Faculté des Sciences
Aix*Marseille Université

DIU EIL
Bloc 3 - Systèmes - Réseaux

Séance 10 – Notions de Sécurité Réseaux

Notes de Cours

1 Préliminaires

1.a Plan de cette partie

- cours (1h30) + TD (0h30)
  - Vocabulaire de la cryptographie
  - Cryptographie Appliquée
  - Certificats
- TP (2h00...)
  - manipulation des primitives cryptographiques
  - certificats

1.b Important

1. La sécurité *n'est pas* un problème technique.
2. *La sécurité d'un ensemble est **égale** à la sécurité de l'élément le plus faible.*

On peut ici mettre de l'authentification par biométrie la plus perfectionnée, la pelouse reste bien accessible, comme en témoigne les traces de roues.

La sécurité informatique consiste notamment à ne pas laisser de pelouse (virtuelle) autour des outils techniques de contrôle d'accès. C'est très difficile et cela dépend davantage de questions organisationnelles que techniques.

Nous allons cependant nous concentrer dans ce cours sur certains de ces outils techniques, notamment à base de cryptographie.

1.c Que faire en pratique ?

- outils techniques
  - cryptographie (bases sur ce cours)
  - outils réseaux : parefeux, VPN, IDS, IPS
- bonnes pratiques
  cf "Guide d'hygiène de l'ANSSI"

1.d Corollaire pour la Sécurité Réseaux

La sécurité des *points terminaux* est aussi importante que la sécurité des transmissions.

1. sniffeur de clavier, chevaux de Troie
2. TEMPEST
3. vol/perte de
   - disque dur, clefs USB
   - portable, mobile, smartphone
4. ingénierie sociale ...

1.e Taxonomie Simplifiée des Menaces

- menaces involontaires :
  - erreur matérielle
  - erreur logicielle
  - erreur humaine

  cf *fiabilité* et *qualité*
- menaces volontaires :
  - attaques malveillantes
  - attaques ciblées avancées
    grands groupes, institutions,...

  cf *sécurité*
- Spécificité des attaques distantes
  modèle client/serveur :
  vulnérabilité du logiciel => *exécution de code arbitraire à distance*

1.f Attaques Réseaux Passives : Ecoute

accès à un contenu réseau dont on n'est pas destinataire

- TCP/IP n'est pas protégé :
  tout équipement réseau intermédiaire peut voir intégralement un paquet :
  - entêtes (*métadonnées* : *qui parle à qui ?*)
  - contenu (*deep packet inspection*)

  => attaque sur les protocoles pour devenir intermédiaire (empoisonnement ARP/DNS)

1.g Attaques Réseaux Actives : Interception

accès *et modification* d'un contenu dont on n'est ni émetteur, ni destinataire
Attaque de l'homme au milieu

1. TCP/IP n'est pas protégé (bis) :
   tout équipement réseau intermédiaire peut voir *et modifier* intégralement un paquet :
   - entêtes
   - contenu
2. Attaque sur les protocoles
   - vol de session
   - rejeu
   - ...
3. Contre-mesures : certificats (utilisabilité discutable)
   cf fin ce cours

1.h Moyens de Protection

- Conception intégrant la sécurité
- Bonne administration
  - Maintenance (mises à jour)
  - Contrôle d'accès
  - Surveillance
  - audit
- Outils techniques
  - Protocoles Réseaux et Cryptographie
    - IPSec et IPv6
    - SSL/TLS
    - WEP et WPA
  - Génie Logiciel
  - Administration
    - contrôle d'accès

  - réseaux privés virtuels
- Applications Réseaux
  - Parefeu
  - Détecteur d'intrusion
  - Veille active

Difficile de concilier utilisabilité et sécurité

2 Rappels Cryptographie

2.a Vocabulaire de la Sécurité Réseaux

Quelles sont les caractéristiques que l'on cherche à protéger

1. *confidentialité* : les données échangées ne peuvent être connues que de l'expéditeur et du destinataire.
2. *intégrité* : toute modification (involontaire ou malveillante) est détectée.
3. *authentification* : les interlocuteurs sont réellement qui ils prétendent être. intégrité des *méta-données*
4. *confidentialité future parfaite* : la découverte future d'un secret partagé ne permet pas de rompre la confidentialité d'anciennes communications.
5. *anonymat* (protection de l'identité) : toute personne observant le trafic ne sait pas qui sont les interlocuteurs d'une conversation.
6. *non-répudiation* : toute partie prenante d'une transaction authentifiée a la garantie que le contrat établi sera respecté (=> *implications juridiques*)

2.b Primitives Cryptographiques

- Petit Historique
- Chiffrement Symétrique
- Fonctions de Hachage
- Chiffrement Asymétrique

2.c Encore Un Peu de Vocabulaire

Protocoles Cryptographiques

Combinaisons de techniques mises en oeuvre pour assurer la sécurité

- mathématiques,
- algorithmique,
- logiciel

2.d De Tout Temps, les Hommes ...

ont cherché à dissimuler des messages, des journaux, ...

- Stéganographie : dissimulation
- Méthode par substitution : A$\rightarrow$B,$B \rightarrow$C,...
  attaque par analyse de fréquence

— méthodes complexes, *paramétrée* par une clef :
3DES, IDEA, Blowfish, AES...

La plupart de ces dernières sont extrêment solides, y compris avec les puissances de calcul actuelles.

2.e Un Algorithme de Chiffrement Totalement Fiable

Il existe un algorithme prouvé mathématiquement (par théorie de l'Information) comme étant sûr à 100% :

Méthode du masque jetable :

— Alice et Bob se choisissent une longue suite de bits *aléatoires* P

— Pour chiffrer un message M, Alice fait un OU EXCLUSIF ($\oplus$) de M avec P

— Pour déchiffrer le message reçu C, Bob effectue la même opération
Comme $\oplus$ est

  — associatif : $\forall X, Y, Z \ (X \oplus Y) \oplus Z = X \oplus (Y \oplus Z)$

  — idempotent : $\forall X, X \oplus X = 0$
  On obtient

— $C \oplus P = (M \oplus P) \oplus P = M$

2.f Problème pour le Masque Jetable

La méthode par masque jetable est malheureusement *impraticable* :

— il faut une *clef* aussi grande que le message à transmettre.

— une *clef* ne peut servir qu'une seule fois.

=> *Problème de la distribution des clefs.*

2.g Les Débuts de la Cryptographie à Clef Publique

En 1975, Diffie et Helman propose un algorithme permettant à deux personnes ne s'étant jamais rencontrée de partager un secret sur un canal public.

D'autres techniques à clefs publiques émergent (*RSA*, *ElGamal*) basées sur des fonctions difficilement inversibles (propriétés algébriques).

Mais le protocole Diffie Helman est vulnérable à l'attaque de l'intercepteur.

=> nécessité d'authentifier les interlocuteurs.

2.h Définitions

Chiffrement symétrique la même clef sert pour le chiffrement et le déchiffrement

Chiffrement assymétrique deux clefs différentes => *la clef de chiffrement peut être publique*

2.i Etat de la Loi (Française)

- Historique *romanesque* :
  - En France, cryptographie forte interdite jusqu'en 1999
  - Aux Etats-Unis,
    - NSA
    - PGP
    - ...
- Le tournant du commerce électronique : (décret 1999)
  clefs symétriques de taille maximale 128 bits
- Déclaration au Ministère de la Défense
- Signatures électroniques ont valeur légale (2001)
- LCEN

Voir http://www.ssi.gouv.fr

3 Chiffrement Symétrique

Fonction de chiffrement à secret partagé.

Alice		**Bob**
clef K		clef K
M		
$C = E_K(M)$	$\xrightarrow[C]{}$	$M = D_K(C)$

Les algorithmes de chiffrements par blocs sont conçus à partir d'opérations élémentaires opérants sur les bits des données :

- permutation (circulaire)
- ou exclusif $\oplus$
- multiplication modulo
- ...

3.a Algorithmes de Chiffrement

Algorithme	**Longueur de clef**
DES (*obsolète*)	56 bits
3DES	168 bits
IDEA	128 bits
Blowfish	variable $\rightarrow$ 448 (128 bits en général)
RC5	variable
CAST-128	de 40 à 128 bits
AES(Rijndael)	variable (128,192,256)

3.b Flots de Données Chiffrés

Les algorithmes précédents chiffrent par blocs.

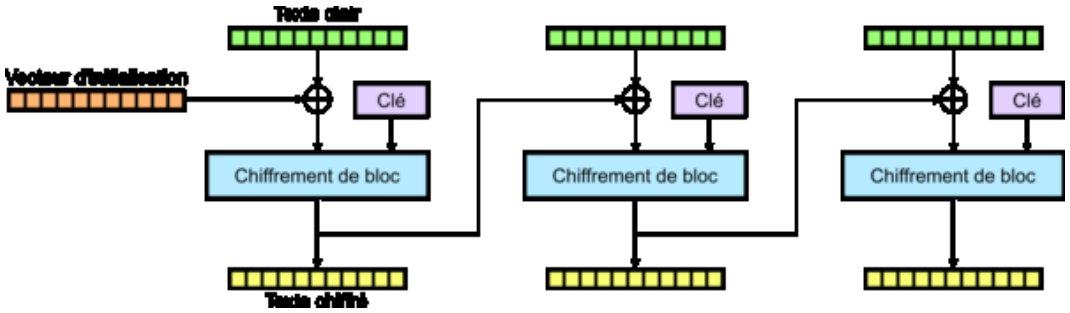
En chiffrant par bloc, un bloc de même contenu est toujours chiffré de la même manière =>

CBC chiffrement par blocs chainés

$$\begin{array}{|c|c|c|} \hline C_i & = & E_K[C_{i-1} \oplus M_i] \\ \hline M_i & = & C_{i-1} \oplus D_K[C_i] \\ \hline \end{array}$$

— avantages : cryptanalyse plus difficile : un bloc de texte clair répété sera chiffré différemment

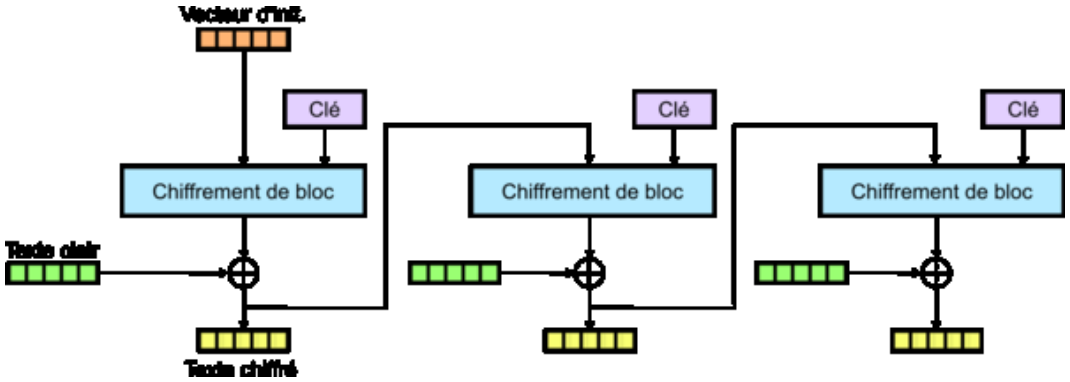
— *inconvénient* blocs de taille fixée => bourrage



CFB chiffrement par rétroaction

$$\begin{array}{rcl} C_i & = & E_K[\widetilde{C_{i-1}}] \oplus M_i \\ M_i & = & E_K[\widetilde{C_{i-1}}] \oplus C_i \end{array}$$

— avantage : données de taille quelconque



3.c Fonctions de Hachages Cryptographiques

Une **fonction de hachage** est une fonction qui à toute donnée fait correspondre un *résumé* de (petite) taille fixe.

Propriétés attendues d'une fonction de **hachage cryptographique** H :

— H est applicable à des données de taille arbitraire
— H produit une sortie de longueur fixe
— $H(x)$ est algorithmiquement facile à calculer
— **inversibilité** : il est *algorithmiquement difficile*, pour tout résumé h, de trouver x tel que $H(x) = h$
— **collision** : il est *algorithmiquement difficile* de trouver x et y, avec $y \neq x$, tels que $H(y) = H(x)$
— **collision à document donné** : pour toute donnée x, il est *algorithmiquement difficile* de trouver $y \neq x$ tel que $H(y) = H(x)$

3.d Exemple de Hachage Elémentaire

— Ou exclusif des blocs : $H(M) = b_1 \oplus \cdots \oplus b_n$ où b_i correspond au i-ème bloc du message.
code de parité sur les bits distants de la taille d'un bloc

— Amélioration possible : chaque bloc est décalé circulairement d'un bit ...

— codes détecteurs d'erreur (comme les CRC des trames réseaux)

3.e Fonctions de Hachages Cryptographiques

Algorithme de hachage	taille donnee	taille hache
MD5 (obsolète)	∞	128 bits
SHA-1 (obsolète)	$2^{64} - 1$	160 bits
SHA-256	$2^{64} - 1$	256 bits
RIPEMD-160	∞	160

— Eté 2005 : MD5 est cassé (collisions)

— 2007 : SHA-1 fragilisé
il est conseillé de passer à la famille SHA-2 (SHA256, SHA512)

— 2008 : lancement d'un concours pour la création d'une nouvelle (famille de) fonction de hachage cryptographique.

— octobre 2012 : résultat du concours Keccak est SHA3

— sept. 2014 : mozilla annonce refuser les certificats avec SHA1 à partir de jan 2017

— fév. 2017 : collision trouvée pour deux PDFs

— 2017 : les principaux navigateurs refusent les certificats avec SHA1

3.f Intégrité par HMAC

un haché cryptographique procure la propriété d'intégrité car il n'est pas aussi facile de modifier le contenu du message et le CRC associé.

On peut également insérer une clef secrète dans l'opération.
Code d'Authentification de Message par Haché Cryptographique :
$H((K \oplus opad).H((K \oplus ipad).msg))$

— H : fonction de hachage,
— K : clef secrète,
— $ipad$: `0x363636...3636`,
— $opad$: `0x5c5c5c...5c5c`,
— msg : le message à authentifier.

— **avantages** :
  — intégrité et authentification
  — exécution plus rapide des algorithmes de hachage
  — pas de restriction légales à l'utilisation

— $clef = grain\ de\ sel$ pour contrer
  — les attaques par dictionnaire et précalcul,
  — le rejeu

HMAC (sécurité sur IP, SSL/TLS, SET)

3.g Cryptographie à Clef Publique

Rappel du Principe :

— A possède une paire $(privee, publique)$,
— *confidentialité* : de B vers A,
  B chiffre avec $publique$,
  A déchiffre avec $privee$.
— *authentification* : de A vers B,
  A chiffre un haché de son message M avec $privee$,
  B déchiffre avec $publique$ **et** vérifie que le résultat est bien le haché de M.

3.h Cryptographie à Clefs Publiques

Algorithme	**Utilisation**
Diffie Helman	Echange de clefs
RSA	chiffrement, signature
El Gamal	chiffrement, signature

Les longueurs de clefs sont variables ($\sim$ 2048 bits pour RSA actuellement, transition vers 8196 bits en cours pour les plus paranoïaques).

3.i Chiffrement à Clef Publique

On considère un groupe fini $(G, \times)$.
Considérons un couple $(K_{\text{priv}}, K_{\text{pub}})$. tel que $taille(G) \mid K_{\text{priv}}K_{\text{pub}} - 1$.

Soit $M \in G$ un message à chiffrer :

— Le chiffrement est $C = M^{K_{\text{pub}}}$,

— Le déchiffrement est $M = C^{K_{\text{priv}}}$,

— car

— $\exists k, k.taille(G) = K_{\text{pub}} \times K_{\text{priv}} - 1$

— $\forall m \in G$,

$$\begin{aligned}(m^{K_{\text{pub}}})^{K_{\text{priv}}} &= m^{K_{\text{pub}}K_{\text{priv}}} \\ &= m^{k.taille(G)+1} \\ &= (m^{taille(G)})^k \times m^1 \\ &= m\end{aligned}$$

3.j Signature avec une Clef Publique

Si on inverse l'ordre d'exponentation (les deux exponentiation *commutent*), on obtient un *schéma de signature* :

— signature d'un haché : $S = h^{K_{\text{priv}}}$

— vérification (publique) : $h \stackrel{?}{=} S^{K_{\text{pub}}}$

On peut utiliser la même clef pour signer et pour le chiffrement dans certains cas (RSA/DSA, ECDSA), c'est risqué pour d'autre (El Gamal).

3.k Authentification

Elle s'effectue par la signature (via K_{priv}) de l'association

— entité

— valeur

Exemple : `zorro@gmail.com` est l'adresse l'électronique de "Monsieur H. Simpson".

3.l Retour sur l'Authentification I

On a :

— un chiffrement symétrique fort : AES

— un schéma d'échange de clefs : Diffie-Helmann

— et le problème est résolu. non ?...

attaque d'interception

3.m Retour sur l'Authentification II

L'authentification est une qualité cruciale des protocoles cryptographiques :

Cela est implémentable de nombreuses manières

— par chiffrement symétrique (secret partagé)

— par cryptographie à clé publique (clé privée/clé publique)

— par fonctions de haché à sens unique + secret partagé

3.n Utiliser des Primitives Cryptographiques

Les algorithmes cryptographiques sont des briques de bases.
Implémentations :
— problème de brevets logiciels
— standard d'implémentation (algos simples)
— standard :
  — algorithmes
  — objets mathématiques

3.o Exemple de Communication Sécurisée

Un protocole typique est
1. échange clef symétrique k par cryptographie publique
2. chiffrement du message avec k
3. signature avec cryptographie publique

3.p Exemple de Communication Sécurisée (TLS simplifié)

A de clefs publique / privée (K_A, K'_A) veut envoyer un message M à B de clefs publique / privée (K_B, K'_B).

— A tire au hasard une clef k (de 128 bits),
— A envoie
— $c = C^{\text{RSA}}_{K_B}(k)$,
— $C = C^{\text{AES}}_{k}(M)$,
— $s = C^{\text{RSA}}_{K'_A}(H^{\text{SHA2}}(M))$.
— B obtient la clef de session $k = D^{\text{RSA}}_{K'_B}(c)$,
— B déchiffre $D = D^{\text{AES}}_{k}(C)$,
— et vérifie la signature avec $D^{\text{RSA}}_{K_A}(s) = H^{\text{SHA2}}(D)$.

3.q Performances

Le coût d'un chiffrage par algorithme à clef public est **beaucoup** plus important que celui d'un chiffrement à clefs symétriques.

=> toujours privilégier les algorithmes symétriques *si* le problème de la distribution des clefs est résolu.

3.r Performances 2

La vitesse de chiffrement en situation réelle est également très importante

— implémentation logicielle
— implémentation FPGA
— implémentation matérielle (instructions CPU, puce dédiée)