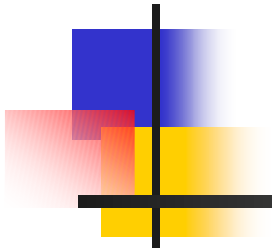


# NSY107 - Intégration des systèmes client-serveur

Cours du 24/06/2006, max 2 heures,

**Thème : Cryptologie**

© Emmanuel DESVIGNE
<emmanuel@desvigne.org>

*Document sous licence libre (FDL)*

# Plan du cours « Cryptologie »

- Introduction, définitions
- Les règles d'or
- Les méthodes classiques de cryptanalyse
- Principaux algorithmes de chiffrement
- Signature électronique
- Certificat, tiers de confiance
- Cryptologie et réseau (802.1X, WEP/PKA, etc.)

# Introduction, définitions [1/16]

- **Cryptographie :** discipline visant à protéger des messages (en assurant confidentialité et/ou authenticité).
- **Cryptanalyse :** discipline visant à étudier et contourner les méthodes issues de la cryptographie.
- **Cryptologie :** science regroupant la cryptographie et la cryptanalyse.

# Introduction, définitions [2/16]

- **Cryptosystème :** terme utilisé en cryptographie pour désigner un ensemble composé d'algorithmes cryptographiques et de tous les textes en clairs, textes chiffrés et clés possibles.
- **Confidentialité :** (déf fournie par l'ISO) fait de s'assurer que l'information n'est seulement accessible qu'à ceux dont l'accès est autorisé.
- **Authenticité :** assurance de l'identité d'une personne ou d'une machine, par ex à l'aide d'une signature numérique.

# Introduction, définitions [3/16]

- **Signature numérique :** mécanisme permettant d'authentifier l'auteur d'un e-document et de garantir son intégrité. Propriétés nécessaires à un mécanisme de signature numérique :
  - doit permettre d'identifier celui qui a signé le doc.
  - doit garantir que le doc. n'a pas été altéré depuis signature
- Pour cela, 3 conditions doivent être réunies :
  - la signature ne doit pas pouvoir être falsifiée ;
  - la signature ne doit pas pouvoir être réutilisable. Elle fait partie du document signé et ne peut être déplacée sur un autre document ;
  - un document signé est inaltérable. Une fois qu'il est signé, on ne peut plus le modifier.

# Introduction, définitions [4/16]

- **Non répudiation :** fait de s'assurer qu'un document ne peut être remis en cause par l'une ou l'autre des parties.
- **Chiffrement (encryption/cryptage) :** transformation d'un message de façon à le rendre incompréhensible.
- **Masque jetable (One-Time Pad) :** longue suite aléatoire et non répétitive de caractères. Utilisé pour chiffrer un document à l'aide d'un « ou exclusif » (XOR). Pour être fiable à 100%, le masque doit être de la même taille que le message à chiffrer.

# Introduction, définitions [5/16]

- **Stéganographie (Steganography) :** mécanisme qui consiste à cacher un message (généralement chiffré) dans un document d'apparence anodine : image, son, texte, ...
- **Cryptogramme (Ciphertext) :** texte chiffré
- **Texte en clair (Plaintext) :** message à chiffrer.

# Introduction, définitions [6/16]

- **Historique :**
  - Le premier « document » chiffré connu remonte au XVIe siècle av. JC : il s'agit d'une tablette d'argile, retrouvée en Irak. Un potier y avait gravé sa recette secrète en supprimant des consonnes et en modifiant l'orthographe des mots
  - Entre le Xe et VIIe siècle av. JC, les Grecs utilisent à des fins militaires une technique de chiffrement par transposition (changement de position des lettres dans le message) : une bande de cuir est enroulée autour d'un bâton (**scytale**, ou **bâton de Plutarque**). Le message est écrit sur la bande, qui est envoyée au destinataire. La sécurité repose sur la connaissance du diamètre du bâton.

# Introduction, définitions [7/16]

- Ve siècle av. JC : les Hébreux utilisent la technique « **Atbash** », basée sur une méthode de substitution alphabétique inversée : le A est échangé avec le Z, le B avec le Y, etc.
- -600 avant JC : Nabuchodonosor. Le roi de Babylone écrivait sur le crâne rasé de ses esclaves, attendait que leurs cheveux repoussent, et les envoyait à ses généraux.

# Introduction, définitions [8/16]

- 1er siècle av. JC : **le code César** = substitution mono-alphabétique*. Consiste à décaler les lettres de l'alphabet d'un nombre n. Par exemple, si on remplace A par E (n=4), on remplace B → F, C → G,… Système peu sûr: 26 lettres = 26 clés possibles

  (*) : notons qu'il existe 4 types de substitutions:

  - **mono-alphabétique :** remplace chaque lettre du message par une autre lettre de l'alphabet
  - **poly-alphabétique :** utilise une suite de chiffres mono-alphabétiques (la clé) réutilisée périodiquement
  - **homophonique :** fait correspondre à chaque lettre du message en clair un ensemble possible d'autres caractères
  - **polygrammes :** substitue un groupe de caractères dans le message par un autre groupe de caractères

# Introduction, définitions [9/16]

- -150 av. JC : **Carré de Polybe**. Dessiner une matrice 5x5. Y déposer les lettres de A à Z (en mettant V & W sur la même case). Chiffrer = remplacer chaque lettre par ses coordonnées dans la matrice. Amélioration par l'introduction d'une clé : si clé = N, commencer à positionner les lettres à partir de la Nème case.

| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | A | B | C | D | E |
| 2 | F | G | H | I | J |
| 3 | K | L | M | N | O |
| 4 | P | Q | R | S | T |
| 5 | U | VW | X | Y | Z |

E=1,5

N=3,4

# Introduction, définitions [10/16]

- 1412 : une encycopédie en 14 volumes expose ttes les connaissances de crypto. du monde arabe : **Subh al-a sha**, écrite par l'Égyptien al-Qalqashandi.
- 1467 : le savant italien **Leone Battista Alberti** expose pour la première fois le chiffrement par substitution polyalphabétique qu'il applique à l'aide d'un disque à chiffrer : il s'agit de remplacer chaque lettre du texte en clair par une lettre d'un autre alphabet et à changer plusieurs fois d'alphabet de substitution au cours du chiffrement

# Introduction, définitions [11/16]

- 1586 : **Le chiffre de Vigenère**. Utilise un mot de passe comme clé. Basé sur une matrice de 26x26. Mettre l'alphabet sur la 1ère ligne, idem sur la 2ème ligne en commençant par B, etc. Le texte chiffré s'obtient en prenant l'intersection, de la ligne qui commence par la lettre à coder, avec la colonne qui commence par la première lettre du mot de passe, et ainsi de suite. Dès que l'on atteint la fin du mot de passe, on recommence à la première lettre. Pour décoder, il suffit de faire la même chose dans l'autre sens. Ne sera décrypté qu'en 1854 !

# Introduction, définitions [12/16]

- **Le « grand chiffre du roi Louis XIV » :** a donné du fil à retordre aux cryptanalystes jusqu'en 1893. Idée : ne pas chiffrer les lettres, mais… les syllabes. Pour le reste, il s'agit d'un simple système de substitution, qui utilise 587 nombres pour coder les syllabes.
- **Navajos :** utilisé par les USA comme technique de chiffrement lors de la GM2. Consiste à traduire les phrases dans la langue des indiens Navajos.

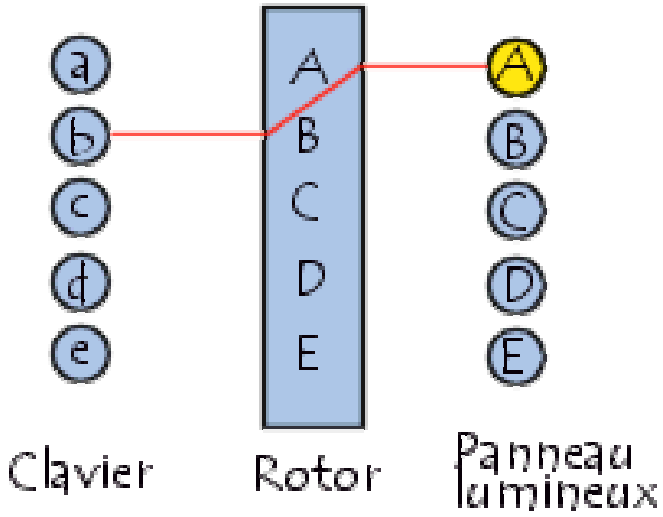
# Introduction, définitions [13/16]

- **Enigma :** machine inventée pas des civils pour un usage civil ; a été rendu célèbre car utilisé par les allemands lors de la GM2 :
  - équipée d'un clavier pour la saisie du message,
  - de différentes roues pour le codage,
  - et d'un tableau lumineux pour le résultat.

  Le codage était symétrique : pour une clé donnée, crypter le texte chiffré nous redonne le texte en clair. Son fonctionnement (source : http://www.commentcamarche.net/crypto/enigma.php3) :

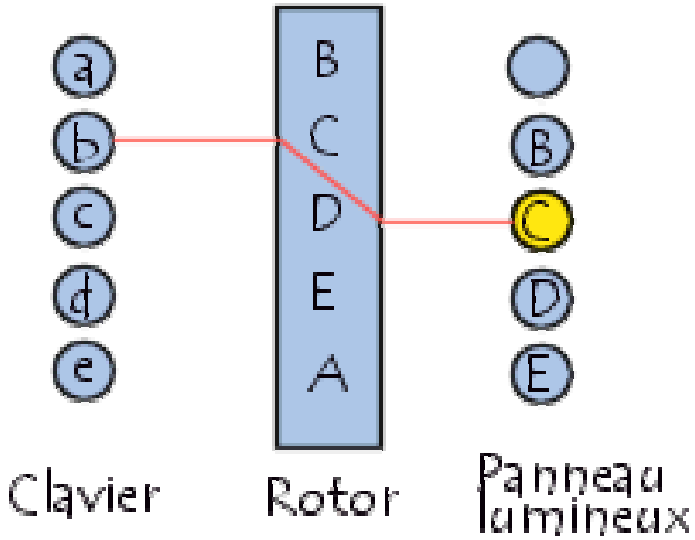
# Introduction, définitions [14/16]

- A chaque pression d'une touche du clavier, une lettre du panneau lumineux s'illuminait. Il y avait ainsi 3 roues de codage, appelées « Brouilleur Rotor », qui reliaient le clavier au panneau lumineux. Par ex, avec un seul rotor, lorsque l'on appuie sur B le courant passe par le rotor et allume A sur le panneau lumineux :



# Introduction, définitions [15/16]

- Pour complexifier la machine, à chaque pression sur une touche, le rotor tourne d'un cran. Après la première pression on obtient donc :



# Introduction, définitions [16/16]

- Suivant les modèles ( M3 ou M4), le système était muni de 3 ou 4 rotors. Les deuxième et troisième rotors avançaient d'un cran quand le précédent faisait un tour complet. Il y avait aussi un tableau de connexion qui mélangeait les lettres de l'alphabet et un réflecteur qui faisait repasser le courant dans les rotors avant l'affichage.
- Au final, pour des machines Enigma équipées pour 26 lettres, il y avait 17 576 combinaisons (26 x 26 x 26) liées à l'orientation de chacun des trois rotors, 6 combinaisons possibles liées à l'ordre dans lequel sont disposés les rotors, soient 100 391 791 500 branchements possibles quand on relie les six paires de lettres dans le tableau de connexions : 12 lettres choisies parmi 26 (*26! /(12!14!)*), puis 6 lettres parmi 12 (12!/6!), et puisque certaines paires sont équivalents (A/D et D/A), il s'agit de diviser par $2^6$.
- Les machines Enigma peuvent donc chiffrer un texte selon $10^{16}$ (17 576 * 6 * 100 391 791 500) combinaisons différentes !

# Les règles d'or [1/2]

- **Ne pas faire confiance à un système dont la sécurité repose sur le fait que l'algorithme de chiffrement est tenu secret (il y a toujours un hacker bon en reverse engineering qui cassera le système)**

# Les règles d'or [2/2]

- Ne pas faire confiance aux algorithmes :
  - Trop jeunes (laissez toujours un peu de temps aux cryptanalyses pour évaluer la qualité du système)
  - Dont on dit qu'ils sont théoriquement « cassables » (même si pour l'heure, personne n'a encore réussi l'exploit, et qu'aucun programme permettant le déchiffrement ne soit diffusé)

# Les méthodes classiques de cryptanalyse [1/6]

- **Attaque par force brute :** méthode utilisée pour trouver un mot de passe ou une clé. Principe : tester, une à une, toutes les combinaisons possibles. Cette méthode de recherche exhaustive ne marche que dans les cas où :
  - Le mot de passe cherché est court,
  - L'algorithme ne prends pas trop de tps de calcul,
  - On sait ce qu'on cherche (marche pour trouver du français, pas un nombre aléatoire).

# Les méthodes classiques de cryptanalyse [2/6]

- **L'attaque par dictionnaire :** méthode utilisée pour trouver un mot de passe ou une clé. Principe : tester une série de mots de passe potentiels, les uns à la suite des autres, en espérant que le mot de passe utilisé pour le chiffrement soit contenu dans le dictionnaire. Marche si :
  - Le mot de passe est « simple ».

# Les méthodes classiques de cryptanalyse [3/6]

- **L'analyse fréquentielle, ou analyse de fréquences :** méthode découverte au IXe siècle. Elle consiste à examiner la fréquence des lettres employées dans un message chiffré. L'analyse fréquentielle est basée sur le fait que, dans chaque langue, certaines lettres ou combinaisons de lettres apparaissent avec une certaine fréquence. Par exemple, en français, le e est la lettre la plus utilisée, suivie du s et du a. Inversement, le w est peu usité. Cette méthode est fréquemment utilisée pour **décoder des messages chiffrés par substitution**.

# Les méthodes classiques de cryptanalyse [4/6]

- **Indice de coïncidence :** technique inventée en 1920 par William F. Friedman. L'indice permet de savoir si un texte a été chiffré avec un chiffre mono- ou poly-alphabétique, en étudiant la probabilité de répétition des lettres du message chiffré. Donne aussi une indication sur la longueur de la clé.

  L'indice se calcule ainsi :

  $$IC = \sum_{q=A}^{q=Z} \frac{n_q(n_q-1)}{n(n-1)}$$

  $IC_{français} \approx 0{,}0746$

  $IC_{anglais} \approx 0{,}0667$

# Les méthodes classiques de cryptanalyse [5/6]

- **Cryptanalyse différentielle :** méthode générique qui consiste en l'étude de la manière dont les différences entre les données en entrée affectent les différences de leurs sorties. En pratique, cette méthode est appliquée à chaque étape de l'algorithme. Beaucoup de méthodes sont issues de cette technique :
  - Cryptanalyse $\chi^2$
  - Attaques boomerang, …

# Les méthodes classiques de cryptanalyse [6/6]

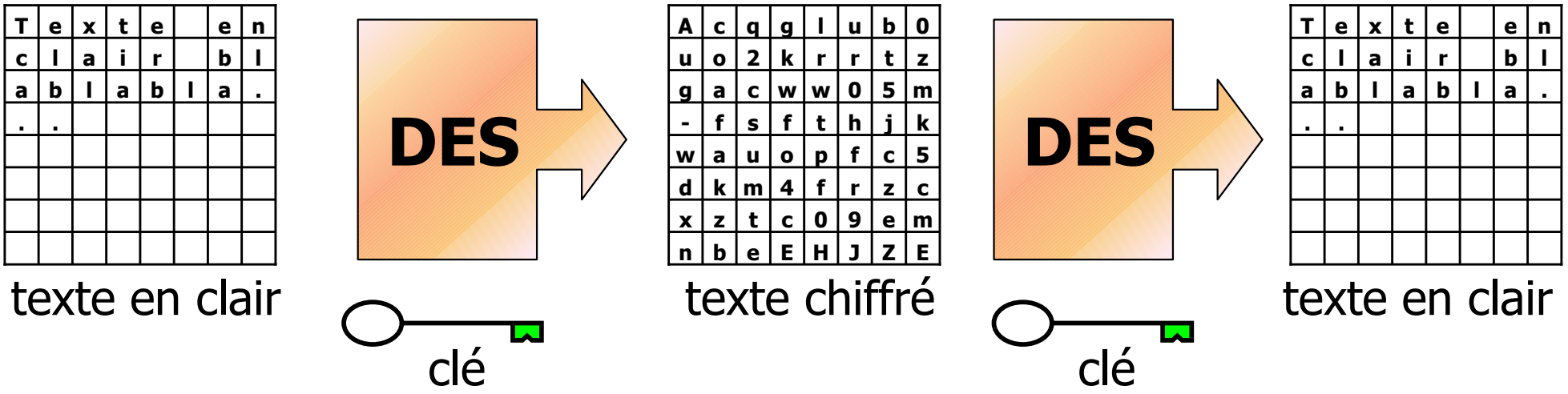
- **Attaque par clé apparentée** :
  - Pour sécuriser les échanges, certains protocoles renouvellent régulièrement leurs clés. Or, parce que les algorithmes sont complexes, il peut arriver que les clés possèdent des propriétés non voulues, et qu'elles ne soient pas si aléatoires que ça. Ex : bits toujours à 1 ou à 0, alternance régulière de séquences de bits, etc.

# Principaux algorithmes de chiffrement [1/12]

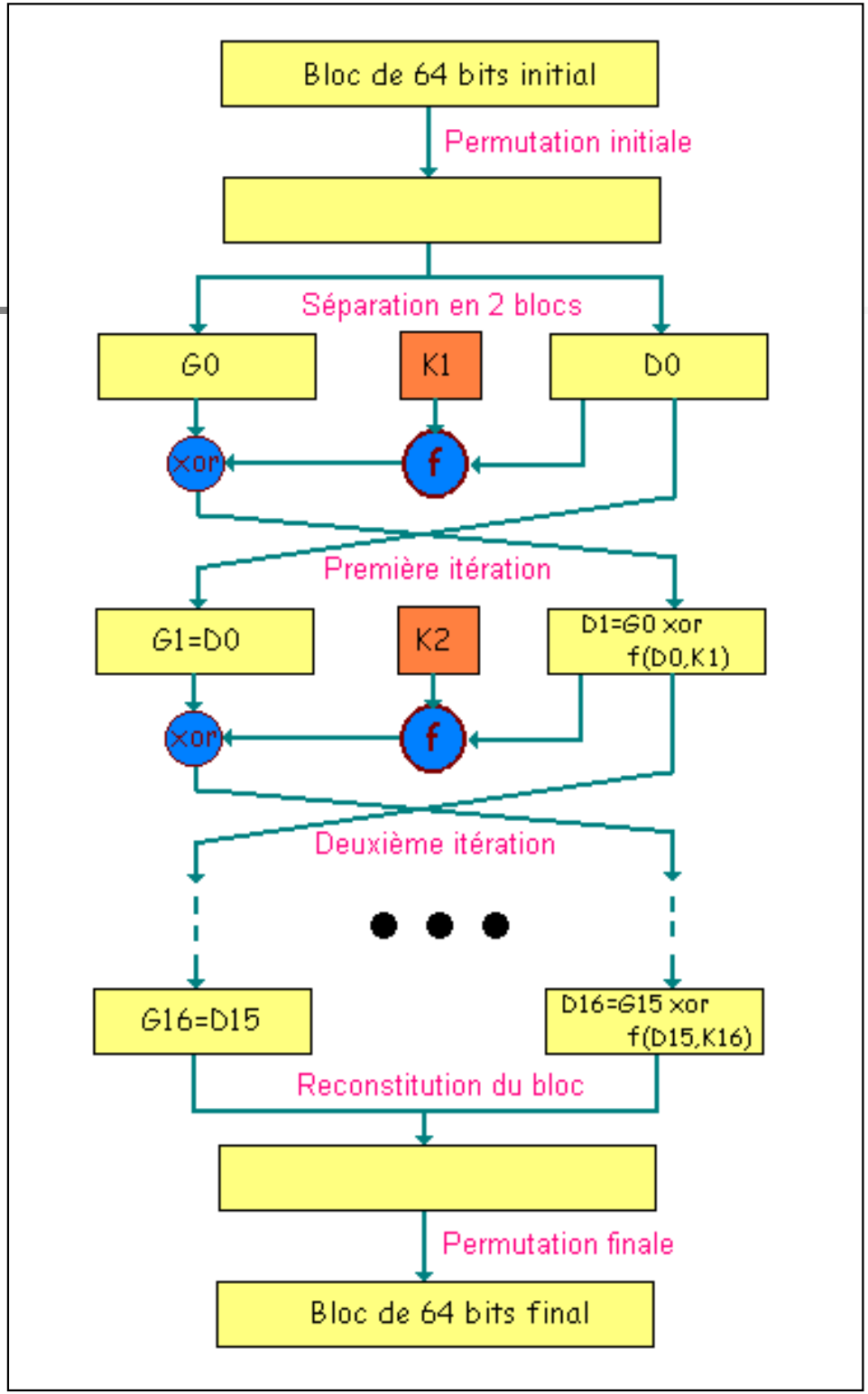
- Les algorithmes de chiffrement :
  - **Chiffrement par blocs** ;
  - **Chiffrement par flots** (chiffrement continu).
- Les deux grandes familles de chiffrement sont :
  - **Chiffrement symétrique** ou **chiffrement à clés secrètes :** si on connaît la clé de chiffrement, on déduit facilement la clé de déchiffrement (dans certains cas, c'est la même)
  - **Chiffrement asymétrique :** utilise une paire de clés (clé publique et clé privée). Tout ce qui est chiffré avec une est déchiffré avec l'autre. La sécurité repose sur le fait que connaissant une clé, il est presque impossible d'en déduire l'autre.

# Principaux algorithmes de chiffrement [2/12]

- Exemple de chiffrement par bloc : DES (Data Encryption Standard), datant de 1975/1977 (IBM). Travaille sur des blocs de 64 bits, et des clés de 56 bits



# Algorithme DES

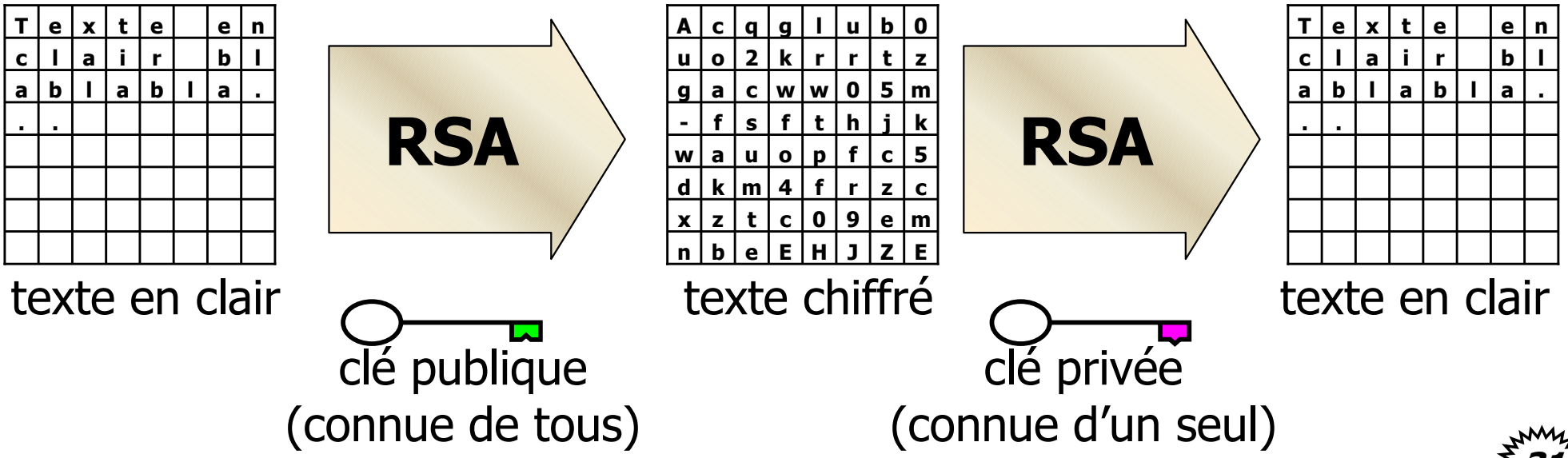


# Principaux algorithmes de chiffrement [4/12]

- Autres algorithmes de chiffrement par blocs à clés secrètes :
  - 3DES (3 x DES, avec une clé 2 à 3 fois plus longue),
  - AES - Advanced Encryption Standard (1999),
  - IDEA,
  - Blowfish,
  - Twofish, etc.
- Algorithmes de chiffrement de flots à clés secrètes :
  - RC4,
  - Py,
  - E0 (utilisé pour sécuriser le bluetooth), etc.

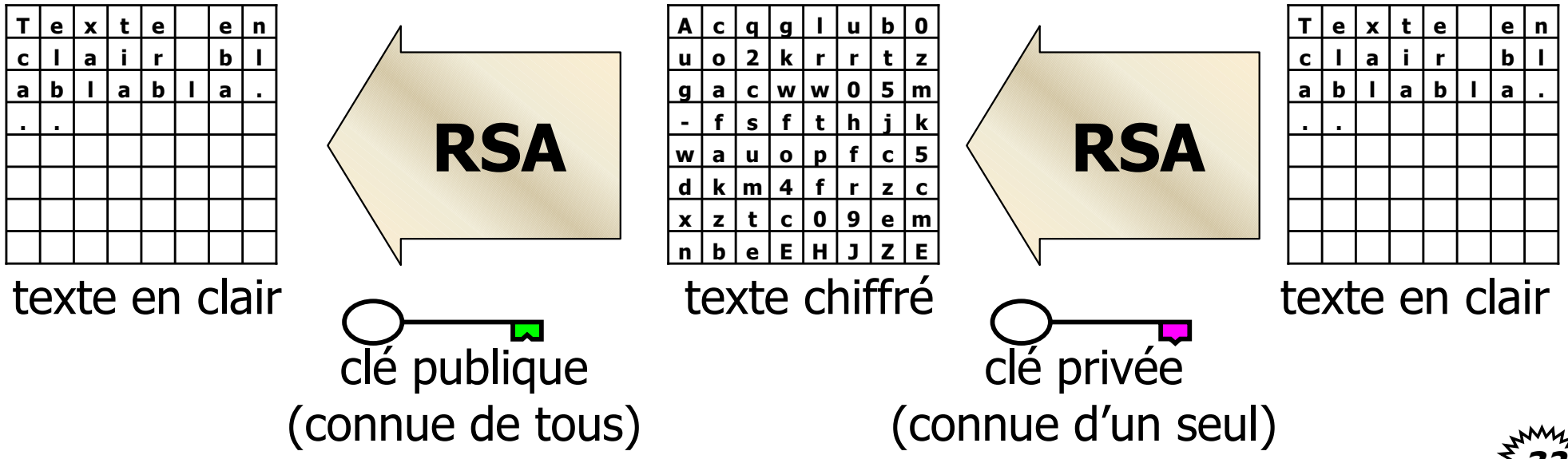
# Principaux algorithmes de chiffrement [5/12]

- Exemple de chiffrement asymétrique : RSA (1997 par Ron **R**ivest, Adi **S**hamir et Len **A**dleman)



# Principaux algorithmes de chiffrement [6/12]

- RSA fonctionne dans les deux sens (application : signature électronique – Cf. prochaines diapos)



# Principaux algorithmes de chiffrement [7/12]

- **Principe de RSA :** comme tous les algorithmes de chiffrement asymétrique, utilise une **fonction à brèche secrète** (procédure mathématique dont qu'il est trivial d'effectuer dans un sens, et dont on ne connaît pas de moyen simple de faire dans l'autre sens). Dans le cas présent, RSA utilise l'anneau $Z/Z_n$, et le petit théorème de fermat. Si p et q sont deux grands nombres premiers :
  - il est simple de calculer $n = p \times q$
  - mais, connaissant n (grand), on ne sait pas calculer simplement sa décomposition en facteur premiers

# Principaux algorithmes de chiffrement [8/1

**Fermat :** si a premier, et x premier avec a, $x^{a-1}$ mod a=1

- RSA :
  - soit p et q deux grands nombres premiers,
  - soit $n=p_x q$,
  - on appelle « fonction de totalisation d'Euler » le nombre $\Phi(n)=(p-1)(q-1)$
  - soit e un entier quelconque premier à $\Phi(n)$
  - soit d calculé pour que ed mod $\Phi(n) = 1$. Autrement dit, ed-1 est multiple de $\Phi(n)$. d se calcule à partir de p, q, et e avec l'algorithme d'Euclide.
  - Le couple K=(d, e) est la clé privée du système, et L=(n, e) est la clé publique.
  - Soit x le nombre à chiffrer (x<n)

# Principaux algorithmes de chiffrement [9/1

**Généralisation de Fermat par Euler :** si x<n et x premier avec n, $x^{\Phi(n)} \bmod n = 1$

- RSA (suite) :
  - fonction de chiffrage (notée C(x) ) :
    $C(x) = x^e \bmod n$
  - fonction de déchiffrage (notée D(y), y étant le texte chiffré, i.e. y=C(x) ) :
    $D(y) = y^d \bmod n$
  - En effet :
    - $D(y) = y^d \bmod n$ ; or, $y = C(x) = x^e \bmod n$
      => $D(y) = (x^e \bmod n)^d \bmod n = x^{ed} \bmod n$
      Or, $ed \bmod \Phi(n) = 1$, ce qui, par définition de l'opération modulo, nous donne $ed = k.\Phi(n)+1$, k étant un entier
      => $D(y) = x^{k.\Phi(n)+1} \bmod n = x.x^{k.\Phi(n)} \bmod n$
      => $D(y) = x.(x^{\Phi(n)})^k \bmod n$ ; Or (Euler) $x^{\Phi(n)} \bmod n = 1$
      => $D(y) = x.1^k \bmod n = x \bmod n = x$ (car x<n) CQFD

# Principaux algorithmes de chiffrement [10/12]

- Exemple avec RSA :
  - p=3, q=7 => n=p.q=3x7=21
  - $\Phi(n)=(p-1)(q-1)=2x6=12$
  - au hasard, e=5 (car 5 premier avec $\Phi(n)$ )
  - recherche de d tel que ed mod 12 = 1, i.e. 5d mod 12=1, i.e 5d = 12.k + 1, k entier. Pour k=2, on a d=5
  - clé publique L=(21,5), clé privée K=(5,5)
  - supposons que le chiffre à transmettre soit 2 :
    - $C(2) = 2^e \bmod n = 2^5 \bmod 21 = 32 \bmod 21 = 11$
    - $D(11) = 11^d \bmod n = 11^5 \bmod 21 = 161051 \bmod 21 = 2$

# Principaux algorithmes de chiffrement [11/12]

- Sécurité de RSA :
  - repose sur la difficulté à factoriser un grand nombre premier. En 2005, le plus grand nombre factorisé par les méthodes générales et l'état de l'art en matière de calculs distribués, était long de 663 bits. Les cléfs RSA sont habituellement de longueur comprise entre 1024 et 2048 bits.
- Autres algorithmes asymétriques :
  - puzzles de Merkle (premier algo rendu public),
  - cryptosystème courbes elliptiques&hyperelliptiques,
  - cryptosystème de Paillier, de Rabin, etc.

# Principaux algorithmes de chiffrement [12/12]

- Avenir de la cryptographie : chiffrement quantique. Il s'agit de mécanismes utilisant les propriétés étranges (pour nous) de la physique quantique. En pratique, l'information est portée par un photon. Certaines méthodes
  - utilisent un seul photon et un canal non sécurisé ;
  - d'autres, utilisent des des paires de photons enchevêtrés (photons nés d'un même événement, dont les propriétés restent identiques, même si quelque chose agit sur eux, même s'ils sont distant de milliers de Km)
- Bien que prometteuses, ces techniques sont délicates à mettre en place

# Signature électronique [1/5]

- Rappels/Prérequis :
  - Fonction de hachage (Cf. checksum) :
    - Fonction mathématique qui, à un ensemble de nombres en entrée, fait correspondre un ensemble de nombres de cardinal plus petit en sortie ;
    - La modification d'un élément en entrée engendre une modification de sa fonction de hachage en sortie.
  - Fonctions de hachage cryptographique :
    - si H(x) est une telle fonction, pour tout y donné, il doit être quasi-impossible de trouver un x tel que H(x)=y ;
    - si y=H(x), et x' ≈ x à un tout petit détail près (ex : changer 1 bit), on doit avoir y'=H(x') très ≠ de y

# Signature électronique [2/5]

- Exemples d'algorithmes de hachage cryptographique :
  - MD5,
  - SHA-1, SHA-256, SHA-512…
  - RIPEMD-160,
  - Whirlpool, etc.
- Une fonction de hachage crypto. suffit-elle pour faire une signature électronique ? Non :
  - possède la propriété « si message change, la signature change »,
  - mais ne permet pas de s'assurer de l'identité du signataire

# Signature électronique [3/5]

- Principe de la signature électronique :
  - Si M est le message à signer,
  - Si $C_{clé}(x)$ est une fonction de cryptage asymétrique, et $D_{clé}(x)$ la fonction de décryptage correspondant,
  - Si H(x) est une fonction de hachage crypto.,
  - Si « *Pub* » est la clé publique du signataire, et « *Priv* » sa clé privée,
  - Alors, la signature S du message M est :

$$S(M) = C_{Priv}(H(M))$$

# Signature électronique [4/5]

- Si un signataire vous envoie un message M accompagné de sa signature S(M) :
  - Pour vérifier que M n'a pas été altéré :
    - calculer H(M)
    - calculer $D_{Pub}(S(M))$ ; ces deux nombres doivent être égaux. En effet :
      $D_{Pub}(S(M)) = D_{Pub}(C_{Priv}(H(M))) = H(M)$
  - Pour vérifier l'identité du signataire :
    - calculer H(M)
    - calculer $D_{Pub}(S(M))$ ; ces deux nombres doivent être égaux. En effet, si H(M) n'a pas été crypté avec la clé privée que seul le signataire connaît, mais avec une clé « *Priv'* » :
      $D_{Pub}(S(M)) = D_{Pub}(C_{Priv'}(H(M))) \neq H(M)$

# Signature électronique [5/5]

- En résumé :
  - Mécanisme de signature électronique = fonction de hachage cryptographique + fonction chiffrage/déchiffrage asymétrique
  - Exemple de signature numérique :
    - MD5 + SHA-1
- **Signature aveugle :** signature effectuée sur un document qui a été masqué (ex crypté) avant d'être signé, afin que le signataire ne puisse prendre connaissance de son contenu. Ex d'utilisation :
  - Vote électronique,
  - Porte-monnaie électronique, …

# Certificat, tiers de confiance [1/11]

- Les chiffrements asymétriques sont fiables (pour l'heure), mais lents.
- Les chiffrements à clés secrètes sont très rapides, et sont fiables, d'autant plus si les clés sont changées régulièrement.
- Technique souvent utilisée :
  - utiliser un algorithme de chiffrement asymétrique pour s'échanger les clés (générées de façon aléatoire) d'un algorithme symétrique,
  - transmettre les messages à l'aide de cet algorithme symétrique & de ces clés,
  - changer les clés régulièrement (ex : toutes les heures).

# Certificat, tiers de confiance [2/11]

- Problématiques liées à cette technique :
  - comment créer des clés réellement aléatoires (par définition, les ordinateurs font des calculs reproductibles ; les générateurs pseudo aléatoires inclus dans les machines, s'ils sont initialisés par les mêmes conditions initiales, donneront toujours les mêmes nombres) ???
  - comment diffuser sûrement les clés publiques de l'algo. asymétrique ??? Ex: si elles sont dispo sur un site Web, comment savoir que le site web n'a pas été piraté (pour remplacer les clés) :
    - Soit en infiltrant le serveur,
    - Ou en changeant le routage (piratage du réseau),
    - Ou en piratant le DNS…

# Certificat, tiers de confiance [3/11]

- Génération de clés aléatoires :
  - Utilise un algorithme de génération de nombres pseudo-aléatoires (basé sur les techniques de permutation des nombres de 1 à N-1 – N étant la base – puis application d'un modulo avec un nombre n, n<N). Or, ces algorithmes sont reproductibles (bien que d'apparence aléatoire, la connaissance de quelques nombres de la suite permet de connaître tous les nombres suivants). Pour éviter ceci, on perturbe l'algorithme avec des événements physiques aléatoires (mouvement de souris, frappe du clavier, etc.)
  - Mieux : il existe des cartes électroniques de génération de nombres vraiment aléatoires, qui utilisent des événements physiques chaotiques (bruit brownien de l'air, ou des électrons dans un semi-conducteur, etc.).

# Certificat, tiers de confiance [4/11]

- Problématique de diffusion des clés :
  - Les clés sont générées et diffusées par… un « tiers de confiance », ou « **autorité de certification** ». Organisme dont le rôle est de :
    - Générer les paires de clés (publique et privée) d'un algorithme asymétrique,
    - Signer (on dit alors « certifier ») les clés,
    - Diffuser les clés publiques,
    - Gérer les liste de révocation (par ex, liste de clés perdues, volées… qui n'ont plus de valeur).

# Certificat, tiers de confiance [5/11]

- Les organismes de certification fournissent les clés publiques dans des « **certificats électroniques** ». Certificat =
  - La clé publique,
  - Algorithmes utilisés (pour chiffrer et signer),
  - Date de génération de la paire de clés,
  - Date de fin de validité du certificat,
  - Des informations sur le porteur de la paire de clés (nom, adresse électronique, titre, n° téléphone, entité qui a délivré le certificat…),
  - La signature de l'organisme de certification.

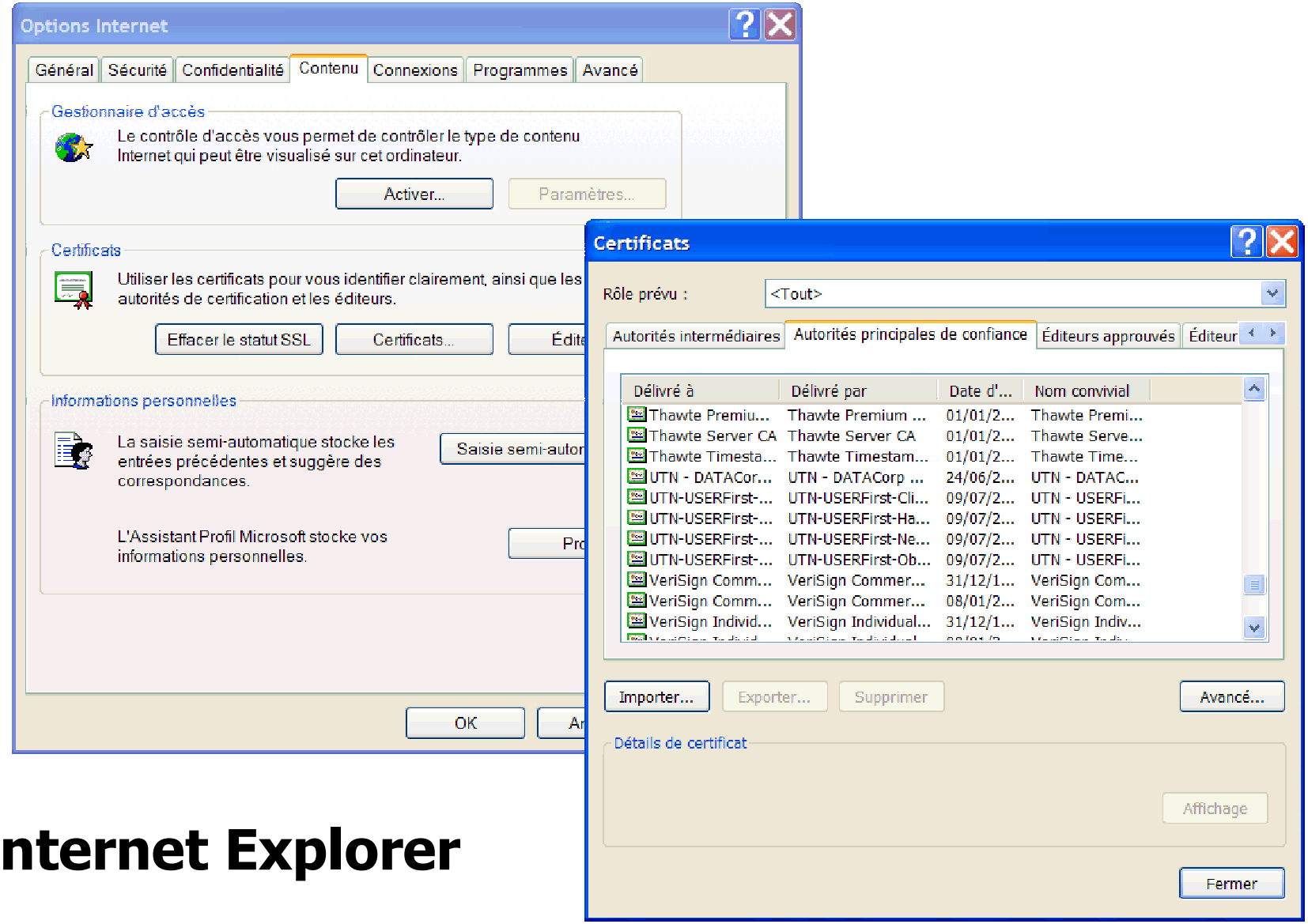
# Certificat, tiers de confiance [6/11]

- Exemples de normes & standards de certificats :
  - Norme X.509 de l'ISO – nous en sommes à la 3ème version – (utilisée par exemple dans les Cartes Professionnelles de Santé, site des impôts, SSL/HTTPS, SSH, etc.),
  - RFC 2440 (certificat utilisé dans le logiciel PGP – premier logiciel de communication sécurisée libre et public, développé par Philip ZIMMERMANN – et autres clones, comme OpenPG).

# Certificat, tiers de confiance [7/11]

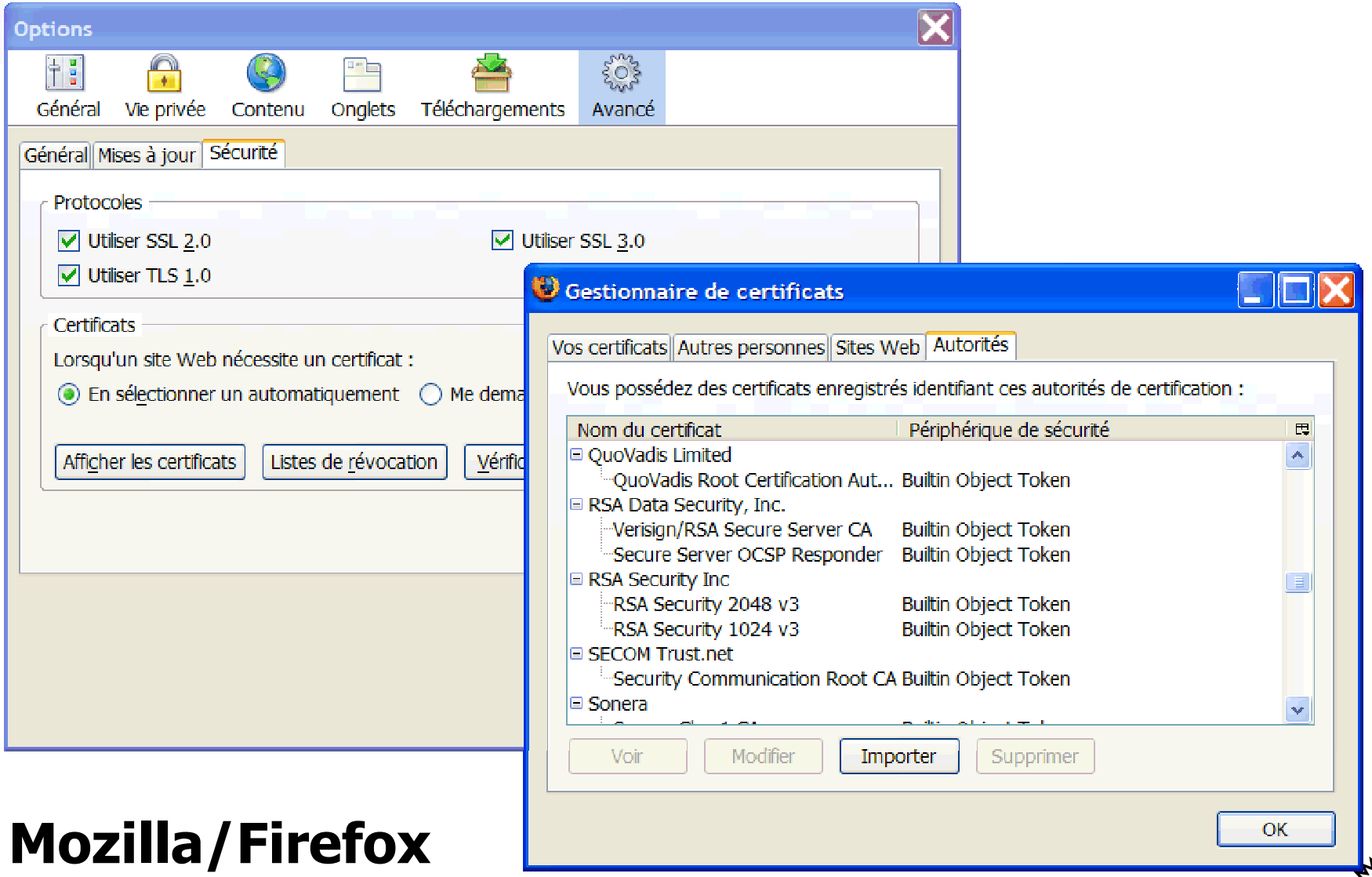
- Nous l'avons vu, le tiers de confiance signe les certificats. Aussi, pour vérifier si un certificat n'est pas frauduleux, il faut vérifier cette signature. Donc, il faut connaître la clé publique de l'organisme de certification. C'est pourquoi les navigateurs et systèmes d'exploitation intègrent une liste de certificats de tiers de confiance : « **certificats racines** ».

# Certificat, tiers de confiance [8/11]




**Internet Explorer**

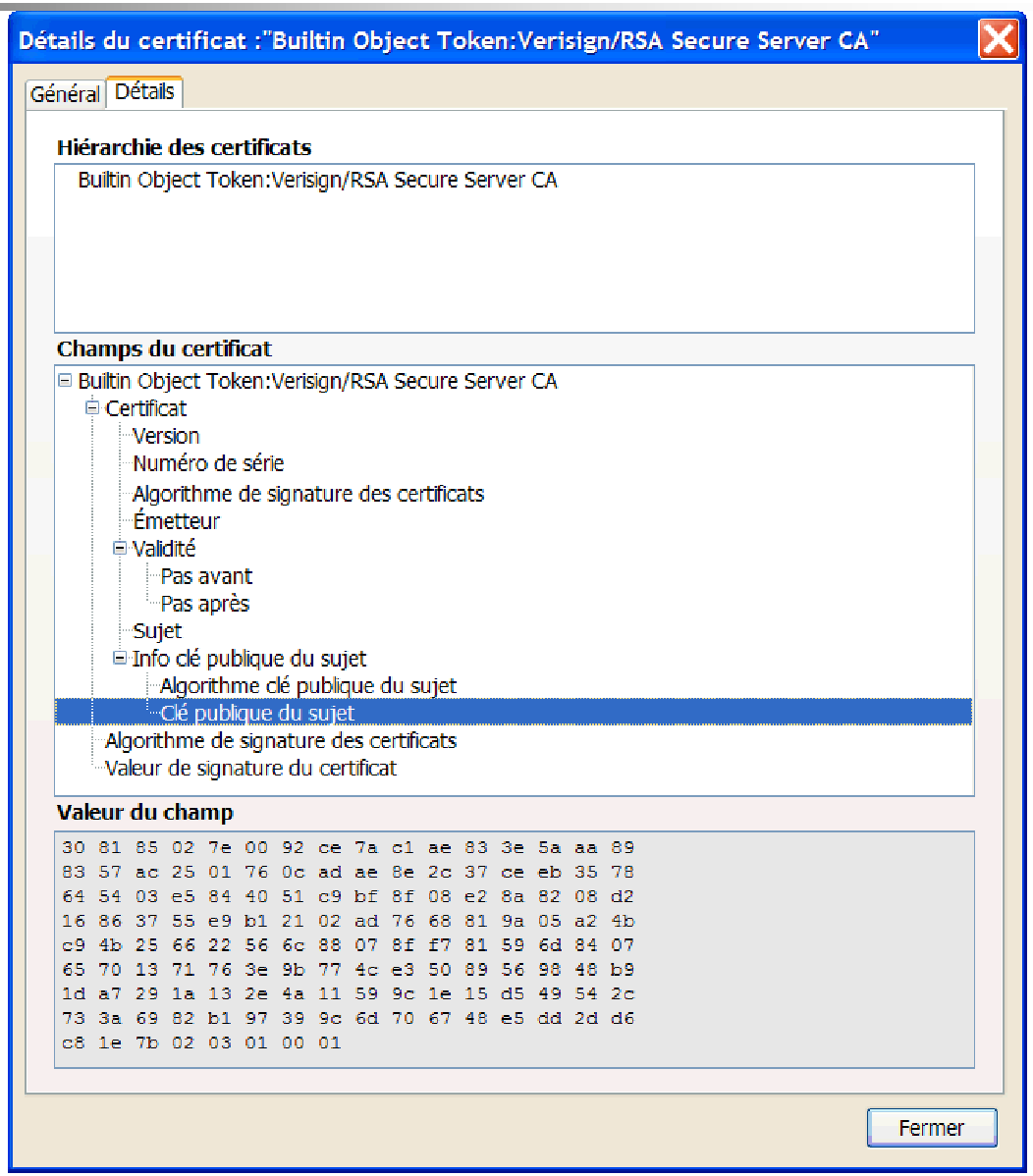
# Certificat, tiers de confiance [9/11]




**Mozilla/Firefox**

# Certificat, tiers de confiance [10/11]

Exemple de certificat racine (vu dans Mozilla/Firefox) :



# Certificat, tiers de confiance [11/11]

- Terminologie :
  - On appelle **PKI** (Public Key Infrastructure), ou **IGC** (Infrastructure de Gestion de Clefs) ou **ICP** (Infrastructure à Clefs Publiques) un ensemble de composants physiques (ordinateurs, cartes à puces…), de procédures (vérifications, validation) et de logiciels en vue de créer et gérer le cycle de vie des certificats numériques. En résumé, il s'agit d'une infrastructure fournissant les services suivants :
    - Enregistrement des utilisateurs,
    - Génération des paires de clés (publique et privée),
    - Génération de certificats,
    - Gestion de la révocation de certificats,
    - Publication des certificats valides et révoqués,
    - Identification et authentification des utilisateurs,
    - Archivage (sécurisé) des certificats.
- Législation :
  - la signature électronique a été légalisée par la loi n° 2000-230 du 13 mars 2000 (parue au JO n°62, 14/03/2000, NOR: JUSX9900020L) + décret d'application du 31 mars 2001

# Cryptologie et réseau [1/3]

- Utilisation de la cryptographie pour sécuriser un réseau filaire :
  - 802.1X (standard défini par l'IEEE en juin 2001, permettant d'authentifier un utilisateur souhaitant accéder à un réseau via à un serveur d'authentification) :
    - Cette authentification utilise le protocole EAP (Extensible Authentication Protocol), défini par l'IETF. Les différents mécanismes EAP inclus dans le programme de certification sont :
      - EAP-TLS,
      - EAP-TTLS/MSCHAPv2,
      - PEAPv0/EAP-MSCHAPv2,
      - PEAPv1/EAP-GTC,
      - EAP-SIM,
      - et d'autres mécanismes propriétaires.
    - Le serveur d'authentification est souvent un serveur RADIUS (RFC 2865 et 2866), mais pas obligatoirement.

# Cryptologie et réseau [2/3]

- Si les serveurs RADIUS sont souvent utilisés avec EAP comme serveur d'authentification, se ne sont pas les seuls mécanismes. Il existe aussi :
  - NTLM (NT Lan Manager) : mécanisme NT4,
  - KERBEROS (créé initialement par le MIT pour échanger les clés dans les serveurs Unix, et repris par exemple par MS Active Directory),
  - HMAC (RFC 2104),
  - GPS (GPS, Girault - Poupard/Paillès - Stern), normalisé ISO/IEC 14888-2,
  - PAP (Password Authentication Protocol) et CHAP (Challenge-Handshake Authentication Protocol), utilisés dans PPP,
  - etc.

# Cryptologie et réseau [3/3]

- L'adaptation de la norme « 802.1X » (développée pour les réseaux Ethernet) au Wifi est la norme « 802.11i »
- Les algorithmes de sécurisation des réseaux Wifi sont :
  - **WEP** (Wired Equivalent Privacy, RC4 avec des clés de 64, 128, ou 256 bits) : cassé ☹, ∀ long. clé, RC4 n'étant pas en cause ;
  - **PSK** (Pre-Shared Key), qui correspond au 802.11i dont la clé est partagée « en dur » entre l'utilisateur et le réseau ;
  - **WPA** (Wifi Protected Access, qui correspond au draft du 802.11i d'avril 2003) et **WPA2** (=802.11i ratifié) :
    - WPA utilise un mécanisme sécurisé (par rapport au WEP) : le protocole TKIP (Temporal Key Integrity Protocol), qui échange de manière dynamique les clés lors de l'utilisation du système. L'algorithme de chiffrement par lot est RC4 ;
    - WPA2 utilise le protocole « CCMP » (Counter-Mode/CBC-Mac protocol), supposé plus sûr que TKIP, et AES pour le chiffrement.
    - WPA et WPA2 utilisent le même type de serveur d'authentification selon les mêmes mécanismes EAP que « 802.1X ».

# Questions/Réponses ???

- That’s all, folks!…