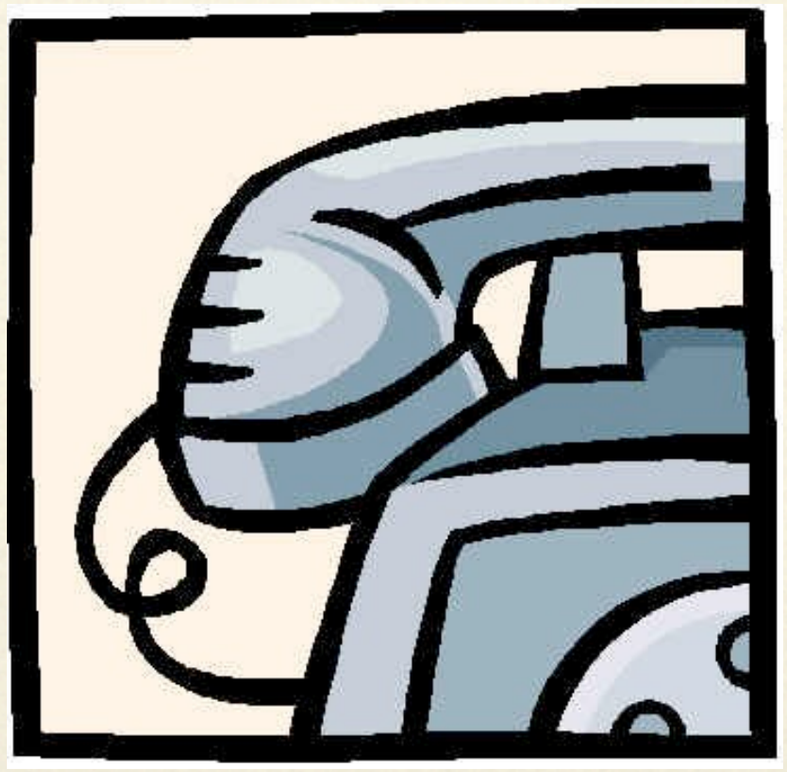
7. Sécurité

M1 Outils de l'Internet
lundi 9 novembre 2009

victor.poupet@lif.univ-mrs.fr

Communications sécurisées



Communication sur Internet

- Discussion
- Courrier
- Lecture de données
- Envoi de données
- Paiement
- ...

Ce que l'on veut pouvoir faire

- Savoir à qui l'on parle (authentification)
- Être sûr que le message envoyé n'est pas modifié (intégrité)
- Faire en sorte que le message échangé ne puisse pas être intercepté (confidentialité)

Contrôle d’accès

Dans la vie de tous les jours

- Restreindre l'accès à une ressource :
  - Code PIN sur un DAB
  - Clé sur une porte de voiture
  - Videur de boîte de nuit

Dans un système informatique

- Des utilisateurs
- Des sujets (processus, en général liés à un utilisateur)
- Des objets (ressources)

Contrôle d'accès

- Authentification (*Authentication*)
- Autorisation (*Authorization*)
- Comptabilité (*Accounting*)

1. Authentification

- Vérifier l'identité de l'utilisateur
- Obtenue à l'aide de mot de passe, ticket à usage unique, certificat numérique, numéro de téléphone, données biométriques, etc.

Critères

- On peut authentifier un utilisateur selon plusieurs paramètres :
  - Ce qu'il sait (mot de passe, chemin secret)
  - Ce qu'il possède (clé, carte)
  - Ce qu'il est (empreinte, voix)
  - Où il est (adresse IP, numéro de téléphone, dans un sous-réseau)

2. Autorisation

- Déterminer si un sujet a accès à l'objet demandé
- Les autorisations dépendent de l’administrateur du serveur
- Parfois l’autorisation est faite à l’aide de tickets (sans authentification)

Principe de moindre privilège

- *principle of least privilege*
- Chaque sujet n'a accès qu'aux objets dont il a besoin pour fonctionner
- Irréalisable en pratique, on se contente d'une faible granularité (invités, utilisateurs, administrateurs, etc.)

Modèles d'autorisation

- Par listes de contrôle d'accès (ACL) :
  Les utilisateurs sont identifiés, on associe à chaque ressource une liste d'utilisateurs autorisés à y accéder
- Par capacité (*capability*) :
  Les sujets ont la capacité d'agir directement sur les ressources, et ils peuvent la transmettre (par exemple un *file descriptor* qui permet d'écrire dans un fichier)

Confused Deputy Problem

- Faille de sécurité
- Idée : un sujet fait effectuer une tâche pour laquelle il n'est pas autorisé par un autre sujet (autorisé) sans qu'il s'en aperçoive
- Problème fréquent lorsque l'on ne respecte pas le principe de moindre privilège

Techniques de contrôle d'accès

- Contrôle d'accès discrétionnaire (DAC) : le propriétaire de chaque ressource décide des autorisations
- Contrôle d'accès obligatoire (MAC) : règles globales sur les autorisations (niveaux de sécurité imposés)
- Contrôle d'accès par rôles (RBAC) : les utilisateurs choisissent ou reçoivent un rôle et ne peuvent effectuer que des tâches autorisées pour ce rôle

3. Comptabilité

- Enregistrer la quantité de service livrée à un utilisateur
- On peut par exemple enregistrer le nom de l'utilisateur, le service fourni, les dates de début et de fin.

Utilité de la comptabilité

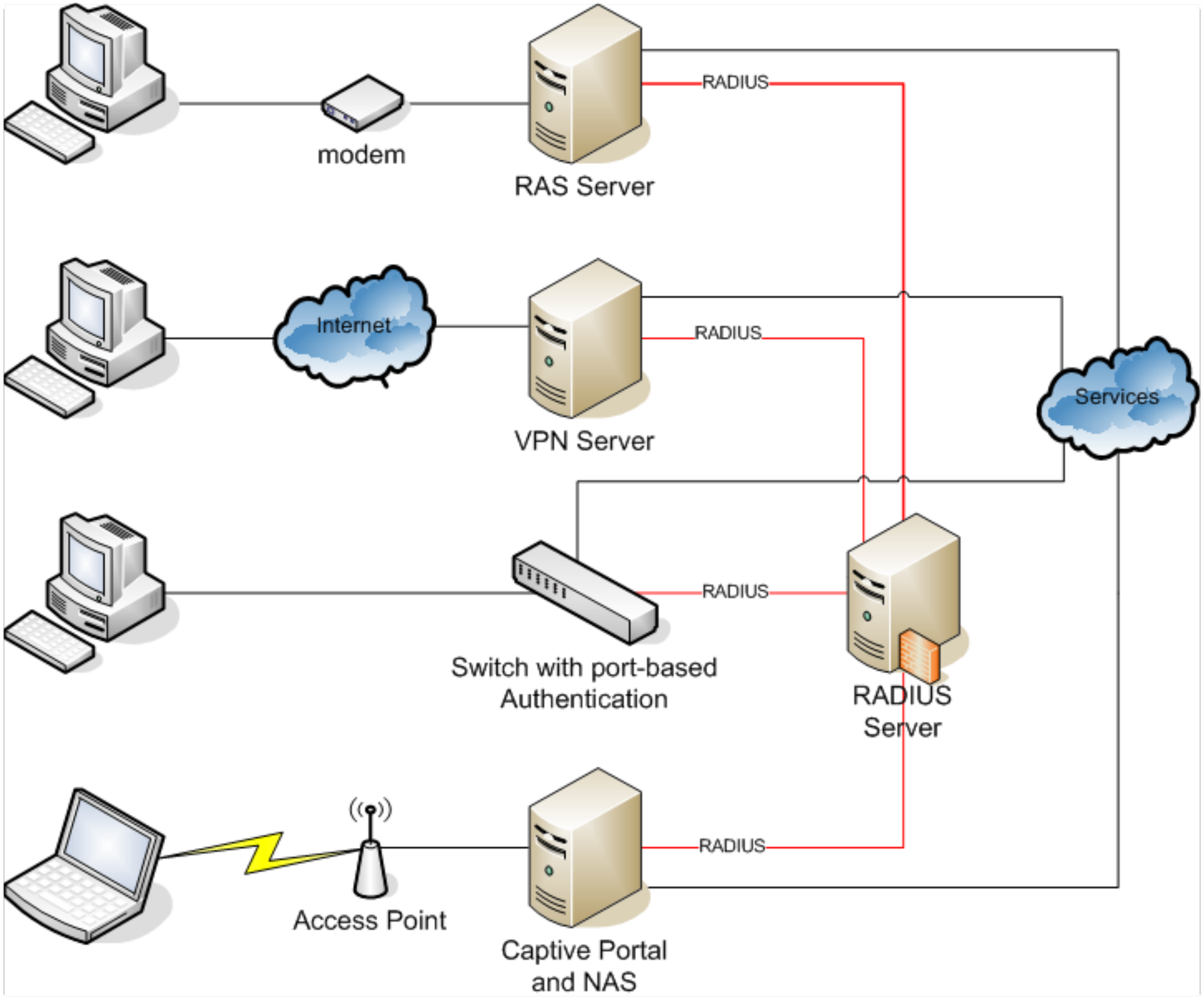
- Statistiques
- Facturation
- Prévisions
- Détection et compréhension de risques de sécurité

RADIUS

- Remote Authentication Dial In User Service
- Protocole AAA
- Client-Serveur en UDP
- Utilisé par les FAI ou les administrateurs réseaux
- Développé en 1991, devenu un standard IETF

Organisation

- Un serveur RADIUS
- Plusieurs passerelles pour accéder au réseau (reliées au serveur principal) :
  - Remote Access Server
  - Virtual Private Network Server
  - Network Switch
  - Network Access Server



Source : http://en.wikipedia.org/wiki/File:Drawing_RADIUS.png

RADIUS : Authentification

- L'utilisateur envoie une requête à un NAS (par PPP)
- Le NAS contacte le serveur RADIUS :
  - informations de créance (mot de passe, certificat, etc.)
  - autres informations (n° de téléphone, adresse IP, etc.)

Network Access Server

- Point d'accès unique à une ressource distante
- Fonctionne comme une passerelle
  - Le client se connecte au NAS
  - Le NAS interroge une autre ressource pour savoir si l'identification du client est valide
  - Le NAS autorise (éventuellement) l'accès à la ressource

Network Access Server

- Le NAS n'effectue ni l'authentification ni l'autorisation
- Exemples :
  - WiFi : Navigateur → NAS → serveur AAA → Internet
  - VoIP : L'adresse IP ou le n° de téléphone est considéré pour savoir si la communication est acceptée

RADIUS : Autorisation

- Le serveur RADIUS vérifie l'identité (PAP, CHAP ou EAP) à l'aide d'informations locales ou distantes (SQL, LDAP, etc.)
- Les autorisations sont déterminées
- La réponse est envoyée au NAS (ainsi que des informations de connexion : IP, temps limite de connexion, etc.)

Roaming

- RADIUS permet l'identification entre différents opérateurs (protocole commun)
- Utilisé par exemple par *eduroam*
- Quand un utilisateur se connecte sur un réseau, le NAS du réseau interroge un serveur RADIUS commun aux réseaux pour obtenir l'authentification
- *username@realm*

RADIUS : Comptabilité

- Début de comptabilité demandée par le NAS au serveur RADIUS avec l'autorisation
- Au cours de la connexion, d'autres enregistrements de comptabilité peuvent être demandés par le NAS
- Fin de comptabilité à la déconnexion

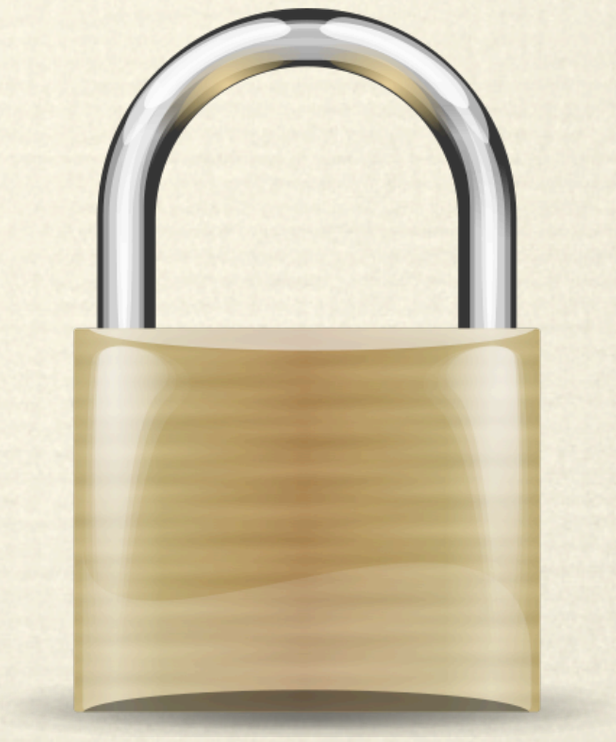
RADIUS

- Echanges entre NAS et RADIUS cryptés par une clé commune et un hash MD5 (éventuellement tunnel IPsec)
- Facilement extensible
- Utilise UDP comme couche de transport

Diameter

- Le successeur de RADIUS
- Fonctionne en TCP
- Supporte TLS
- Pas vraiment rétro-compatible, mais le passage de RADIUS à Diameter est assez simple
- Plus facile à étendre

Un peu de cryptographie



PAP

- *Password Authentication Protocol*
- Authentification par mot de passe
- On envoie l'identifiant et le mot de passe en clair

A n'utiliser que lorsque la connexion est sécurisée...

CHAP

- *Challenge Handshake Authentication Protocol*
- RFC 1994
- Schéma d'authentification de serveurs PPP
- Clé partagée
- La vérification peut être répétée à n'importe quel moment

CHAP

- Le serveur peut lancer un *défi* (*challenge*) au client :
  - Génération et envoi d'une chaîne de caractères
  - Le client envoie le MD5 de la concaténation de la chaîne et du mot de passe
  - Le serveur effectue le même codage et vérifie l'égalité des deux résultats

CHAP : Caractéristiques

- Résistant aux attaques par *replay*
- Les deux partis doivent connaître la clé commune
- La clé n'est jamais transmise sur le réseau
- Seul le client est authentifié

MS-CHAP

- Variante développée par Microsoft
- Le mot de passe n'est pas stocké en clair sur le serveur
- Peut être utilisé pour une authentification à double sens

MS-CHAP (en gros)

- Le serveur envoie un *challenge*
- Le client effectue un hash de ce *challenge* avec un hash de son mot de passe et ajoute une chaîne
- Le serveur vérifie que le *challenge* a bien été hashé, et renvoie un hash de la chaîne transmise par l'utilisateur

Extensible Authentication Protocol

- RFC 3748
- Modèle d'authentification
- Principalement utilisé dans les réseaux sans fil (utilisé dans la norme WPA)
- Définit le format des messages, c'est la méthode choisie qui encapsule ensuite les messages
- Il existe de nombreuses méthodes

LEAP

- *Lightweight Extensible Authentication Protocol*
- Méthode propriétaire (CISCO)
- Version modifiée de MS-CHAP
- Les informations ne sont pas très protégées
- Très vulnérable si la clé n'est pas complexe

EAP-TLS

- EAP-Transport Layer Security
- Standard IETF (RFC 2716)
- Protocole TLS
- Utilisation de clés publiques pour communiquer avec le serveur d'authentification

EAP-MD5

- IETF (RFC 3748)
- Faible niveau de sécurité
- Vulnérable aux attaques par dictionnaire
- Authentification du client mais pas du serveur

EAP-PSK

- *Pre-Shared Key* (clé échangée au préalable)
- RFC (4764)
- Sécurisé
- Ne nécessite pas de cryptographie à base de clé publique
- Pas toujours pratique d'échanger la clé

Et les autres...

- EAP-TTLS
- EAP-IKEv2
- PEAP
- EAP-FAST
- EAP-SIM
- EAP-AKA
- ...

Transport Layer Security

- Anciennement *Secure Sockets Layer* (SSL)
- Protocole de cryptographie
- Authentification mutuelle avant de communiquer (PKI)
- Basé sur l'algorithme RSA

TLS : Développement

- SSL v1 : Netscape. Jamais publié.
- SSL v2 : 1995. Nombreuses failles.
- SSL v3 : 1996.
- TLS : IETF (RFC 2246), 1999.

TLS : Fonctionnement

1. Négociation des algorithmes utilisés
2. Echanges des clés et authentification
3. Cryptage symétrique et authentification du message

TLS : Handshake

- Le client se connecte au serveur, propose une liste d'algorithmes supportés
- Le serveur choisit l'algorithme
- Le serveur envoie un certificat (nom, CA, clé publique)
- Le client génère une chaîne aléatoire, la crypte à l'aide de la clé publique du serveur et l'envoie
- Les deux participants peuvent alors générer les clés à partir de la chaîne transmise

TLS : Sécurité

- Le client vérifie l'identité du serveur par le CA
- L'algorithme commun le plus sûr est utilisé
- Compteur de messages, utilisation des MAC
- Le message concluant la poignée de mains envoie un hash des messages échangés
- Deux hachages sont utilisés en parallèle (MD5 et SHA-1)

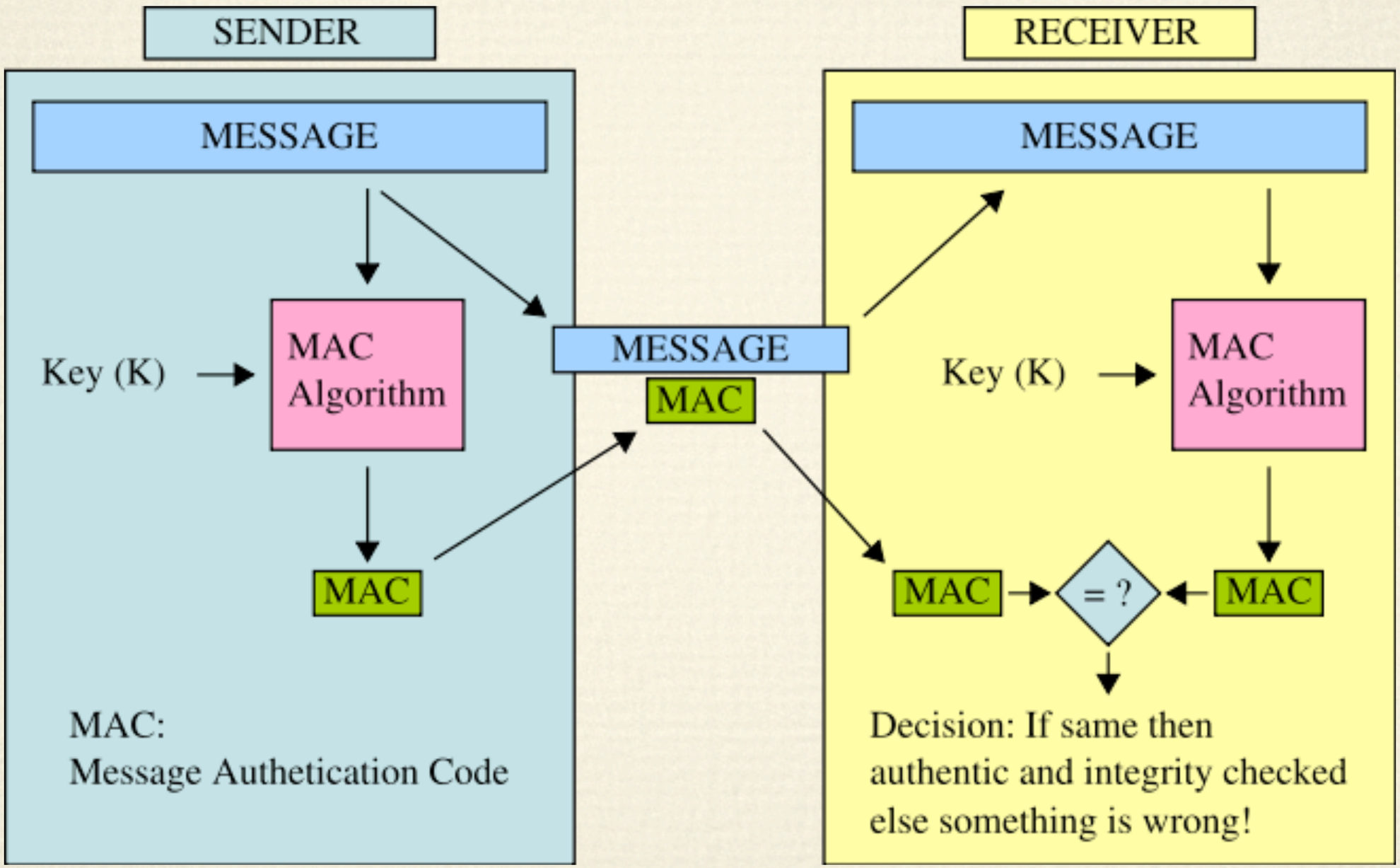
Failles de SSL v2

- La même clé est utilisée pour l'authentification et le cryptage
- Le gouvernement américain limite la sécurité des MAC (40 bits)
- La fin de la communication n'est pas explicite (fermeture TCP)

Message Authentication Code

- Chaîne de caractères permettant d'authentifier un message
- Certifie l'intégrité et l'authenticité
- Pas besoin de crypter la communication (peut être fait après le cryptage)
- Utilise une clé secrète commune
- Version sans clé : message integrity code (MIC)

Message Authentication Code

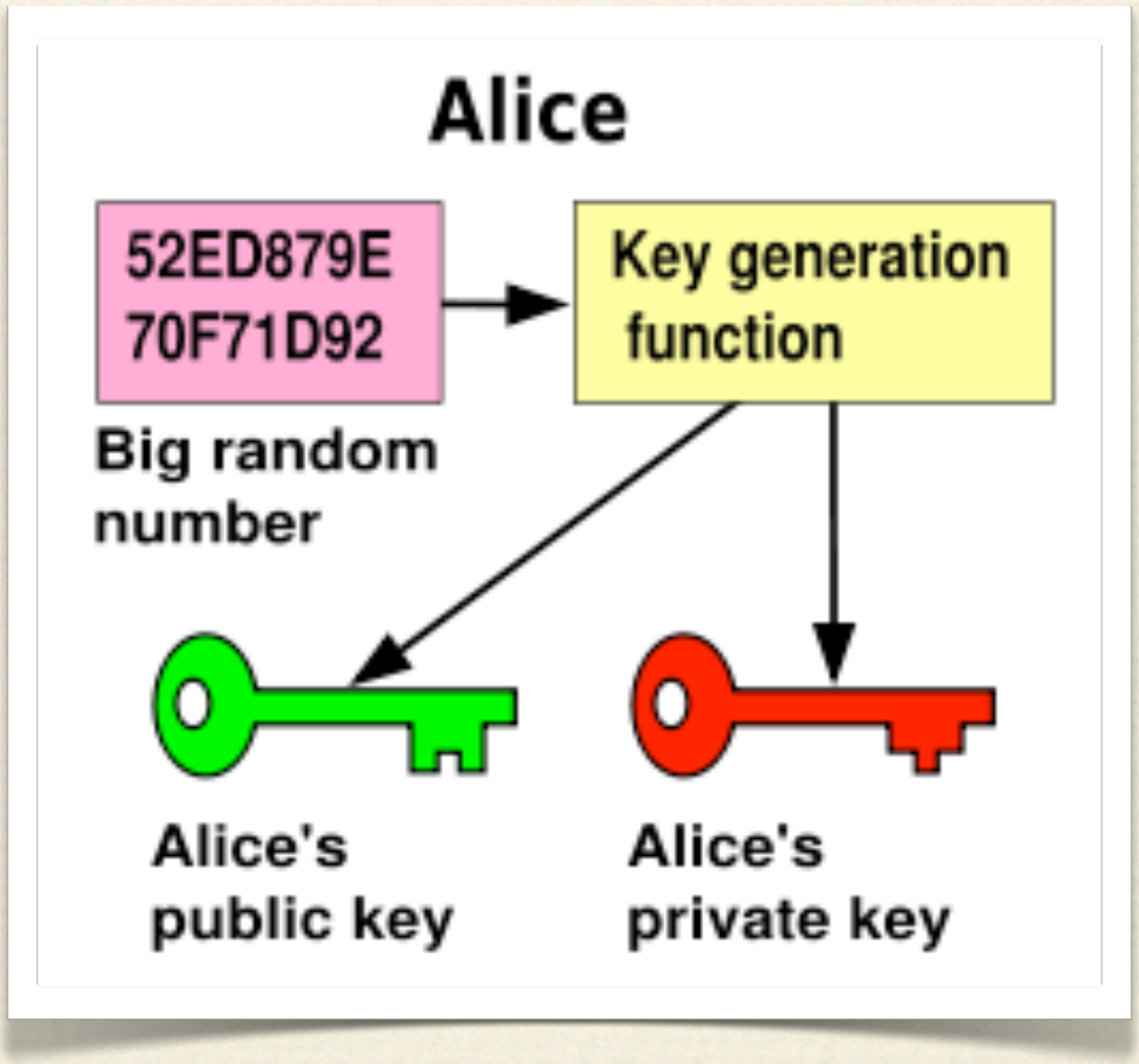


Source : http://en.wikipedia.org/wiki/Message_authentication_code

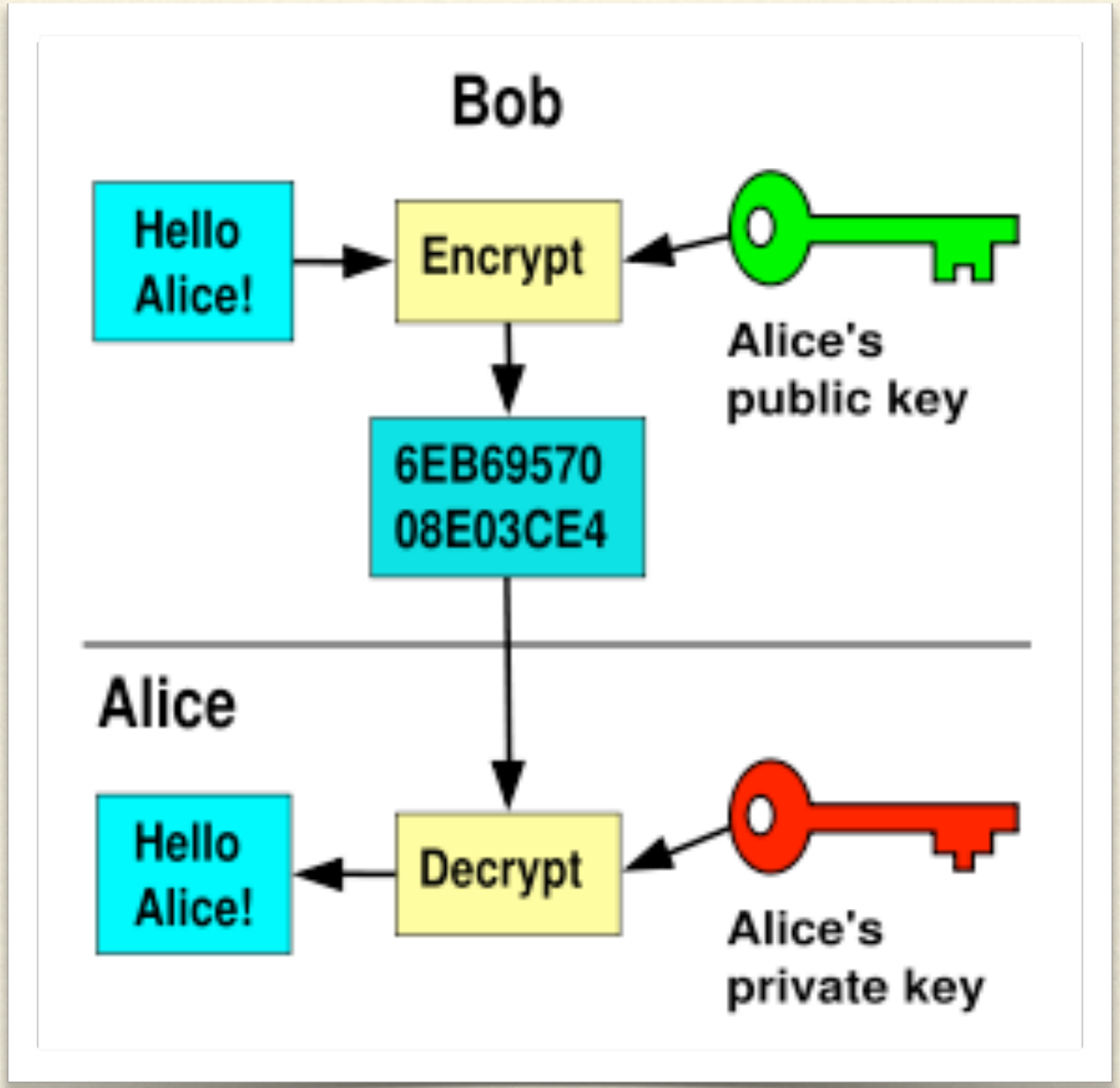
Clés publiques/privées

- Chaque utilisateur a une clé publique et une clé privée (secrète)
- Par la magie des fonctions *one-way* on peut crypter avec une clé et décrypter avec l'autre tout en gardant le secret

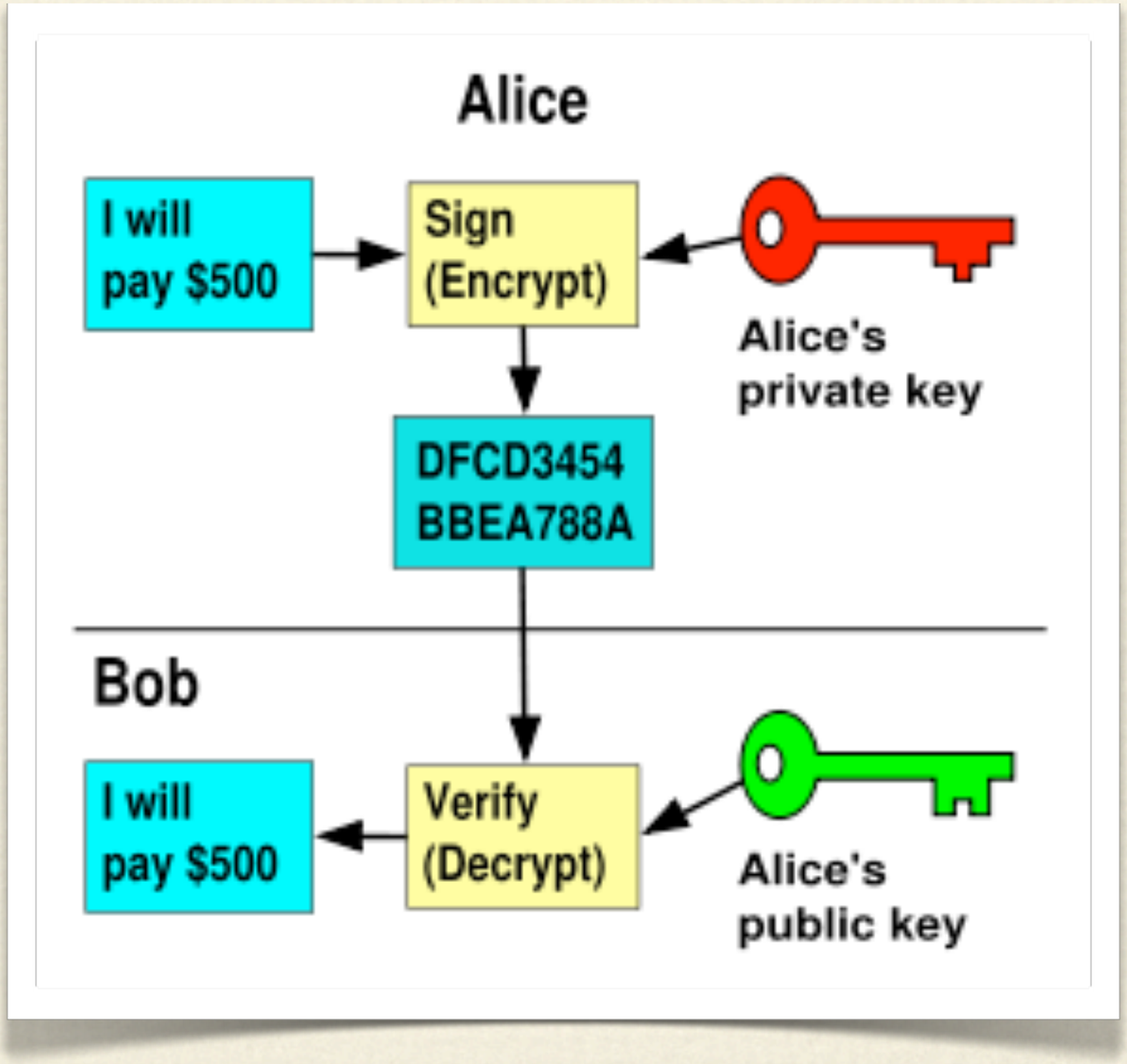
Clés publiques/privées



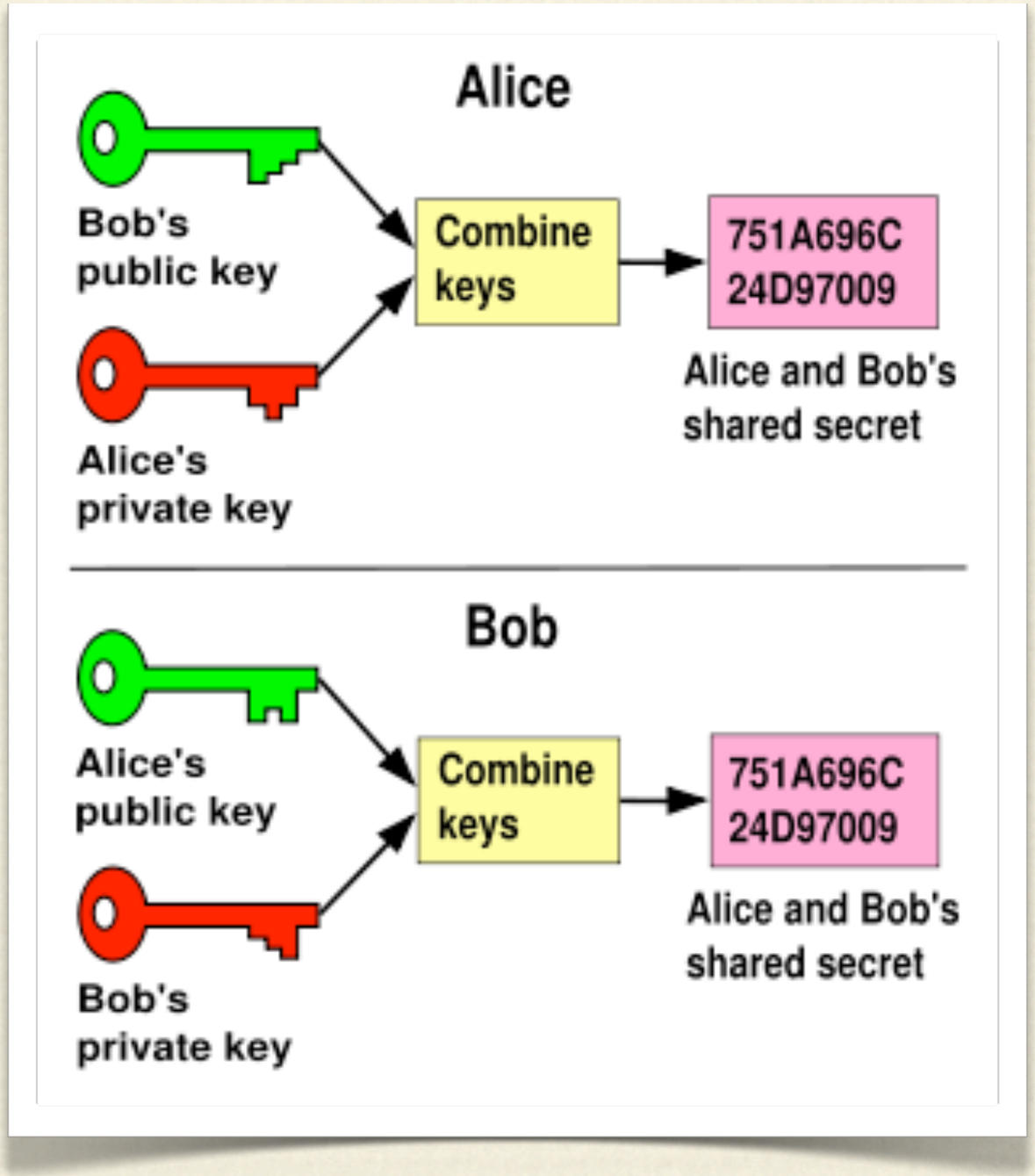
Clés publiques/privées



Clés publiques/privées



Clés publiques/privées



Quelques difficultés

- Comment change-t-on de clé ?
- Comment ajoute-t-on un utilisateur ?
- Comment sait-on que la clé publique d'Alice est bien celle d'Alice ?

Vue d’ensemble

Public Key Infrastructure

- Chaque utilisateur génère un couple de clés privée/publique
- L'utilisateur enregistre sa clé publique auprès d'une autorité de certification (CA)
- L'enregistrement est assuré par une autorité d'enregistrement (RA)
- L'utilisateur reçoit un certificat (infalsifiable) qui lie sa clé à son identité

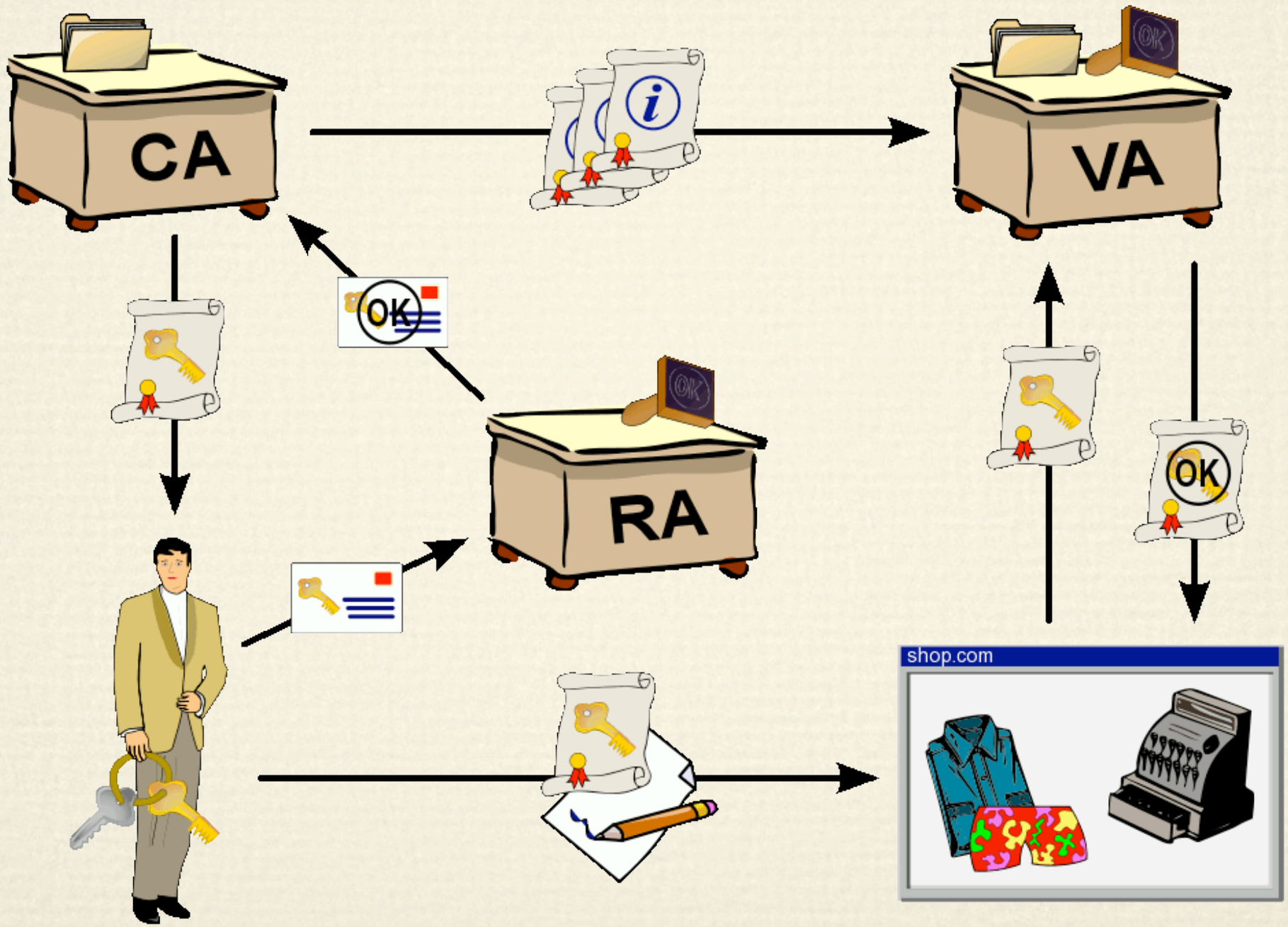
Public Key Infrastructure

- Lors d'une communication, l'utilisateur transmet sa clé publique ainsi que son certificat
- Le correspondant peut alors vérifier la validité du certificat auprès d'un validateur (VA)

Certificats

- La clé publique
- L'identité du propriétaire de la clé
- Une date d'expiration
- L'adresse du serveur de révocation
- La signature numérique, encodée à l'aide de la clé privée du CA

Schéma de fonctionnement



http://en.wikipedia.org/wiki/Image:Public-Key-Infrastructure.svg

PKI : Hiérarchie

- Les utilisateurs sont enregistrés chez leur CA “local” (entreprise)
- Les informations sont souvent liées à un annuaire LDAP
- Les CA sont eux-mêmes enregistrés chez des CA de plus haut niveau...

PKI : Révocation

- Que faire quand un certificat n'est plus valide ?
- Tenir à jour une liste de certificats révoqués (CRL)
- Ces CRL doivent être accessibles à tout moment (risque d'attaque *DoS* sur le serveur)
- Les CRL correspondent aux CA (et sont authentifiées à l'aide du certificat du CA)

PKI : Révocation

- Alternative aux CRL : *Online Certificate Status Protocol*
  - Requête de validité d'un certificat à un serveur dédié
  - Moins d'encombrement réseau
  - Les CRL restent confidentielles

Web of Trust

"As time goes on, you will accumulate keys from other people that you may want to designate as trusted introducers. Everyone else will each choose their own trusted introducers. And everyone will gradually accumulate and distribute with their key a collection of certifying signatures from other people, with the expectation that anyone receiving it will trust at least one or two of the signatures. This will cause the emergence of a decentralized fault-tolerant web of confidence for all public keys."

Phil Zimmermann, 1992

Web of Trust

- Système de PGP, OpenPGP, etc.
- Etablir le lien entre un utilisateur et sa clé publique
- Système décentralisé (pas de CA)
- Les utilisateurs transmettent leurs listes de confiance

Web of Trust

- Lorsqu'un utilisateur accepte de faire confiance à un certificat, il le signe
- Un utilisateur peut donner des règles qui permettent de faire confiance à des certificats (ou le faire manuellement)
- Problème de révocation (latence) et difficulté d'insertion

Chemins de confiance

- Principe des graphes “petit-monde” : on obtient rapidement des chaînes de confiances disjointes (donc meilleure confiance)
- Événements organisés pour faire se rencontrer des utilisateurs qui se signent mutuellement leurs clés (avec preuve d’identité)

Kerberos

- Système d'authentification (comme PKI et PGP)
- Cryptographie par clés symétriques
- Nécessite un tiers de confiance

Kerberos : Histoire

- Développé au MIT pour sécuriser le projet Athena
  - Versions 1 à 3 internes au MIT
  - v4 : fin des années 1980
  - v5 : 1993, RFC 1510
  - Maintenu et distribué librement par le MIT

Kerberos : principe

- Alice contacte le serveur d'authentification en lui transmettant un secret commun
- Le serveur envoie un ticket à Alice
- Ces tickets servent à prouver à Bob que l'authentification a bien eu lieu
- Alice peut utiliser son ticket pour en obtenir d'autres (afin de rester authentifiée plus longtemps)

Kerberos : Inconvénients

- Le serveur d'authentification doit être disponible à tout moment (possibilité d'utiliser plusieurs serveurs)
- Les horloges des clients et du serveur doivent être synchronisées (les tickets ont une faible durée de vie)
- Toutes les clés sont sur le serveur