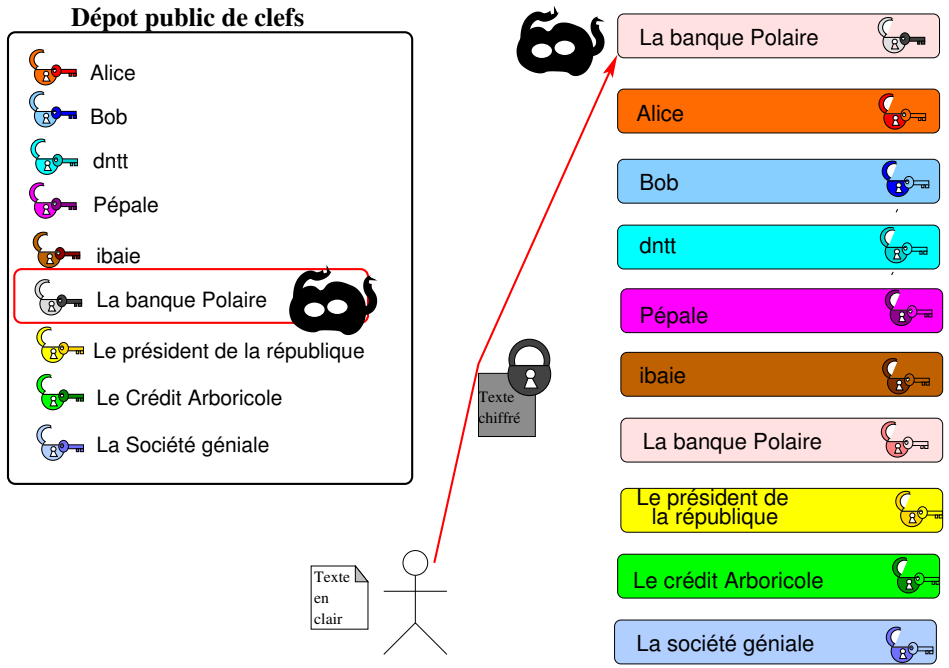
Peut-on faire confiance à l'organisme qui distribue les clefs ?

DEPARTEMENT Sciences Informatiques

Problèmes de la distribution de clefs publiques

Comment être **certain** que la clef publique récupérée est bien celle de l'entité (la personne, l'entreprise) avec laquelle on veut communiquer ?

- Chiffrement asymétrique = basé sur la distribution de clés publiques (Annuaire, serveur de clefs)
- rien ne garantit que la clé est bien celle de l'utilisateur a qui elle est sensé être associée
- tout repose sur la confiance dans la provenance de la clef publique



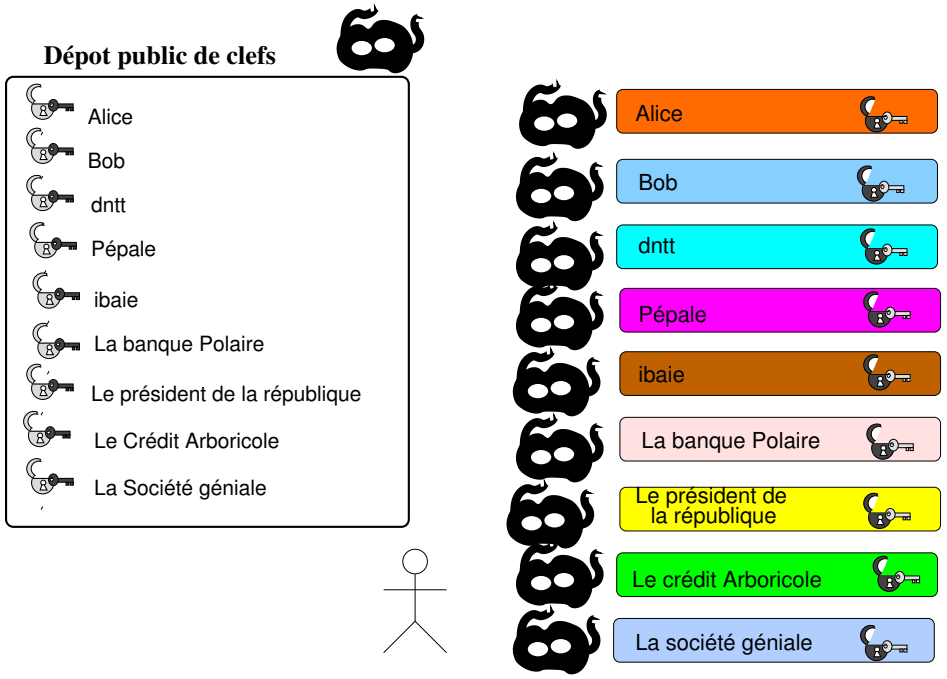
Peut-on faire confiance à l'organisme qui distribue les clefs ?

DÉPARTEMENT Sciences Informatiques

Problèmes de la distribution de clefs publiques

Comment être **certain** que la clef publique récupérée est bien celle de l'entité (la personne, l'entreprise) avec laquelle on veut communiquer ?

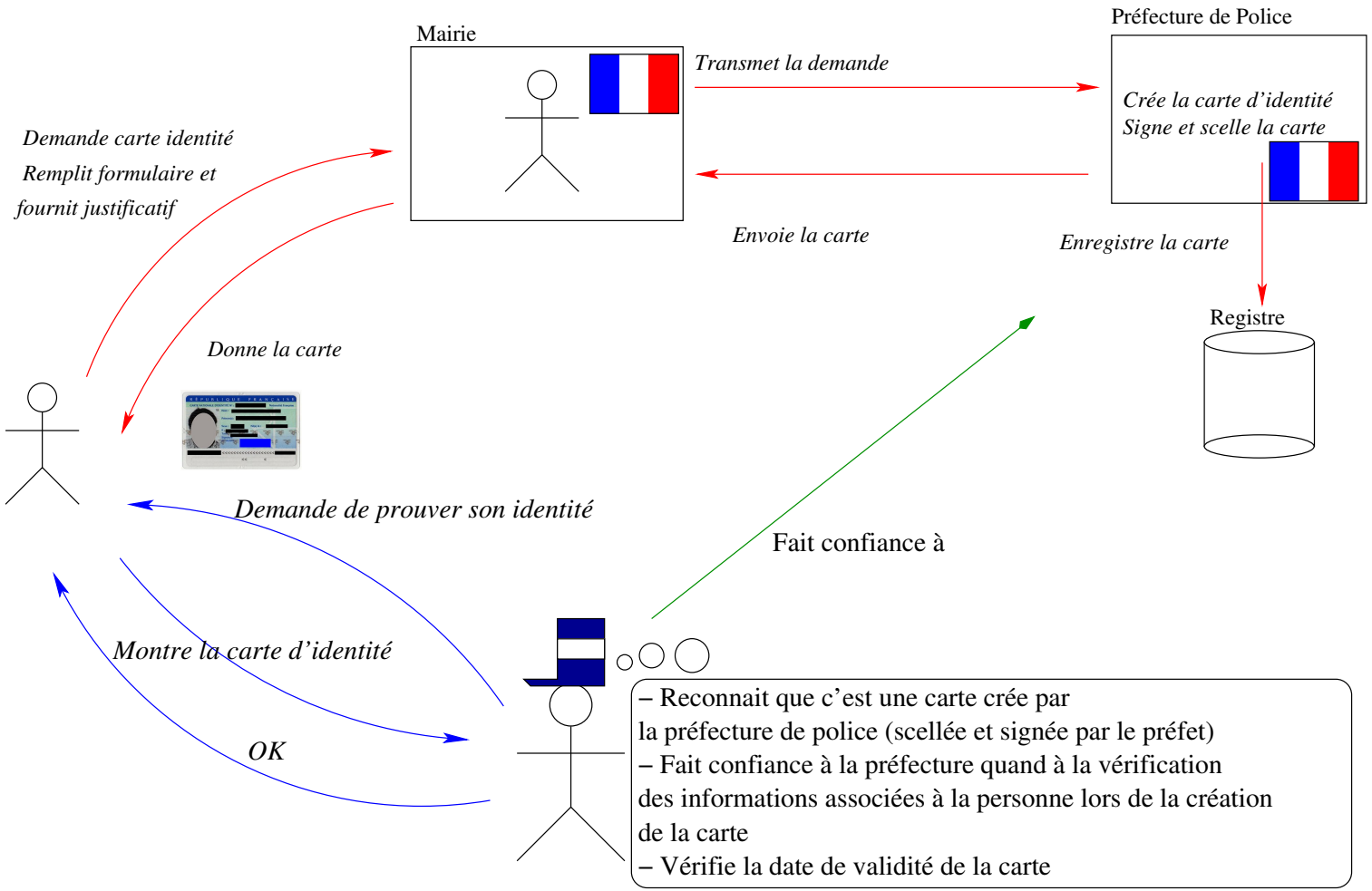
- Chiffrement asymétrique = basé sur la distribution de clés publiques (Annuaire, serveur de clefs)
- rien ne garantit que la clé est bien celle de l'utilisateur a qui elle est sensé être associée
- tout repose sur la confiance dans la provenance de la clef publique



Peut-on faire confiance à l'organisme qui distribue les clefs ?

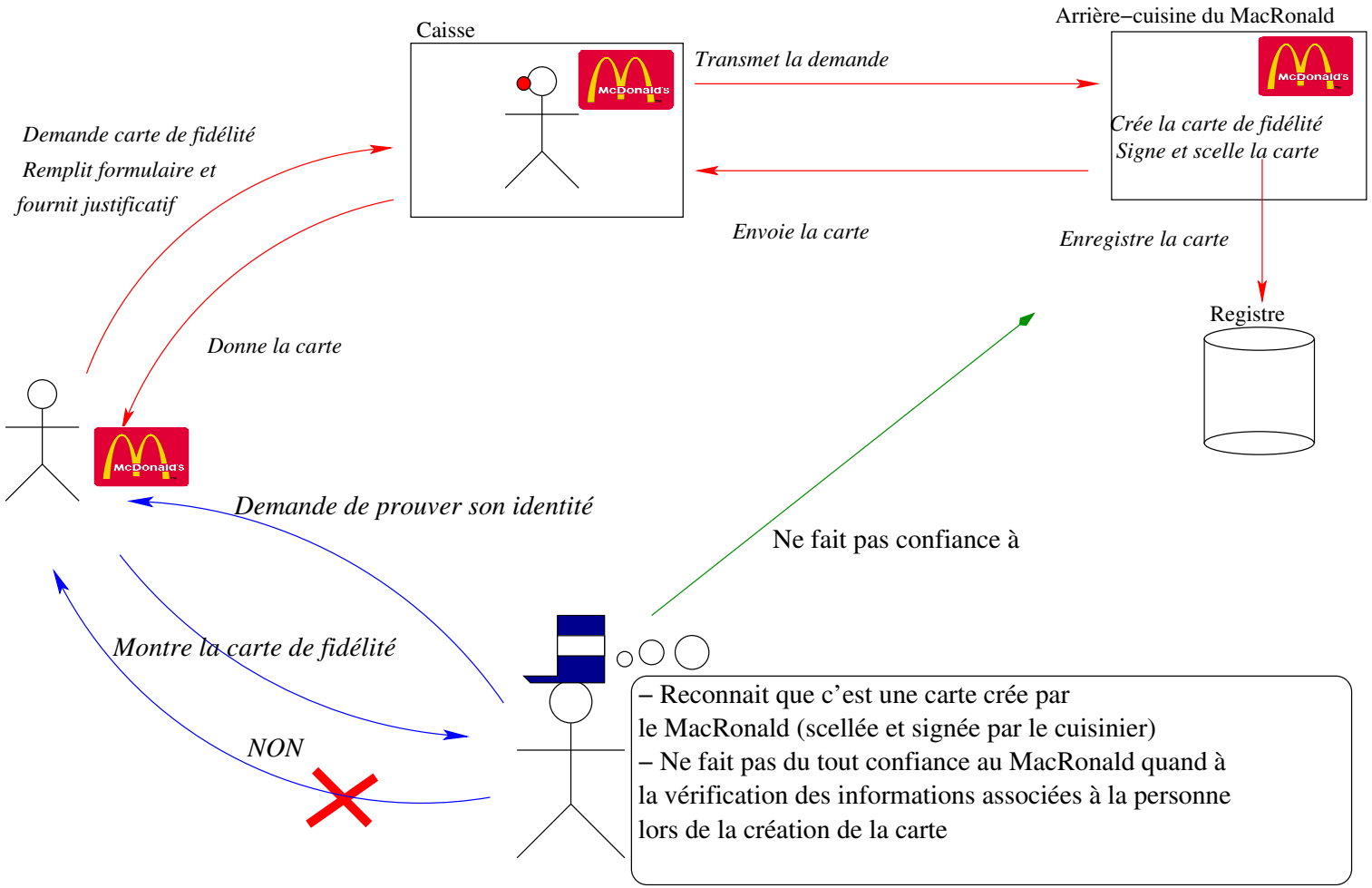
Sciences Informatiques

Analogie : Délivrance et utilisation d'une carte d'identité



DÉPARTEMENT Sciences Informatiques

Analogie : Délivrance et utilisation d'une carte d'identité



DÉPARTEMENT Sciences Informatiques

Analogie : Carte Nationale d'Identité (CNI)

N° CNI

Signature

Scellé

Identité (nom, prénom, date de naissance, lieu de naissance)

Description (sexe, taille, adresse)

Bande de lecture optique (Synthèse des informations)

Date d'émission

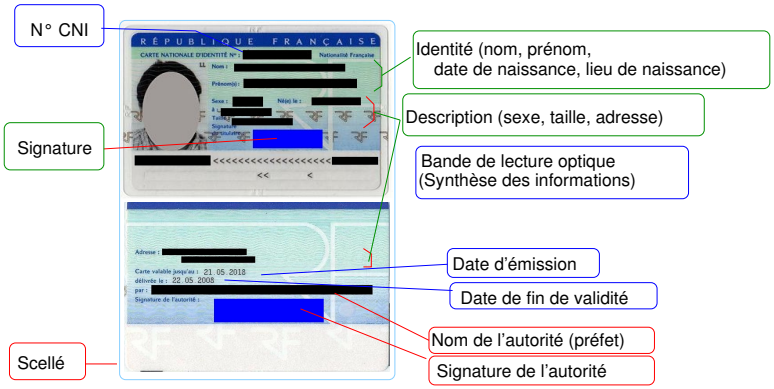
Date de fin de validité

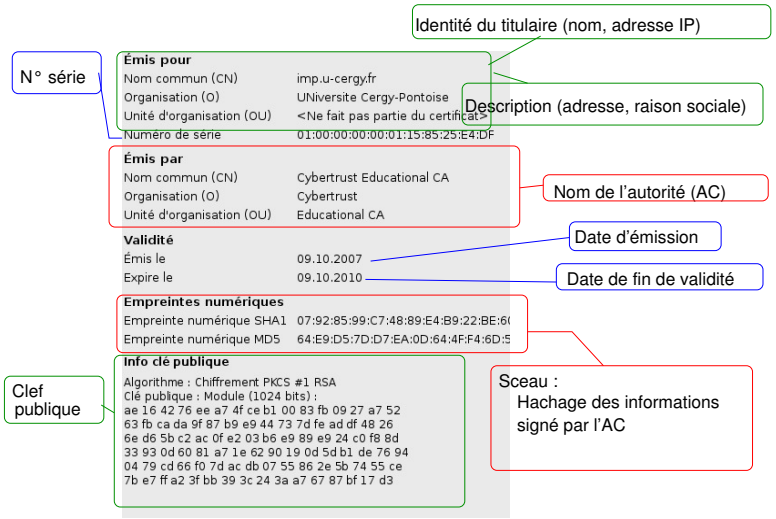
Nom de l'autorité (préfet)

Signature de l'autorité

DÉPARTEMENT Sciences Informatiques

Analogie : Carte Nationale d'Identité (CNI) / Certificat X509





DÉPARTEMENT Sciences Informatiques

But d'une architecture PKI

Délivrer des certificats numériques qui offrent les garanties suivantes lors des transactions électroniques :

- confidentialité : seul le destinataire légitime du message pourra le lire
- authentification : l'identité de l'émetteur est garantie
- intégrité : Garantie qu'un message expédié n'a pas été altéré, accidentellement ou intentionnellement ;
- non-répudiation : l'auteur du message ne peut pas renier son message.

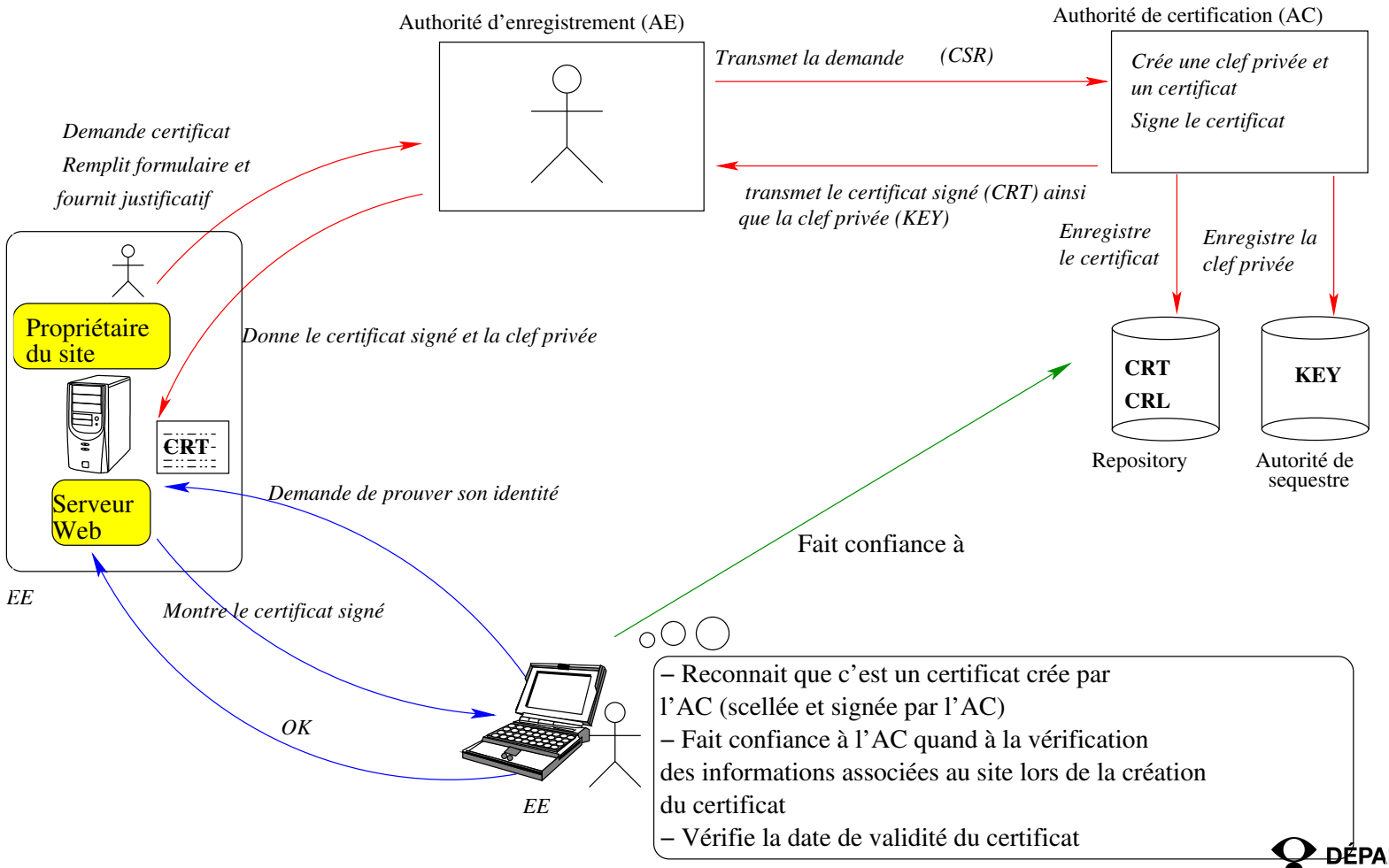
DÉPARTEMENT Sciences Informatiques

PKI (Public Key Infrastructure) / IGC (Infrastructure de Gestion de Clef)

- L'entité finale (EE : End Entity) : L'utilisateur ou le système qui sujet d'un certificat
- L'autorité d'enregistrement (AE/RA) : effectue les vérifications d'usage sur l'identité de l'utilisateur. Fait la demande de certificat et donne le certificat signé à l'utilisateur.
- L'autorité de certification (AC/CA) : signe les demandes de certificat (CSR) et les listes de révocation (CRL)
- L'autorité de dépôt (Repository) : stocke les certificats numériques et les listes de révocation (CRL).
- L'autorité de séquestre (Key Escrow) : stocke de façon sécurisée les clés de chiffrement qui ont été générées par l'IGC, pour pouvoir les restaurer le cas échéant.

DÉPARTEMENT Sciences Informatiques

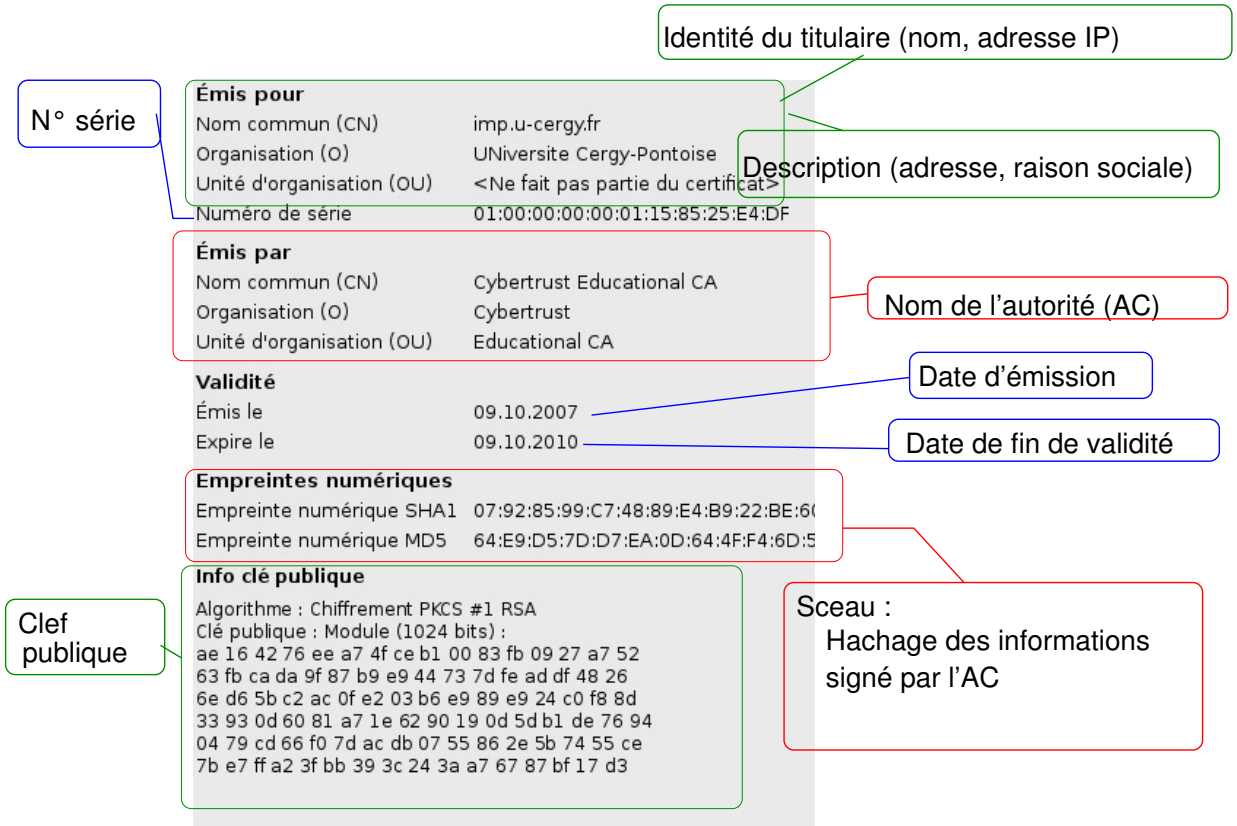
PKI (Public Key Infrastructure) / IGC (Infrastructure de Gestion de Clef)



DÉPARTEMENT Sciences Informatiques

Certificat électronique

- Carte d'identité électronique, composée de la clé publique du porteur et d'informations relatives à ce dernier.
- Délivré par un tiers de confiance, appelé "autorité de certification" (AC), qui, par sa signature, en garantit l'authenticité.



DÉPARTEMENT Sciences Informatiques

Certificat numérique X.509

- Version
- Numéro de série
- Algorithme de signature du certificat
- Nom du signataire du certificat
- Validité (dates limite)
  - Pas avant
  - Pas après
- Détenteur du certificat
- Informations sur la clé publique :
  - Algorithme de la clé publique
  - Clé publique proprement dite
- Identifiant unique du signataire (optionnel, à partir de X.509 v2)
- Identifiant unique du détenteur du certificat (optionnel, à partir de X.509 v2)
- Extensions (optionnel, à partir de X.509 v3)

DÉPARTEMENT Sciences Informatiques

Format des clefs

- binaire encodé DER (Definite Encoding Rules) (ASN.1 BER)
- Encodage PEM (Privacy Enhanced Mail) base 64 encodée base64 de la forme DER munie de lignes d'en-tête et de pied de page

```
-----BEGIN XXXXXX KEY-----
-----END XXXXXX KEY-----
```

- Encodage XML

```
<RSAKeyPair>
  <Modulus>...<Modulus>
  <Exponent>...</Exponent>
  <P>...</P>
  <Q>...</Q>
  <DP>...</DP>
  <DQ>...</DQ>
  <InverseQ>... </InverseQ>
  <D></D>
</ RSAKeyPair>
```

DÉPARTEMENT Sciences Informatiques

Format des certificats X509

Encoder des certificats X509

- Encodage DER (Definite Encoding Rules) en notation ASN.1
  Extensions usuelles : .der, .cer, .crt, .cert
- Encodage PEM (Privacy Enhanced Mail) encodée base64 de la forme DER munie de lignes d'en-tête et de pied de page

  ```
  -----BEGIN X509 CRL-----
  -----END X509 CRL-----
  ```

  Extensions usuelles : .pem, .cer, .crt, .cert

DÉPARTEMENT Sciences Informatiques

Autorité d'enregistrement (AE) et Classe de certificat

1. L'AE vérifie l'identité de l'utilisateur : 4 classes de certificats en fonction des vérifications effectuées auprès de l'autorité d'enregistrement :
   - classe 1 : adresse e-mail du demandeur requise ;
   - classe 2 : preuve de l'identité requise (photocopie de carte d'identité par exemple) ;
   - classe 3 : présentation physique du demandeur obligatoire.
   - classe 3+ : identique à la classe 3, mais le certificat est stocké sur un support physique (clé USB à puce, ou carte à puce ; exclut donc les certificats logiciels)
2. l'AE génère le certificat et demande à l'AC de le signer (Certificate Signing Request - CSR).
3. l'AE donne le certificat signé (Certificate - CRT) à l'utilisateur.

DÉPARTEMENT Sciences Informatiques

Autorité de Certification (AC)

- Traite les demandes de signature de certificats (Certificate Signing Request - CSR) qui lui sont transmis par l'AC en les signant à l'aide de sa propre clef publique.
- Enregistre le certificat dans un repository
- Enregistre (éventuellement) la clef privée dans un dépôt sequestré (obligation légale en France) ou la détruit.

Synthèse : Procédure de génération de certificat

1. L'AC possède une paire de clefs asymétrique privée p_{AC} et publique P_{AC}.
2. À la demande, il génère une paire de clefs asymétrique privée p_x et publique P_x pour la machine x.
3. Il crée un certificat $cert_x$ composé de :
   - un numéro de série
   - l'identification de l'algorithme de signature
   - la désignation de l'autorité de certification émettrice du certificat
   - la période de validité au-delà de laquelle il sera suspendu ou révoqué
   - le nom du titulaire de la clé publique (x)
   - l'identification de l'algorithme de chiffrement et la valeur de la clé publique P_x.
   - des informations complémentaires optionnelles
   - l'identification de l'algorithme de signature et la valeur de la signature numérique.
4. Il signe le certificat $cert_x$ à l'aide de sa propre clef privée p_{AC}
5. Il envoie la clef privée p_x et le certificat $cert_x$ à x
6. Il détruit la clef privée p_x de x (ou la stocke dans un dépôt sequestré).

DÉPARTEMENT Sciences Informatiques

Exemple d'utilisation du certificat

L'utilisateur accède à un site marchand, le site marchand montre son certificat
L'utilisateur vérifie le certificat du site marchand :

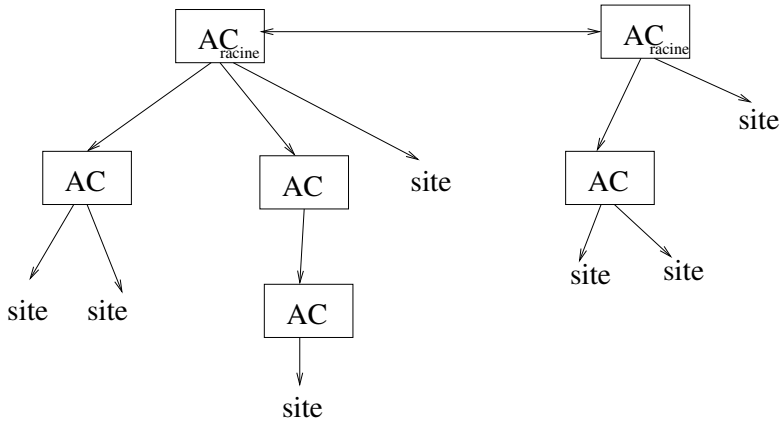
1. il regarde la date de validité du certificat
2. il vérifie que l'adresse du site correspond bien à l'adresse indiquée par le certificat
3. il vérifie que le certificat a bien été scellée par une AC à laquelle l'utilisateur a confiance.
   - l'utilisateur utilise la clef publique de l'AC pour déchiffrer l'empreinte chiffrée du certificat
   - il calcule l'empreinte du certificat
   - il compare les empreintes.

Problème

Dans quelles autorités de certification (AC) avoir confiance ?

Certification croisée et hiérarchique

- une AC émet un certificat pour un autre AC : elle engage sa responsabilité quand elle signe un certificat, mais selon les termes de la politique de certification qu'elle a définie.
- si on fait confiance à une AC, alors on fait confiance aux AC qu'elle a certifié...
- ... et ceci récursivement.



Problème

Mais à quelles AC fait on confiance au départ ?

DÉPARTEMENT Sciences Informatiques

AC de confiance

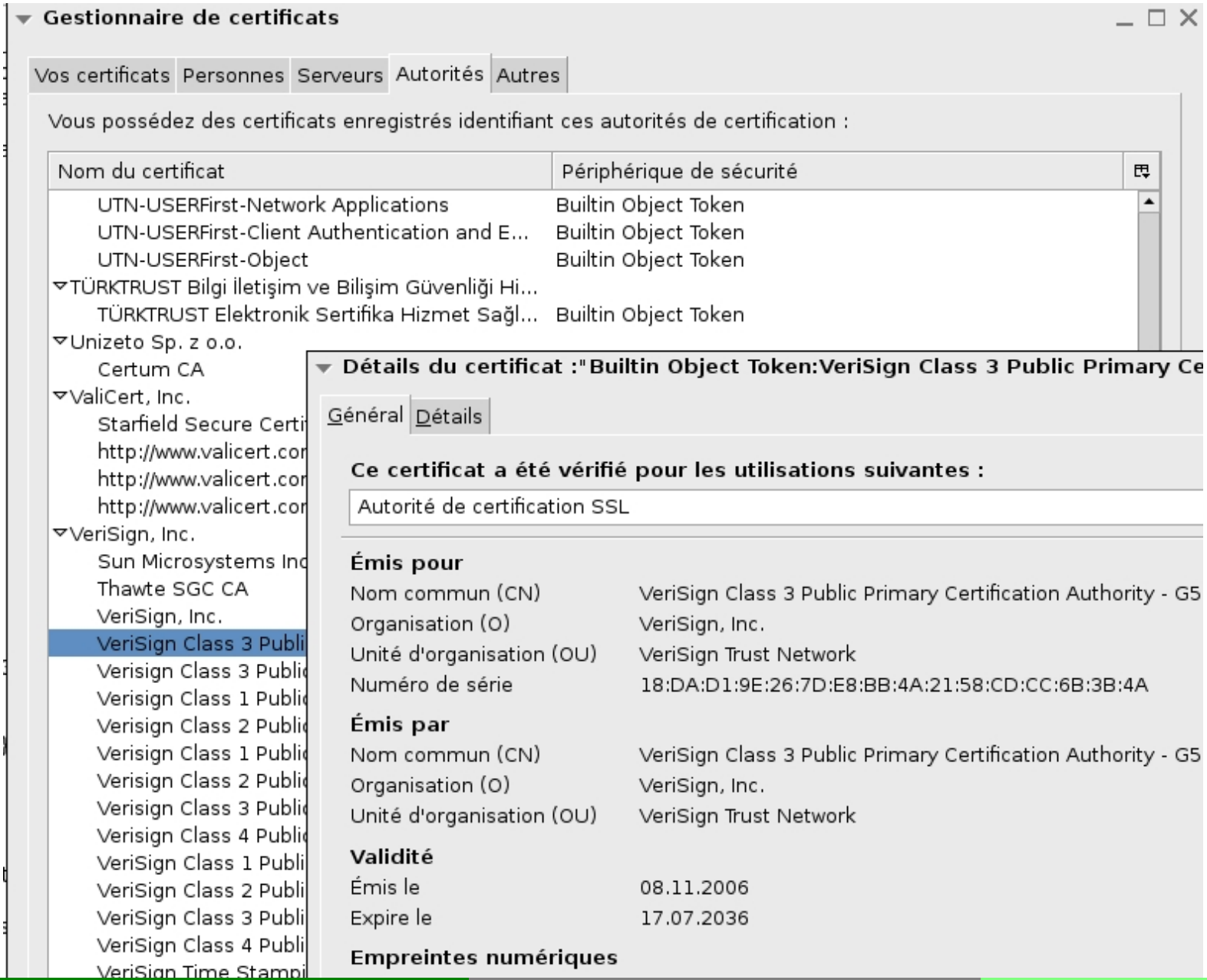
Problème

A quelles AC fait on confiance au départ ?

- on les définit explicitement (on importe le certificat du ou des AC) : par exemple l'AC maintenu par notre entreprise, une AC avec laquelle notre entreprise s'est personnellement engagé, etc.
- on utilise une liste d'AC qui ont "pignon sur rue". Par exemple, la liste des AC par défaut fourni dans les navigateurs web (firefox, internet explorer, opera, etc.) les plus courants :
  AOL Time Warner, beTRUSTed, Certplus, COMODO CA, ComSign, Cybertrust, Digicert, Entrust, Equifax, GeoTrust, GlobalSign, GTE Corp, Microsec, Quo Vadis, Root CA, RSA Security, SECOM Trust, SecureTrust, Sonera, SwissCom, TDC, Thawte, TÜRKTRUST, Unizeto, ValiCert, VeriSign, VISA, Wells Fargo, WISeKey, XRamp, etc.

DÉPARTEMENT Sciences Informatiques

Liste d'AC commerciaux par défaut sous firefox 3.0.6



DÉPARTEMENT Sciences Informatiques

Révocation de certificat

Certaines raisons peuvent amener à révoquer un certificat :

- perte de la clef privée (effacement accidentel, crash disque, etc.)
- compromission de la clef privée (piratage)
- disparition du titulaire (fermeture de l'entreprise, etc.)

DÉPARTEMENT
Sciences Informatiques

Comment révoquer ?

- Le certificat existant est bien signé par l'AC
- La date d'expiration n'est pas encore atteinte.

$\Rightarrow$ Pas de moyen de savoir que le certificat n'est plus valide.
Solution : l'AC doit maintenir une liste des certificats révoqués.

DÉPARTEMENT
Sciences Informatiques

Certificate Revocation List (CRL)

Liste de certificats révoqués sous forme de paires :
(numéro de série du certificat révoqué ; motif éventuel de révocation)

- Liste CRL envoyée sous format DER ou PEM.
- Récupération des CRL difficilement automatisable : grosse liste de CRL $\Rightarrow$ gros traffic pour le client (l'acheteur).
- CRL (mal) perçue comme une liste de "mauvais vendeurs"...

DÉPARTEMENT Sciences Informatiques

Certificate Revocation List (CRL)

- Protocole de vérification **en ligne** de certificat.
- encodés en ASN.1 transportés par différents protocoles applicatifs (SMTP, LDAP, HTTP, etc.)
- Serveur OCSP ou répondeur OCSP appelé aussi Autorité de Validation (VA).
- Le client ne communique plus qu'avec la VA
  - L'empreinte du certificat du vendeur est transmise par requête OCSP au VA par OCSP.
  - Le VA consulte l'AC (validité + liste de révocation)
  - la réponse OCSP est renvoyée.
- Le client n'a pas à faire la vérification sur la CRL (ni à récupérer la liste)
- La VA ne considère que la dernière mise à jour du certificat (pas de publication de "mauvais" certificats
- la VA peut faire payer le vendeur pour ce service...

DÉPARTEMENT Sciences Informatiques

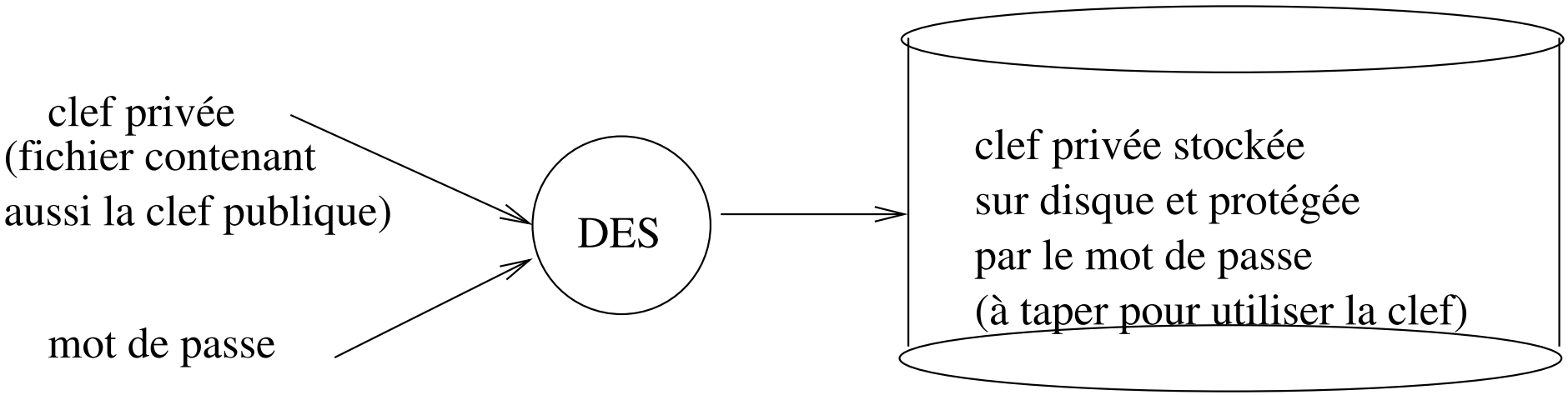
Synthèse : Comment vérifier la validité d'un certificat ?

- on regarde si le certificat a été validé par une AC
- on vérifie que l'AC est une AC à laquelle on fait confiance
  - le certificat a bien été signé avec la signature privée de l'AC. (on doit pouvoir le déchiffrer avec la clef publique de l'AC).
  - l'AC est dans notre liste d'AC auxquelles on fait confiance, et dont on a enregistré le certificat (et donc aussi la clef publique) auparavant.
- on vérifie que la date de validité du certificat est toujours bonne
- on vérifie que le certificat ne se trouve pas dans la liste des certificats révoqués CRL.

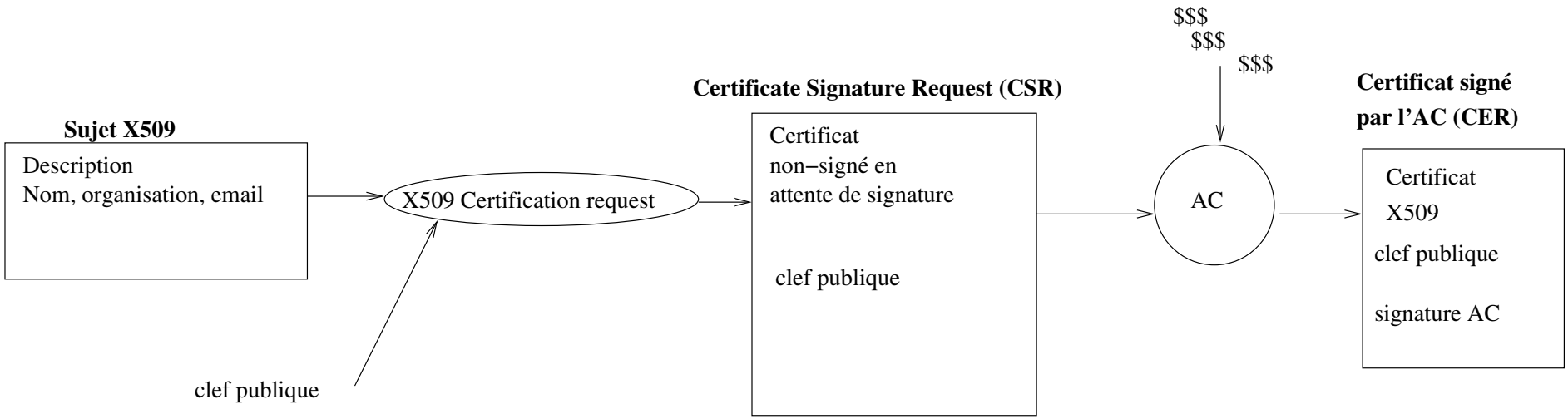
ou si le protocole OCSP est supporté :

- on demande au VA auquel on fait confiance de valider le certificat en lui envoyant son empreinte.

DÉPARTEMENT Sciences Informatiques



DÉPARTEMENT
Sciences Informatiques



DÉPARTEMENT
Sciences Informatiques

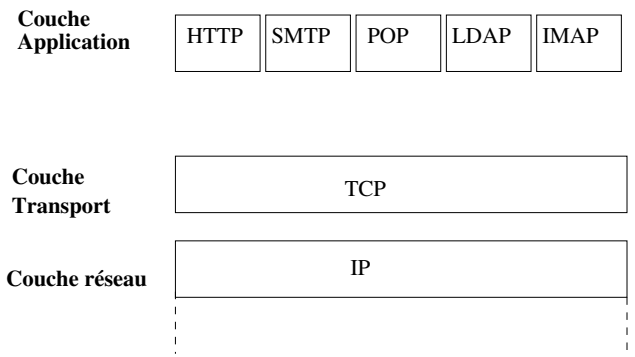
DÉPARTEMENT
Sciences Informatiques

Applications

- Protocole SSL/TLS
  - Web sécurisé : HTTPS
  - envoi de courrier : SMTPS
  - accès distant à la messagerie : POPS, IMAPS
  - VPN : IPSec
  - Niveau applicatif : utilisateur/mot de passe
- Chiffrement et/ou signature du courrier électronique : S/MIME

SSL/TLS

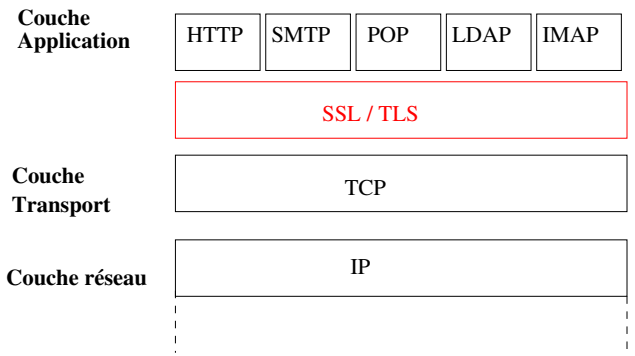
- développé à l'origine par Netscape (SSL version 2 et SSL version 3)
- renommé en Transport Layer Security (TLS) par l'IETF suite au rachat du brevet de Netscape par l'IETF en 2001.
- TLS/SSL s'insère entre la couche réseau TCP/IP et la couche application
- Pas (ou peu) de modifications aux logiciels client et serveur et au protocole applicatif.
- Tunnel SSL sécurisé



DÉPARTEMENT Sciences Informatiques

SSL/TLS

- développé à l'origine par Netscape (SSL version 2 et SSL version 3)
- renommé en Transport Layer Security (TLS) par l'IETF suite au rachat du brevet de Netscape par l'IETF en 2001.
- TLS/SSL s'insère entre la couche réseau TCP/IP et la couche application
- Pas (ou peu) de modifications aux logiciels client et serveur et au protocole applicatif.
- Tunnel SSL sécurisé



DÉPARTEMENT Sciences Informatiques

Etablissement de connexion SSL

1. Authentification du serveur auprès du client (certificat serveur)
2. Choix d'un algorithme de chiffrement symétrique acceptable par le client et le serveur pour l'établissement de la connexion sécurisée
3. Optionnellement authentification du client auprès du serveur (certificat client)
4. Echange des secrets partagés nécessaires à la génération d'une clé secrète (clé de session) pour le chiffrement symétrique ;
5. Etablissement d'une connexion SSL chiffrée à clé secrète.

DÉPARTEMENT
Sciences Informatiques

Crédits I

- http ://fr.wikipedia.org/wiki/Infrastructure_à_clés_publiques
- TLS (RFC 2246, RFC 4346, RFC 5246)
- DTLS (RFC 4347)
- Public-Key Infrastructure (X.509)
  (http ://www.ietf.org/dyn/wg/charter/pkix-charter.html)
-

DÉPARTEMENT
Sciences Informatiques