

# Sécurité des Systèmes d'Information

Eric Alata    eric.alata@laas.fr

Yves Deswarte    yves.deswarte@laas.fr

Vincent Nicomette    vincent.nicomette@laas.fr

INSA de Toulouse

22 novembre 2011

## Système d'information

Un système d'information est l'ensemble des éléments participant au traitement, à la gestion et à la transmission d'informations entre les membres d'une communauté.

## Sécurité

La sécurité des systèmes d'information est l'ensemble des moyens permettant d'assurer les propriétés de confidentialité, d'intégrité et de disponibilité des informations.

Un peu d'histoire

Les propriétés de la sécurité

Les attaques

Les défenses

La protection des systèmes informatiques

# Sommaire

## Un peu d'histoire

Les propriétés de la sécurité

Les attaques

Les défenses

La protection des systèmes informatiques

# Sommaire

## Un peu d'histoire

La scytale – *v<sup>e</sup> siècle avant J.C.*

Le chiffre de César – *I<sup>e</sup> siècle avant J.C.*

Le chiffre de Vigenère – *XVI<sup>e</sup> siècle après J.C.*

Le chiffre de Marie Stuart – *XVI<sup>e</sup> siècle après J.C.*

La machine Enigma – *XX<sup>e</sup> siècle après J.C.*

Kevin Mitnick – *25 décembre 1995*

A nos jours

# Sommaire

## Un peu d'histoire

La scytale – *v<sup>e</sup> siècle avant J.C.*

Le chiffre de César – *i<sup>e</sup> siècle avant J.C.*

Le chiffre de Vigenère – *xvi<sup>e</sup> siècle après J.C.*

Le chiffre de Marie Stuart – *xvi<sup>e</sup> siècle après J.C.*

La machine Enigma – *xx<sup>e</sup> siècle après J.C.*

Kevin Mitnick – *25 décembre 1995*

A nos jours

# La scytale – v<sup>e</sup> siècle avant J.C.

- Les spartiates, contexte militaire
- Besoin en communication durant la bataille
  - Permettre aux généraux d'échanger des messages à travers le champ de bataille
  - Empêcher les ennemis de lire le contenu des messages



# La scytale – v<sup>e</sup> siècle avant J.C.

- Description dans une œuvre de Plutarque[Plu]

Je  
dois dire ce que c'est que la scytale. Quand un général part pour une  
expédition de terre ou de mer, les éphores prennent deux bâtons  
ronds, d'une longueur et d'une grandeur si parfaitement égales, qu'ils  
s'appliquent l'un à l'autre sans laisser entre eux le moindre vide. Ils

223

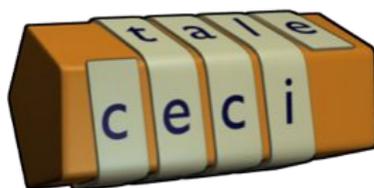
LYSANDRE.

gardent l'un de ces bâtons et donnent l'autre au général; ils appellent  
ces bâtons scytales. Lorsqu'ils ont quelque secret important à faire  
passer au général, ils prennent une bande de parchemin, longue et  
étroite comme une courroie, la roulent autour de la scytale qu'ils ont  
gardée, sans y laisser le moindre intervalle, en sorte que la surface du  
bâton est entièrement couverte. Ils écrivent ce qu'ils veulent sur cette  
bande ainsi roulée, après quoi ils la déroulent, et l'envoient au gé-  
néral sans le bâton. Quand celui-ci la reçoit, il ne peut rien lire, parce-  
que les mots tous séparés et épars, ne forment aucune suite. Il prend  
donc la scytale qu'il a emportée, et roule autour la bande de parche-  
min, dont les différents tours, se trouvant alors réunis, remettent les  
mots dans l'ordre où ils ont été écrits, et présentent toute la suite de  
la lettre. On appelle cette lettre scytale, du nom même du bâton,  
comme ce qui est mesuré prend le nom de ce qui lui sert de mesure.

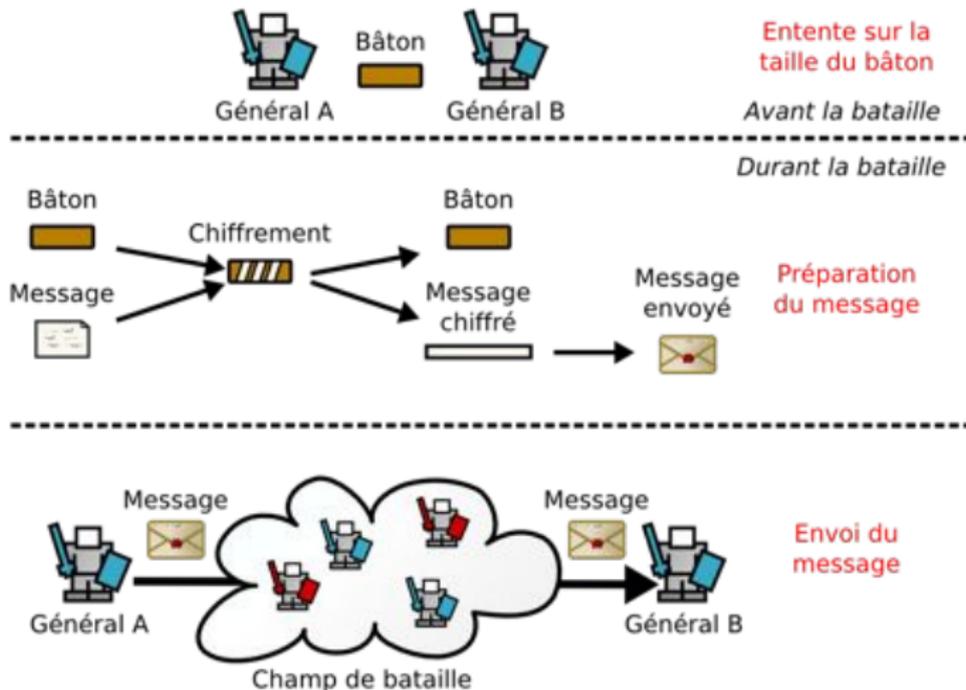
# La scytale – v<sup>e</sup> siècle avant J.C.

- Dispositif de chiffrement par transposition
  - Bâton ~ clef
  - Lanière de cuir ~ support du message
- Exemple avec une scytale à 5 caractères par contour

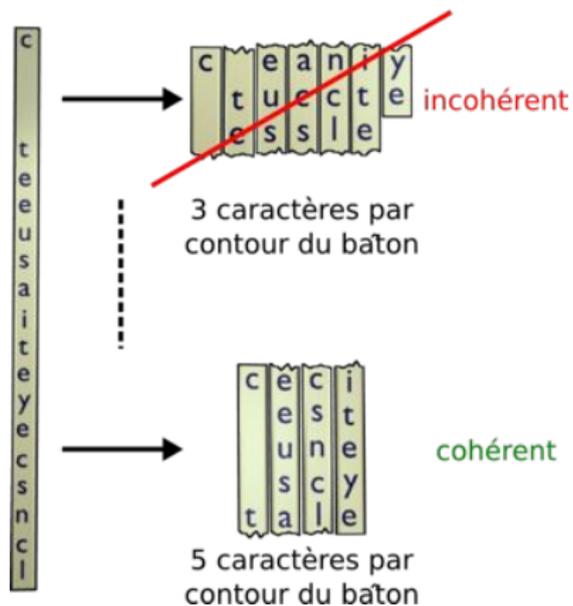
Message clair	ceci est une scytale
Message chiffré	c teusaiteyecsncl



# La scytale – *v<sup>e</sup> siècle avant J.C.*



# La scytale – v<sup>e</sup> siècle avant J.C.



# Sommaire

## Un peu d'histoire

La scytale – *v<sup>e</sup> siècle avant J.C.*

**Le chiffre de César – *I<sup>e</sup> siècle avant J.C.***

Le chiffre de Vigenère – *xvi<sup>e</sup> siècle après J.C.*

Le chiffre de Marie Stuart – *xvi<sup>e</sup> siècle après J.C.*

La machine Enigma – *xx<sup>e</sup> siècle après J.C.*

Kevin Mitnick – *25 décembre 1995*

A nos jours

# Le chiffre de César – 1<sup>e</sup> siècle avant J.C.

- Décalage de chaque lettre du message clair d'une distance fixe



- Soient  $n$  la distance de décalage,  $x$  la lettre à coder/décoder

$$\text{Codage} \quad C_n(x) = (x + n) \bmod 26$$

$$\text{Décodage} \quad D_n(x) = (x - n) \bmod 26$$

- Exemple

Message clair	chiffre de cesar
Distance 1	dijggsf ef dftbs
Distance 2	ejkhhtg fg eguct

- Utilisé par Jules César lors de la Guerre des Gaules avec  $n = 3$

# Le chiffre de César – 1<sup>e</sup> siècle avant J.C.

- Analyse fréquentielle possible (fonction de la langue)
  - Certaines lettres sont plus employées que d'autres
  - La fréquence d'une lettre dans un message égale la fréquence de son image dans le message chiffré

$$f(x, M) = f(C_n(x), C_n(M))$$

- Liste des lettres alphabétiques de la plus fréquente à la moins fréquente dans un texte français

EAISTNRULODMPCVQGBFJHZXYKW

- Exemple

atnqf zs fzywj hmnkkwj ij hjxfw xn kfhnqj f hfxxjw

# Le chiffre de César – 1<sup>e</sup> siècle avant J.C.

- Analyse fréquentielle possible (fonction de la langue)
  - Certaines lettres sont plus employées que d'autres
  - La fréquence d'une lettre dans un message égale la fréquence de son image dans le message chiffré

$$f(x, M) = f(C_n(x), C_n(M))$$

- Liste des lettres alphabétiques de la plus fréquente à la moins fréquente dans un texte français

EAISTNRULODMPCVQGBFJHZXYKW

- Exemple

atnqf zs fzywj hmnkkwj ij hjxfw xn kfhnqj f hfxxjw  
voilà un autre chiffre de césar si facile à casser

# Sommaire

## Un peu d'histoire

La scytale – *v<sup>e</sup> siècle avant J.C.*

Le chiffre de César – *i<sup>e</sup> siècle avant J.C.*

**Le chiffre de Vigenère – *xvi<sup>e</sup> siècle après J.C.***

Le chiffre de Marie Stuart – *xvi<sup>e</sup> siècle après J.C.*

La machine Enigma – *xx<sup>e</sup> siècle après J.C.*

Kevin Mitnick – *25 décembre 1995*

A nos jours

# Le chiffre de Vigenère – XVI<sup>e</sup> siècle après J.C.

- Blaise de Vigenère (1523 – 1596), diplomate français
- Amélioration du chiffre de César  $\Rightarrow$  substitution polyalphabétique
  - Une lettre de l'alphabet peut être chiffrée de plusieurs manières différentes
  - La clé est représentée par une chaîne de caractères
  - Un caractère de la clef = une distance

$$A=0 \quad B=1 \quad C=2 \quad \dots$$

- Répétition de la clé, si nécessaire
- Exemple

<i>Message clair</i>	c	h	i	f	f	r	e	d	e	v	i	g	e	n	e	r	e
<i>Clé</i>	u	n	e	c	l	e	u	n	e	c	l	e	u	n	e	c	l
<i>Décalage</i>	20	13	4	2	11	4	20	13	4	2	11	4	20	13	4	2	11
<i>Message chiffré</i>	w	u	m	h	q	v	y	q	i	x	t	k	y	a	i	t	p

# Le chiffre de Vigenère – XVI<sup>e</sup> siècle après J.C.

Comment obtenir le message clair avec, seulement, le message chiffré ?

Message chiffré esmhqvgxipeexpgnpgjtjhcifpzkrppvg

- Trois motifs : vg ( $\Delta = 27$ ), ip ( $\Delta = 21$ ) et pg ( $\Delta = 3$ )
  - La distance entre les répétitions d'un motif est multiple de 3
- ⇒ La clé a vraisemblablement une taille de 3

Analyse	e h g p x n j h f k p g
fréquentielle	s q x e p p t c p r p
	m v i e g g j i z i v

Message clair	chiffrementavec	lechiffredevigenere
Clé	cle	cleclecleclecleclecleclecleclecleclecle
Message chiffré	esmhqvgxipeexpgnpgjtjhcifpzkrppvg	

# Sommaire

## Un peu d'histoire

La scytale – *v<sup>e</sup> siècle avant J.C.*

Le chiffre de César – *1<sup>e</sup> siècle avant J.C.*

Le chiffre de Vigenère – *xvi<sup>e</sup> siècle après J.C.*

**Le chiffre de Marie Stuart – *xvi<sup>e</sup> siècle après J.C.***

La machine Enigma – *xx<sup>e</sup> siècle après J.C.*

Kevin Mitnick – *25 décembre 1995*

A nos jours

# Le chiffre de Marie Stuart – XVI<sup>e</sup> siècle après J.C.

- Tentative d'assassinat de la reine d'Angleterre
  - Marie Stuart (en prison) et ses complices, conspirateurs
- Chiffrement des correspondances avec une nomenclature
  - 1 symbole pour chaque lettres de l'alphabet (sauf j, v et w)
  - 36 symboles pour les mots usuels
  - 4 symboles nuls
  - 1 symbole indiquant la répétition du symbole suivant

a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	x	y	z
o	†	∧	‡	α	□	θ	∞	ı	ö	κ		∅	∇	∫	∩	f	Δ	ε	c	7	8	9

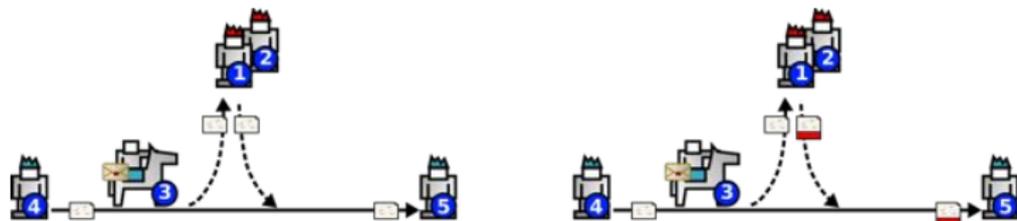
Nulles ff. | . | . | d.

Dowbleth σ

and	for	with	that	if	but	where	as	of	the	from	by	
2	3	4	4	4	3	∫	∫	∩	∩	∩	∩	
so	not	when	there	this	in	wich	is	what	say	me	my	wyrt
∫	x	∫	∫	∫	∫	∫	∫	∫	∫	∫	∫	∫
send	lre	receave	bearer	I	pray	you	Mte	your	name	myne		
∫	∫	∫	∫	∫	∫	∫	∫	∫	∫	∫		

# Le chiffre de Marie Stuart – XVI<sup>e</sup> siècle après J.C.

- Complot déjoué[dM][Lab]
  - ① Francis Walsingham : maître-espion de la reine d'Angleterre
  - ② Thomas Phelippes : chiffreur, déchiffreur du maître-espion
  - ③ Gilbert Gifford : messenger de Marie Stuart et agent double
  - ④ Marie Stuart : comploteur
  - ⑤ Anthony Babington : comploteur
- Comment identifier tous les comploteurs ?



*"Francis Walsingham est l'homme au milieu"*

# Sommaire

## Un peu d'histoire

La scytale – *v<sup>e</sup> siècle avant J.C.*

Le chiffre de César – *i<sup>e</sup> siècle avant J.C.*

Le chiffre de Vigenère – *xvi<sup>e</sup> siècle après J.C.*

Le chiffre de Marie Stuart – *xvi<sup>e</sup> siècle après J.C.*

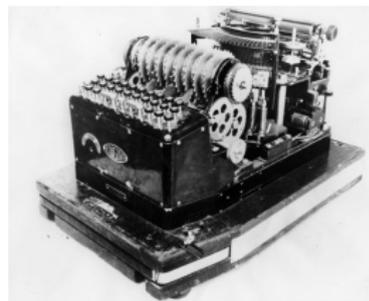
**La machine Enigma – *xx<sup>e</sup> siècle après J.C.***

Kevin Mitnick – *25 décembre 1995*

A nos jours

# La machine Enigma – *XX<sup>e</sup> siècle après J.C.*

- Arthur Scherbius (1878 – 1929), ingénieur en électricité allemand
- Machine électromécanique inventée en 1919, à but commercial
- Employée par l'armée allemande (U-Bot)
- Automatisation du chiffrement par substitution
  - Un clavier de 26 lettres
  - Un cadran lumineux de 26 lettres
  - Des rotors (en général 3)
  - Un réflecteur



# La machine Enigma – *xx<sup>e</sup> siècle après J.C.*



- Clé de chiffrement
  - Disposition et orientation des rotors
  - Connexions entre lettres de l'alphabet
- A chaque pression d'une touche
  - Une des ampoules du cadran lumineux s'allume
  - Le rotor le plus à droite pivote
  - A chaque tour complet d'un rotor, le rotor à sa gauche pivote
- La bombe de Turing : découverte de la clé en 1 heure

# Sommaire

## Un peu d'histoire

La scytale – *v<sup>e</sup> siècle avant J.C.*

Le chiffre de César – *1<sup>e</sup> siècle avant J.C.*

Le chiffre de Vigenère – *xvi<sup>e</sup> siècle après J.C.*

Le chiffre de Marie Stuart – *xvi<sup>e</sup> siècle après J.C.*

La machine Enigma – *xx<sup>e</sup> siècle après J.C.*

**Kevin Mitnick – 25 décembre 1995**

A nos jours

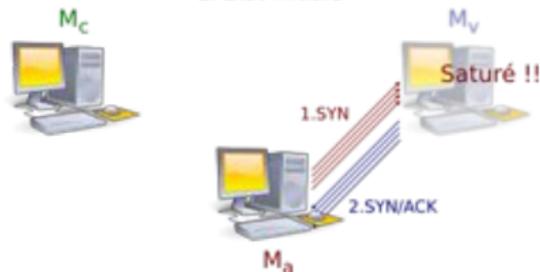
# Kevin Mitnick – 25 décembre 1995

- Attaque du 25 décembre 1994
- Protagonistes de l'affaire
  - Kevin Mitnick : pirate informatique américain
  - Tsutomu Shimomura : expert américain en sécurité informatique
- Configuration
  - $M_a$ , attaquant, San Francisco
  - $M_v$ , victime, San Diego
  - $M_c$ , cible, San Diego
  - Liaison de confiance  $M_v \rightarrow M_c$
- Objectif
  - Empêcher  $M_v$  de dialoguer avec  $M_c$
  - Permettre à  $M_a$  de répondre à la place de  $M_c$
- Techniques employées
  - Déni de service ou saturation de service (*SYN Flooding*)
  - Usurpation d'adresse IP (*IP Spoofing*)

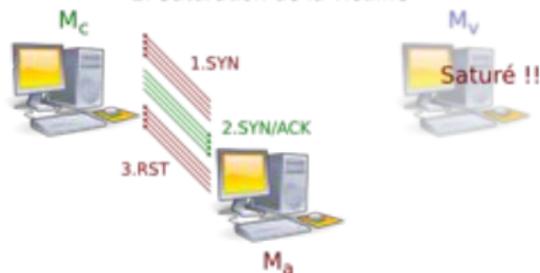
## Kevin Mitnick – 25 décembre 1995



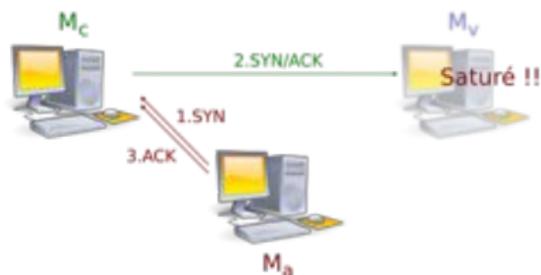
1. Etat initiale



2. Saturation de la victime



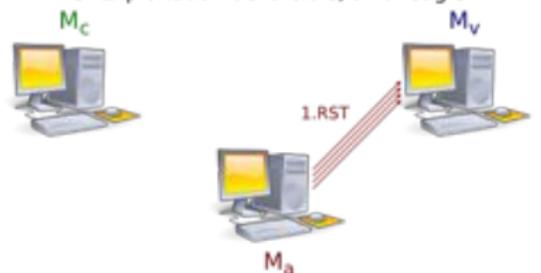
3. Analyse du comportement de la cible



4. Ouverture d'une connection avec la cible, à l'aveugle



5. Exploitation de la cible, à l'aveugle



6. Libération de la victime

# Sommaire

## Un peu d'histoire

La scytale – *v<sup>e</sup> siècle avant J.C.*

Le chiffre de César – *1<sup>e</sup> siècle avant J.C.*

Le chiffre de Vigenère – *xvi<sup>e</sup> siècle après J.C.*

Le chiffre de Marie Stuart – *xvi<sup>e</sup> siècle après J.C.*

La machine Enigma – *xx<sup>e</sup> siècle après J.C.*

Kevin Mitnick – *25 décembre 1995*

**A nos jours**

# A nos jours

- 1949 John Von Neumann, logiciels autocopiés
- 1960 ingénieurs des laboratoires Bell, Core war
- 1984 Scientific American, guide pour fabriquer ses propres virus
- 1986 Les frères Alvi, virus Brain
- 1988 Robert Morris, fraude informatique
- 2000 Virus "I Love You"

# Sommaire

Un peu d'histoire

Les propriétés de la sécurité

Les attaques

Les défenses

La protection des systèmes informatiques

# Sommaire

## Les propriétés de la sécurité

# Définition

## Definition

Sécurité = confidentialité + intégrité + disponibilité

- Perte de confidentialité = divulgation non-autorisée d'information
- Perte d'intégrité = altération de l'information
- Perte de disponibilité = interruption d'accès à l'information ou interruption du service d'un système d'information

*For most distributed systems, the security objectives of confidentiality, integrity, and availability of information apply. A loss of confidentiality is the unauthorized disclosure of information. A loss of integrity is the unauthorized modification or destruction of information. A loss of availability is the disruption of access to or use of information or an information system.[NIS]*

# Confidentialité

## Definition

La **confidentialité** est la propriété d'une information de ne pas être révélée à des utilisateurs non autorisés à la connaître.

- Empêcher les utilisateurs de lire une information confidentielle, sauf s'ils y sont autorisés
- Empêcher les utilisateurs autorisés à lire une information de la divulguer à des utilisateurs non-autorisés

# Intégrité

## Definition

L'**intégrité** est la propriété d'une information d'être exacte.

- Empêcher une modification (création, destruction) indue de l'information (incorrecte ou par des utilisateurs non-autorisés)
- Faire en sorte qu'aucun utilisateur ne puisse empêcher la modification légitime de l'information
- Faire en sorte que l'information soit créé/saisie
- Vérifier qu'elle est exacte (correspond au monde réel)

# Disponibilité

## Definition

La **disponibilité** est la propriété d'une information d'être accessible quand on en a besoin.

- Fournir l'accès à l'information pour que les utilisateurs autorisés puissent la lire ou la modifier
- Faire en sorte qu'aucun utilisateur ne puisse empêcher les utilisateurs autorisés d'accéder à l'information (lire ou modifier)

Disponibilité à long terme = Pérennité

# L'information

## Definition

Une information est composée de données et méta-données.

- **Données** : captées ou générées, traitées, stockées, transmises, affichées
- **Méta-données** : créées et utilisées par les services sous-jacents

Une méta-donnée est une donnée à un niveau inférieur

# Autres propriétés

## Anonymat

Confidentialité de (identité de l'utilisateur)

## Protection de la vie privée

Confidentialité de (identité de l'utilisateur + données personnelles)

## Authenticité d'un message

Intégrité de (contenu + identité de l'émetteur + date + ...)

## Authenticité d'un document

Intégrité de (contenu + identité du créateur + date + ...)

## Authenticité d'un utilisateur

Intégrité de (identité)

# Autres propriétés

## Imputabilité

Disponibilité de (qui + quoi + quand + où + ...) d'une action

## Non-répudiation d'origine

Disponibilité de (identité de l'émetteur + ...) +  
intégrité du (contenu)

## Non-répudiation de réception

Disponibilité de (identité du récepteur + ...) +  
intégrité du (contenu)

## Protection de la propriété intellectuelle

Confidentialité de (contenu) +  
intégrité du (contenant)

# Besoins de sécurité selon les secteurs

- Défense, gouvernement  
Confidentialité  $\gg$  intégrité, disponibilité
- Finance  
Intégrité  $\gg$  disponibilité  $>$  confidentialité
- Autres (industrie, administrations, médecine, ...)  
Ça dépend

$\Rightarrow$  Besoin de définir les spécificité de l'application

$\Rightarrow$  Politique de sécurité

# Les moyens pour la sécurité

## *Eviter les fautes*

### Prévention des fautes

Comment empêcher que des fautes surviennent ou soient introduites

### Elimination des fautes

Comment réduire la présence (en nombre ou en gravité) des fautes

## *Accepter les fautes*

### Tolérance aux fautes

Comment fournir un service conforme à la fonction en dépit des fautes

### Prévision des fautes

Comment estimer la présence, la création et les conséquences des fautes

# Sommaire

Un peu d'histoire

Les propriétés de la sécurité

**Les attaques**

Les défenses

La protection des systèmes informatiques

# Sommaire

## Les attaques

Les attaquants et leurs motivations

Classification des attaques

# Sommaire

## Les attaques

Les attaquants et leurs motivations

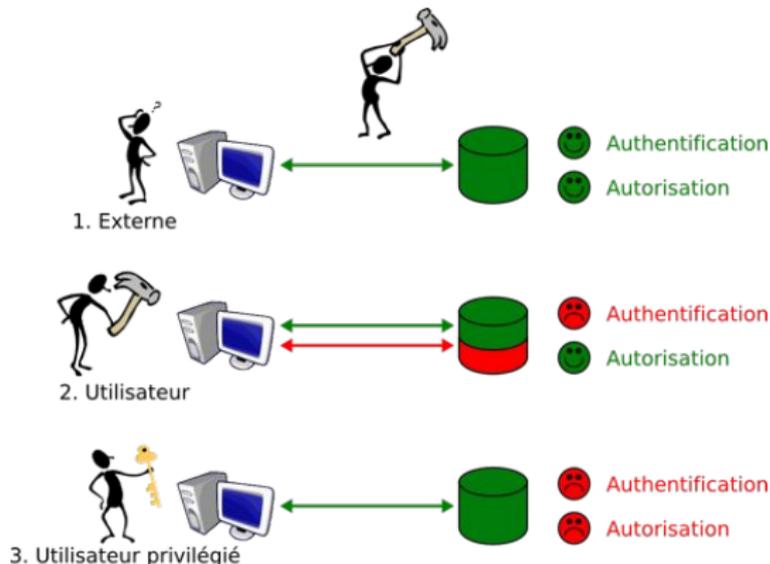
Classification des attaques

# Les attaquants et leurs motivations

- **Jeu** : explorer les limites, éprouver et étendre ses connaissances, découvrir de nouvelles failles, améliorer la sécurité : “hackers”
- **Emulation, sectarisme** : groupe de hackers : “exploits”
- **Vandalisme** : montrer sa force, punir : “web defacing”, virus, vers, ...
- **Politique, idéologie** : ex. CCC, 600 sites danois “défigurés” en février 2006
- **Vengeance** : ex. SCORES
- **Profit** : **espionnage, extorsion de fonds** : concurrence déloyale, crime organisé, espionnage international (attaques probablement chinoises contre des sites gouvernementaux des USA, GB, Allemagne, France, ...)
- **Guerre informatique, terrorisme** : 2007 DDoS contre des sites estoniens, 2008 contre des sites géorgiens, ...
- **Sensibilisation, lobbying**
- **Protection abusive** : ex. SONY

# Les attaquants et leurs motivations

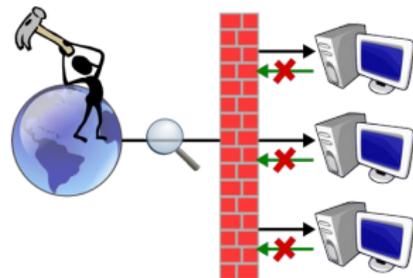
Qui sont les "intrus" ?



80% des fraudes sont "autorisées"

# Les attaquants et leurs motivations

- Organisation
    - Seul ? Groupe ?
  - Compétence
    - Novice ? Averti ? Expert ?
  - Comportement
    - Discret ? Ostensible ?
- ⇒ Utilisation de “pots de miel”



# Sommaire

## Les attaques

Les attaquants et leurs motivations

Classification des attaques

# Classification des attaques(1)

- Ecoute passive (**confidentialité**)  
Accès sans modification à des informations générées, transmises, stockées ou affichées sur des composants vulnérables
  - *sniffing, snooping, eavesdropping, wiretapping*, réutilisation de mémoire (buffers, fichiers temporaires, supports magnétiques), analyse de trafic, effet Van Heck (**TEMPEST**), *key logger*, ...  
<http://lasecwww.epfl.ch/keyboard>



Effet Van Heck

## Classification des attaques(2)

- Interception (**intégrité**)  
Modification d'informations transmises
  - Modification de messages, rejeu, éblouissement
- Cryptanalyse (**confidentialité**)  
Obtenir des informations secrètes (messages en clair, clés, algorithme de chiffrement) à partir des informations publiques (cryptogrammes)
  - Identification de collisions dans MD5 en 2004[WFLY04], utilisé pour signer, par exemple, les fichiers téléchargés
- Répudiation (**intégrité**)  
Refuser de reconnaître une opération qu'on a effectuée
  - Répudiation d'origine, de réception

# Classification des attaques(3)

- Déduction par inférence, furetage (**confidentialité**)  
Obtenir des informations secrètes (par exemple, des données personnelles) à partir des informations auxquelles on a accès (par exemple, statistiques)
- Déguisement (**masquerade**) (**intégrité**)  
Se faire passer pour quelqu'un d'autre (tromper l'authentification, s'il y en a ... )
  - *Phishing* → obtenir des renseignements personnels
  - *Scam* (Fraude 4-1-9) → escroquer
  - *WiPhishing* : hotspots WiFi ouverts
  
  - *Attaques homographes* → cacher les vraies URL
  - *IP spoofing* (contre-mesure : *ingress filtering*)
  - *DNS poisoning* : URL ↔ @IP
  - *ARP poisoning* : @MAC ↔ @IP

## Cas particulier de déguisement : *phishing*

- Hameçonnage de mots de passe : autrefois par téléphone  
Maintenant : (courriel, blog, news, IRC, MSN Messenger...)  
avec **URL cliquable** pointant vers une autre URL, ou (SMS)
- 49 084 sites actifs en juin 2009, durée de vie moyenne : 1 jour à 1 mois  
<http://www.antiphishing.org> et <http://phishery.internetdefence.net>

- En 2007, 3,6  $m^{ns}$  d'américains piégés, 3,2  $m^{ds}$  de dollars de perte
- Exemple, en 2005 : Croix-rouge et cyclone Katrina, grippe aviaire, ...
- Octobre 2009 : *Des fraudeurs se font passer pour les services des impôts sur internet*[L'E09]

Avril 2009		Mai 2009		Juin 2009	
USA	66,24%	USA	68,65%	Suede	46,18%
Chine	8,00%	Chine	6,33%	USA	42,39%
Suede	7,76%	Canada	6,15%	Canada	3,52%
Canada	2,67%	Allemagne	2,24%	Chine	1,57%
Allemagne	1,97%	Royaume-Uni	1,60%	Allemagne	0,88%
Royaume-Uni	1,02%	Suede	1,29%	Royaume-Uni	0,54%
Pays-Bas	0,98%	Russie	1,23%	France	0,53%
Coree	0,90%	France	1,19%	Coree	0,40%
France	0,86%	Coree	1,17%	Pays-Bas	0,39%
Russie	0,85%	Pays-Bas	1,01%	Russie	0,28%

# Cas particulier de déguisement : *phishing*

From remboursement@impots.gouv.fr Mon Oct 5 09:17:46 2009  
Return-Path: <remboursement@impots.gouv.fr>  
Reply-To: <remboursement@impots.gouv.fr>  
From: "L'administration Fiscale" <remboursement@impots.gouv.fr>  
Subject: Notification d'impôt  
Date: Mon, 5 Oct 2009 02:55:50 -0400  
Content-Type: text/html; charset="Windows-1251"  
X-Spam-Status: Yes  
X-Spam-Score: 9.706 (\*\*\*\*\*)



DIRECTION GENERALE DES FINANCES PUBLIQUES

05/10/2009

Notification d'impôt - Remboursement

---

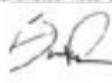
Après les derniers calculs annuels de l'exercice de votre activité, nous avons déterminé que vous êtes admissible à recevoir un remboursement d'impôt de € 178,80.

S'il vous plaît soumettre la demande de remboursement d'impôt et nous permettre de 10 jours ouvrables pour le traitement.

Pour accéder au formulaire pour votre remboursement d'impôt, [cliquez ici](#)

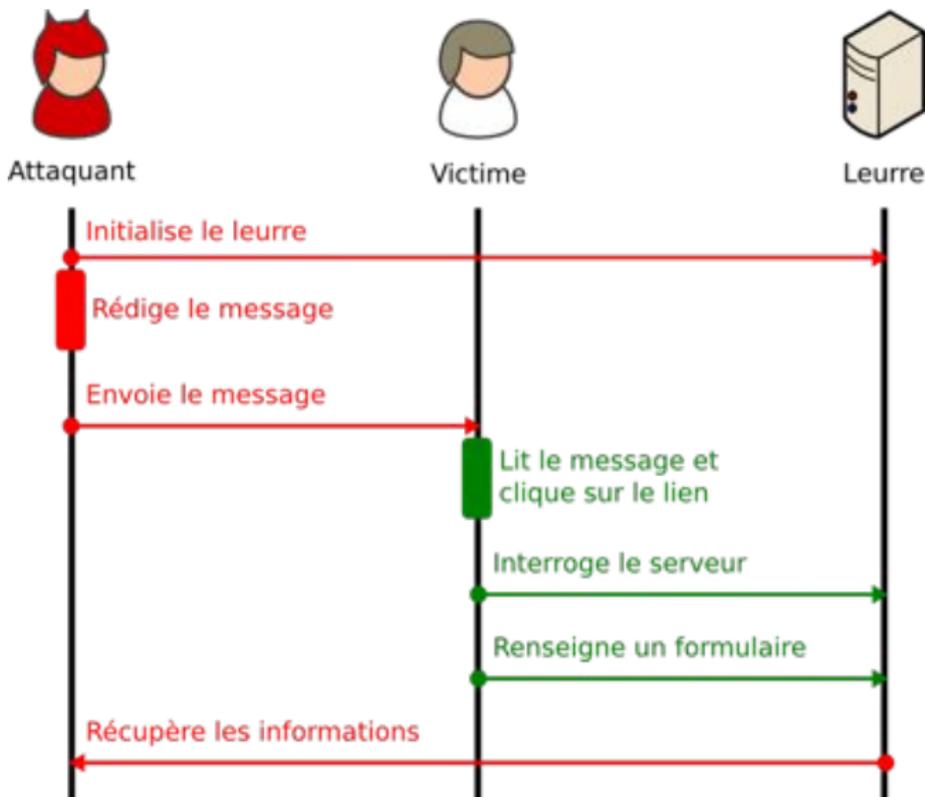
Un remboursement peut être retardé pour diverses raisons. Par exemple la soumission des dossiers non valides ou inscrites après la date limite.

Le Conciliateur fiscal adjoint



Philippe BERGER

# Cas particulier de déguisement : *phishing*



# Comment cacher la vraie URL ?

- *Open Redirect* : redirection vers une URL sans validation

<http://unsite.com/redirect?url=http://unstie.com>

```
http://cgi4.ebay.com/ws/eBayISAPI.dll?MfcISAPICommand=RedirectToDomain&
DomainUrl=http%3A%2F%2F%32%31%31%2E%31%37%32%2E%39%36%2E%37%2F
UpdateCenter%2FLogin%2F%3FMfcISAPISession%3DAAJbaQqzeHAAeMWZlHh1WXS2A1B
XVShqAhQRfhgTDrferHCUrstpAisNRqAhQRfhgTDrferHCUrstpAisNRpAisNRqAhQRfhgT
DrferHCUQRfqzeHAAeMWZlHh1WXh
```

- *Pharming* : tromper la traduction DNS
  - *DNS poisoning*
  - Modifier la table /etc/hosts
- *Phishing* et HTTPS
  - MD5 cassé ⇒ possibilité de forger des certificats
  - Vérifier la présence du cadenas ne suffit plus

# Comment cacher la vraie URL ?

- Attaques homographes : obtenir un nom de domaine en caractères UNICODE non-latins (ex. : cyrillique) qui s'affiche comme le site cible : **eBay.com** ≠ **eBay.com**
- Attaques dites d'erreur typographiques : par exemple, `goggle.com` ou `google.com`  
Plus subtil, pages d'erreur *DNS* redirigées par le *FAI* vers un site publicitaire hacké, *Wildcard DNS*  
`www.schneier.com/blog/archives/2008/04/hacking_isp_err.html`
- Attaques par modification de logiciel
  - Noyau ou interception des appels systèmes
  - Modification des applications : browser (java, plug-ins), mail.

# Attaques dites d'erreur typographiques

ryanair.com → <http://www.ryanair.com/site/FR/>

The screenshot shows the French version of the Ryanair website. At the top, there is a navigation bar with the Ryanair logo and a language selector set to 'France (Français)'. Below this is a menu with categories like ACCUEIL, LOCATION DE VOITURE, HÔTELS À BAS PRIX, etc. The main content area is divided into several sections:

- Left sidebar:** A large red banner with the text 'AUCUNE TAXE OU CHARGE'. Below it, a section titled 'RÉSERVEZ AVANT 14.10.09!' with a list of destinations from Marseille: Biarritz (Gratuit), Brest (Gratuit), Lille (Gratuit), Nantes (Gratuit), Paris-Beauvais (Gratuit), and Tours (Gratuit). At the bottom of this sidebar are buttons for 'GÉRER MA RÉSERVATION' and 'ENREGISTREMENT EN LIGNE'.
- Center:** A search form for flights with fields for 'Origine', 'Destination', 'Date de départ', and 'Date de retour'. It also includes options for 'Aller-retour' or 'Aller simple' and a 'RECHERCHER DES VOLS' button. Below the search form is a 'NEWS UPDATES' section with a link for 'Informations Importantes: Enregistrement En Ligne'. A large black box with 'BOOK NOW!' and 'RYANAIR.COM' is also present.
- Right sidebar:** Promotional banners for 'Location de Voiture' (rental car), 'Hôtels à Bas Prix' (hotels), and 'Bus Aéroport' (airport bus).
- Bottom:** A 'SERVICES' section with two sub-sections: 'PRIX GARANTI' (Price Guarantee) and 'LOCATION DE VOITURE' (Car Rental).

# Attaques dites d'erreur typographiques

ryamair.com → <http://www.searchnut.com/?domain=ryamair.com>

**ryamair.com** Search the Web:

[Ryanair](#) | [Airline](#) | [Aer Lingus](#) | [Airline Tickets](#) | [Ryanair Cheap Flights](#) | [Low Cost Airlines](#) | [Cheap Flights](#)

**Ryanair**

- [Airline](#)
- [Aer Lingus](#)
- [Airline Tickets](#)
- [Ryanair Cheap Flights](#)
- [Low Cost Airlines](#)
- [Cheap Flights](#)
- [Aviation](#)
- [Plane](#)
- [Europe Airline](#)
- [Air Travel](#)



**Aviation**

- [Commercial Aviation](#)
- [Aviation Services](#)
- [Aero](#)
- [Aviation Information](#)
- [Aviation Service](#)

**Plane**

- [Pilot](#)
- [Airline Tickets](#)
- [Train](#)
- [Jet](#)
- [Helicopter](#)

**Low Cost Airlines**

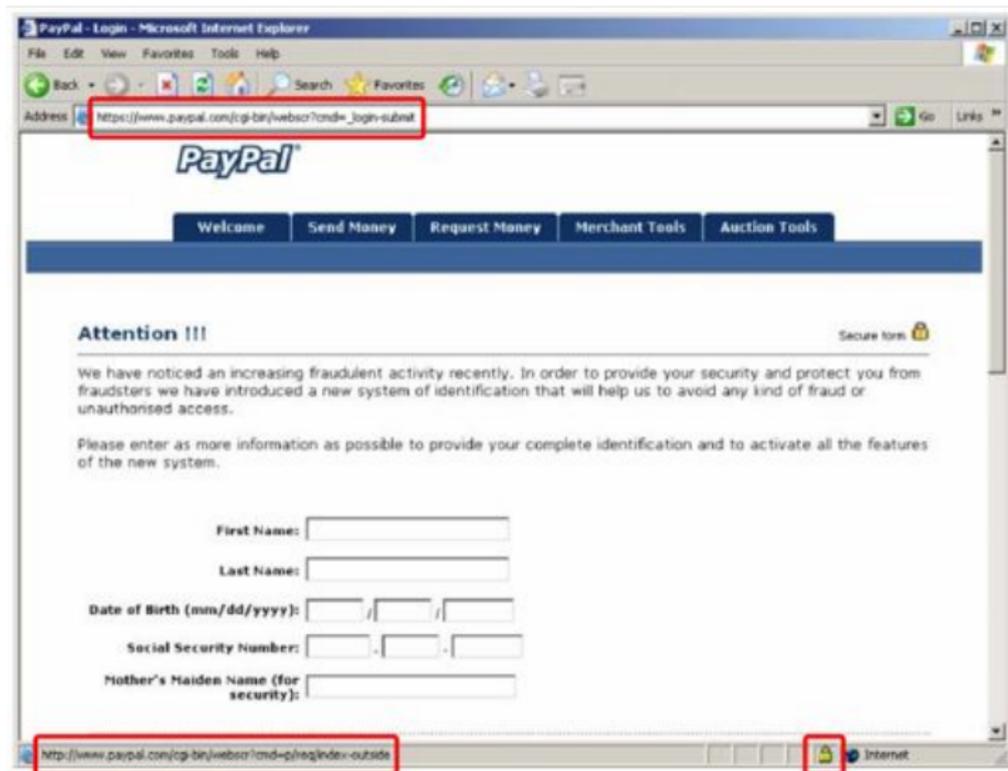
- [Discount Airline](#)
- [Europe Airline](#)
- [Low Cost Air](#)
- [Cheap Flights](#)
- [Cheap Airline](#)

# Pharming : tromper la traduction DNS



# Attaques par modification de logiciel

https://www.paypal.com



http://www.paypal.com

## Cas particulier de déguisement : Scam

Date: Thu, 24 Sep 2009 11:31:29 -0700 (PDT)  
From: Linda Fastus <ms\_lindafastus@info2link.biz>  
Subject: FROM MRS LINDA FASTUS SCHWARZ

X-Spam-Status: Yes  
X-Spam-Score: 12.088 (\*\*\*\*\*)

FROM MRS LINDA FASTUS SCHWARZ  
ABIDJAN COTE D'IVOIRE  
PLEASE EMAIL BACK

DEAREST IN CHRIST,

KNOW THAT THIS MAIL MAY REACH YOU BY SURPRISE.AS WE DONT KNOW OURSELF PREVIOUSLY.I AM THE ABOVE NAME PERSON FROM INDIA. I AM MARRIED TO MR FASTUS SCHWARZ ;WHO WAS THE AMBASSADOR OF JAMAICA FOR NINETEN YEARS IN COTE DIVOIRE.WE WERE MARRIED FOR FIFTEEN YEARS WITHOUT A CHILD. HE DIED IN DECEMBER 27TH 2004 AFTER A BRIEF ILLNESS THAT LASTED FOR ONLY TWO WEEKS

BEFORE HIS DEATH WE ARE HAPPY HUSBAND AND WIFE CHRISTIAN FAMILY. SINCE HIS DEATH IDECDED NOT TO REMARRY OR GET A CHILD OUTSIDE MY MATRIMONIAL HOME WHICH THE BIBLE IS AGAINST. WHEN MY LATE HUSBAND WAS ALIVE, HE DEPOSITED THE SUM OF **USD \$3.2.7MILLION** TWELVE MILLION SEVEN HUNDRED THOUSAND U.S.DOLLARS INTO A BOX FOR SECURITY REASON AND THE MONEY STILL WITH THE SECURITY COMPANY HERE IN ABIDJAN COTE D'IVOIRE.

MEANWHILE, I HAVE NOT TELL ANY BODY THE CONTENT OF THIS DEPOSIT IN THE SECURITY COMPANY.I AM TELLING YOU THE CONTENT REASON THAT I WANT YOU TO ASSIST ME USE THE FUND FOR THE WORK OF GOD. EVEN DO YOU ARE NOT A CHRISTAIN.THAT IS NOT A PROBLEM .WHAT I WANT IS FOR YOU TO USE IT AND HELP THE HELPLESS PEOPLE AROUND YOU .TO HELP THE ORPHANAGES, WIDOWS, AND MOTHERLES CHILDRENS

RECENTLY, MY DOCTOR TOLD ME THAT I HAVE SERIOUS SICKNESS WHICH IS CADIAC PROBLEM.THE ONE THAT DISTURBS ME MOST IS MY STROKE SICKNESS HAVING KNOWN MY CONDITION I DECIDED TO DONATE HIS FUND TO YOU TO UTILIZE THIS MONEY ACCORDING TO MY DIRECTION AND THE WILL OF GOD.

PLEASE DO GIVE URGENT RESPONSE TO THIS MAIL WITHOUT ANY DELAY.

I WANT TO GIVE YOU NUMBER TO CALL ME BUT I DONT WANT IN A WAY MY HUSBAND RELATIONS WILL KNOW THAT I AM GIVING YOU THIS MONEY. I HAVE SISTER NURSE WHO IS FEARFUL TO THE LORD THAT WILL BE HELPING GIVING YOU INFORMATION OF THIS DEPOSIT.HER NAME ID SISTER (CHANTAL KONE)

SO PLEASE I AM WAITING FOR YOUR URGENT REPLY SO THAT I CAN GIVE YOU ALL THE INFORMATION ABOUT THIS MONEY AND THE SECURITY COMPANY WERE IT WAS DEPOSITED BY MY LATE HUSBAND ,

REMAIN BLESSED ALWAYS  
YOURS SISTER IN CHRIST  
MRS LINDA FASTUS SCHWARZ

# Classification des attaques(4)

- Canaux cachés (**covert channels**) (**confidentialité**)  
Communiquer (*high* → *low*) par des moyens non-surveillés
  - Canaux de stockage (ou canaux mémoires)
  - Canaux temporels
  - Autres : stéganographie, modulation analogique, canaux de fuite indirects, ...
- Canaux de fuite (**side channels**) (**confidentialité**)  
Obtenir des informations cachées (*high*) de façon détournée
  - Exemples avec les cartes à puce : analyse de la consommation de courant d'alimentation (simple SPA, différentielle DPA)
  - Captation : microscope à balayage, micro-sondes, rayonnement électromagnétique, ...
  - Injection de faute : micro-sondes, impulsions électro-magnétiques (y compris lumineuses), rayonnement nucléaire, ...
  - Analyse logicielle (fuites, analyse temporelle)

# Classification des attaques(5)

- Porte dérobée (**Trapdoor** / **Backdoor**)  
(**confidentialité**, **intégrité**, **disponibilité**)  
Contourner les mécanismes de protection
  - Authentification (Turing Award de Ken Thompson), autorisation
  - Exemple : *l'œuf du coucou*, Clifford Stoll, 1986
  - *Rootkits*
    - Utilisation d'une porte dérobée pour devenir *root* (escalade de privilège)
    - Modification du noyau, appels systèmes ou commandes (ps, w, netstat, ...)
    - Installation d'une porte dérobée pour un accès plus facile (ex. à distance)
    - Installation de logiciels malveillants, invisibles au niveau utilisateur

# Classification des attaques(6)

- *Spyware* (**confidentialité, disponibilité**)  
80% des PC professionnels infectés (estimation Webroot août 2005)
  - *Keyloggers, screen grabbers, sniffers*, analyse de fichiers, ...
  - Installés par des vers, des pages Webs minées, du *phishing*, freeware, ...
  - Octobre 2005 : Intermix paie 7,5  $m^{ns}$  de dollars pour un retrait de plainte pour *spyware*
- *Spyware légitime?*
  - Rapports d'anomalies
  - Mises à jour automatiques
  - Vérification de versions (anti-piratage)
  - Détection de tricherie (World of Warcraft)

# Classification des attaques(7)

- Bombe logique (**confidentialité, intégrité, disponibilité**)  
Déclencher des dégâts sur un événement particulier
  - Divulgence d'information confidentielle (ex. virus SirCam)
  - Destruction, modification de données/programmes (disques, audit)
  - Diffusion de fausses informations (ex. diagnostic)
  - Dégâts matériels (ex. virus Tchernobyl)
  - Installer un *zombie*, *spammer*, *spyware*, ...
- Logiciels malveillants (**confidentialité, intégrité, disponibilité**)  
(*malware* / maliciels : *rootkits*, *zombies*, ...)
  - Furtivité (*stealth*)
  - Escalade de privilèges (jusqu'à *root*)
  - Installation de portes dérobées, de bombes logiques, de *spyware*, ...
  - Pages Web *minées* (→ firewalls)
  - Exécution de scripts HTTP

→ Correction des failles pour protéger le zombie contre d'autres pirates !

## Exemple de pages Web minées

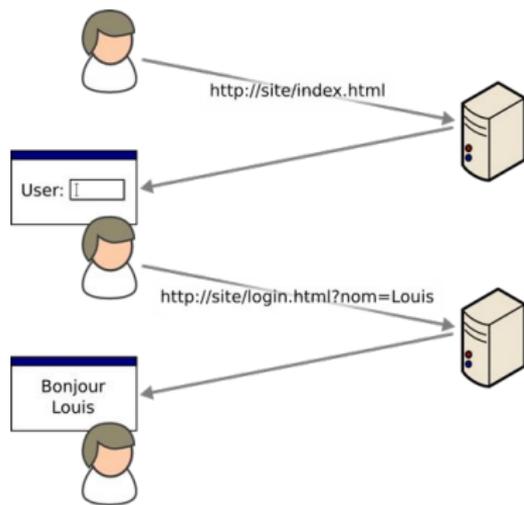
- Décembre 2005 : 2 *zero-day exploits* (MS05-054 et MS06-001) sur plus de 1500 sites Web : *Broad Proliferation of Crimeware Sites Exploiting WMF Image-Handling Vulnerabilities*

*The websites ... are using the exploit to distribute Spyware applications and other Potentially Unwanted Software. The user's desktop background is replaced with a message warning of a spyware infection and a "spyware cleaning" application is launched. This application prompts the user to enter credit card information in order to remove the detected spyware. . . In addition, a mail relay is installed on the infected computer and it will begin sending thousands of SPAM messages.*

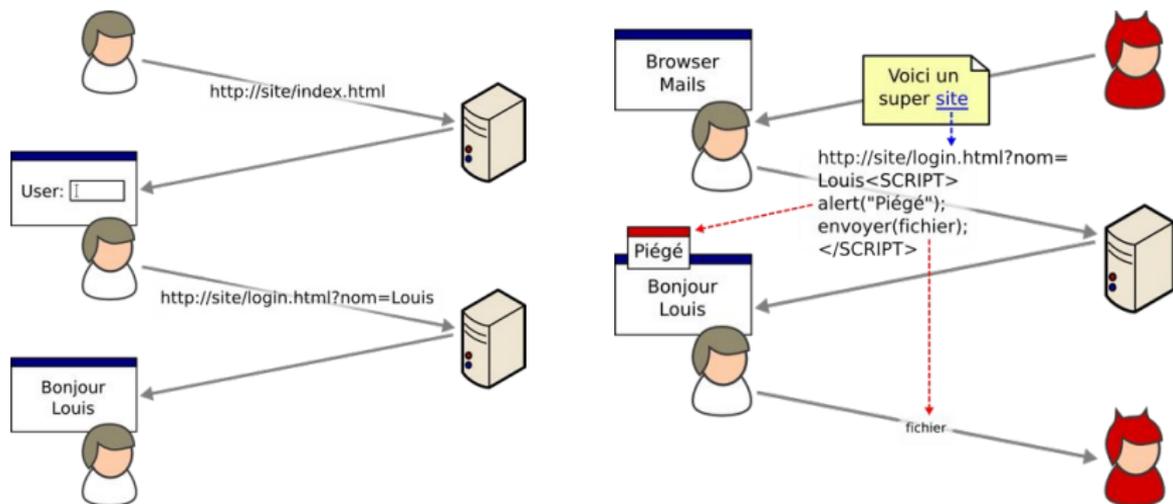
[www.antiphishing.org](http://www.antiphishing.org)



# Le Cross Site Scripting (XSS)



# Le Cross Site Scripting (XSS)



[www.cert.org/advisories/CA-2000-02.html](http://www.cert.org/advisories/CA-2000-02.html)

- Le pirate peut communiquer le lien soit par *phishing* soit indirectement via, par exemple, un *blog* ou *forum* d'un serveur innocent
- La victime lit le message avec un navigateur configuré pour permettre l'exécution de scripts

## Classification des attaques(8)

- Cheval de Troie (**confidentialité, intégrité, disponibilité**)  
Fonction illicite cachée dans un programme apparemment bénin
  - Divulgence (*SoBig, P2P-Winny*) ou modification d'information, bombe logique
  - Exemples : disquette AIDS (1989), PGPCoder (ransomware), pages web *minées*

Sur [www.hifocus.net](http://www.hifocus.net), le 05/03/05 :

*... , je suis allé télécharger winrar sur  
[http://www.01net.com/telecharger/windows/Utilitaire/  
compression\\_et\\_decompression/fiches/2257.html](http://www.01net.com/telecharger/windows/Utilitaire/compression_et_decompression/fiches/2257.html) ... et j'ai mis  
une heure pour désinfecter le pc. Pour un logiciel payant, ce  
n'est quand même pas normal de choper un virus. Il y avait  
entre autres **Trojan-Spy.Banker.EA** ...*

## Informations sur *Trojan.Banker.FA*

*Trojan.Banker.FA est un cheval de Troie voleur de mots de passe prenant pour cible les clients d'une banque brésilienne.*

*Trojan.Banker.FA surveille l'accès Internet de l'utilisateur. Lorsque certains sites bancaires sont visités sur Internet, le cheval de Troie affiche un faux écran de connexion afin de tromper l'utilisateur pour qu'il saisisse ses détails. Trojan.Banker.FA transmet ensuite les détails qu'il a subtilisés à une adresse électronique brésilienne.*

*Trojan.Banker.FA peut aussi télécharger et installer d'autres logiciels associés. Lorsqu'il est exécuté, Trojan.Banker.FA se copie dans le dossier système Windows sous le nom de CARTAO.EXE et, pour être exécuté au démarrage du système, paramètre l'entrée de registre suivante :*

```
HKCU\Software\Microsoft\Windows\CurrentVersion  
\Runcartao<système>\cartao.exe
```

## Phishing utilisé pour télécharger un cheval de Troie

*In July, Websense® Security Labs discovered a new malicious website, which distributed malicious code that installs a Trojan Horse on end-users' machines. This potentially occurs without user interaction.*

*The site appeared to be mirroring a World Cup 2006 Soccer website with the exception that they have a lead story regarding the now infamous, Zinedine Zidane head butt incident from the World Cup final against Italy.*

*Upon visiting any of the pages on the site, end-users were potentially infected with a Trojan Horse downloader. This Trojan Horse downloads additional payload code from the site. The site was using the underground "Web Attacker" toolkit (discussed in an earlier alert*

*<http://www.websense.com/securitylabs/alerts/alert.php?AlertID=472>). The Web Attacker toolkit is sold on a Russian website and costs anywhere from \$20 to \$300. This toolkit allows users to install code that exploits users based on their browser types. The installed code includes one of five different variants, including exploits for old and new vulnerabilities.*

*This site was hosted in the United States.*

*[www.antiphishing.org](http://www.antiphishing.org) – Juillet 2006*

# Exemple de pages Web minées

The screenshot shows the FIFA World Cup 2006 Germany website. At the top, there is a banner with the logo and the text "FIFA WORLD CUP GERMANY 2006". Below the banner, there is a navigation menu with links: Home, News List, Match Schedule, Results. The main content area features a "World Cup 2006 Top Story" section with the headline "What did Materazzi say to Zidane?". Below the headline is a small image showing Zidane and Materazzi on the field. To the right of the main content, there is a sidebar with a list of "FIFA World Cup 2006 Champions" and "Teams that did not qualify".

**FIFA World Cup 2006 Champions**

- Italy
- Second Place
- France
- Third Place
- Germany
- Fourth Place
- Portugal

**Teams that did not qualify**

- Brazil
- England
- Ukraine
- Argentina
- Spain
- Ghana
- Switzerland
- Australia
- Netherlands
- Ecuador
- Mexico
- Sweden
- Poland
- Costa Rica
- Paraguay

**World Cup 2006 Top Story**

## What did Materazzi say to Zidane?

**PARIS - The Zinedine Zidane mystery is not quite solved yet.**

In his first, highly awaited comments since the World Cup final, the French soccer star only partly explained what caused him to react in fury and head-butt an Italian opponent: repeated harsh insults about his mother and sister.

But Zidane didn't go into specifics about what Marco Materazzi said. Materazzi swears he never insulted Zidane's mother. And FIFA is still investigating.

## Classification des attaques(9)

- Virus (**confidentialité, intégrité, disponibilité**)  
Segment de code qui, lorsqu'il est exécuté, se reproduit en s'attachant à un autre programme (système ou application), éventuellement porteur d'une bombe logique
  - Période d'incubation
  - Seulement sur des fichiers de programme
  - Propagation par échange de support ou par réseau
  - Exemples : *Brain*, *Vendredi 13*, *macrovirus*, etc.
- Ver (*worm*) (**confidentialité, intégrité, disponibilité**)  
Programme autonome, capable de se répliquer et de se propager, éventuellement porteur d'une bombe logique
  - Mail (sendmail, outlook, majordomo, etc.) débordement de buffer (IIS, SQL Server, LSASS, etc.), Instant Messaging (IRC, AOL, Yahoo!, MSN, etc.), P2P, etc.
  - Exemples : Xerox, CHRISTMAS, Robert T. Morris, ILOVEYOU, Code Red, Slammer, etc.

# Classification des attaques(10)

- Dénier de service (*DoS : denial of service*) (**disponibilité**)  
Empêcher les utilisateurs légitimes d'accéder aux informations ou aux services auxquels ils ont droit
  - *flooding, smurfing (ICMP echo requests)*, DDoS (par *botnets*)
    - Ver de Morris (novembre 1988)
    - DDoS : février 2000 (Amazon, CNN, eBay, etc.), octobre 2002 (DNS), juin 2004 (Akamai), février 2007 (DNS),
    - Février 2006 : un espagnol condamné à 2 ans de prison et 1,4  $m^{ns}$  d'euros pour un DDoS ayant affecté 1/3 des utilisateurs d'ISP espagnols en 2003 (vengeance contre son exclusion d'un IRC)
    - Octobre 2006 : 4 russes condamnés à 8 ans de prison pour avoir extorqué plus de 4  $m^{ns}$  de dollars à des casinos et bookmakers sous menace de DDoS
  - *Spamming*
    - Escroquerie britannique aux noms de domaines : 1,6  $m^{ns}$  de livres détournés, mille à 5  $m^{ns}$  de mails par victimes, 6 ans de prison
    - Un spammer condamné à payer 5,6  $m^{ns}$  de dollars à AOL

# Classification des attaques(10)

- Attaques complexes
  - *Spam/Blogs* → *Vers/Chevaux de Troie* → Botnets (IRC, P2P, etc.)  
→ *spamming*, *phishing*, DDoS, chantage, serveur illégal, etc.
    - Octobre 2005 : 3 Hollandais arrêtés pour avoir pris le contrôle de 1.500.000 machines (zombies) avec un ver
    - Mai 2006 : 1 californien de 20 ans, Jeanson James Ancheta, condamné à 57 mois de prison pour avoir créé 400.000 zombies (contrôlés par IRC) et s'être fait payer 100.000 dollars par des sites publicitaires
    - En 2007, *Storm Worm* : estimation : 1 à 50  $m^{ns}$  de zombies, utilisés pour spam (record : 57  $m^{ns}$  le 22/08/07) + DDoS de sites de spam, anti-spam ou de contre-mesures [www.schneier.com/blog/archives/2007/10/the\\_storm\\_worm.html](http://www.schneier.com/blog/archives/2007/10/the_storm_worm.html)
  - Attaques ciblées (*harponnage*, *spear phishing*) : un cheval de Troie spécifique, difficile à détecter, visant une compagnie (exemple : espionnage industriel en Israël), ou une agence gouvernementale particulière, etc.  
Autre exemple : cibler une banque particulière : Auto-XSS + *session hijacking*

# Exemple d'harponnage (1)

**December 4, 2007**

## **MIS Sounds Alarm on Internet Spying from China**

Someone in MIS is [pissed off](#) at China:

In an unprecedented alert, the Director-General of MIS sent a confidential letter to 300 chief executives and security chiefs at banks, accountants and legal firms this week warning them that they were under attack from "Chinese state organisations."

[...]

Firms known to have been compromised recently by Chinese attacks are one of Europe's largest engineering companies and a large oil company, The Times has learnt. Another source familiar with the MIS warning said, however, that known attacks had not been limited to large firms based in the City of London. Law firms and other businesses in the regions that deal even with only small parts of Chinese-linked deals are being probed as potential weak spots, he said.

A security expert who has also seen the letter said that among the techniques used by Chinese groups were "custom Trojans", software designed to hack into the network of a particular firm and feed back confidential data. The MIS letter includes a list of known "signatures" that can be used to identify Chinese Trojans and a list of internet addresses known to have been used to launch attacks.

A big study gave warning this week that Government and military computer systems in Britain are coming under sustained attack from China and other countries. It followed a report presented to the US Congress last month describing Chinese espionage in the US as so extensive that it represented "the single greatest risk to the security of American technologies."

[www.schneier.com/blog/archives/2007/12/mi5\\_sounds\\_alar.html](http://www.schneier.com/blog/archives/2007/12/mi5_sounds_alar.html)

## Exemple d'harponnage (2)

Publié le 04/01/2008 à 08:28 Le Point.fr

### L'armée sud-coréenne subit des attaques informatiques

Guerric Poncet

Le ministère de la Défense de la Corée du Sud est en ébullition : des soldats ont subi des attaques informatiques sur leurs ordinateurs et des pirates ont réussi à voler des données personnelles. Selon le porte-parole du ministre, cité par l'AFP, "aucune donnée militaire n'a été dérobée". Mais rien ne permet d'en être sûr, les soldats ayant pu violer les règles de sécurité informatique en enregistrant des données confidentielles sur leurs postes personnels.

Grâce à un e-mail, intitulé avec intelligence "État actuel des capacités militaires de la Corée du Nord" et infecté par un cheval de Troie (programme malicieux destiné à "ouvrir les portes" de l'ordinateur), les pirates ont parfaitement réussi leur coup. Officiellement, l'attaque provenait "de l'étranger", sans plus de précisions. Plus bavarde, la presse ne cesse de parler de la Chine, qui développe une véritable force d'attaque cybernétique depuis plusieurs années. En novembre 2006, un rapport du Congrès des États-Unis s'inquiétait de la formation par Pékin d'unités de combat informatique (NET Force), capables d'infiltrer n'importe quel ordinateur, de Wall Street au Pentagone. Un cri d'alarme à prendre toutefois avec des pincettes, dans un contexte où le Congrès hésitait à accorder des crédits supplémentaires pour la protection de la sécurité nationale.

Immédiatement, le ministère de la Défense sud-coréen a rappelé à ses troupes les règles élémentaires de la sécurité informatique : utiliser un antivirus à jour ainsi qu'un pare-feu, et ne pas ouvrir n'importe quel contenu. Dans ce pays où 70 % de la population utilise Internet (contre 40 % en Europe et 12 % en Chine), cette affaire a créé un scandale et personne ne veut en rester là.

Toutes les structures étatiques du monde sont soumises à des attaques informatiques. Parfois l'intensité ou la gravité de celles-ci permettent de parler d'acte de guerre informatique. Par exemple, la mise à genoux de l'infrastructure informatique estonienne en 2001 a été considérée comme le premier événement du genre. Les attaques répétées menées contre des soldats de l'armée sud-coréenne ces derniers jours sont inquiétantes mais ne peuvent pour l'heure être qualifiées d'actes de guerre. Cela ressemble plutôt à un entraînement.

# Les gains financiers

- Pour les pirates qui contrôlent un ordinateur
  - Utilisation de numéros de cartes de crédit
  - Chantage, extorsion de fonds, espionnage industriel, etc.
  - Spéculation en bourse : *pump and dump scams* (*spam*, VoIP), exemple : [www.investopedia.com/ask/answers/05/061205.asp](http://www.investopedia.com/ask/answers/05/061205.asp)
  - Connexion à des lignes téléphoniques payantes
  - Accès à des comptes (banques, retraites, paypal, e-Bay, FAI, opérateurs téléphoniques, hotspots, etc.)
  - Vente d'adresses e-mails : exemple 28 000 dollars pour 92 *m<sup>ns</sup>* d'adresses mail (AOL)
  - Services payants (exemples : porno, films piratés, etc.) + spammers, etc.
  - *Click fraud* (relais de publicité) : exemple : 100 mille dollars avec 400.000 zombies
  - Location de botnets, spammers, etc.

# Les principales failles exploitables

[cwe.mitre.org/top25](https://cwe.mitre.org/top25)

- API : Débordement de buffers, heap, stack (exemple : return-to-libc attacks), entiers, etc.
- API : pas de contrôle de type ou vérification de format insuffisante : SQL injection, PHP, etc.
- Utilisation non-prévue → Fuzzing
- Race conditions
- Contrôle d'origine insuffisant : *cross-scripting*, applets, plug-ins, extensions, certificats, etc.

# Sommaire

Un peu d'histoire

Les propriétés de la sécurité

Les attaques

**Les défenses**

La protection des systèmes informatiques

# Sommaire

## Les défenses

Cryptographie

Prévention et élimination des vulnérabilités

Cloisonnement

Audit

Détection d'intrusions

# Différents volets de la sécurité

- Sécurité physique  
Protection des locaux contre incendie, inondation, etc.  
Contrôle des accès physiques
- Sécurité du personnel (pas la CHS)  
Règles liées aux conditions de travail pour les personnels internes (employés, intérimaires, stagiaires, etc.) et externes (visiteurs, maintenance, sous-traitants, etc.), y compris dans des circonstances particulières : embauche, départ, grève, etc.

## Différents volets de la sécurité

- Sécurité procédurale  
Procédures pour la gestion du SIC : enregistrement (et effacement) des utilisateurs, sauvegardes, maintenance, installation et mises à jour de matériels et de logiciels, etc.
- Sécurité technique  
C'est tout ce qu'on va voir maintenant.

# Sommaire

## Les défenses

Cryptographie

Prévention et élimination des vulnérabilités

Cloisonnement

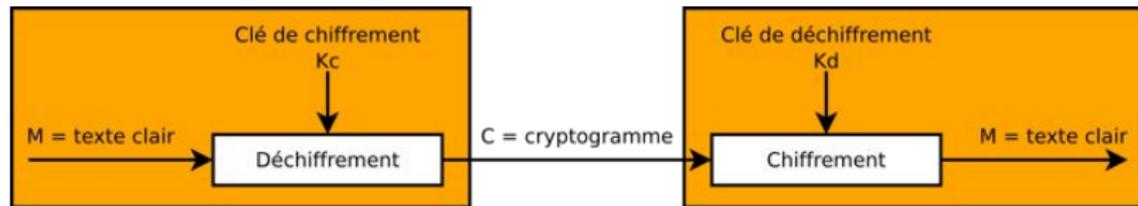
Audit

Détection d'intrusions

# Terminologie

- Cryptologie = cryptographie + cryptanalyse
  - Cryptographie, du grec κρυπτος (caché) et γραθειν (écrire)  
Ecrire des messages incompréhensibles par des tiers
  - Cryptanalyse  
Découvrir le(s) secret(s), décrypter
- A ne pas confondre avec stéganographie
  - Du grec στεγανο (dissimuler)
  - Encre sympathique
  - Filigranes (tatouages)
- Chiffre, chiffrement (pas chiffrage ni cryptage), déchiffrement, clair, cryptogramme

# Chiffrement (confidentialité)



- Notation

Chiffrement :  $C = \{M\}_{Kc}$       Déchiffrement :  $M = [C]_{Kd}$

- Confidentialité

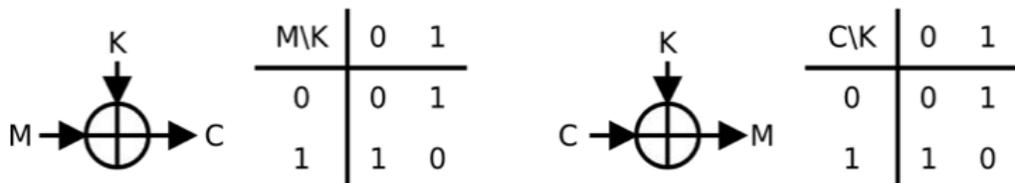
- Sans connaître  $Kd$ , il doit être “impossible” de retrouver  $M$
- Il doit être “impossible” de trouver  $Kd$ , même en connaissant  $C$  et  $M$  (attaque par *clair connu*)
- Il doit être “impossible” de trouver  $Kd$ , même en choisissant  $M$  (attaque par *clair choisi*)

## Chiffrement symétriques : $K_c = K_d$

- Tous les chiffres connus jusqu'en 1976 !
- Exemples
  - DES (1976)
    - Clés de 56 bits (plus 8 bits de parité pour tester l'intégrité de la clé)
    - Blocs de 64 bits
  - AES (2000)
    - Clés de 128, 192, 256 bits
    - Blocs de 128 bits

# One-time pad

- La clé est une chaîne de bits aléatoire aussi longue que le message, algorithmique : ou-exclusif



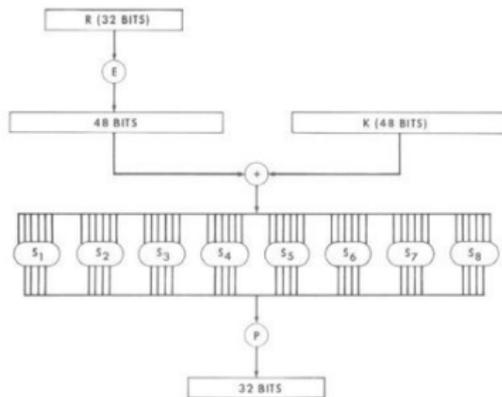
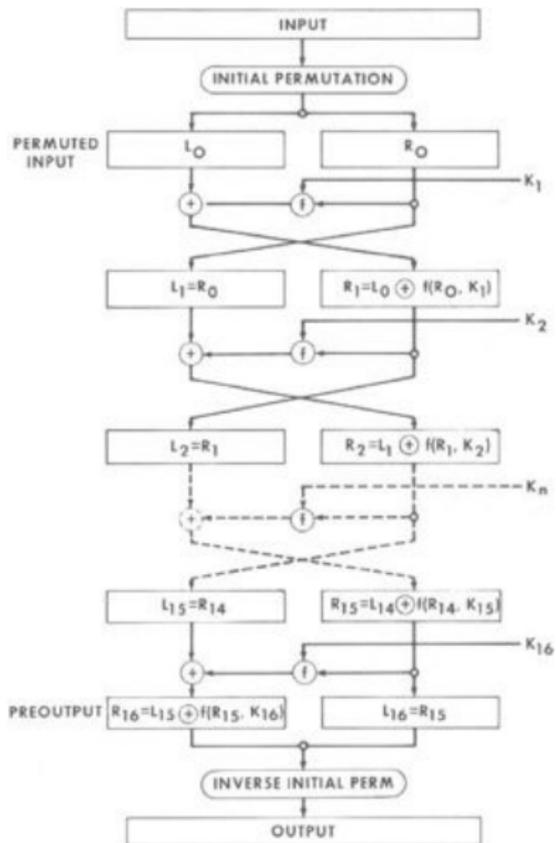
- D'après la théorie de l'information (Shannon), c'est un chiffre incassable (si la clé n'est pas réutilisée !)

# DES, Data Encryption Standard

[csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf](http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf)

- ① Diversification de la clé  $\rightarrow$  16 sous clés  $K_{1..16}$  de 48 bits  
*Chaque  $K_i$  est composé de 48 bits de  $K$  pris dans un certain précis*
- ② Fractionnement du texte en blocs  $B_{1..n}$  de 64 bits
- ③ Pour chaque bloc  $B_j$ 
  - ① Permutation initiale du bloc  $B_j$
  - ② Découpage du bloc  $B_j$  en parties gauche  $G_0$  et droite  $D_0$
  - ③ Pour chaque sous clé,  $K_i$ 
    - ①  $G_i = D_{i-1}$
    - ②  $D_i = G_{i-1} \oplus f(D_{i-1}, K_i)$
  - ④ Reconstitution du bloc  $B'_j$  à partir de  $G_{16}$  et  $D_{16}$
  - ⑤ Permutation initiale inverse du bloc  $B'_j$

# DES, Data Encryption Standard



Fonction  $f(R, K)$

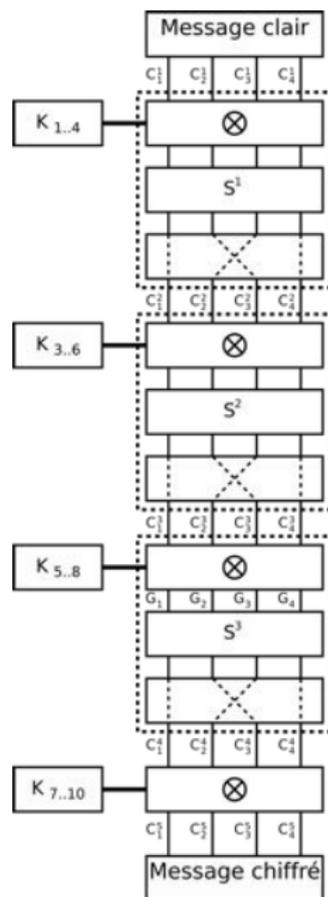
IP

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Permutation initiale

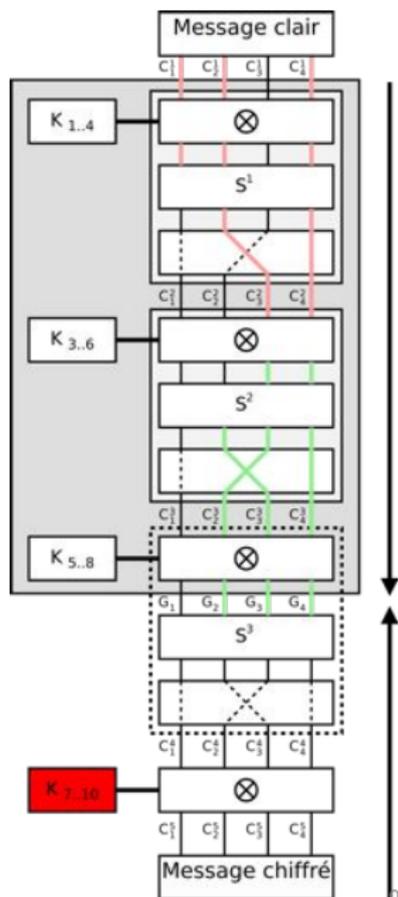
# Cryptanalyse linéaire

- Attaque à clairs connus
  - $\mathcal{T} = \{(\text{message}_i, \text{ciphertext}_i)\}$
- L'attaquant dispose de l'algorithme et recherche la clé
- Exploitation d'un manque de non linéarité pour établir des approximations
  - ⇒ Réduction de l'espace de recherche



# Cryptanalyse linéaire – démarche

- Approximation d'une partie de l'algorithme
    - Indépendante de la clé, pour réduire l'espace de recherche
    - Dépendante des messages clairs et de l'entrée du dernier bloc (tous les *rounds* sauf le dernier)
  - Attaque par force brute sur la partie non approximée de l'algorithme
- ⇒ Trouver une partie de la clé  $K$ , correspondant au dernier *round*

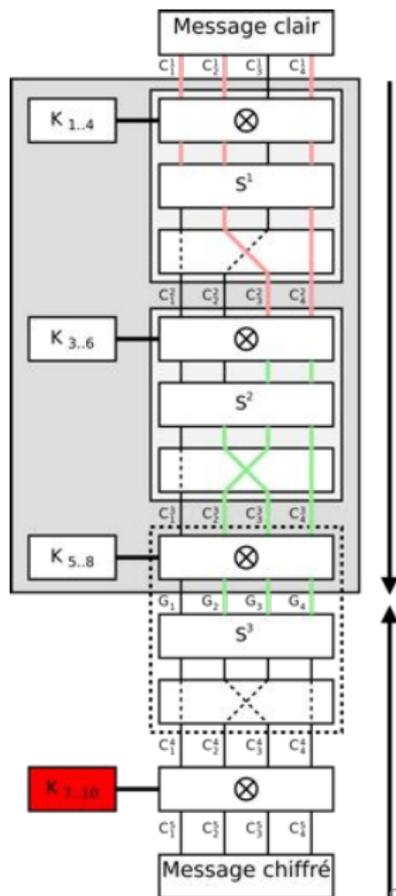


# Cryptanalyse linéaire – algorithme

## Version simplifiée

```

for candidat = 0 ..  $2^4 - 1$  do
  scorecandidat ← 0
  for all (message, ciphertext) ∈  $\mathcal{T}$  do
    valeur ← ciphertext ⊕ candidat
    valeur ← (Permute3)-1(valeur)
    valeur ← (S3)-1(valeur)
    Test si l'approximation est vérifiée par
    le couple valeur, message
    if Approximation_Avérée(valeur, message) then
      scorecandidat ← scorecandidat + 1
    end if
  end for
end for
résultat ← argmaxx abs(scorex - | $\mathcal{T}$ |/2)
  
```



# Cryptanalyse linéaire

- Approximation linéaire des premiers *rounds*

$$\left(\bigoplus_{i=1}^4 a_i \oplus C_i^1\right) \oplus \left(\bigoplus_{i=1}^4 b_i \oplus C_i^4\right) \oplus \text{Constante} = 0$$

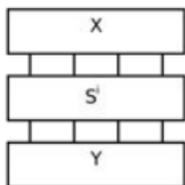
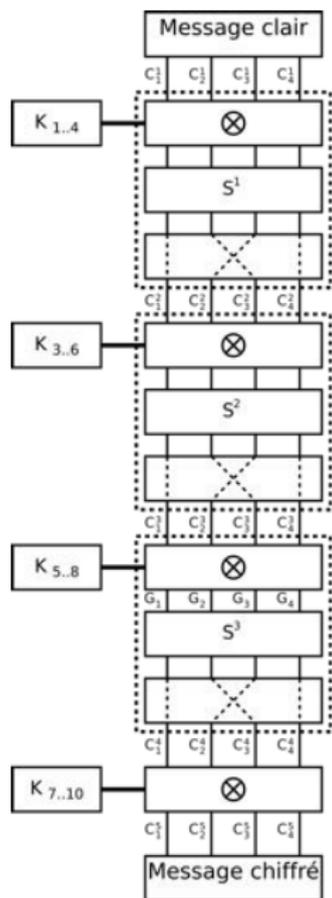
$$\mathcal{P}\left[\left(\bigoplus_{i=1}^4 a_i \oplus C_i^1\right) \oplus \left(\bigoplus_{i=1}^4 b_i \oplus C_i^4\right) \oplus \text{Constante} = 0\right] = 1/2 + \epsilon$$

Biais  $\epsilon$ ,  $a_i$  et  $b_i$  sont des masques

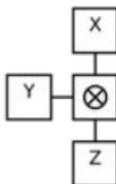
# Cryptanalyse linéaire

- Approximation linéaire des premiers *rounds*
  - Dernier *round* non approximé : toute ou partie de la clé correspondante va être recherchée
  - Pour chaque clé candidate au dernier *round*, il est possible de calculer  $C_i^4$  correspondant
  - La clé vraisemblablement utilisée pour chiffrer les message; en ciphertext; est celle pour laquelle l'approximation est la meilleure ...

# Cryptanalyse linéaire – exemple



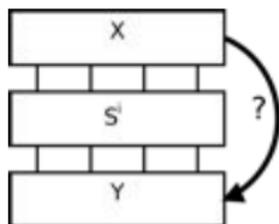
XXXX	YYYY
0000	0011
0001	1000
0010	1001
0011	0101
0100	1011
0101	0001
0110	0010
0111	0100
1000	0000
1001	1111
1010	0111
1011	0110
1100	1110
1101	1010
1110	1100
1111	1101



XY	Z
00	0
01	1
10	1
11	0

- Plusieurs *rounds*
- Fonction non linéaire  $S^i$ , ne peut pas être exprimée sous la forme de xor

# Cryptanalyse linéaire – exemple, table des approximations



$(a_1, a_2, a_3, a_4)$

$$N((a_1, a_2, a_3, a_4), (b_1, b_2, b_3, b_4)) = \{ (X_1, X_2, X_3, X_4), (Y_1, Y_2, Y_3, Y_4) / \bigoplus_{i=1}^4 a_i X_i \oplus \bigoplus_{i=1}^4 b_i Y_i = 0 \}$$

	$(b_1, b_2, b_3, b_4)$															
	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0000	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001	0	0	-2	2	2	-2	0	0	0	0	-2	2	2	6	0	0
0010	0	0	-2	2	4	0	2	2	-2	-2	4	0	2	-2	0	0
0011	0	0	0	0	-2	-2	-2	6	2	2	2	2	0	0	0	0
0100	0	-2	0	2	0	2	4	2	2	0	-2	0	-2	0	-2	4
0101	0	-2	2	0	2	0	0	-2	2	0	0	6	0	-2	2	0
0110	0	2	2	4	0	2	-2	0	0	2	-2	0	4	-2	-2	0
0111	0	2	0	-2	2	0	-2	0	4	-2	0	-2	2	0	2	4
1000	0	-2	2	0	4	-2	-2	0	2	0	0	-2	-2	0	-4	-2
1001	0	-2	-4	-2	2	0	-2	0	-2	4	-2	0	0	-2	0	2
1010	0	-2	0	2	0	-2	0	2	0	-2	-4	-2	0	-2	4	-2
1011	0	-2	-2	4	-2	0	-4	-2	0	-2	2	0	-2	0	0	2
1100	0	0	2	-2	0	0	-2	2	-4	-4	-2	2	0	0	-2	2
1101	0	0	0	0	-2	6	2	-2	0	0	0	0	2	-2	-2	2
1110	0	4	-4	0	0	0	0	0	2	-2	-2	2	-2	-2	-2	-2
1111	0	-4	-2	-2	-2	2	0	0	2	-2	0	0	4	0	-2	-2

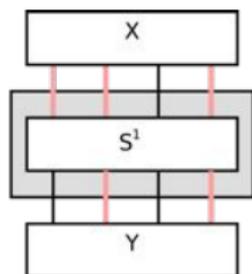
XXXX	YYYY	XXXXY	⊕
0000	0011	00001	1
0001	1000	00100	1
0010	1001	00001	1
0011	0101	00111	1
0100	1011	01001	0
0101	0001	01101	1
0110	0010	01000	1
0111	0100	01110	1
1000	0000	10000	1
1001	1111	10111	0
1010	0111	10011	1
1011	0110	10110	1
1100	1110	11010	1
1101	1010	11100	1
1110	1100	11010	1
1111	1101	11111	1

XXXX	YYYY	XXXXY	⊕
0000	0011	00011	0
0001	1000	01000	1
0010	1001	10001	0
0011	0101	11101	0
0100	1011	00011	0
0101	0001	01001	0
0110	0010	10010	0
0111	0100	11100	1
1000	0000	00000	0
1001	1111	01111	0
1010	0111	10111	0
1011	0110	11110	0
1100	1110	00110	0
1101	1010	01010	0
1110	1100	10100	0
1111	1101	11101	0

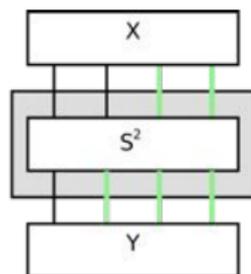
XXXX	YYYY	XXXXY	⊕
0000	0011	00011	0
0001	1000	01100	0
0010	1001	00101	0
0011	0101	01001	0
0100	1011	10111	0
0101	0001	11001	1
0110	0010	10010	0
0111	0100	11000	0
1000	0000	00000	0
1001	1111	01111	0
1010	0111	00011	0
1011	0110	01010	0
1100	1110	10110	1
1101	1010	11110	0
1110	1100	10100	0
1111	1101	11101	0

XXXX	YYYY	XXXXY	⊕
0000	0011	00011	0
0001	1000	11001	0
0010	1001	01011	0
0011	0101	10111	0
0100	1011	01011	0
0101	0001	10011	1
0110	0010	00000	0
0111	0100	10101	0
1000	0000	00000	0
1001	1111	11111	0
1010	0111	00111	0
1011	0110	10101	0
1100	1110	01101	1
1101	1010	11001	0
1110	1100	01101	0
1111	1101	11111	0

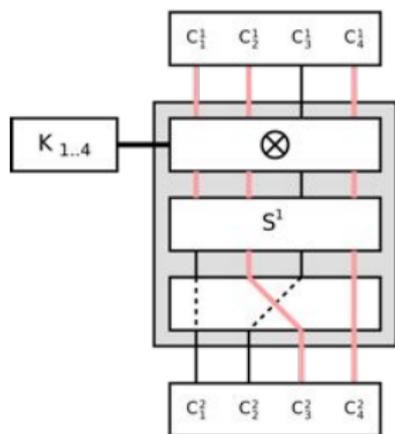
# Cryptanalyse linéaire – exemple, approximation



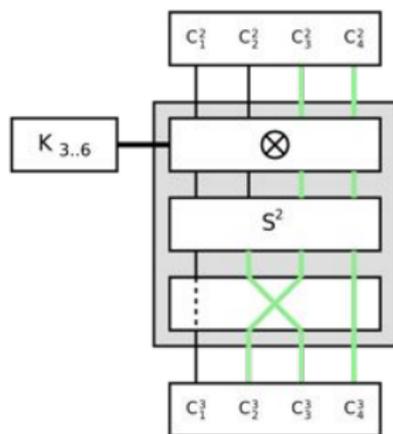
$$X_1 \oplus X_2 \oplus X_4 \oplus Y_2 \oplus Y_4 = 0$$



$$X_3 \oplus X_4 \oplus Y_2 \oplus Y_3 \oplus Y_4 = 0$$

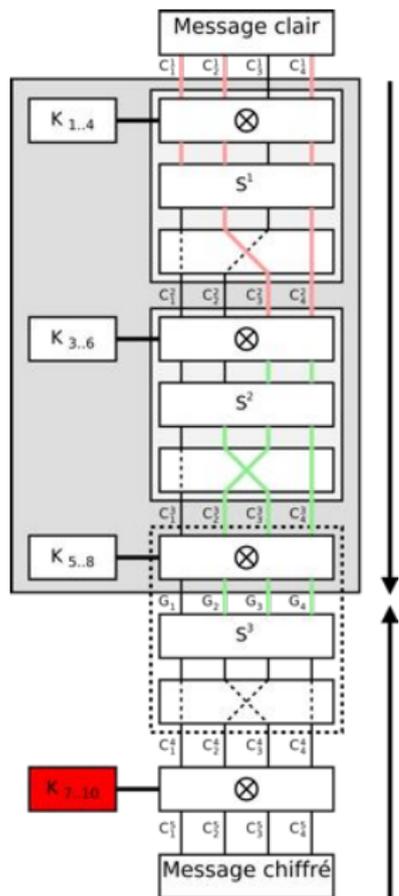


$$C_1^1 \oplus C_2^1 \oplus C_4^1 \oplus C_3^2 \oplus C_1^2 \oplus K_1 \oplus K_2 \oplus K_4 = 0$$



$$C_3^2 \oplus C_4^2 \oplus C_2^3 \oplus C_3^3 \oplus C_1^3 \oplus K_3 \oplus K_4 = 0$$

# Cryptanalyse linéaire – exemple, algorithme

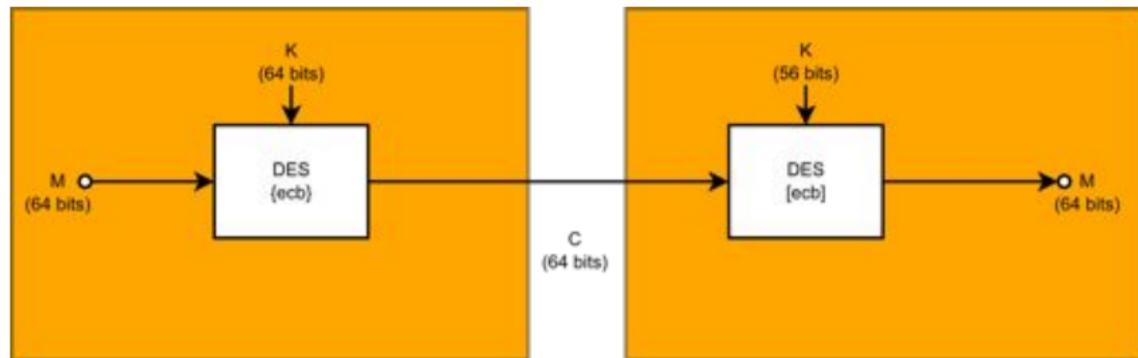


$$\begin{aligned}
 c_1^1 \oplus c_2^1 \oplus c_3^1 \oplus c_4^1 \oplus K_1 \oplus K_2 \oplus K_4 &= 0 \\
 c_3^2 \oplus c_4^2 \oplus c_3^3 \oplus c_4^3 \oplus K_3 \oplus K_4 &= 0 \\
 c_1^1 \oplus c_2^1 \oplus c_4^1 \oplus c_3^3 \oplus c_4^4 \oplus K_1 \oplus K_2 \oplus K_4 \\
 \oplus c_3^3 \oplus c_4^4 \oplus K_3 \oplus K_4 &= 0 \\
 c_2^2 \oplus c_3^3 \oplus c_4^4 \oplus G_2 \oplus G_3 \oplus G_4 \oplus K_6 \oplus K_7 \oplus K_8 &= 0 \\
 c_1^1 \oplus c_2^1 \oplus c_4^1 \oplus G_2 \oplus G_3 \oplus G_4 \\
 \oplus K_3 \oplus K_4 \oplus K_1 \oplus K_2 \oplus K_4 \oplus K_6 \oplus K_7 \oplus K_8 &= 0 \\
 \text{Constante} \\
 c_1^1 \oplus c_2^1 \oplus c_4^1 \oplus G_2 \oplus G_3 \oplus G_4 &= 0
 \end{aligned}$$

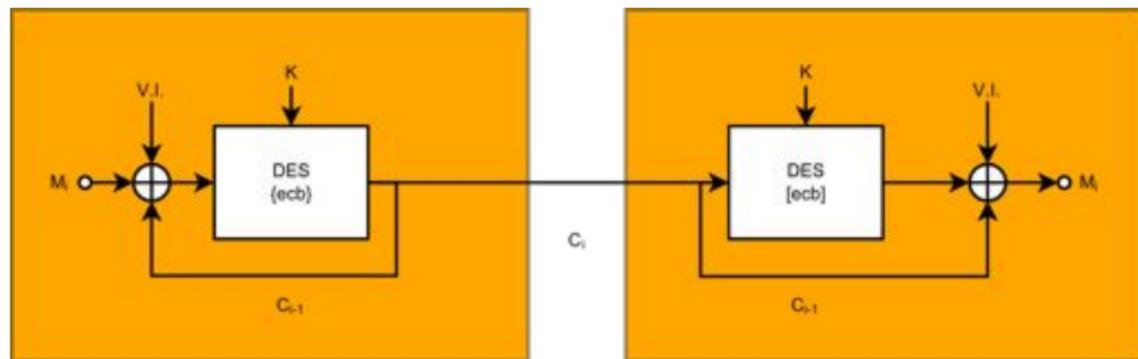
```

for k = 0 .. 2^4 - 1 do
  s_k ← 0
  for all (m, c) ∈ T do
    g ← c ⊕ k
    g ← (Permute3)-1(g)
    g ← (S3)-1(g)
    if g2 ⊕ g3 ⊕ g4 ⊕ m1 ⊕ m2 ⊕ m4 = 0 then
      s_k ← s_k + 1
    end if
  end for
end for
r ← argmax_x abs(s_x - |T|/2)
    
```

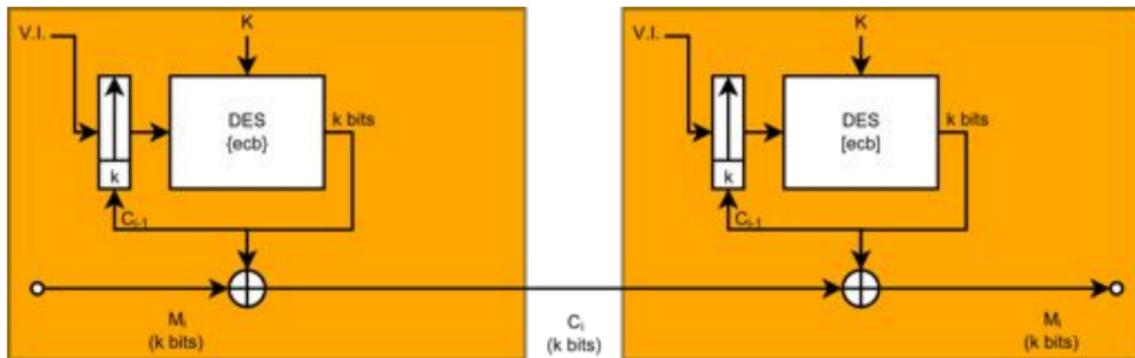
# DES, mode ECB (Electronic-Code-Book)



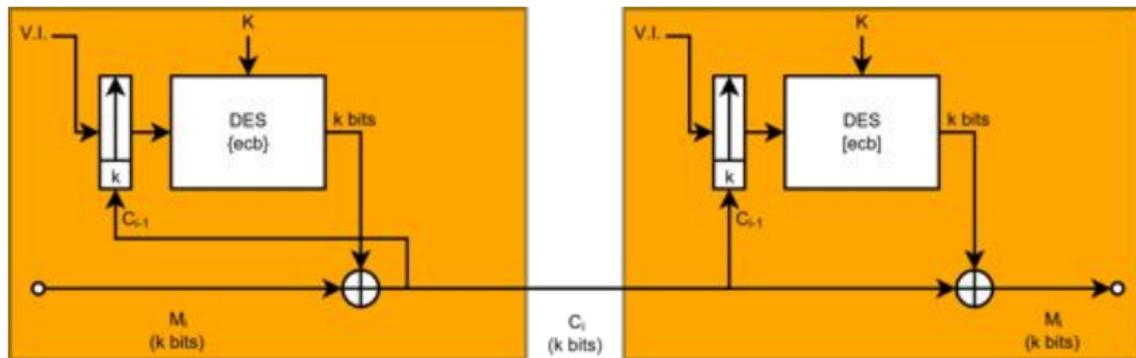
# DES, mode CBC (Cipher-Block-Chaining)



# DES, mode OFB (Output-Feedback-Block)

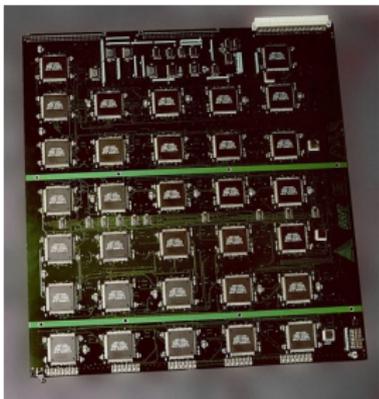


# DES, mode CFB (Cipher-Feedback-Block)



## DES, cryptanalyse

- Clé sur 56 bits  $\Rightarrow 2^{56}$  clés possibles
  - Possibilité d'attaques par brute force
    - *Deep Crack* – Cryptography Research, Advanced Wireless Technologie, EFF
    - 29 cartes de 64 puces (1 856 puces spécialisées pour le DES)
    - 90  $m^{ds}$  de clés testées par seconde
- $\Rightarrow$  Environ 5 jours pour tester toutes les possibilités



# Avantages des chiffres symétriques

- Rapides
  - Exemple avec AES
    - Jusqu'à 100 Gb/s sur du matériel spécifique
    - Jusqu'à 250 Mb/s avec du logiciel (MavBook Pro)
- Clés *courtes* : typiquement 80 bits pour résister aux attaques *brutales* (aujourd'hui)  
[www.rsa.com/rsalabs/node.asp?id=2103](http://www.rsa.com/rsalabs/node.asp?id=2103)
  - DES (ECB) cassé en octobre 1997 (22h avec un matériel spécifique)
  - RC5-56 cassé en octobre 1997 (250j sur Internet)
  - RC5-64 cassé en juillet 2002 (1757j sur Internet)
- Pratiques pour chiffrer des fichiers personnels  
→ pas de clé à partager

# Problèmes des chiffres symétriques

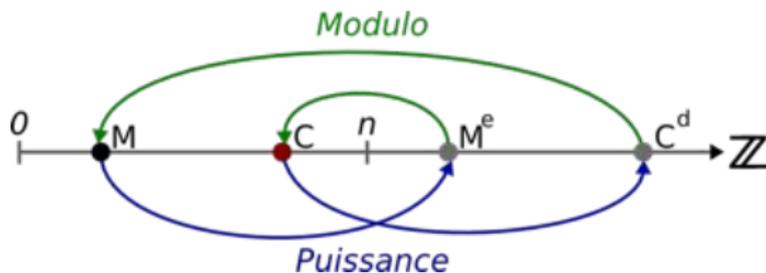
- Communications : clé secrète partagée  
Il faut que l'émetteur et le récepteur se fassent confiance, et gardent soigneusement la clé secrète
- Comment distribuer/renouveler la clé ?
  - Chiffrer la nouvelle clé de session avec l'ancienne
  - Chiffrer la clé de session avec une clé spécifique de chaque matériel  
⇒ site de confiance (répertoire)
  - Cryptographie quantique
  - Utiliser un système à clé publique (Diffie-Hellman)

## Chiffres à clé publique : $K_c \neq K_d$

- Connaissant  $K_c$ , il est “impossible” de trouver  $K_d$ 
  - $K_d$  est “privé” : seul celui qui connaît  $K_d$  peut déchiffrer
  - $K_c$  est public : tout le monde peut chiffrer → répertoire de clés publiques
- Exemples
  - RSA (1978) → difficulté de factoriser de grands nombres[RSA78]
  - El Gamal (1985) → difficulté de calcul des logarithmes discrets[EG85]

# RSA – Rivest, Shamir, Adleman

- Création des clés
  - Choisir  $p$  et  $q$  deux nombres premiers distincts
    - ⇒ Calculer le *module de chiffrement*  $n$ ,  $n = p \cdot q$
    - ⇒ Calculer l'*indicatrice d'Euler* de  $n$ ,  $\phi(n) = (p - 1) \cdot (q - 1)$
  - Choisir l'*exposant de chiffrement*  $e$ , un entier premier avec  $\phi(n)$ ,
    - ⇒ Calculer l'*exposant de déchiffrement*  $d$ ,  $e \cdot d \equiv 1 \pmod{\phi(n)}$
- *Algorithme d'Euclide étendu*
  - ⇒  $Kc = \{e, n\}$      $Kd = \{d, n\}$
- Chiffrement :  $C = M^e \pmod n$ , avec  $M < n$
- Déchiffrement :  $M = C^d \pmod n$
- Décomposition de  $n$  en produit de facteurs premiers  $\rightarrow p$  et  $q$   $\mathcal{O}(e^n)$



# RSA – Rivest, Shamir, Adleman

- Exemple

- Création des clés

- $p = 5, q = 11$

- ⇒  $n = p \cdot q = 5 \cdot 11 = 55$

- ⇒  $\phi(n) = (p - 1) \cdot (q - 1) = (5 - 1) \cdot (11 - 1) = 4 \cdot 10 = 40$

- $e = 3$

- ⇒  $d = 27$  Vérification :  $d \cdot e \stackrel{?}{\equiv} 1 \pmod{\phi(n)}$

$$d \cdot e = 3 \cdot 27 = 81 = 2 \cdot 40 + 1 \equiv 1 \pmod{40}$$

- ⇒  $Kc = \{n, e\} = \{55, 3\}$      $Kd = \{n, d\} = \{55, 27\}$

- Chiffrement de  $M = 19$

- $C = M^e \pmod{n} = 19^3 \pmod{55} = 6859 \pmod{55} = 39$

- Déchiffrement de  $C = 39$

- $M = C^d \pmod{n} = 39^{27} \pmod{55} = 39$

- $39^{27} = 9093778876146525519753713411306280250639479$

## Avantages des chiffres à clé publique

- Pas de confiance mutuelle entre émetteur et récepteur
- Gestion de clé “facile”
  - Répertoire public de clés publiques ou distribution entre pairs
  - La clé privée ne doit “jamais” être transmise
- Possibilité d'utilisations nouvelles : distribution de clés symétriques, signatures, certificats, etc.

# Distribution de clés symétriques

- Exemple : Alice génère aléatoirement une clé de session  $K$  (symétrique) et la chiffre avec la clé publique de Bob
- Exemple : Diffi-Hellmann
  - Alice (A) et Bob (B) souhaitent communiquer
 

A	↔	B	Alice et Bob se mettent d'accord sur un nombre premier $p$
A	↔	B	Alice et Bob conviennent d'une racine primitive $g$
A		B	Alice choisit un nombre secret $0 \leq a \leq p - 1$
A	→	B	Alice envoie la valeur $g^a \bmod p$ à Bob
A		B	Bob choisit un nombre secret $0 \leq b \leq p - 1$
A	←	B	Bob envoie la valeur $g^b \bmod p$ à Alice
A		B	Alice calcule la clé secrète $K = (g^b \bmod p)^a \bmod p$
A		B	Bob calcule la clé secrète $K = (g^a \bmod p)^b \bmod p$
  - Eve écoute les transmissions
    - Eve connaît  $p, g, g^a \bmod p, g^b \bmod p$
    - Peut-il calculer  $a$  et  $b$  ?
    - $a = \log_g(g^a)$  et  $b = \log_g(g^b) \pmod p$
    - Problème du logarithme discret

# Problèmes des chiffres à clé publique

- Calculs complexes : lents ( $\sim 1$  Mbits/s), clé longue (1024 ou 2048 bits), sauf avec des courbes elliptiques ( $\sim 160$  bits)

## Records actuels

- RSA 200, 200 chiffres (2005) : 663 bits (BSI, U.Bonn, CWI)
  - RSA 640/173 (2005) : 4,5 mois à 80 opteron 2,2 GHz (BSI, U.Bonn)
  - Logarithme discret 613 bits (2005) : 17 jours à 64 Itanium2 (Bull, U. Versailles)
  - Certicom ECC2-109 (2004) : 15 mois à 2900 calculateurs
- Problèmes spécifiques
    - Intégrité des répertoires de clés publiques
    - Durée de vie des clés
    - Révocation
    - Nécessité de partager des clés privées ?
    - Limitation des algorithmes, par exemple : chiffrer un petit  $M$  par RSA

# Factorisation – Défis

[www.rsa.com/rsalabs/node.asp?id=2092](http://www.rsa.com/rsalabs/node.asp?id=2092)

[www.crypto-world.com/FactorWorld.html](http://www.crypto-world.com/FactorWorld.html)

Nombre	Nombre de décimales	Date	Vitesse	Algorithme
C116	116	1990	275 MIPS années	mpqs
RSA-120	120	06/1993	830 MIPS années	mpqs
RSA-129	129	04/1994	5000 MIPS années	mpqs
RSA-130	130	04/1996	1000 MIPS années	gnfs
RSA-140	140	02/1999	2000 MIPS années	gnfs
RSA-155	155	08/1999	8000 MIPS années	gnfs
C158	158	01/2002	3,4 Pentium 1GHz années	gnfs
RSA-160	160	03/2003	2,7 Pentium 1GHz années	gnfs
RSA-576	174	12/2003	13,2 Pentium 1GHz années	gnfs
C176	176	05/2005	48,6 Pentium 1GHz années	gnfs
RSA-200	200	05/2005	121 Pentium 1GHz années 55 Opteron 2,2GHz années	gnfs
RSA-640	193	11/2005	30 Opteron 2,2GHz années	gnfs

## Factorisation – Défis *RSA-640*

- Durée : 4,5 mois
- Matérielle : 80 Opterons 2,2GHz
- Vitesse : 30 Opteron 2,2GHz années
- Le nombre : 193 chiffres – 640 bits
- La factorisation

3107418240490043721350750035888567930037346022842  
 7275457201619488232064405180815045563468296717232  
 8678243791627283803341547107310850191954852900733  
 7724822783525742386454014691736602477652346609

=

1634733645809253848443133883865090859841783670033  
 092312181110852389333100104508151212118167511579

x

1900871281664822113126851573935413975471896789968  
 515493666638539088027103802104498957191261465571

# Evolution prévisible ?

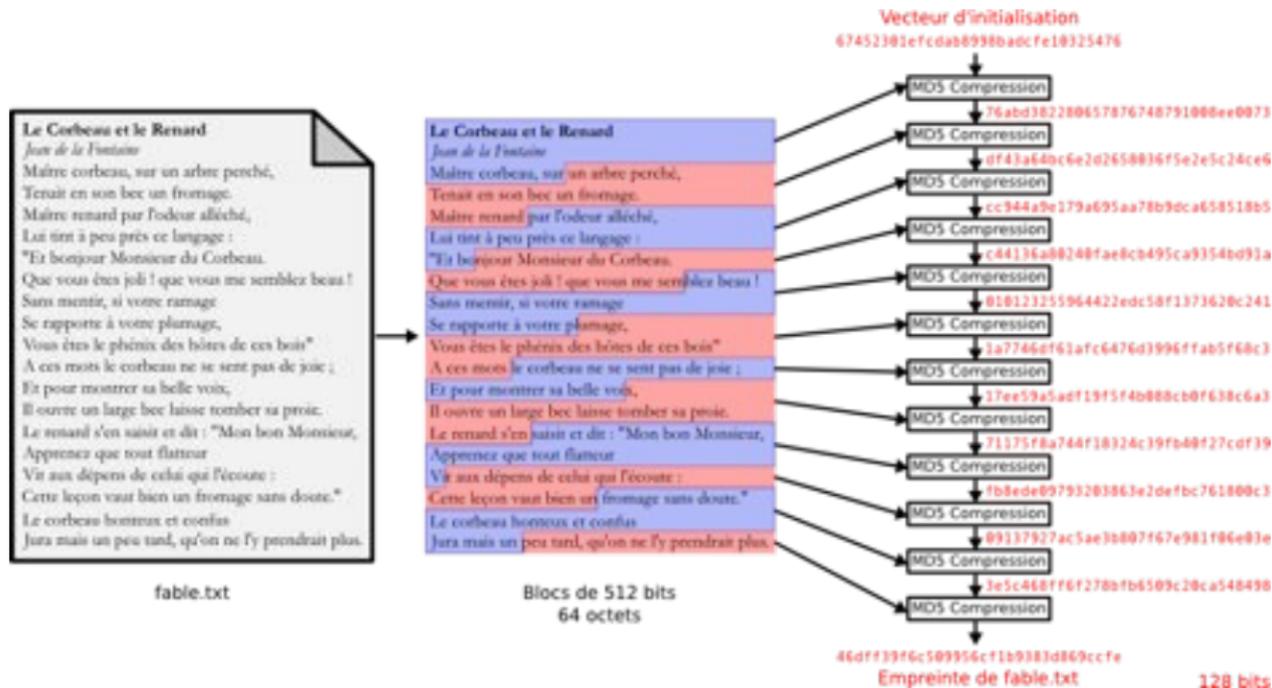
TODO

# Fonctions de hachage → empreinte, condensat

- *One-way hash function*  $\mathcal{H}$ 
  - L'empreinte  $\mathcal{H}(M)$  est de taille fixe  $n$ , quelque soit la longueur de  $M$
  - Si 1 bit de  $M$  est changé, environ  $n/2$  bits de  $\mathcal{H}(M)$  changent
  - Connaissant  $M$ , il est **facile** de calculer  $\mathcal{H}(M)$
  - **Collisions** :  $M \neq M', \mathcal{H}(M) = \mathcal{H}(M')$   
 $|M|$  non borné et  $|\mathcal{H}(M)|$  fixe  $\Rightarrow \exists$  un nombre infini de collisions
  - **Sécurité** : sauf attaque par force brute ( $\sim 2^n$  essais)
    - Connaissant  $x < 2^n$ , il est **"impossible"** de trouver  $M$  tel que  $\mathcal{H}(M) = x$
    - Connaissant  $M$ , il est **"impossible"** de trouver  $M'$  tel que  $M \neq M'$  et  $\mathcal{H}(M) = \mathcal{H}(M')$
    - Il est **très difficile** ( $\sim 2^{n/2}$  essais) de trouver  $M$  et  $M'$  tel que  $M \neq M'$  et  $\mathcal{H}(M) = \mathcal{H}(M')$
- Exemples : *DES – CBC* (64 bits), *MD5* (128 bits), *SHA-1* (160 bits)

## MD5

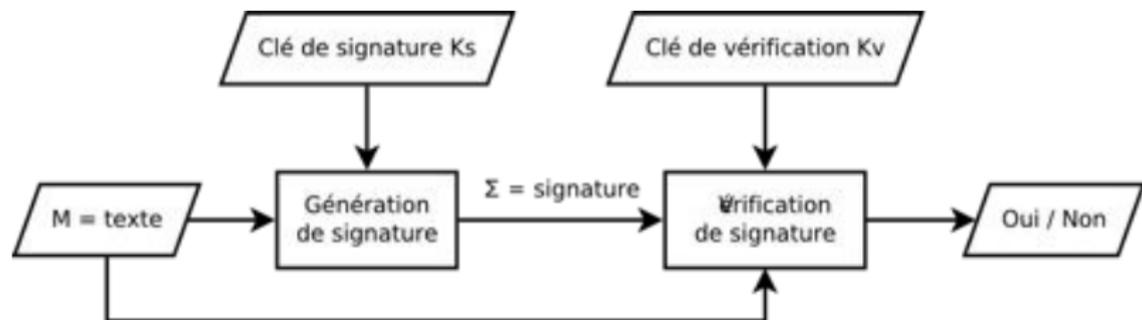
- $M$  est découpé en  $z$  blocs de 512 bits,  $m_1, m_2, \dots, m_z$   
 $h_1 = \mathcal{F}(\text{constante}, m_1)$ ;  $h_2 = \mathcal{F}(h_1, m_2)$ ; ...;  $h_z = \mathcal{F}(h_{z-1}, m_z) = \mathcal{H}(M)$



# Fonctions de hachage – Application : Intégrité

- Communications : contre **interception et modification**  
Transmettre le message et l’empreinte par des canaux indépendants
- Fichiers : détection de modification (*Tripwire*[KS94])
  - Sur une machine correcte, calculer les empreintes des fichiers stables (OS, programmes, configuration, etc.) et les stocker de manière sûre (par exemple, chiffrées)
  - Périodiquement, ou en cas de doute, ou au démarrage, recalculer les empreintes et les comparer (sur une machine saine)
- **Faiblesse découverte récemment (MD5, etc.)**
  - [www.stachliu.com/research\\_collisions.html](http://www.stachliu.com/research_collisions.html)
  - 2006 : collision MD5 en 3/4 d’heure sur Pentium 4 1,6GHz ( $\sim 50$  bits)
  - 2009 : collision SHA-1 en  $\sim 2^{52}$  essais (théoriquement)
- RIPEMD (128 bits, 160 bits, 256 bits), SHA-256, SHA-512, Whirlpool (512 bits)

# Signature (intégrité)



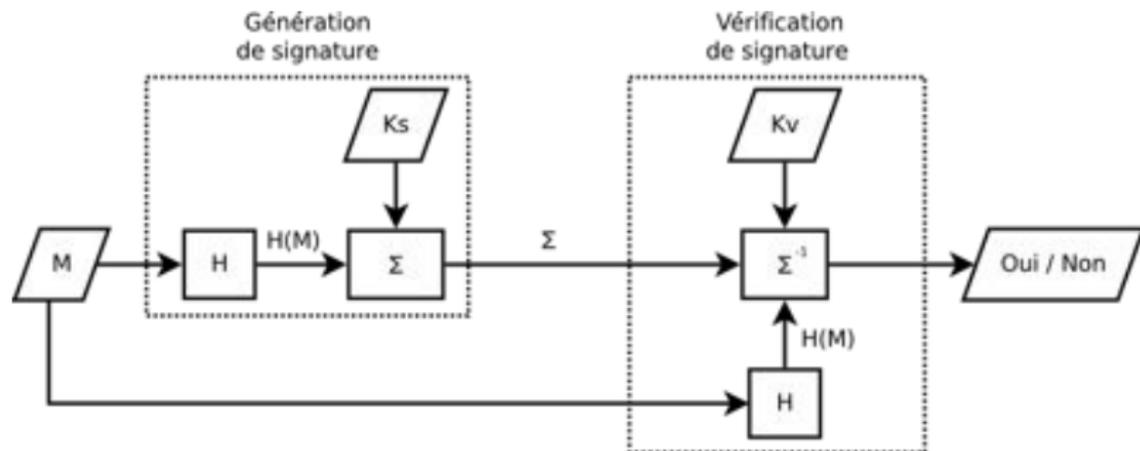
- $K_s$  = clé de signature     $K_v$  = clé de vérification
- Intégrité
  - Sans connaître  $K_s$ , "impossible" de générer une signature valide
  - Il est "impossible" de trouver  $K_s$ , connaissant  $M$  et  $\Sigma$  (clair connu)
  - Il est "impossible" de trouver  $K_s$ , choisissant  $M$  (clair choisi)
- Pratique :  $\Sigma$  est de taille fixe et relativement petit, quelque soit la taille de  $M$

# Signature symétriques – $K_s = K_v$ : secrètes !

- *MAC : Message Authentication Code*
- Exemples
  - *CBC-MAC* : Dernier bloc du *DES* en mode *CBC*
  - $\Sigma = \{\mathcal{H}(M)\}_{K_s} \stackrel{?}{=} \Sigma' = \{\mathcal{H}(M)\}_{K_v}$
  - *H-based MAC* :  $\Sigma = \mathcal{H}(K_s \cdot M) \stackrel{?}{\rightarrow} \Sigma' = \mathcal{H}(K_v \cdot M)$  **Faiblesse !**
  - Variante :  $\Sigma = \mathcal{H}(K_s \cdot \mathcal{H}(K_s \cdot M)) \stackrel{?}{\rightarrow} \Sigma' = \mathcal{H}(K_s \cdot \mathcal{H}(K_s \cdot M))$
- Inconvénients
  - Signataire et vérificateur doivent se faire confiance
  - Répudiation possible  $\Rightarrow$  la signature n'est pas valable devant un juge

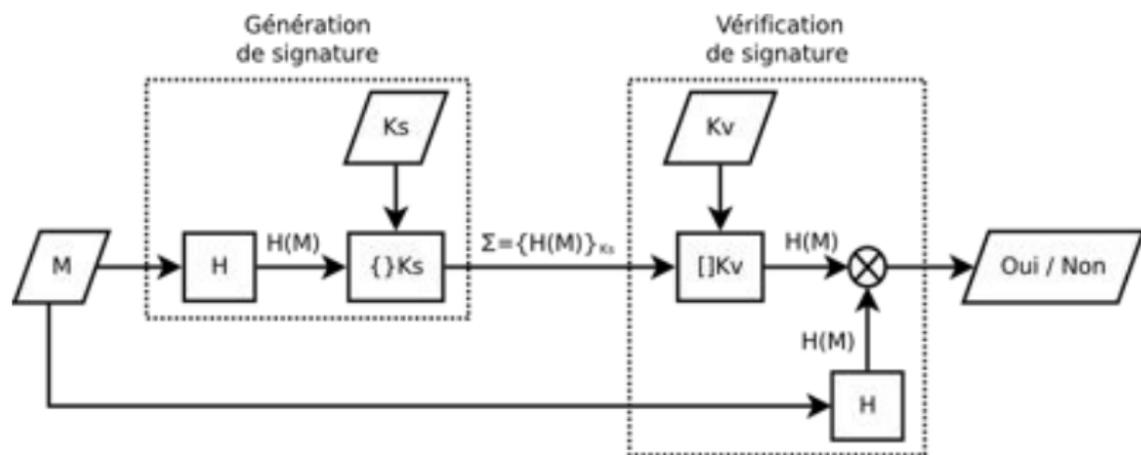
# Signatures à clé publique – $K_s \neq K_v$

- Exemple : *DSA*
  - Fonction de hachage : *SHA-1*
  - Signature / Vérification : ElGamal



# Signatures à clé publique – $K_s \neq K_v$

- Exemple : *RSA*
  - $K_s$  = clé de signature = clé de chiffrement  $K_c$  privée
  - $K_v$  = clé de vérification = clé de déchiffrement  $K_d$  publique



## Propriétés des signatures à clé publique

- Vérifiables par des tiers : preuve de responsabilité du signataire  
*la clé de signature ne doit **jamais** être transmise*
- Peuvent servir à sécuriser les répertoires de clés publiques  
Infrastructure de gestion de clés (*IGC* ou *PKI*)
  - Chaque entrée de répertoire est signée par une *autorité de certification* (*AC* ou *CA*)
  - Les clés publiques des autorités de certification sont dans un répertoire, chacune signée par une *AC* de plus haut niveau, etc.
- **Attention** : être sûr de ce qu'on signe !  
*What you sign is **not necessarily** what you see*

# Propriétés des signatures à clé publique

Julius. Caesar  
Via Appia 1  
Rome, The Roman Empire

May, 22, 2005

To Whom it May Concern:

Alice Falbala fulfilled all the requirements of the Roman Empire intern position. She was excellent at translating roman into her gaul native language, learned very rapidly, and worked with considerable independence and confidence.

Her basic work habits such as punctuality, interpersonal deportment, communication skills, and completing assigned and self-determined goals were all excellent.

I recommend Alice for challenging positions in which creativity, reliability, and language skills are required.

I highly recommend hiring her. If you'd like to discuss her attributes in more detail, please don't hesitate to contact me.

Sincerely,

Julius Caesar

Julius. Caesar  
Via Appia 1  
Rome, The Roman Empire

May, 22, 2005

Order:

Alice Falbala is given full access to all confidential and secret information about GAUL.

Sincerely,

Julius Caesar

# Propriétés des signatures à clé publique

Julius, Caesar  
Via Appia I  
Rome, The Roman Empire

May, 22, 2005

To Whom it May Concern:

Alice Falbala fulfilled all the requirements of the Roman Empire intern position. She was excellent at translating roman into her gaul native language, learned very rapidly, and worked with considerable independence and confidence.

Her basic work habits such as punctuality, interpersonal deportment, communication skills, and completing assigned and self-determined goals were all excellent.

I recommend Alice for challenging positions in which creativity, reliability, and language skills are required.

I highly recommend hiring her. If you'd like to discuss her attributes in more detail, please don't hesitate to contact me.

Sincerely,

Julius Caesar

Julius, Caesar  
Via Appia I  
Rome, The Roman Empire

May, 22, 2005

Order:

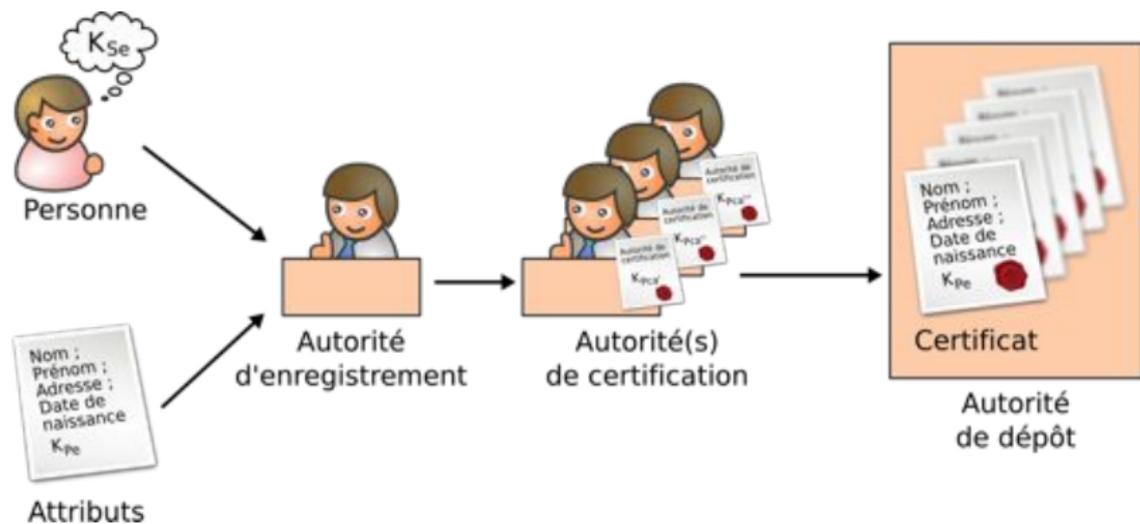
Alice Falbala is given full access to all confidential and secret information about GAUL.

Sincerely,

Julius Caesar

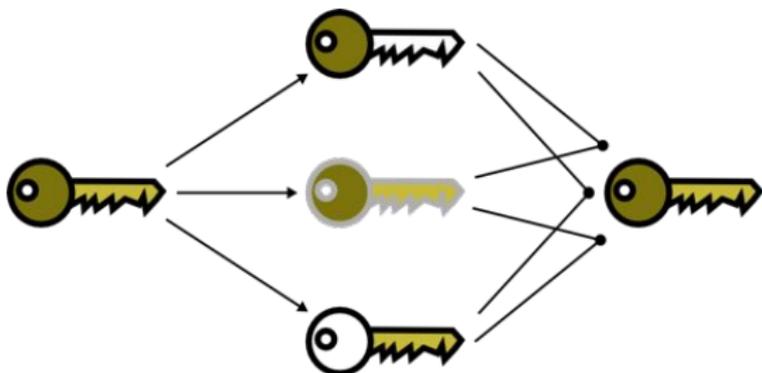
```
$ cat letter_of_rec.ps | openssl md5
a25f7f0b29ee0b3968c860738533a4b9
$ cat order.ps | openssl md5
a25f7f0b29ee0b3968c860738533a4b9
$ diff order.ps letter_of_rec.ps
Binary files order.ps and letter_of_rec.ps differ
```

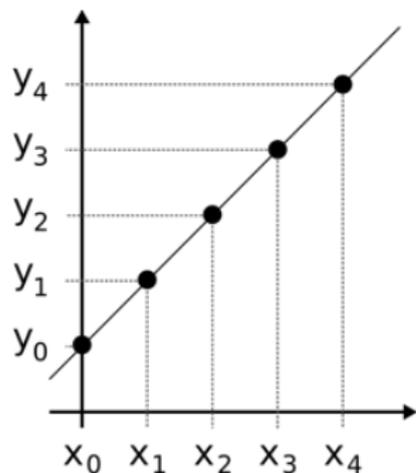
# Certificats et PKI – exemple X509



# Schémas à seuil

- Stocker  $K$  sous la forme d'un ensemble de valeurs  $K_i$  (images), telles que :
  - $S$  images permettent de reconstruire le secret ( $S = \text{seuil}$ )
  - $S - 1$  images n'apportent aucune information
- Si on sait générer  $N$  images (avec  $N > S$ ), alors on tolère de perdre jusqu'à  $N - S$  images



Schémas à seuil – exemple avec  $S = 2$ 

$$P(x) = \{a \cdot x + b\}$$

$$y_0 = a \cdot 0 + b$$

$$y_1 = a \cdot 1 + b$$

$$y_2 = a \cdot 2 + b$$

...

$$y_m = a \cdot m + b$$

- A partir de deux points quelconque on sait calculer  $a$  et  $b$ 
  - ⇒ avec 2 images quelconques, on reconstruit le secret
  - ⇒ avec une seule image, on n'a rien

## Schémas à seuil – généralisation à polynôme de degré $n$

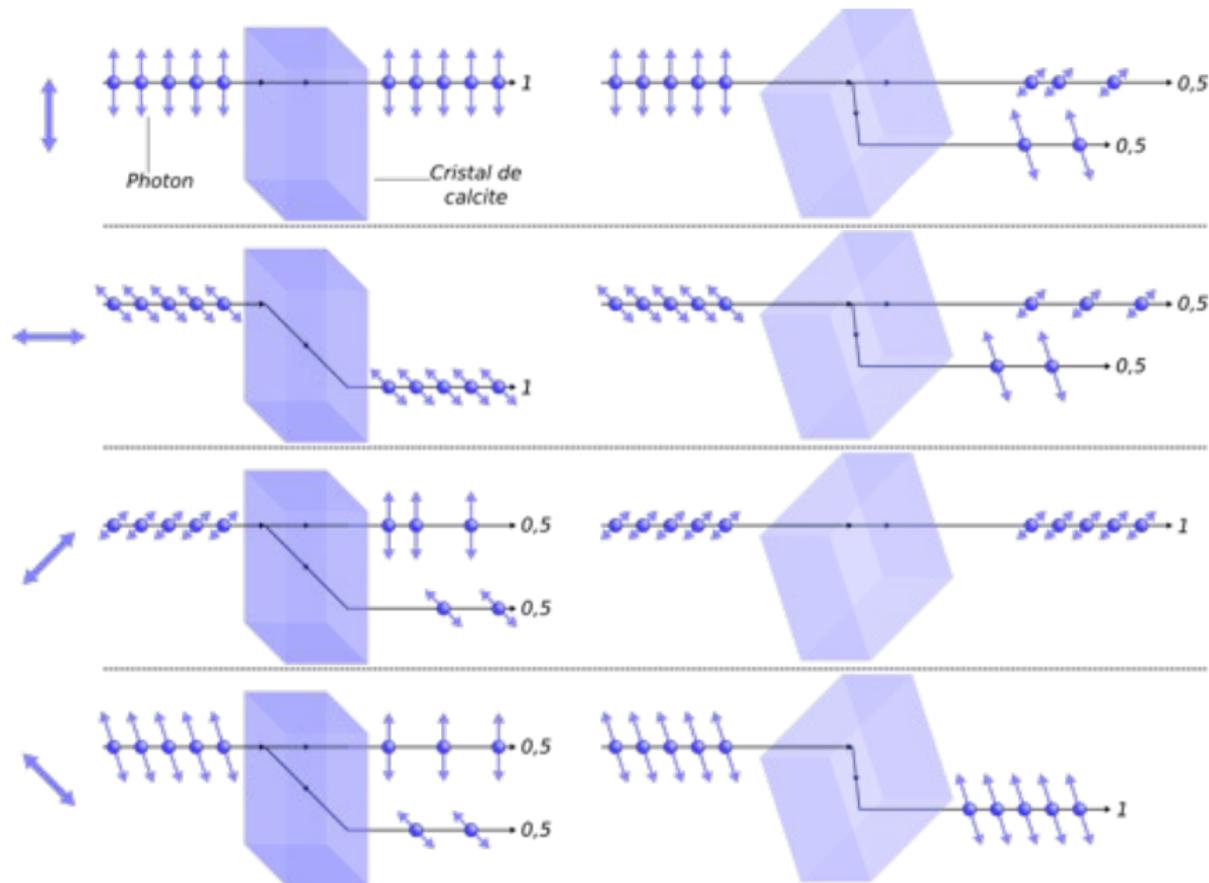
- $P(x) = a_0 + a_1 \cdot x + a_2 \cdot x^2 + \dots + a_n \cdot x^n$
- Si l'on connaît  $S = n + 1$  points, on sait recalculer les coefficients du polynôme ( $S$  équations à  $S$  inconnues) : interpolation polynomiale de Lagrange
- Les calculs se font modulo  $q$ , dans un corps de Galois (avec  $q$  premier) :  $GF(q) = \{0, 1, 2, \dots, q - 1\}$

$$P(x) = \sum_{l \leq i \leq k} y_i \prod_{l \leq j \neq i \leq k} \frac{(x - x_j)}{(x_i - x_j)} \pmod{q}$$

# Quantum Key Distribution[?]

- Sécurité calculatoire des systèmes cryptographiques
  - *Logarithme discret, factorisation en nombre premiers, etc.*
  - Les performances des machines augmentent (ordinateur quantique)  
⇒ Sécurité à long terme menacée
- Le code de Vernam
  - Le seul permettant d'établir un canal inconditionnellement sûr
  - Utilisation du *ou-exclusif*
  - La clé ne doit être utilisée qu'une seule fois
  - Problème majeure :  $|K| \geq |M|$ 
    - Par exemple, si Alice veut communiquer 1 Go de données à Bob, elle doit échanger avec Bob et de manière **sûre**, une clé de taille 1 Go
- Du côté de la mécanique quantique
  - Théorème de *non-clonage*[WZ82]
    - Impossibilité de cloner des états quantiques inconnus
  - Application aux photons → cryptographie quantique

# Quantum Key Distribution[?]



# Quantum Key Distribution[?]

## Correspondances

0	↔	↗, ↔
1	↔	↘, ↓

### Alice (A) se prépare à l'envoi

A choisit $n$ bits aléatoirement	1	0	1	1	0	0	1	1
A choisit $n$ polarisateurs aléatoirement	+	×	×	+	×	+	+	×

### Bob (B) se prépare à la réception

B choisit $n$ polarisateurs aléatoirement	+	×	+	+	×	×	+	×
---	---	---	---	---	---	---	---	---

### Alice communique son secret à Bob (*canal quantique*)

A envoie les photons	↑↓	↗	↘	↑↓	↗	↔	↑↓	↘
B détecte les photons	↑↓	↗		↑↓	↗	↗	↑↓	↘

### Alice et Bob comparent leurs connaissances (*canal public*)

B envoie les polarisateurs des photons détectés	+	×		+	×	×	+	×
A indique les polarisateurs corrects	✓	✓		✓	✓	✗	✓	✓
A et B partage la chaîne secrète	1	0		1	0		1	1

### Alice et Bob évaluent la probabilité d'avoir été espionné (*canal public*)

B sacrifie 1/3 des bits correctement reçus	1			0				1
A confirme la validité des bits sacrifiés	✓			✓				✓

### Résultat

A et B partagent un secret		0		1			1	
----------------------------	--	---	--	---	--	--	---	--

## Quantum Key Distribution[?] – avec écoute

## Correspondances

0  $\Leftrightarrow$  ↗, ↔1  $\Leftrightarrow$  ↘, ↓

## Alice (A) se prépare à l'envoi

A choisit $n$ bits aléatoirement	1	0	1	1	0	0	1	1
A choisit $n$ polarisateurs aléatoirement	+	×	×	+	×	+	+	×

## Bob (B) se prépare à la réception

B choisit $n$ polarisateurs aléatoirement	+	×	+	+	×	×	+	×
---	---	---	---	---	---	---	---	---

## Alice communique son secret à Bob (canal quantique)

A envoie les photons	↕	↗	↘	↕	↗	↔	↕	↘
B détecte les photons	↕	↗		↕	↗	↗	↕	↗

## Alice et Bob comparent leurs connaissances (canal public)

B envoie les polarisateurs des photons détectés	+	×		+	×	×	+	×
A indique les polarisateurs corrects	✓	✓		✓	✓	✗	✓	✓
A et B partagent la chaîne secrète	1	0		1	0		1	0

## Alice et Bob évaluent la probabilité d'avoir été espionné (canal public)

B sacrifie 1/3 des bits correctement reçus	1			0				0
A confirme la validité des bits sacrifiés	✓			✓				✗

## Résultat

A et B partagent un secret		0		1			1	
----------------------------	--	---	--	---	--	--	---	--

## Autres fonctions cryptographiques – sujets

- Cryptographie à clés publiques basées sur l'identité
- Stéganographie
- *Watermarking*
- Génération de nombres aléatoires et pseudo-aléatoires
- Générateur de nombres premiers
- Ecrous (*key escrow*)
- Vote
- Horodatage
- Cryptanalyse
- Protocoles

# Sommaire

## Les défenses

Cryptographie

Prévention et élimination des vulnérabilités

Cloisonnement

Audit

Détection d'intrusions

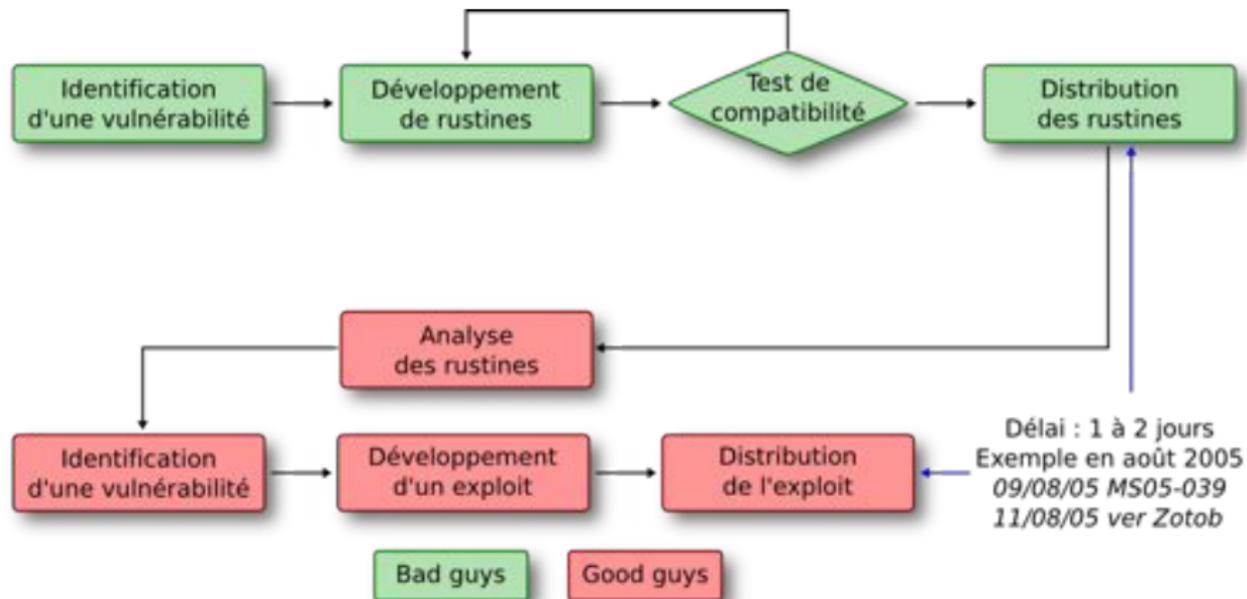
# Prévention des vulnérabilités

- Vulnérabilités = fautes de conception ou de configuration
- Les systèmes commerciaux actuels sont trop complexes pour être sans fautes
- Objectifs divergents
  - Disponibilité / sécurité (*TCP/IP*)
  - Rentabilité-efficacité / sécurité
- Il existe des outils pour éviter d'introduire des vulnérabilités classiques (par exemple des débordements de tampons)

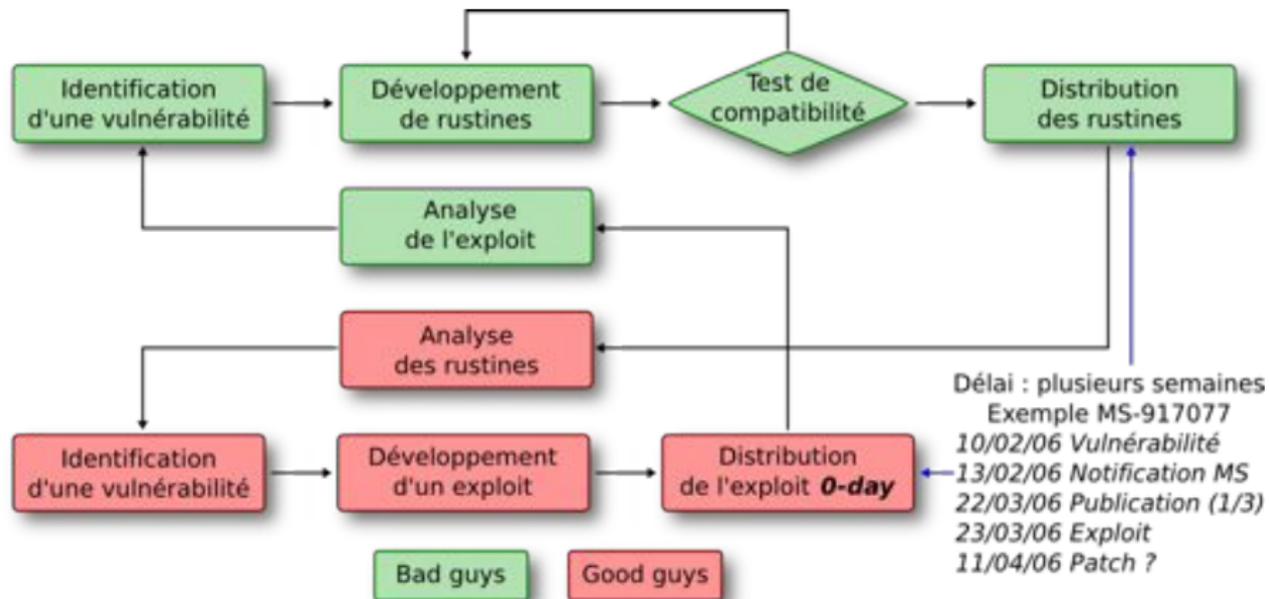
# Elimination des vulnérabilités

- Cycle habituel
  - Identification d'une nouvelle vulnérabilité
  - *Exploit*
  - *Patches* (rustines)
  - Nouvelle version
- **Mais**
  - Nombreuses alertes → quelles sont celles qui sont pertinentes ?
  - Certains *patches* sont imparfaits → élimination d'une fonctionnalité indispensable
  - Certaines applications indispensables ne sont plus compatibles

# Cycle de vie des patches



# Cycle de vie des exploits



# Sommaire

## Les défenses

Cryptographie

Prévention et élimination des vulnérabilités

**Cloisonnement**

Audit

Détection d'intrusions

# Cloisonnement

- Empêcher toute communication/interaction qui n'est pas nécessaire
  - Isoler les systèmes de développement des systèmes opérationnels, les systèmes de surveillance des systèmes surveillés
  - Fragmenter et disséminer l'information, séparer les pouvoirs
- Pare-feux
  - Filtrer les adresses sources/destination ( $IP + n^{\circ}$  port), entrée/sortie
  - Traduction d'adresse (*NAT*)
  - Mandataire d'application (*proxy*) pour vérifier les protocoles d'application
    - Liaison avec *IDS stateful*
    - Option : outil *anti-reconnaissance*, *Intrusion Prevention System (IPS)*

# Sommaire

## Les défenses

Cryptographie

Prévention et élimination des vulnérabilités

Cloisonnement

**Audit**

Détection d'intrusions

# Audit – journalisation

- Enregistrer toutes les opérations liées à la sécurité (réussies ou non)
  - Connexion/déconnexion d'utilisateurs
  - Création/modification/destruction d'informations de sécurité
    - Droits d'accès
    - Mots de passe
    - Enregistrements d'audit
    - ...
  - Changement de privilèges
- Informations enregistrées
  - Date, heure
  - Identité de l'utilisateur
  - Type d'opération, référence des objets
  - ...

# Sommaire

## Les défenses

Cryptographie

Prévention et élimination des vulnérabilités

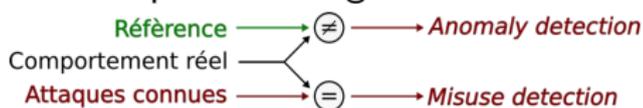
Cloisonnement

Audit

Détection d'intrusions

# Détection d'intrusion – IDS

- Principe : détection d'erreurs dues à des intrusions
- Deux familles de techniques : analogie avec les détecteurs de virus



- **Anomaly detection** : par discrimination entre les comportements normaux (utilisateurs non-malveillants) et les comportements anormaux (intrus) : **profils statistiques**, **systèmes experts**, **systèmes immunitaires**, etc.
- **Misuse detection** : par reconnaissance de **signatures** correspondant à des attaques connues (*stateless*, *stateful*)
- Implémenté dans chaque ordinateur (*host-based IDS*) ou sur des machines observant le réseau (*network-based IDS*)
- Problème
  - Taux de fausses alarmes (*false positives*)
  - Taux de non détection (*false negatives*)
- Les autres mécanismes de détection d'erreurs peuvent aussi être efficaces vis-à-vis des intrusions

# Sommaire

Un peu d'histoire

Les propriétés de la sécurité

Les attaques

Les défenses

La protection des systèmes informatiques

# Sommaire

## La protection des systèmes informatiques

Politiques et modèles de sécurité

Authentification des utilisateurs

# Sommaire

La protection des systèmes informatiques

Politiques et modèles de sécurité

Authentification des utilisateurs

# Politiques de sécurité

## Politique de sécurité

Une politique de sécurité est l'ensemble des lois, règles et pratiques qui régissent la façon dont l'information sensible et les autres ressources sont gérées, protégées et distribuées à l'intérieur d'un système spécifique.[?]

- Modèle de sécurité → formalisme mathématique

# Politiques de sécurité

- **Objectifs à satisfaire**, par exemple :
  - **Confidentialité** : le dossier médical ne peut être consulté que par le patient et son ou ses médecins traitants
  - **Intégrité** : un chèque de plus de 1000 € doit être signé par le Président et le Trésorier
  - **Disponibilité** : si la carte et le *PIN* sont valides, le distributeur de billets doit fournir l'argent dans les 30 secondes
- **Règles**, par exemple :
  - Un fichier ne peut être lu que par les utilisateurs autorisés par le propriétaire du fichier
  - Un message de type *chèque de plus de 1000 €* n'est valide que s'il est signé par  $P_1$  et  $T_2$  et que les signatures sont valides
  - L'insertion d'une carte lance automatique l'action, etc.

# Cohérence d'une politique

- La politique est cohérente si, partant d'un état quelconque où les objectifs sont satisfaits, il n'est pas possible d'atteindre, en respectant les règles, un état où ils ne sont plus satisfaits
- Intérêts d'un modèle formel
  - Décrire de manière précise les objectifs et les règles
  - Prouver des propriétés sur la politique (cohérence, complétude, etc.) et sur son implémentation par le système informatique

# Politique, protection et contrôle d'accès

- Les règles doivent être mises en œuvre par des mécanismes (matériels, logiciels)
- Facile à imaginer pour les règles du type "*il est permis de ...*" ou "*il est interdit de ...*" → **mécanismes de protection**  
instructions privilégiées, contrôle d'accès à la mémoire, contrôle à l'ouverture de fichiers, etc.  
→ **autorisation : confidentialité, intégrité**
- Difficile pour les règles du type "*il est obligatoire de ...*" ou "*il est recommandé de ...*"  
→ **actions automatiques, gestion des ressources, etc : intégrité, disponibilité**

# Politique d'autorisation

- Un **sujet** a un **droit d'accès** sur un **objet**
  - ⇔ le sujet est autorisé à exécuter la méthode d'accès sur cet objet
    - Sujet : processus qui s'exécute pour le compte d'un utilisateur
    - Utilisateur : personne physique ou service identifié dans le système
    - Objet : conteneur d'information, défini par un nom, un état et des méthodes, par exemple : fichier, périphérique, processus, etc.

# Modèle HRU

- L'état de sécurité du système est défini par :
  - $D$  : l'ensemble de tous les droits
  - $S$  : l'ensemble des sujets courants
  - $O$  : l'ensemble des objets courants,  $S \subseteq O$
  - $A$  : l'ensemble des droits courants de chaque sujet sur chaque objet  
 $A$  est représenté par une matrice avec une ligne par sujet  $s_i$  et une colonne pour chaque objet  $o_j$   
 $A_{ij} = d_{ij}$  avec  $d_{ij} \subseteq D$

$$(s_i, o_j, d_k) \text{ est vrai} \Leftrightarrow s_i \text{ a le droit } d_k \text{ sur } o_j$$

$$d_{ij} = \{d_k \in D \mid (s_i, o_j, d_k)\}$$

# Politique d'autorisation discrétionnaires

## *DAC : Discretionary Access Control*

- Les droits d'accès à chaque information sont manipulés par le responsable de l'information (généralement le propriétaire), à sa discrétion
- Les droits peuvent être accordées à chaque utilisateur individuellement ou à des groupes d'utilisateurs ou les deux

# Politique d'autorisation discrétionnaires

- Exemple : protection des fichiers UNIX

- Règles

- Un utilisateur peut créer librement des fichiers dont il devient propriétaire  
 $(A, F, \text{créer}) \xrightarrow{A} (A, F, \text{propriétaire}) \wedge (A, F, \text{écrire}) \wedge (A, F, \text{lire})$
- Les droits d'accès à un fichier sont définis librement par le propriétaire : par exemple, il peut décider quels utilisateurs sont autorisés à lire le fichier  
 $(A, F, \text{propriétaire}) \xrightarrow{A} (B, F, \text{lire})$

- Objectif

- Un utilisateur non-autorisé à lire un fichier ne peut obtenir aucune information contenue dans le fichier (même avec la complicité d'un utilisateur autorisé) → impossible à garantir*

- Exemple

- $S = \{s_1, s_2, s_3\}$

- $O = \{f_1, f_2\}$

- $D = \{\text{propriétaire, lire, écrire}\}$

- $A = \{(s_1, f_1, \text{propriétaire})\}$

- (2 et 6)  $(s_1, f_1, \text{propriétaire}) \xrightarrow{s_1} (s_2, f_1, \text{lire})$

- (1 et 2)  $(s_2, f_2, \text{créer}) \xrightarrow{s_2} (s_2, f_2, \text{écrire}) \wedge (s_3, f_2, \text{lire})$

- (7 et 8)  $(s_2, f_1, \text{lire}) \wedge (s_2, f_2, \text{écrire}) \wedge (s_3, f_2, \text{lire}) \xrightarrow{s_2} (s_3, k(f_1), \text{lire})$

# Inconvénient des politiques *DAC*

- Possibilité d'**abus de pouvoir**  
(par malveillance ou par maladresse)
  - S'il est possible pour un utilisateur légitime d'accéder à certains objets ou d'en modifier les droits d'accès, il est possible qu'un *Cheval de Troie* en fasse de même
  - Si un utilisateur a le droit de lire une information, il a (en général) automatiquement le droit de la divulguer à n'importe qui
  - Il est difficile de corriger les effets d'une divulgation

# Politique d'autorisation obligatoires

## *MAC : Mandatory Access Control*

- Des règles incontournables sont imposées, en plus des règles discrétionnaires
- Exemple : politique de confidentialité multi-niveau *militaire*  
Des **classes** sont assignées aux utilisateurs (**habilitation**) et aux objets (**classification**)

Une classe est définie par :

Un niveau (ordonné)    Un compartiment = {catégories}

Non-classifié

Cryptographie

Confidentiel

Nucléaire

Secret

OTAN

Très secret

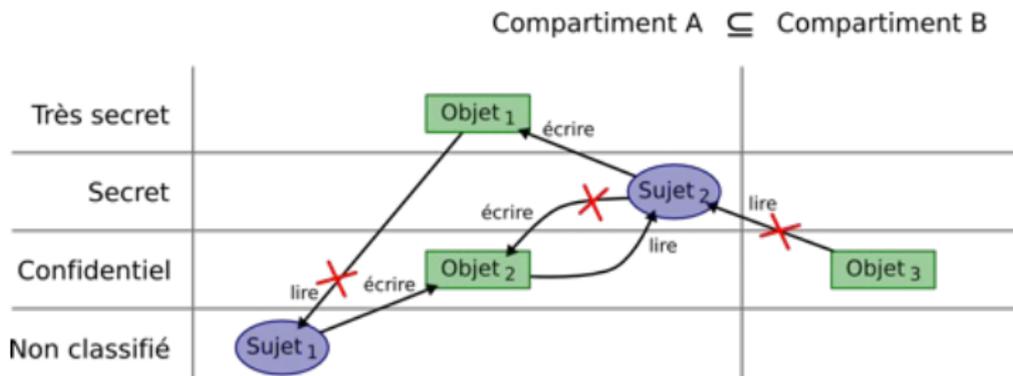
Irak

...

# Politique de *Bell-LaPadula* (confidentialité)

- A chaque sujet  $s_i$  correspond une habilitation  $h(s_i)$  avec un niveau  $n_i$  et un compartiment  $\Sigma_i$
- A chaque objet  $o_j$  correspond une classification  $c(o_j)$  avec un niveau  $n_j$  et un compartiment  $\Sigma_j$
- Règle simple :  
 $(s_i, o_j, \text{lire}) \Rightarrow n_j \leq n_i \wedge \Sigma_j \subseteq \Sigma_i \quad (h(s_i) \text{ domine } c(o_j))$
- Règle étoile :  
 $(s_i, o_j, \text{lire}) \wedge (s_i, o_k, \text{écrire}) \Rightarrow n_j \leq n_k \wedge \Sigma_j \subseteq \Sigma_k \quad (c(o_k) \text{ domine } c(o_j))$
- Propriété :  
 Si  $h(s_n)$  ne domine pas  $c(o_i)$ , il n'existe pas de suite telle que  
 $(s_l, o_i, \text{lire}) \wedge (s_l, o_j, \text{écrire}) \wedge (s_m, o_j, \text{lire}) \wedge \dots \wedge (s_x, o_k, \text{écrire}) \wedge (s_n, o_k, \text{lire})$   
 Interdire à tout sujet d'obtenir des informations d'un objet de niveau supérieur à son habilitation  
 $\Rightarrow$  Pas de fuite d'information possible

# Inconvénients de Bell-LaPadula

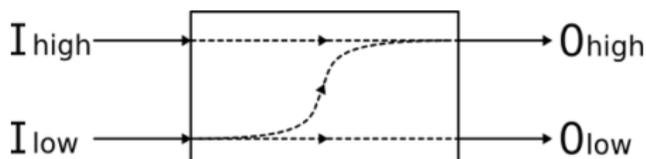


- Surclassification : au fur et à mesure que l'information est traitée, sa classification augmente

⇒ *Trusted process* pour déclassifier

# Autres politiques de confidentialité

- *Muraille de Chine*  
Chez les agents de change (conflits d'intérêt)
- Modèle de non interférence :  $O_{low}$  ne dépend pas de  $I_{high}$

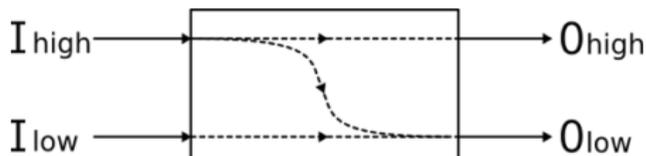


# Politique de *Biba* (intégrité)

- Multiple niveaux d'intégrité (crédibilité, vérification, etc.)
  - A chaque sujet  $s_i$  correspond un niveau  $is(s_i)$
  - A chaque objet  $o_j$  correspond un niveau  $io(o_j)$
- Règles
  - $(s_i, o_j, \text{observer}) \Rightarrow is(s_i) \leq io(o_j)$
  - $(s_i, o_j, \text{modifier}) \Rightarrow io(o_j) \leq is(s_i)$
  - $(s_i, s_j, \text{invoquer}) \Rightarrow is(s_j) \leq is(s_i)$
- Propriété : empêcher la *contamination* des niveaux élevés : diffusion de fausses informations, propagation d'erreur, etc.
- Inconvénient : dégradation progressive des niveaux d'intégrité

# Autres politiques d'intégrité

- Politique de *Clark-Wilson* (transactions financières)
  - Deux niveaux d'intégrité
    - *UDI* (données non contraintes)
    - *CDI* (données contraintes), vérifiables par des *IVP* (*Integrity Verification Procedures*), ne pouvant être manipulées que par des *TP* (*Transformation Procedures*) certifiées
  - Règles : listes de relations autorisées
    - $CDI \leftrightarrow TP$ , Utilisateurs  $\leftrightarrow TP$  (avec éventuellement *separation of duty*),  $UDI \leftrightarrow CDI$  par ( $TP + IVP$ )
- Modèle de non-interférence :  $O_{high}$  ne dépend pas de  $I_{low}$



# Politiques basées sur les rôles

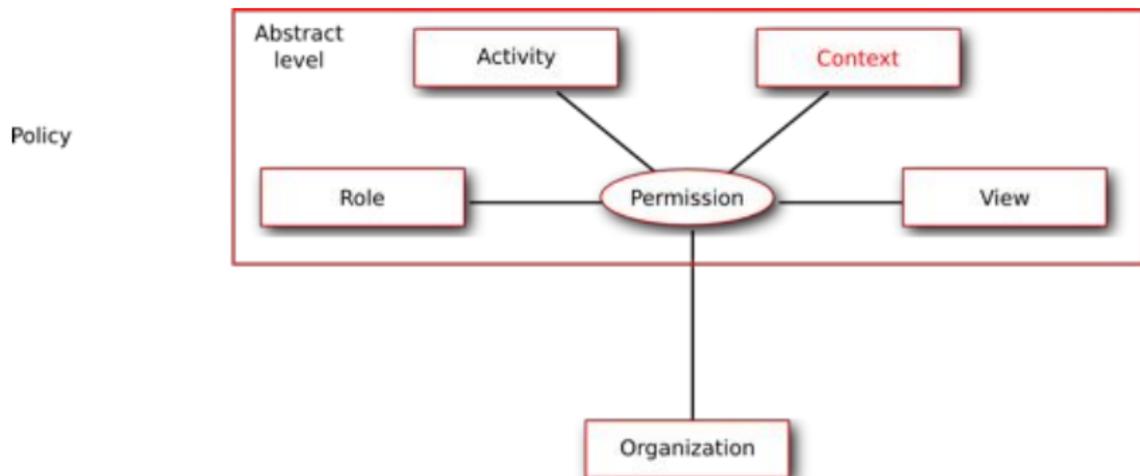
## *RBAC : Role-Based Access Control*

- On définit des rôles, pour représenter des fonctions dans l'organisation
- On associe à chaque rôle les privilèges (ensemble de droits) nécessaires pour remplir la fonction
- On associe à chaque utilisateur le(s) rôle(s) qu'il peut jouer dans l'organisation
- Administration facilitée
  - Les rôles et leurs privilèges changent rarement
  - Il suffit d'identifier les rôles que peut jouer un utilisateur

# OrBAC – Organization-Based Access Control

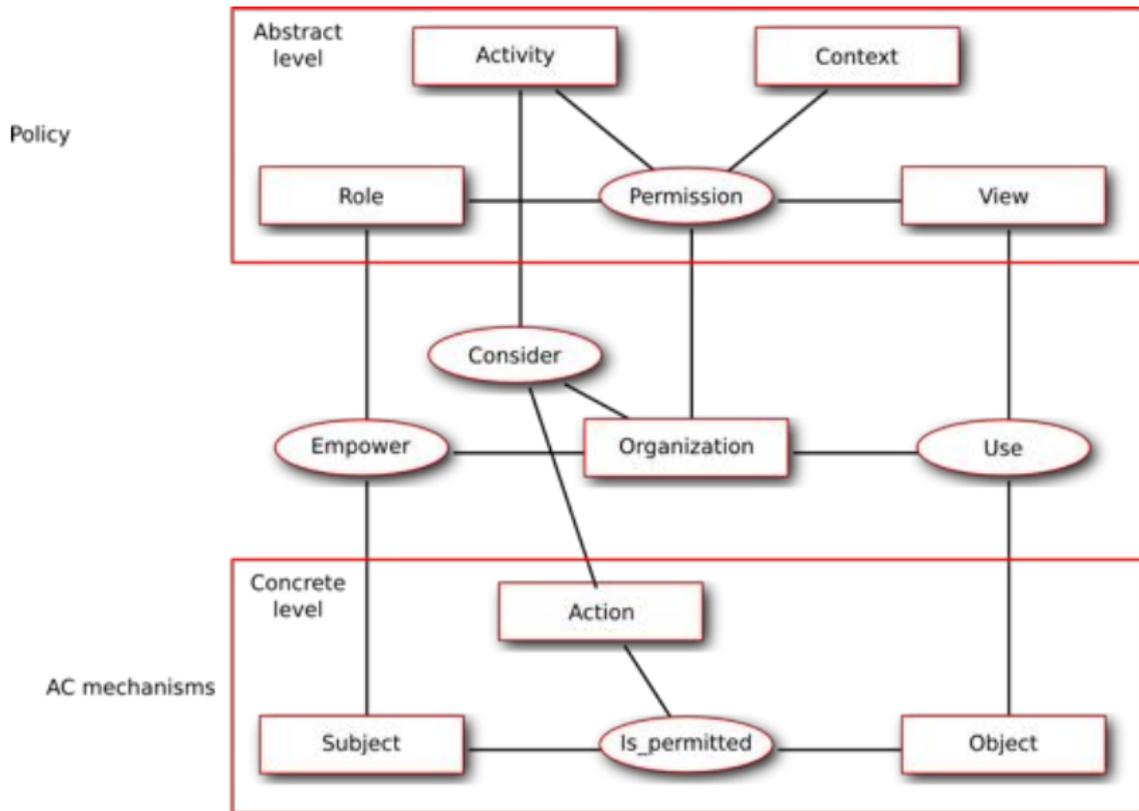
- Modèle conçu dans *MP6*  
Modèles et politiques de sécurité des systèmes d'information et de communication en santé et social  
*Ernst & Young, IRIT, LAAS-CNRS, ONERA, ENST-Bretagne, etc.*
- Abstractions
  - User → rôle
  - Objet → vue
  - Action → activité
- Liaisons entre niveaux abstrait (**politique**) et concret (**mécanismes de contrôle d'accès**) : définies par l'organisation
- Règles
  - Définies au niveau abstrait
  - Permissions/interdictions + obligations
  - Validées par le contexte (concret)

# OrBAC – Organization-Based Access Control



- La politique est définie au niveau abstrait → définition de règles
  - Permission (Organization, Role, Activity, View,  $B(\text{context})$ )
  - Interdiction (Organization, Role, Activity, View,  $B(\text{context})$ )
  - Obligation (Organization, Role, Activity, View,  $B(\text{context})$ )

# OrBAC – Organization-Based Access Control



# Sommaire

## La protection des systèmes informatiques

Politiques et modèles de sécurité

Authentification des utilisateurs

# Authentification des utilisateurs

## *Nécessaire pour l'autorisation et l'audit*

- **Authentification** = identification + vérification de l'identité
- **Identité** = information (non confidentielle) **spécifique** à une personne, connue au moins par elle et par le vérificateur : nom, numéro, etc.
- **Vérification** de la correspondance entre l'identité et la personne, en utilisant :
  - Quelque chose qu'elle connaît (mot de passe, informations personnelles, etc.) ou qu'elle sait faire (reconnaissance de forme, association d'idées, etc.)
  - Quelque chose qu'elle possède : badge, carte à puce, etc.
  - Quelque chose qui lui est propre (biométrie) : empreinte digitale, signature, voix, fond de l'œil, iris, forme de la main, etc.
  - Ou plusieurs de ces moyens : carte à puce + PIN, etc.

# Méthodes de vérification

- Secret partagé entre la personne et le vérificateur (mot de passe, informations personnelles, etc.)
- Secret correspondant à une caractéristique biométrique (stockée par le vérificateur ou non), non falsifiable, non rejouable
- Secret connu par la personne, vérifié par des informations ou protocoles publics (sans apport de connaissance, *zero-knowledge*)

# Qualité de l'authentification

- La qualité des systèmes d'authentification dépend du taux d'acceptation à tort (*false acceptance rate*, *FAR*) et du taux de rejet à tort (*false rejection rate*, *FRR*)

Empreinte digitale  $FAR \approx 10^{-6}$   $FRR \approx 10^{-3}$  (SAGEM, Compaq, NEC)  
Iris  $FAR \approx 10^{-12}$   $FRR \approx 10^{-4}$  (Sensor, IriScan)

- Il faut distinguer si la victime dont on prend l'identité est consentante (par exemple, transmet volontairement son mot de passe) ou non : supériorité des systèmes biométriques
- Mais, les systèmes biométriques ont des limitations : falsification (prothèses), handicapés, acceptation sociale, difficulté de révocation, etc.

# Références I

 M. de Marlès, *Histoire de marie stuart, reine d'ecosse.*

 Taher El Gamal, *A public key cryptosystem and a signature scheme based on discrete logarithms*, Proceedings of CRYPTO 84 on Advances in cryptology (New York, NY, USA), Springer-Verlag New York, Inc., 1985, pp. 10–18.

 Gene H. Kim and Eugene H. Spafford, *The design and implementation of tripwire : a file system integrity checker*, CCS '94 : Proceedings of the 2nd ACM Conference on Computer and communications security (New York, NY, USA), ACM, 1994, pp. 18–29.

 A. Labanoff, *Lettres, instructions et mémoire de marie stuart, reine d'ecosse.*

 L'Expansion.com, *Des fraudeurs se font passer pour les services des impots sur internet*, 6 Octobre 2009.

## Références II

-  *National institute of standards and technology*, Site Internet, <http://www.itl.nist.gov/lab/bulletns/bltnjun06.htm>.
-  Plutarque, *Vie de lysandre*, Les Vies des hommes illustres.
-  R. L. Rivest, A. Shamir, and L. Adleman, *A method for obtaining digital signatures and public-key cryptosystems*, Commun. ACM **21** (1978), no. 2, 120–126.
-  Xiaoyun Wang, Dengguo Feng, Xuejia Lai, and Hongbo Yu, *Collisions for hash functions MD4, MD5, HAVAL-128 and RIPEMD*, 2004, URL : <http://eprint.iacr.org/2004/199/>.
-  W. K. Wootters and W. H. Zurek, *A single quantum cannot be cloned*, Nature **299** (1982), 802–803.