



Cours 14

Crypto

Cryptographie

- Définition
 - Science du **chiffrement**
 - Meilleur moyen de protéger une information = la rendre **illisible** ou **incompréhensible**
- Bases
 - Une **clé** = chaîne de nombres binaires (0 et 1)
 - Un **Algorithme** = fonction mathématique qui va combiner la clé et le texte à crypter pour rendre ce texte illisible

Cryptographie

Chiffrement Symétrique

- Une **Clé Secrète** (Unique) partagée entre les 2 parties qui sert pour le **chiffrement** et le **déchiffrement** du message



Cryptographie

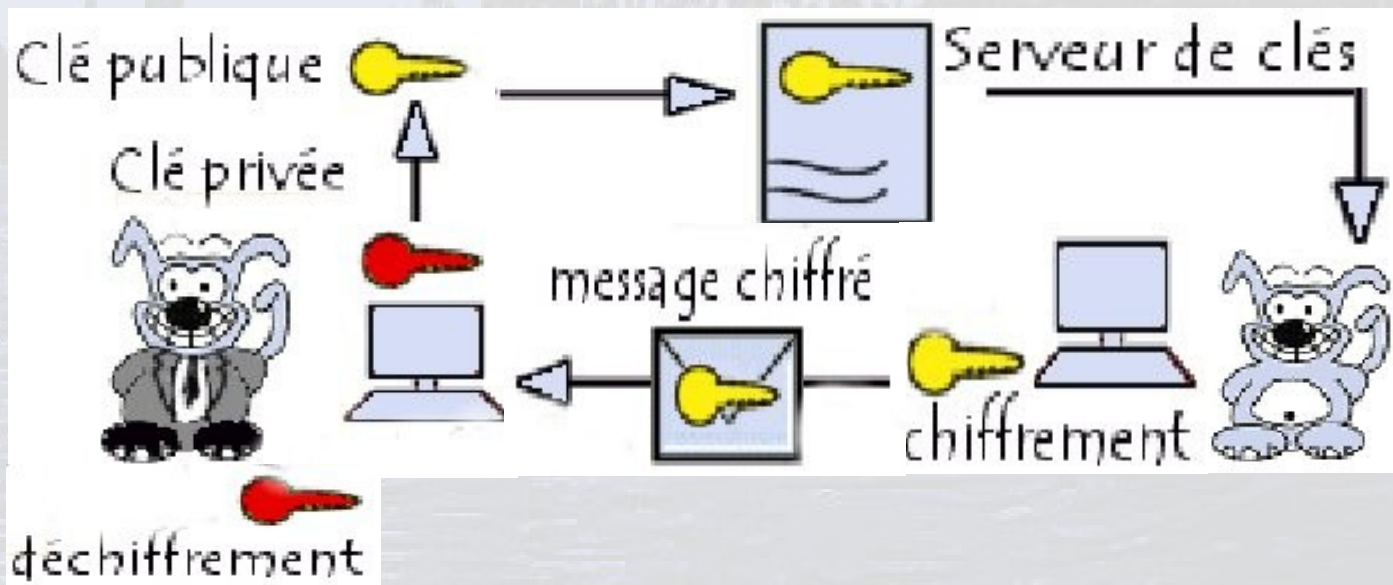
Chiffrement Symétrique

- Algorithmes utilisant ce système :
 - **DES** (Data Encryption Standard, très répandu) : les données sont découpées en blocs de 64 bits et codées grâce à la clé secrète de 56 bits propre à un couple d'utilisateurs
 - **IDEA, RC2, RC4**
- Avantage :
 - **Rapide**
- Inconvénients :
 - Il faut autant de **paires de clés** que de couples de correspondants
 - La **non-répudiation** n'est pas assurée. Mon correspondant possédant la même clé que moi, il peut fabriquer un message en usurpant mon identité
 - **Transmission** de clé

Cryptographie

Chiffrement Asymétrique

- Clé publique
 - Sert à chiffrer le message
- Clé privée
 - Sert à déchiffrer le message



Cryptographie

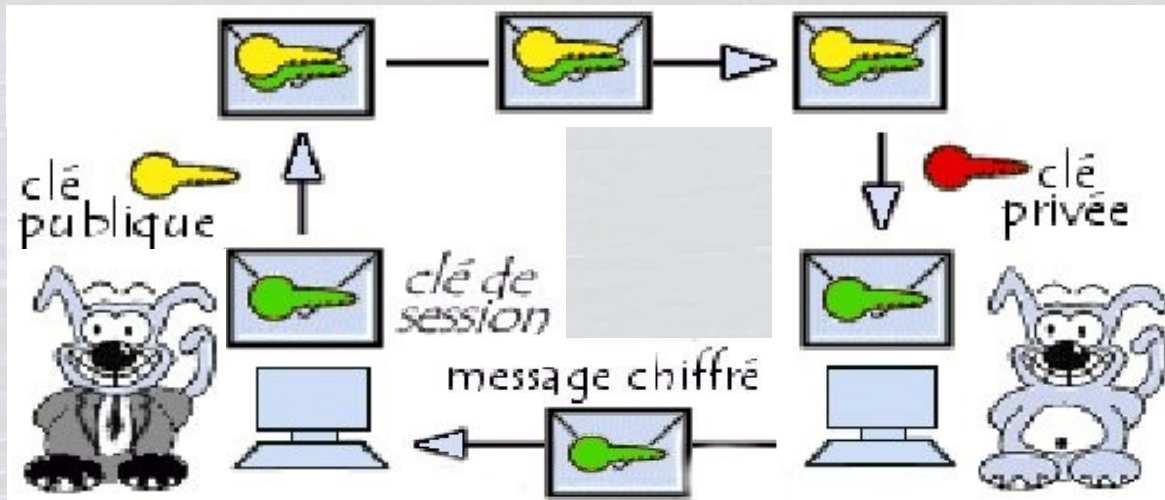
Chiffrement Asymétrique

- Algorithmes utilisant ce système :
 - RSA (Rivest, Shamir, Adelman)
 - DSA
 - ElGamal
 - Diffie-Hellmann
- Avantage :
 - pas besoin de se transmettre les clés au départ par un autre vecteur de transmission.
- Inconvénient :
 - Lenteur

Cryptographie

Combinaison des 2 Chiffrements

- Chiffrement symétrique
 - Problèmes d'échanges de clés
 - Chiffrement asymétrique
 - Problème de lenteur
- combinaison des 2 = clé de session



Authentification

Définition

- La **personne** à qui j'envoie un message crypté est-elle bien celle à laquelle je pense ?
- La **personne** qui m'envoie un message crypté est-elle bien celle à qui je pense ?

Authentification

Technique d'Identification

- **Prouveur**
 - Celui qui s'identifie, qui prétend être...
- **Vérifieur**
 - Fournisseur du service
- **Challenge**
 - Le **Vérifieur** va lancer un challenge au **prouveur** que ce dernier doit réaliser

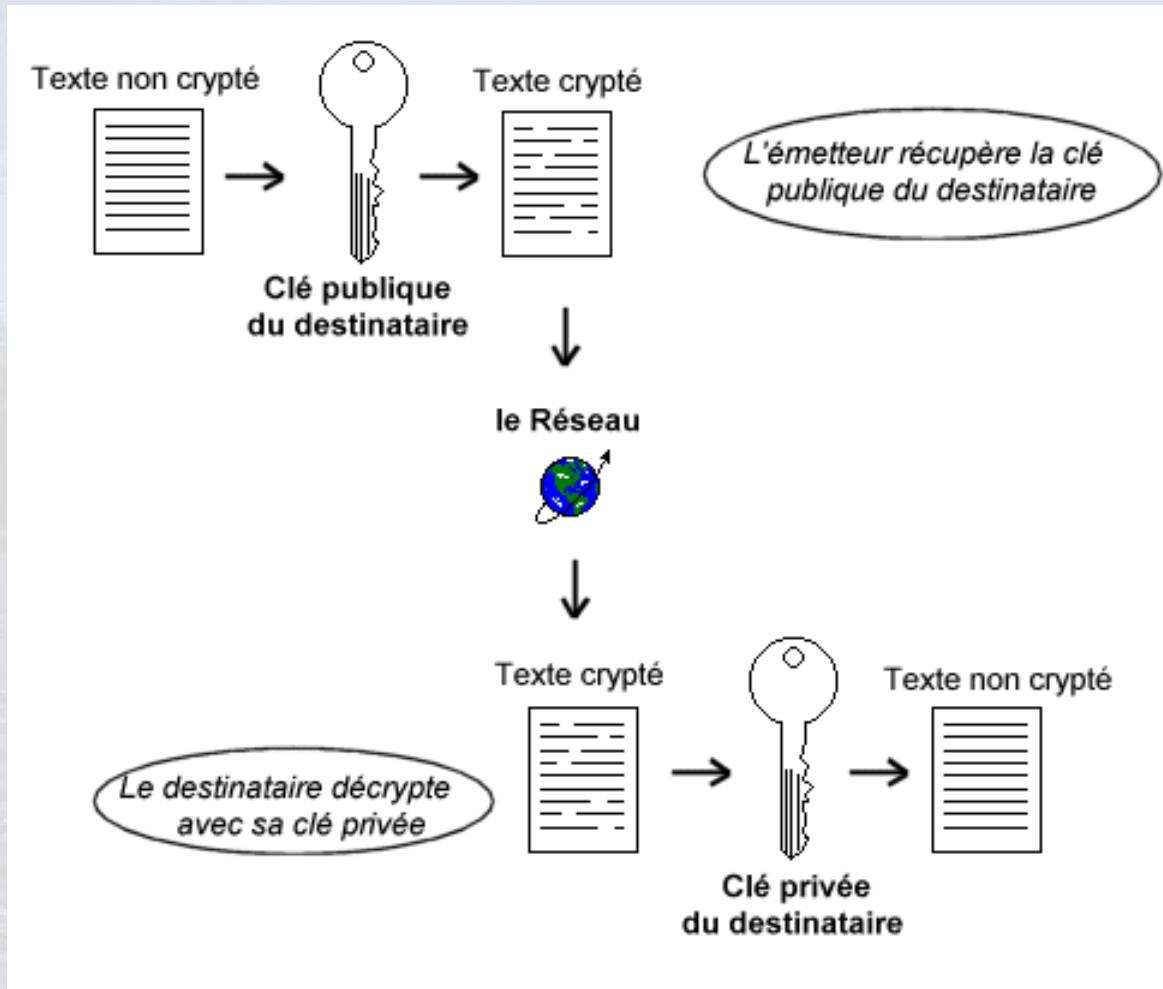
Technique A Clé Publique

Principe

- Algorithme RSA = Réversible
 - $((\text{Mess})_{C_{Pu}})_{C_{Pr}} = ((\text{Mess})_{C_{Pr}})_{C_{Pu}}$
- Confidentialité
- Authentification

Technique A Clé Publique

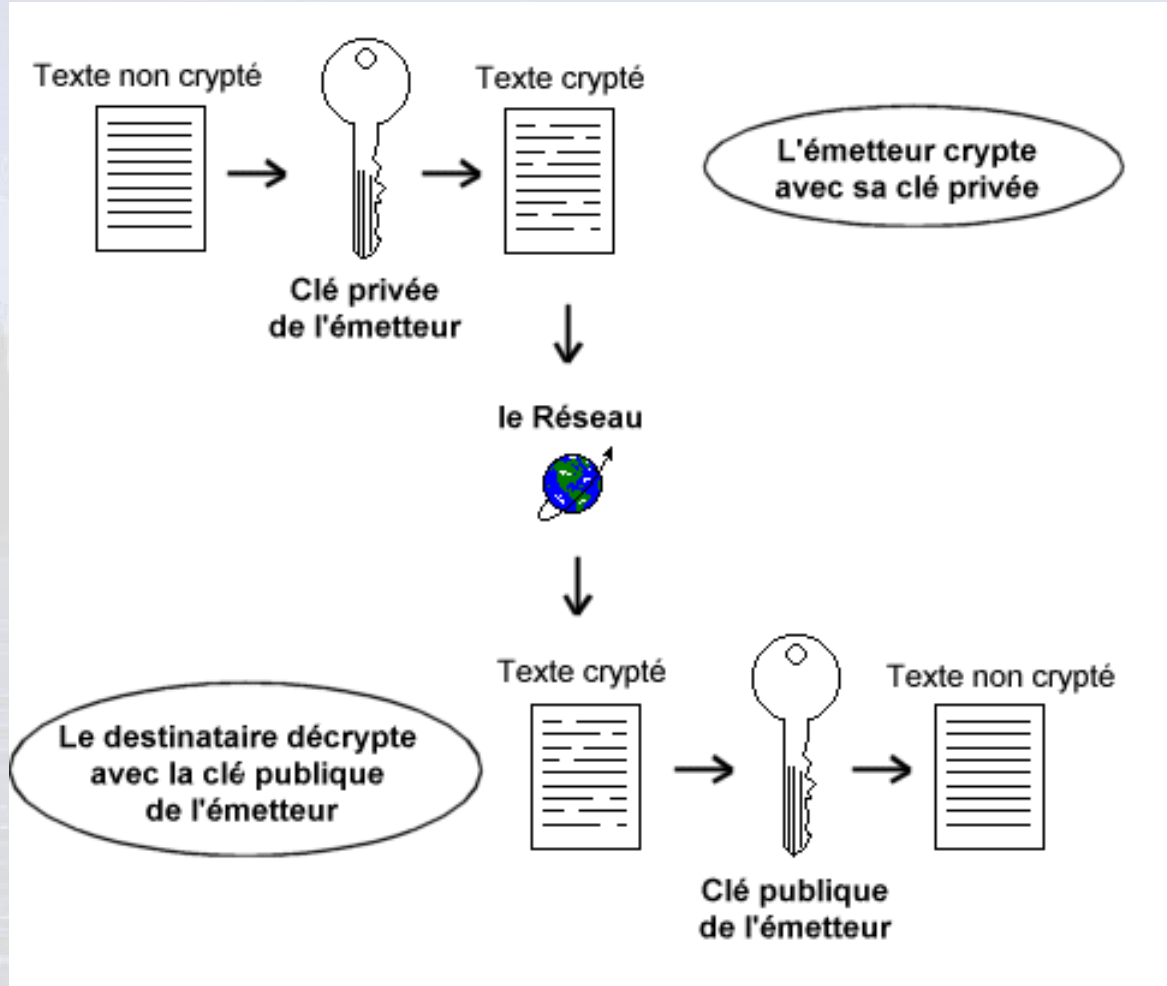
Confidentialité



Le Texte est totalement **confidentiel** car le destinataire est le seul à avoir la **clé privée**

Technique A Clé Publique

Authentification

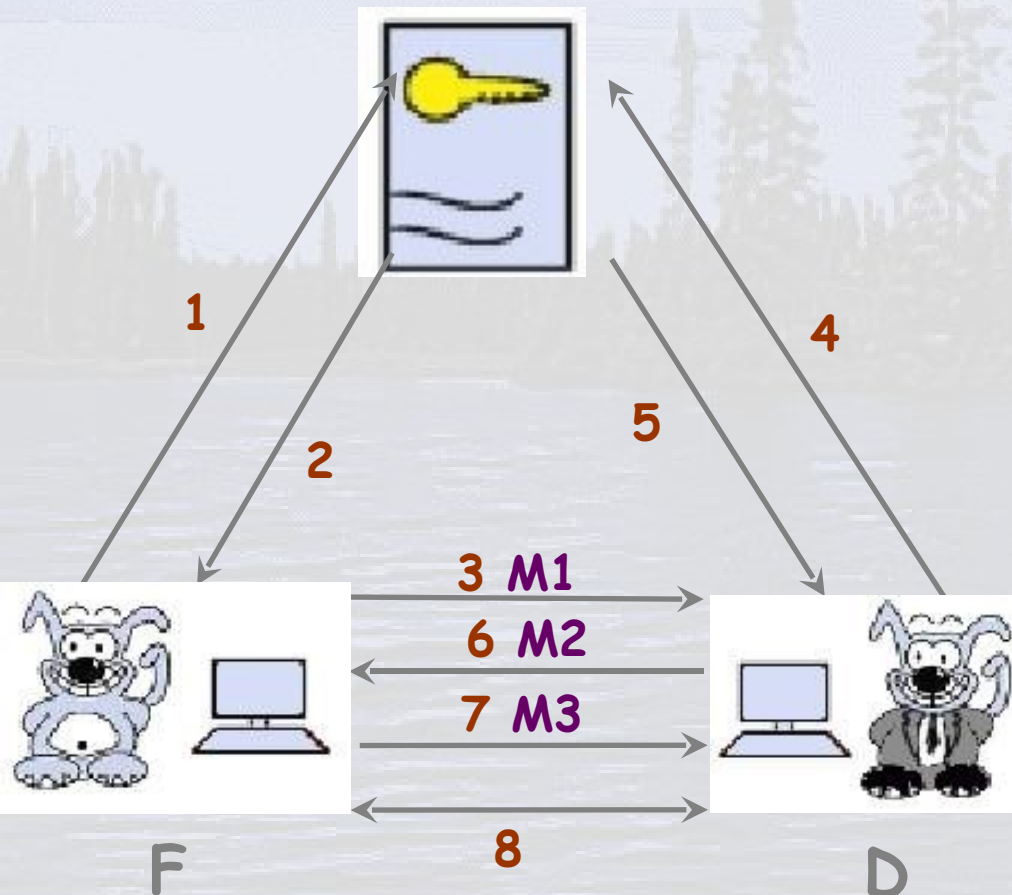


On est sûr de **l'identité** de l'émetteur car il est le seul à pouvoir chiffrer un message avec cette **clé privée**

Technique A Clé Publique

Protocole

Serveur d'authentification - Annuaire
(Clé Publiques de F & D...)



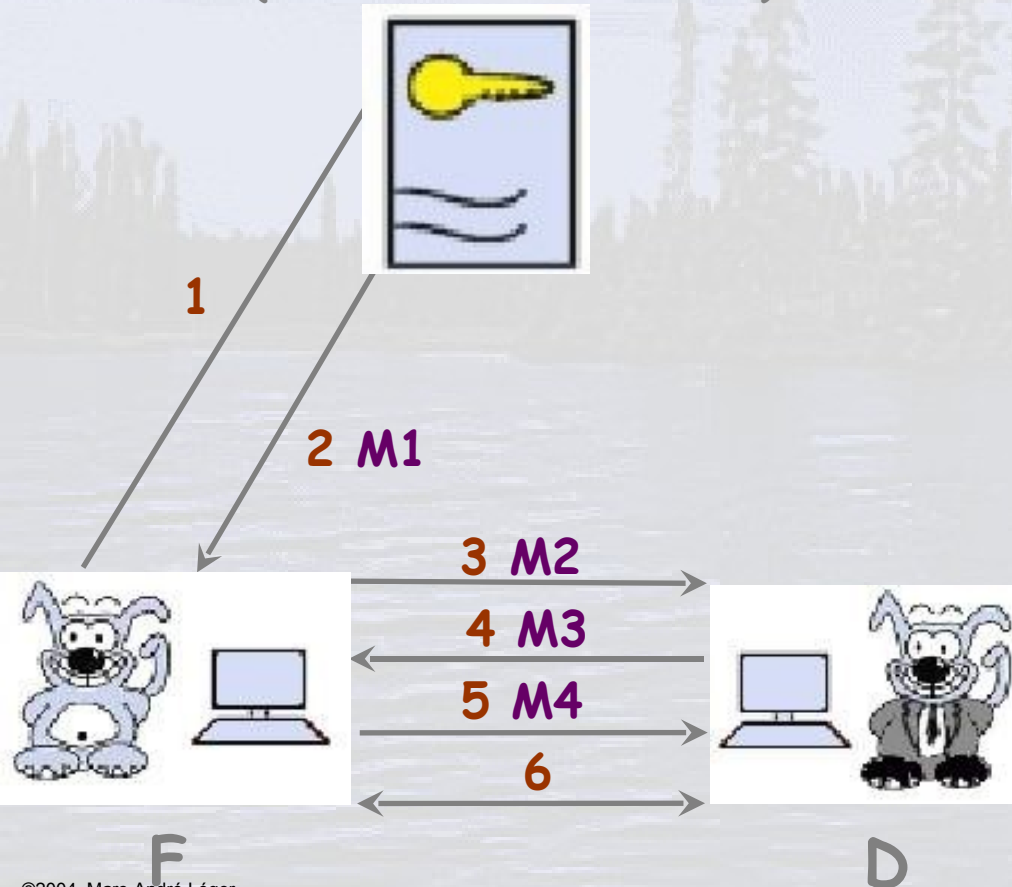
- 1) F demande la *Clé Publique* de D
- 2) S envoie la *Clé Publique* de D à F
- 3) F envoie le « challenge » à D: Décrypte mon message $M1(I_f)$ et renvoie mon I_f pour me le prouver!
- 4) D décrypte $M1$ et demande à S la *Clé Publique* de F
- 5) S envoie la *Clé Publique* de F à D
- 6) A son tour D envoie un « challenge » à F: Décrypte mon message $M2(I_f, I_d)$ et renvoie mon I_d !
- 7) F décrypte $M2$ et renvoie $M3(I_d)$ à D pour lui montrer qu'il y est arrivé
- 8) F & D peuvent maintenant par ex s'envoyer des messages en créant une *Clé Privée* à partir de (I_f, I_d)

Technique A Clé Secrète

Protocole de Needham – Schroeder

Serveur d'authentification - Annuaire

(Clés Secrètes de F & D)



- 1) F demande une *Clé de Session* pour pouvoir parler avec D
- 2) S envoie à F **M1** crypté par la *Clé Secrète* de F:

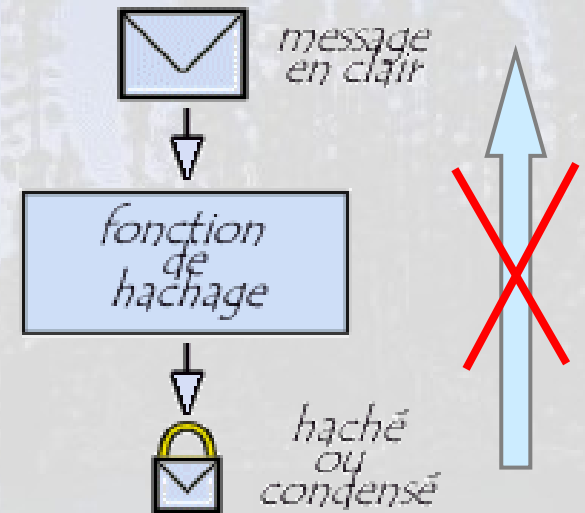
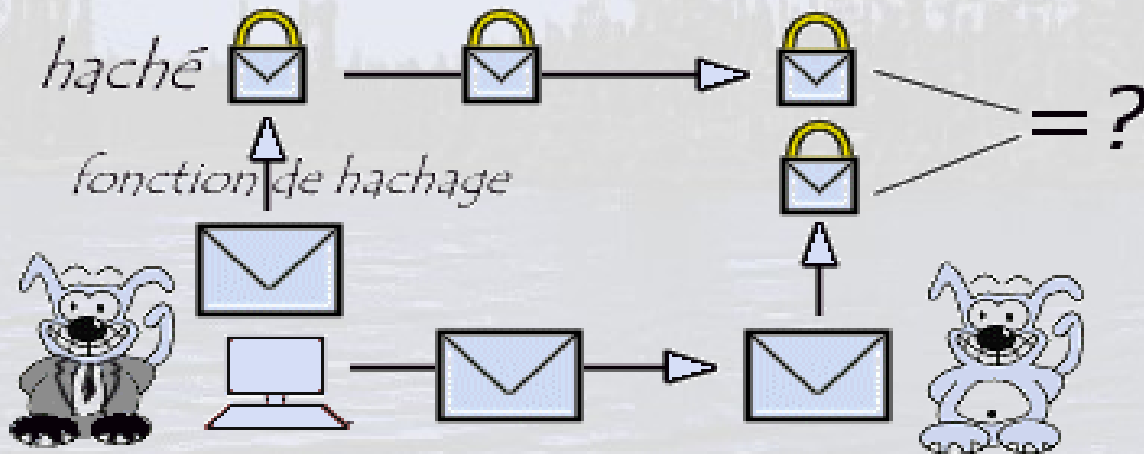
M1 = une *Clé de Session* **CSfd** en clair et une cryptée par la *Clé Secrète* de D (**CSfd**)**CPd**

- 3) F envoie le « challenge » à D: Décrypte mon message **M2**((**CSfd**)**CPd**) et renvoie un **Id** crypté par **CSfd**
- 4) D décrypte **M2** et envoie son « challenge » : Décrypte mon message **M3**((**Id**)**CSfd**) et renvoie **Id-1**
- 5) F décrypte **M3** et renvoie **M4**((**Id-1**)**CSfd**)
- 6) F & D peuvent donc s'envoyer des messages avec la *Clé de Session* (**MESSAGE**)**CSfd**

Signature électronique (1)

– Comment savoir que le message n'a pas été **altéré** ?

→ **fonction de hachage**

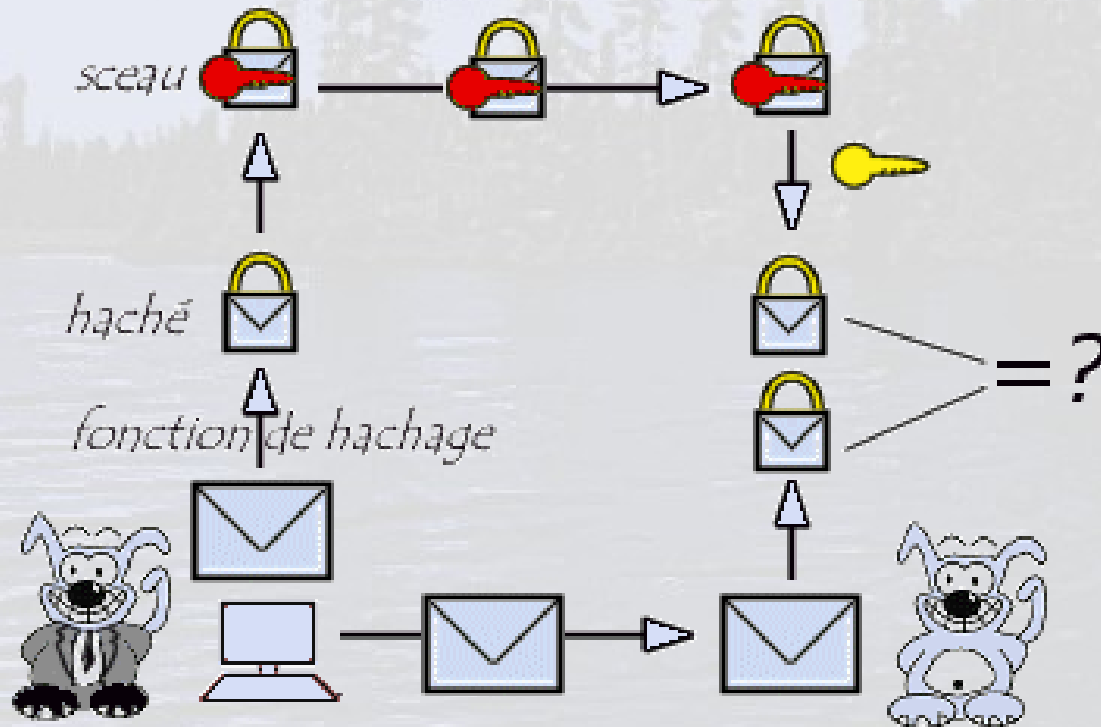


– algorithmes de hachage les plus utilisés: **MD5** (128 bits) et **SHA** (160 bits)

Signature électronique (2)

-Pb du hachage : on est pas sur de
l'expéditeur

→ **Scellement des données**



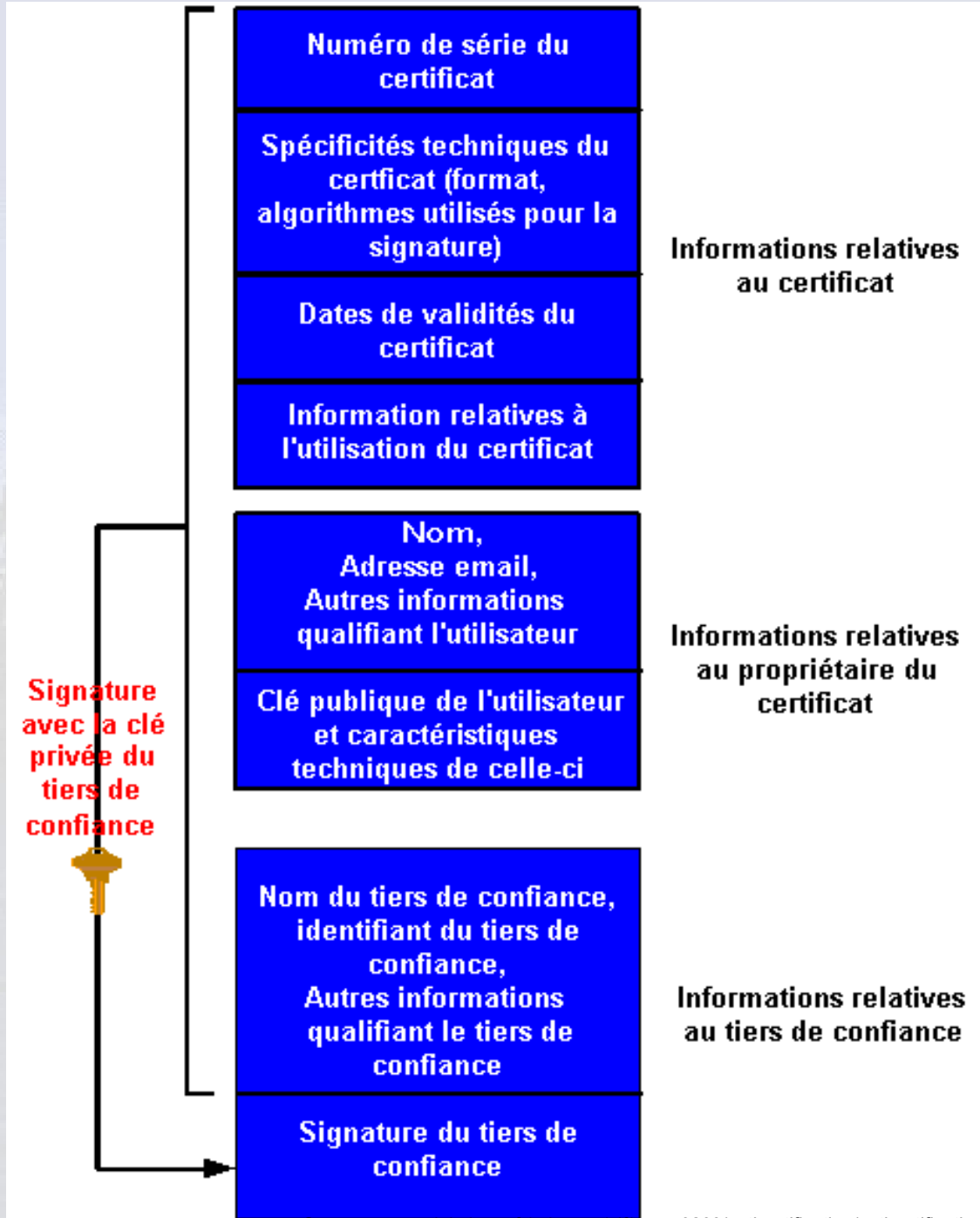
Certification

Principes

- Besoins
 - Chiffrement asymétrique = basé sur la distribution de **clés publiques** (Annuaire)
 - rien ne **garantit** que la clé est bien celle de l'utilisateur a qui elle est associée...
 - **Certificats**
- Certificats
 - **Carte d'identité électronique**, composée de la **clé publique du porteur** et d'**informations** relatives à ce dernier.
 - Délivré par une autorité appelée **tiers de confiance**, qui, par sa signature, en garantit l'**authenticité**.

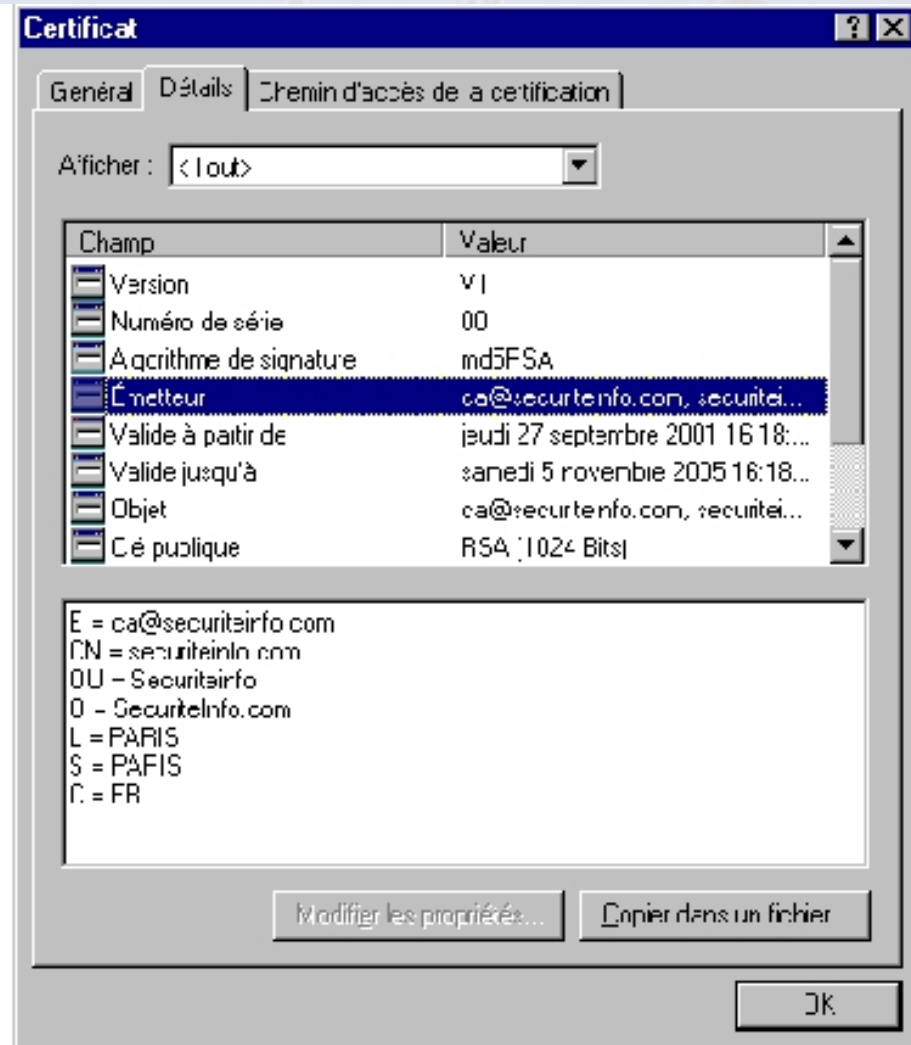
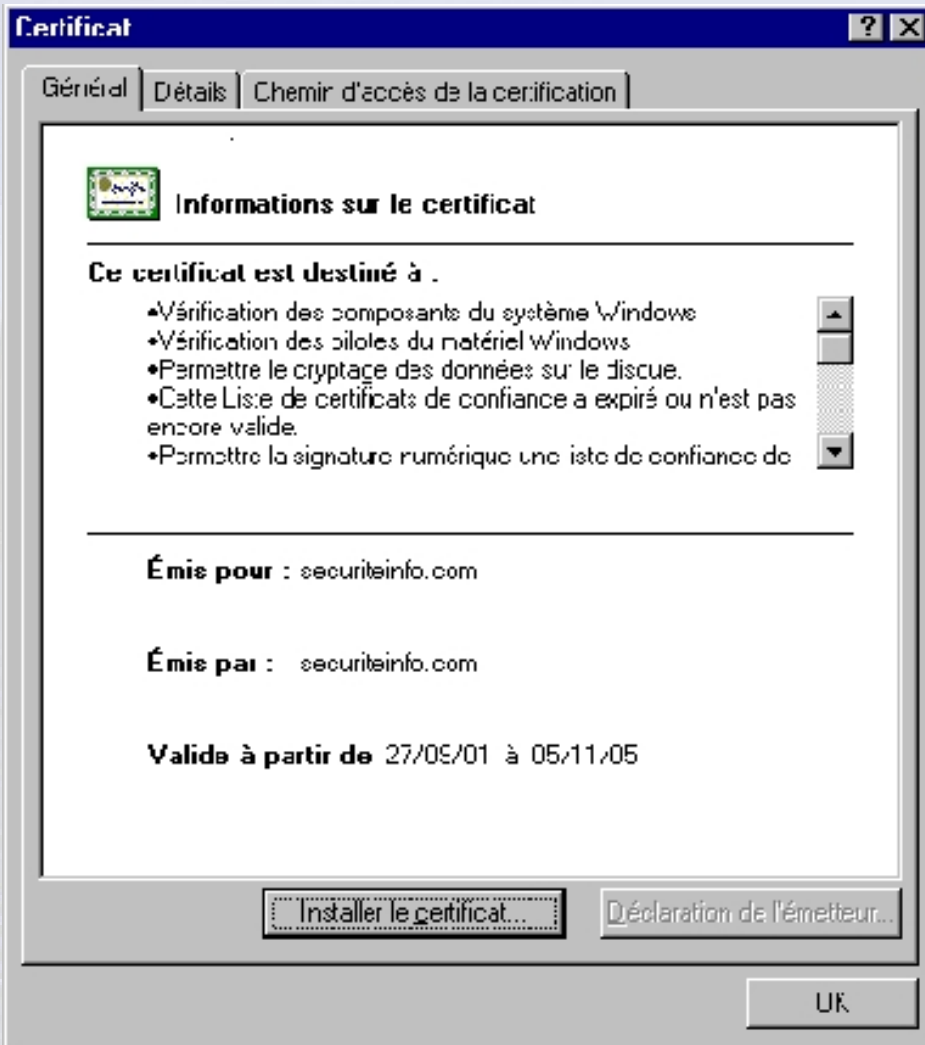
Certification

Certificat X509



Certification

Exemple Certificat X509



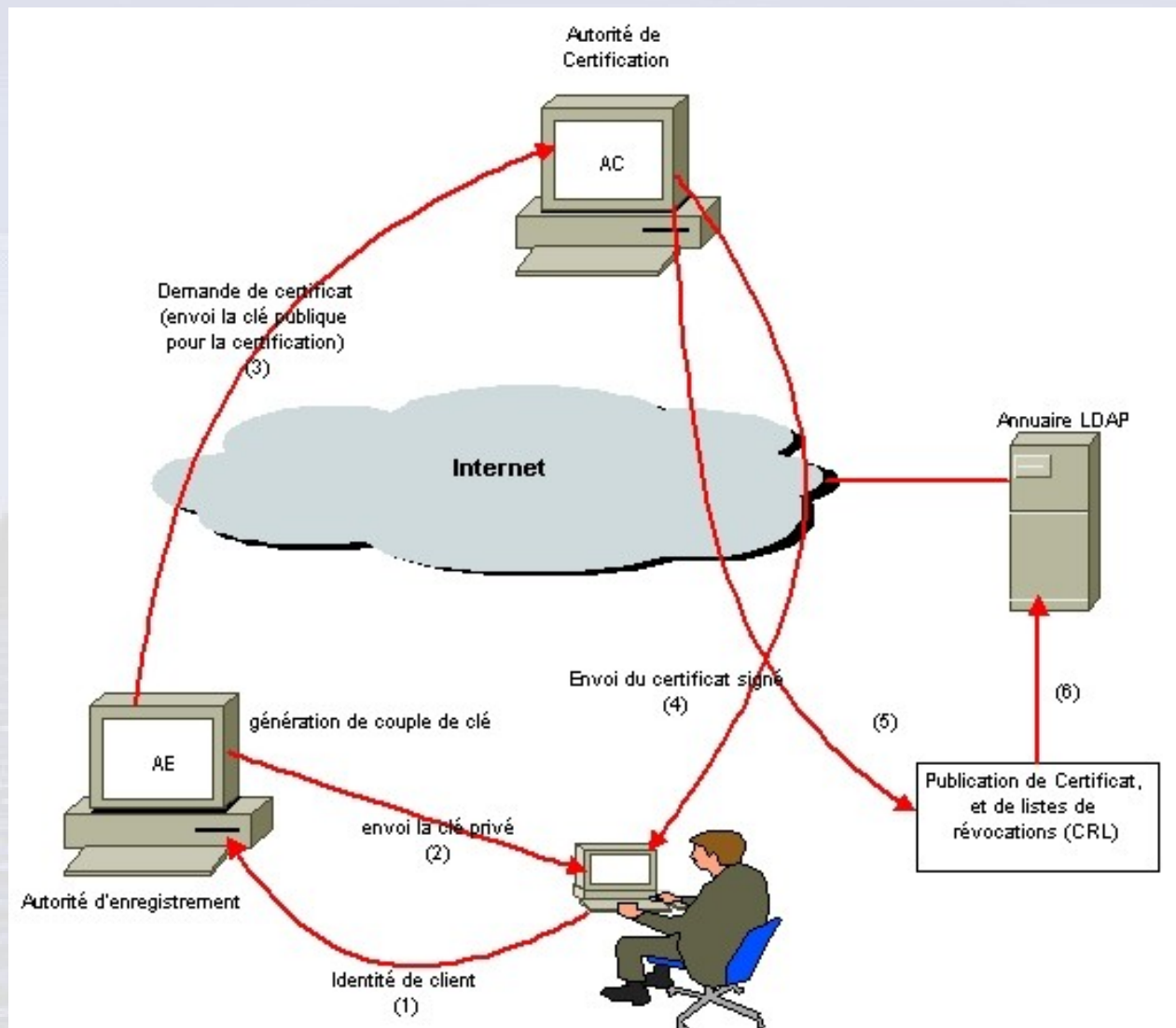
Certification

PKI (Public Key Infrastructure)

IGC (Infrastructure de Gestion de Clés)

- Système permettant la gestion de clés de chiffrement et la délivrance de certificats numériques
- Repose sur l'utilisation de la cryptographie à clé publique

PKI - Organisation



PGP (Pretty Good Privacy)

Introduction

- PGP est un **cryptosystème** (système de chiffrement)
- inventé par **Philip Zimmermann**, un analyste informaticien
- Il est très **rapide et sûr** ce qui le rend quasiment impossible à **cryptanalyser**

PGP

- Principes

- Hybride = Repose sur la Combinaison de la **cryptographie à clé publique** et la **cryptographie à clé secrète**

- Étapes du chiffrement

- PGP crée une **clé secrète IDEA** de manière aléatoire, et chiffre les données avec cette clé.
- PGP chiffre la **clé secrète IDEA** précédemment créée au moyen de la **clé RSA publique** du destinataire

- Étapes du Déchiffrement

- PGP déchiffre la **clé secrète IDEA** au moyen de la **clé RSA privée**.
- PGP déchiffre les données avec la **clé secrète IDEA** précédemment obtenue.

PGP

Fonctionnalités

- Signature électronique et vérification d'intégrité de messages
- Chiffrement des fichiers locaux : fonction utilisant IDEA.
- Génération de clefs publiques et privées
- Gestion des clefs:
 - Distribution de la clé publique aux personnes voulant envoyer un message
- Certification de clefs:
 - Ajout d'un sceau numérique pour garantir l'authenticité des clés publiques
- Révocation, désactivation, enregistrement de clefs

PGP

Format des certificats

- Le **numéro de version** de PGP
 - Version de pgp avec lequel a été créé le certificat
- La **clef publique** du détenteur du certificat:
 - Partie publique de la bi-clé
- Les **informations** du détenteur du certificat
 - nom, ID utilisateur, photographie, etc.
- La **signature numérique** du détenteur du certificat :
 - = auto signature = signature effectuée avec la clef privée correspondant à la clef publique associée au certificat.
- La **période de validité** du certificat:
 - Dates/heures de début et d'expiration du certificat
- L'**algorithme de chiffrement symétrique**:
 - CAST, IDEA ou DES

PGP

PGP versus X509

	PGP	X509
Autorité de certification	Tous les utilisateurs	1 seule
Signature Numérique	Plusieurs	1 seule
Détenteur de clé	Plusieurs	1 seul
Révocation	Émetteur + ceux ajoutés par l'émetteur comme autorité de révocation	Émetteur Seul

Microsoft .NET Passport

- service en ligne gratuit
- permet de se connecter (en toute sécurité ?) à n'importe quel service ou site Web Passport participant
- Utilisation d'une adresse de messagerie et d'un mot de passe unique

Microsoft .NET Passport

- Contenu obligatoire
 - Email (nom d'utilisateur)
 - Mot de passe
- Contenu optionnel
 - Phrase de rappel
 - Clé de sécurité
 - Numéro de mobile
 - Date de naissance, coordonnées
 - Informations bancaires

Microsoft .NET Passport

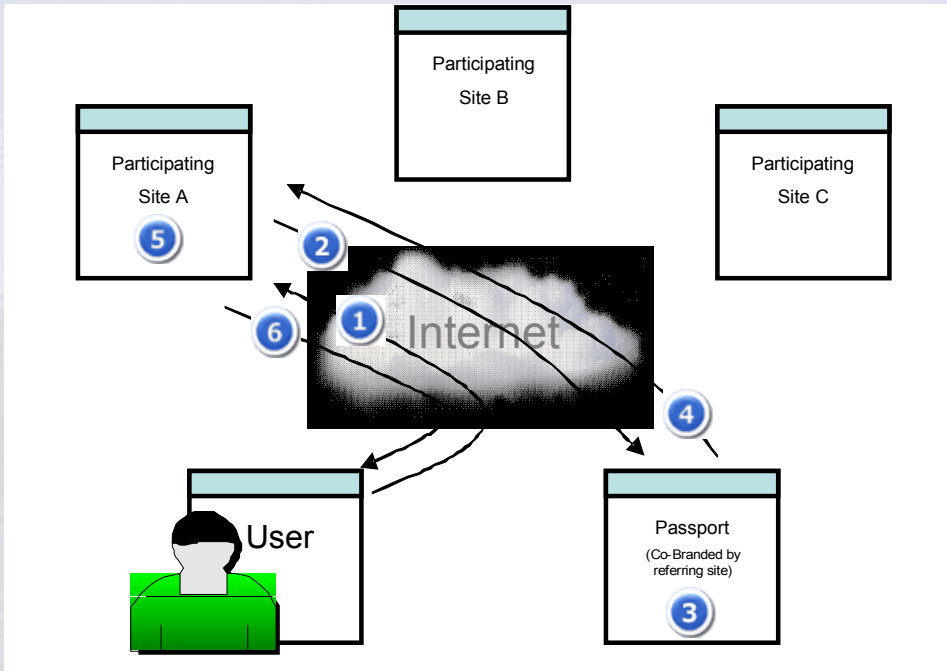
-L'utilisateur contacte un site

-L'utilisateur est redirigé sur le site Passport

-L'utilisateur s'authentifie et reçoit un cookie chiffré

-L'utilisateur est redirigé vers le premier site qui lit le cookie

-L'utilisateur reste authentifié pour tout autre site



Participating Site

Internet

Kerberos

Introduction

- Conditions de fonctionnement
 - Les serveurs ne font aucune confiance aux clients
 - Les clients n'accordent qu'une confiance limitée aux serveurs
 - Authentification contrôlée par des serveurs spécialisés

Kerberos

Service d'Authentification

- Pré-requis
 - Le serveur Kerberos détient les mots de passe utilisateurs
 - Le serveur détient la clé privée du serveur de tickets
 - Le serveur de tickets détient les clés privés de tous les serveurs

Kerberos

Lexique

- Ticket

Caractérise une session entre un client C et un serveur S

$T_{cs} = \{S, C, adr, Td, durée, Kcs\}Ks$

- Adr : adresse IP du client
- Td : heure de début de session
- Durée : durée max de session
- Kcs : clé de session partagée par C et S
- Ks : clé permanente (secrète) de S

Kerberos

Lexique

- Authentifieur

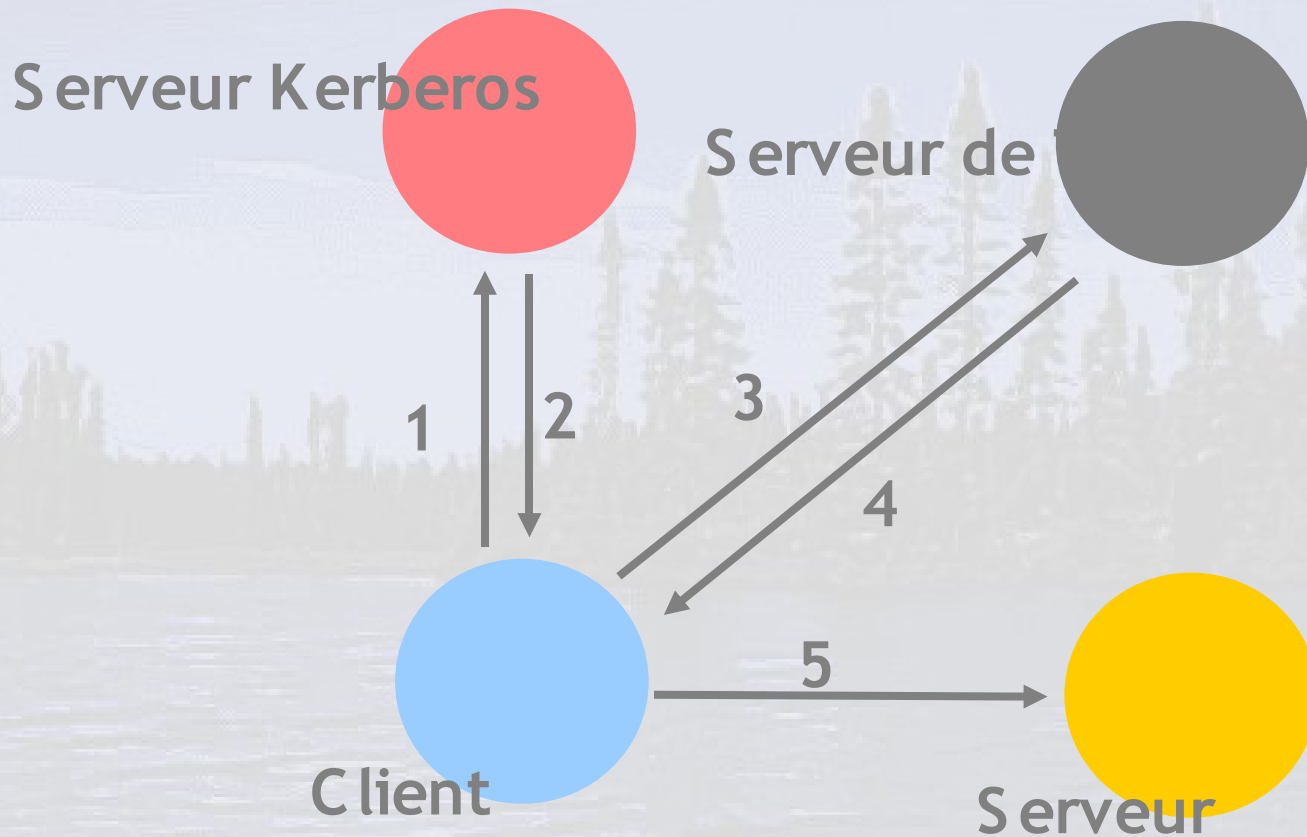
caractérise le client à un instant, vis à vis d'un serveur

$$Acs(t) = \{C, adr, t\}K_{Cs}$$

- Engendré par le client
- Permet une authentification permanente par le serveur

Kerberos

Service d'Authentification



Le serveur de tickets renvoie un ticket pour la discussion
Le client envoie sa requête et le ticket et un ticket pour le serveur
Le message contient le ticket et un authentificateur chiffré
par la clé de session
Le serveur de tickets - Client le tout chiffre par le mot
de passe client

SSL (Secure Sockets Layer)

- Définition
 - « Couche de Sockets Sécurisée »
 - Protocole d'échange de données au dessus de TCP/IP qui assure:
 - Confidentialité des échanges entre 2 applications
 - Authentification des serveurs
 - Indépendant du protocole Utilisé (HTTP, FTP, ...)

SSL (Secure Sockets Layer)

- Principe
 - Utilise **RSA** (clé publique) pour s'échanger des clés **DES** (clé Secrète)
 - Protocole de négociation (choix clés)
 - Protocole d'échange (chiffré par DES)
 - Authentifie un navigateur, pas une personne
- Compatibilité
 - Presque Tous les Navigateurs
 - Affichage du cadenas en bas pour les sites Sécurisés
 - Un serveur sécurisé possède une URL commençant par **https://**

SSL

Phase de Négociation

- Authentification
 - Utilise des **certificats** émis par une autorité de certification
 - Authentifier le **serveur** vis à vis du client (navigateur)
 - Authentifier le **navigateur** vis à vis du serveur
- Génération des clés de session
 - Technique à **clé publique** vue précédemment
 - Création des **clés de session**
- Fin de négociation
 - Client & serveur sont authentifiés mutuellement
 - Ils ont leurs **clés secrètes** pour la phase d'échange